FRAME LINE LTD.
Development of an Invoicing and Reporting System

28th March 2011

Prepared by: Gillian Lane  Supervisor: Denise Leahy
ABSTRACT

Frame Line Ltd, a professional photography company, requested a fully automated invoicing and reporting system in order to reduce the time wasted by manual invoice generation. The primary objective of the project was to facilitate this request by building a system in Microsoft Access using Visual Basic and SQL Code. The secondary objective of the project was to provide the client with a training programme for new staff.
PREFACE

The project was requested by Frame Line Ltd., a professional photography company. Prior to starting the project a requirements document was developed in order to ensure the client, supervisor and student had the same expectations of the automatic invoicing and reporting system and the training programme.

The business of Frame Line Ltd can be split into three sectors; photo centres which are part of the Fujifilm enterprise (Bray, Dalkey and Shankill), a corporate photography service (Studio Lab) and an online processing facility (Print Digital 4 U).

Due to the segmented nature of Frame Line Ltd’s business, a number of issues arose during the system development phase in relation to data importation. These issues were overcome by performing a thorough review of Frame Line’s current data systems and client enquiry.

The invoicing and reporting system developed, automatically gathers all the relevant data from files that Fujifilm provides for Frame Line Ltd. The system uses Visual Basic code to manipulate the data into the relevant format to provide automated invoices and reports through a user-friendly interface. The system also provides a training programme for new and current staff.

The system issues identified in the development phase were resolved and Frame Line Ltd has implemented the invoicing and reporting system. As evidenced by system testing, all requirements have been met.

This project was challenging but extremely rewarding to complete. A considerable amount of experience has been gained from performing this task. Though it seemed an impossible task at times, the system has been implemented into Frame Line and all requirements were met.

I would like to thank Mr. Ian MacMahon for his interminable cooperation and perseverance to complete this project. I would also like to thank Mr. MacMahon for the opportunity to complete this project. I would like to thank Mr. Michael Walsh and Mr. Louis Asslett for their support and advice with this project, Mr. Eoin O’Reilly for his patience and diligence and Mr. Ciaran Tobin for his invaluable assistance with Visual Basic coding.

Should the client require any assistance in the future with implementing the recommendations outlined at the end of this report, they may contact Gillian Lane at laneg@tcd.ie without hesitation. It has been a pleasure to develop this system and I look forward to working with Mr. MacMahon in the future.

Finally, I would like to thank my project supervisor, Mrs. Denise Leahy. Her continuous encouragement and direction ensured the project was completed successfully.
# FRAME LINE LTD.
Development of an Invoicing and Reporting System

*March 2011*

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>NO.</th>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Client</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Project Background</td>
<td>2</td>
</tr>
<tr>
<td>1.3</td>
<td>Terms of Reference</td>
<td>3</td>
</tr>
<tr>
<td>1.4</td>
<td>Summary of the remaining chapters</td>
<td>3</td>
</tr>
<tr>
<td>2.</td>
<td>SYSTEM OVERVIEW</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Objectives</td>
<td>4</td>
</tr>
<tr>
<td>2.2</td>
<td>Technical Environment</td>
<td>5</td>
</tr>
<tr>
<td>2.3</td>
<td>System Overview Diagram</td>
<td>6</td>
</tr>
<tr>
<td>3.</td>
<td>DESCRIPTION OF WORK DONE</td>
<td>8</td>
</tr>
<tr>
<td>3.1</td>
<td>Methodology</td>
<td>8</td>
</tr>
<tr>
<td>3.2</td>
<td>Analysis</td>
<td>8</td>
</tr>
<tr>
<td>3.3</td>
<td>Design</td>
<td>13</td>
</tr>
<tr>
<td>3.4</td>
<td>Database Development</td>
<td>14</td>
</tr>
<tr>
<td>3.5</td>
<td>Application Development</td>
<td>15</td>
</tr>
<tr>
<td>3.6</td>
<td>Testing</td>
<td>18</td>
</tr>
<tr>
<td>4.</td>
<td>CONCLUSIONS AND RECOMMENDATIONS</td>
<td>19</td>
</tr>
<tr>
<td>4.1</td>
<td>Conclusions</td>
<td>19</td>
</tr>
<tr>
<td>4.2</td>
<td>Recommendations</td>
<td>20</td>
</tr>
</tbody>
</table>
## APPENDICES

<table>
<thead>
<tr>
<th>NO.</th>
<th>CONTENT</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.</td>
<td>Original Project Outline</td>
<td>A.1</td>
</tr>
<tr>
<td>B.</td>
<td>Interim Report</td>
<td>B.1</td>
</tr>
<tr>
<td>C.</td>
<td>Technical Commentary</td>
<td>C.1</td>
</tr>
<tr>
<td>D.</td>
<td>User Manual</td>
<td>D.1</td>
</tr>
<tr>
<td>E.</td>
<td>Design Documentation</td>
<td>E.1</td>
</tr>
<tr>
<td>F.</td>
<td>Sample Reports and Screen shots</td>
<td>F.1</td>
</tr>
<tr>
<td>G.</td>
<td>Source Code</td>
<td>G.1</td>
</tr>
<tr>
<td>H.</td>
<td>Implementation and Test Documentation</td>
<td>H.1</td>
</tr>
</tbody>
</table>

REFERENCES
1. INTRODUCTION

The primary objective of the project was to build a system that will enable Frame Line Ltd to automatically generate invoices and reports. Chapter 1 outlines the client’s business model, the background to the project and the terms of reference. Chapter 2 goes on to examine the requirements of the new system including the technical environment and a diagrammatical illustration of the system. The work carried out in order to ensure the objectives of the project were met, is outlined in Chapter 3. This chapter also documents the testing performed subsequent to the systems implementation to ensure all requirements were met. Finally the conclusions and recommendations are documented in Chapter 4.

1.1 Client

This section will introduce two companies, Fujifilm and Frame Line Ltd. Frame Line Ltd. is a photography company that runs three photo centres. The photo centres provide a photography service based on Fujifilm produce. For the purpose of this report, Frame Line Ltd. will be referred to as “the client” or “Frame Line”.

Frame Line Ltd:
Frame Line is a professional photography company that runs three photography stores, a professional photography service and an online processing facility linked to the three stores. The company comprises Bray, Shankill and Dalkey Photo Centres, Studio Lab and Print Digital 4 U [1]. Bray, Shankill and Dalkey Photo Centres are stores in which Fujifilm products are sold. Frame Line produces a range of products from photographs, cameras and personalised gifts. Studio Lab is aimed at corporate customers. It provides a professional photography service such as photographing brochure and product images [2]. Print Digital 4 U is the website that allows full time access to Studio Lab and Bray, Shankill and Dalkey Photo Centres.

Frame Line provides Fujifilm Ireland with a wholesale photo-gift processing facility. Fujifilm has an online ordering system which sends gift orders directly to a main computer at Shankill Photo Centre, one of the three stores of Frame Line. All gift orders that Frame Line produce for other Fujifilm stores are invoiced and forwarded to Head Office, Fujifilm Ireland. Shankill Photo Centre is a wholesaler for Fujifilm Ireland Ltd. Because of this, Shankill Photo Centre receives all gift orders from kiosks all over Ireland. The Fujifilm hierarchy can be seen in Appendix C.1, p. C.1.

Fujifilm Ireland:
Fujifilm is known as “the world’s largest photographic and imaging company” [3]. The company is renowned for innovative work in areas such as motion picture film and computer media. Their Irish website can be found at Fujifilm.ie. Their products include photo development, cameras, camera accessories and business systems. Fujifilm have 169 stores in Ireland, operated by individuals and multiple chains, situated in 24 of the 32 counties of Ireland. Each location produces photos and sells Fujifilm products.
Fujifilm have approximately 1500 kiosks situated around Ireland, located in chain stores such as Harvey Norman, McCabes Pharmacy Group and Health Express Group and other owner operated stores. These kiosks act as an automated sales attendant. The consumer can order Fujifilm prints and gifts directly from these free-standing units in stores. The kiosks use software that was developed by an Australian company called Whitech. Their software is used for the up loader and the downloader that Fujifilm Ireland uses to operate the kiosks. These free standing units are connected to the Fujifilm network. The kiosks use an up loader as a means of sending information from a local system to a remote system, for example Fujifilm Head Office upload information onto the kiosks. They also make use of a downloader as a means of receiving information from a remote system such as Fujifilm Head Office. Fujifilm use this way of transferring information between Head Office and the franchises also. When a new product is put on the market, Fujifilm can up load the information and the kiosks can receive it using a downloader. These kiosks currently offer 16 products that Frame Line’s Shankill Photo Centre produces. Frame Line agrees the prices of these products with Fujifilm Ireland. Once the prices are agreed, they are inputted into all of the kiosks around Ireland using this up load and download system. Fujifilm have kiosks in the North of Ireland also and so the current systems account for Sterling as well as Euro.

1.2 Project Background

The client has noted that a lot of time is wasted on the generation of manual invoices. The existing system requires that Frame Line generates their invoices by hand at the end of each month. The client believes that this process is out of date and urgently requires automation. Along with this issue, Frame Line need to be able to provide Fujifilm Head Office with monthly reports, such as the number of each product sold. This information will allow Frame Line to see trends in their orders, such as a peak at Christmas time. The third process that Frame Line has been looking to update is their process of training staff. New staff need to be able to learn quickly and also learn by doing, this creates a requirement for a training program to be developed. As a result of these issues, Frame Line would like a system that integrates all three aspects. The three aspects of this project are discussed below in more detail.

**Invoices**

Invoices are generated by Frame Line for every order they complete. Currently, a member of staff spends approximately two days at the end of each month manually producing invoices for all of the orders that were completed in that month. The client wishes to fully automate this process.

**Reports**

In order to keep track of their sales, the client would like to be able to produce reports on the products sold. These reports will be used by both Frame Line and also by Fujifilm Head Office. The client would like to able to generate both monthly and trend reports over a specified period of time.
Training Program

Bringing in new staff to Frame Line, currently means that they are thrown in at the deep end. They must learn-by-doing and be very adaptable. If a staff member needs to learn how to make a new product, they must look it up online. The client would like a training program that provides the information needed to make all the products that they provide. This also means that products may be added, changed and deleted from the system. This will ensure that all staff follow the same training standard and are kept up to date with new products.

1.3 Terms of Reference

The following terms of reference were agreed with the client on 25/11/2010 (Appendix B, p. B.1). Automation of their invoicing process, reporting functionality and a training program are the main requirements. Based on these requirements, the terms of reference are:

Primary
- To analyse the current system, including data sources, reporting and invoicing methods
- To create an automated system for reporting and invoicing
- To implement the system in Frame Line
- To prepare supporting documentation for the new system
- To train staff in the use of the new system

Secondary
- To create a customer database from previous orders
- To develop a training system for current and future staff

1.4 Summary of the remaining chapters

- Chapter 2: System Overview sets out the purpose of the system and uses diagrams such as the system overview and an entity-relationship diagram to describe the structure of the system.
- Chapter 3: Description of Work Done describes the steps taken to complete the project and the problems that were encountered. This chapter goes on to examine the methodology, design, development and testing stages of the project.
- Chapter 4: Conclusions and Recommendations outlines the conclusions that were established upon completion of the project as well as some recommendations which were noted during the project and which should provide additional benefit to the business for the future.
2. **SYSTEM OVERVIEW**

This chapter will provide an overview of the system. It will discuss the objectives of the new system, including the limitations of the current system and the objectives of the system to be developed. It will also examine the technical environment required for the new system from two perspectives: system requirements and user requirements.

2.1 **Objectives**

Frame Line has requested two main features of this project: an invoicing and reporting system and a training program. The following section will discuss both of the aspects separately.

**Summary of Current Invoicing and Reporting Process**

Frame Line currently generates invoices manually at the end of each month. This process is costly to the company and can take up to two days to complete. The orders arrive at Shankill photo centre’s main computer and are moved to the COMPLETED folder once the products are completed. When the orders arrive on the main computer, an order form is automatically generated detailing the order. These order forms are used to generate the invoices. At the end of every month, all of the order forms are collected and manually inputted into MS Word. The details of the invoice such as store name, products ordered, price and total value are manually typed into MS Word. Once the invoices are generated manually, the order folders are moved to the ARCHIVED folder. During the Christmas season, Frame Line could receive thousands of orders, which all are manually invoiced.

Currently Frame Line cannot produce management reports such as sales per product and sales per store on their orders.

**Summary of Current Training Process**

Frame Line do not currently have a training program on site for new and current staff. When a new member of staff joins Frame Line, they must learn how to produce up to 16 different products by observation. Currently a new member of staff is shown how to create the gifts by a current member of staff. Once they have been shown, the new member must practice what they have been shown. Once they are sure they can produce the gifts to Frame Line’ standards, they may complete a product that has been ordered. If a member of staff forgets how to make a gift, they resort to consulting a website [4]. At this website the staff can find instructions on how to produce each individual gift. Permission to use content from the website can be found in Appendix C.7, p. C.12.

**Limitations of Current Invoicing and Reporting System**

The current invoicing system is time-consuming and costly to Frame Line. A staff member must be assigned to invoicing customers at the end of every month. Because this process could take up to two days, this is costing Frame Line unnecessarily. The member of staff has to be paid to generate these invoices when they should be producing products that are being ordered.
This system is also very vulnerable to human error. When manually creating hundreds of invoices, the person performing the task may find the job tedious and so be prone to making mistakes. This could lead to further problems such as an incorrect invoice being sent, or an invoice being sent to the wrong person. This could be costly to Frame Line and this issue needs to be eliminated.

Limitations of Current Training Program
The current training program causes significant stress for employees. A new employee must be adaptable and a quick learner if they are to succeed working for Frame Line. The lack of support for employees exacerbates the problem. The employees do not have an easily accessible source of information. The information needs to be accessed quickly as staff do not have time to waste when producing merchandise for customers.

Objectives of the New System
The purpose of the new system is to eliminate the issues discussed above. This includes automating the invoicing process, producing a functionality to generate reports, capture data for invoices more accurately and provide training support for new and current staff. This system will increase the efficiency of the staff for two processes: invoicing customers and training staff. The advantages of an automatic system over the current system are as follows:

- One-click generation of accurate invoices and reports
- Less manual input needed for invoicing resulting in a less time consuming process and reduced human error
- Quick access to product instructions and information for staff
- Links to the internet for further information on products
- Ease of navigation between training program, invoices, reports and importing data
- User-friendly interface for staff that may not have a high level of computer skills
- Reduction in duplicate data and also the use of paper
- Improved use of data provided to Frame Line by Fujifilm Ireland
- Importation of data is automated and tracked to ensure data is not duplicated

2.2 Technical Environment

This section will discuss the technical layout of the new system. The system is made up of a set of RAW tables, LIVE tables, queries, reports and a number of forms which make up the user interface. The system is a fully implemented Microsoft Access database which is located on the client’s computers in Shankill photo centre. The system may be accessed by all employees. The client was willing to consider the use of an off-the-shelf package, but after further study into the suitability for this project, as discussed in Chapter 3.2, a custom built system was selected. MS Access 2007 was selected as the platform for this project.
System Requirements
The client currently has MS Access 2007 on the computers in Shankill photo centre. This means that the client did not need to invest any money in order for this system to be developed. The system currently occupies under 20MB of disk space, including some test data. When the system starts to import orders directly from the COMPLETED folder, the size will grow. It will be up to the client how he wants to maintain the size of the system. This may require the deletion of records after a number of years of the system being in use. The system also requires that a scheduled task be set in order to automatically import the data from the order folders. A scheduled task is a common feature on computers since Windows 95 Plus! became available in 1995 [5]. This means that the client must have an operating Windows system that is later than this package.

User Requirements
The current system is of file type .accdb – this file type is compatible with MS Access 2007. This file type is not compatible with MS Access packages before 2007 and so this version is required. Although MS Access is not available on Macintosh or Linux operating systems, there are Open Source applications which will open this file if necessary. The client uses Windows XP operating system for all their computers in Shankill photo centre so this will not be a problem for this project.

2.3 System Overview Diagram

A system overview diagram can be seen in figure 1. This diagram outlines how a user can navigate through the system. The homepage is indicated by the red icon at the left-hand side of the image. From here, the options and forms are mapped out. A data flow diagram may also be found in Appendix E.1, p. E.1. This diagram shows the flow of data from an order being placed either online or on a kiosk. The diagram then shows how the data is passed through and invoices and reports are generated at the end.
Fig. 1 System Overview Diagram
3. DESCRIPTION OF WORK DONE

This chapter will discuss the work done to develop the system. It will start by examining the methodologies used and then the analysis stage will be described. Following from that the design and development stages will be outlined. Finally a description of the testing performed on the system will be explained.

3.1 Methodology

This project required a lot of communication with the client. The input of Frame Line was crucial for establishing the background of the project, defining the meaning of data variables as well as the production of reports and invoices. Interviews were used to determine the background of Frame Line and to specify the requirements. Communication with the client was also kept through email and over the phone. Once the requirements were agreed, the next stage was to choose a software package to complete the project. After selecting a means of developing the system, the data was studied in great detail. Once the data had been formatted to the correct design, the system could then be developed, using the waterfall model. This type of methodology allowed for careful consideration of the data and applications of the system, as well as input from the client on an on-going basis.

3.2 Analysis

Assessing the current system

The first step taken was to meet with the client in Shankill Photo Centre and understand how Frame Line's processes work. This involved the client giving a description of where Frame Line comes in Fujifilm's organisational hierarchy. The client demonstrated how an order arrives on the main computer and the order form is generated. Upon receiving an order, the client proceeded to explain how the invoices were generated, manually at the end of each month. The client explained that the process was time consuming and costly to the business. The client then expressed a wish to develop an automated invoicing system. Issues such as reporting and the training program were also stated at this stage.

Once this stage of the analysis was completed, a document outlining the client's background and the problems associated with it was drawn up. This document was completed on 25th November 2010 and was presented to the client. The client assessed the document and made some minor changes to the document. The final document can be found in Appendix C.2, p.C.2. Once this initial analysis was concluded, the data was then looked at in detail. This research is summarised below.

- Generating invoices manually was time consuming and costly
- Head Office required specific reports on the number of products sold per store and the number of orders of each product
- Necessity for a training program for staff
- Client records are not stored or used for forecasting, marketing etc.
- Trend reports cannot be generated on the current system
Software Selection
The client was willing to consider using either software already being used within Frame Line or to purchase an off-the-shelf package if it was required. In order to receive the greatest benefit from the software, a suitability analysis was prepared. This analysis compared some popular off-the-shelf packages against a proposed custom built system.

Each system was given a rating from 1 to 4 to determine firstly, the expense and secondly some characteristics that are necessary for the completion of this assignment. A lower value meant that the software package was well suited to this project. The characteristics that were used to determine the suitability are as follows:
- Ease of use
- Maintainability
- Customisation
- Ability to integrate a CRM (Customer Relationship Management application)
- Ability to integrate a training program for staff
- Ability to archive files after use

The client currently has Microsoft Office 2007 on the system. For this reason, MS Access 2007 was chosen as the package that would be used for a custom built system, if it proved to be the most suitable option in the analysis. Each package was a given a rating for each factor of the analysis. Internet comparison websites such as toptenreviews.com [6], were used as the reference point for this stage. The ratings were then added up for each software package. The lowest total would be associated with the most suitable software to use for this system. The analysis resulted in a custom built system being selected. A custom built system proved to be a lot more flexible when it came to integrating features such a training program and archiving data. It was also the most affordable, since the client would not have to purchase a new software package. The details of this analysis can be found in Appendix C.3, p.C.7. Other factors such as technical support and security issues were also taken into account.

Data Analysis
Once the requirements were specified and agreed, and the software to build the system was chosen, the next step involved close examination of the data. Once an order is generated, the main computer at Shankill Photo Centre receives a computer folder containing three files. These files contain all the information about an individual order. Because there are two ways in which Frame Line receive orders (online and kiosk), there are also two types of folder that they receive. Both types of folder are named with the JOBID of the order. A kiosk order folder is indicated by the format 1125611014JOB17320, whereas an online order folder is indicated by the format 1125600000JOB~IREAAA021610. This means that an online order is indicated by the characters “~IREAAA0” inside the folder name. These folders are kept in a folder called PROCESSING. Once the orders are completed by the shop staff, the files then move to the COMPLETED folder. This folder is where the data is collected for
the database. Once the orders are invoiced by the system that was developed for this project, the client requires the folders to be moved to a folder named ARCHIVED.

Inside a kiosk order folder there are a textfile (Info.txt), and two XML files (InfoXML.xml and Summary.xml). A textfile is a computer file that contains letters and numbers. An XML file is an abbreviation of “extensible mark-up language”. Inside an online order folder there is a textfile also (Info.txt) and one XML file (InfoXML.xml). It was agreed with the client to use the InfoXML.xml file to extract the data that was needed for the database, as it was common to both kiosk and online orders.

On further study of these files, it was discovered that there were two different InfoXML.xml files, with different formats. One of the formats corresponded to kiosk orders and the other to online orders. Because the format between the two types of orders differed, this would cause issues when trying to import the data into Microsoft Access. An example of the variables in a kiosk InfoXML.xml file can be found in Appendix C.4, p. C.9, while an online InfoXML.xml example file can be found in Appendix C.5, p. C.10. The client was then asked to choose which variables he required to be in the database. This was done because the XML files have over 270 variables corresponding to each order. In order to determine which variables were required for the database, an Excel spreadsheet was sent to the client with all of the variables listed on it, separating an online order from a kiosk order. The client returned the spreadsheet with the required variables indicated. This resulted in approximately 70 variables being required for the database. The variables required can be seen in red in Appendix C.4 and C.5, p. C.9 and C.10 respectively.

When this document was received, the variables that were indicated to be required were looked at in detail. Each variable was given a meaning based on the values that were entered into them. Once this task was completed, it was discovered that JOBID was the only variable that uniquely identified an order.

Importing the relevant data into Microsoft Access was then evaluated. The format of the XML files meant that the data automatically set up five tables in the database: Data, Job, Picture, Product and Record. Each table had a set of variables associated with them. A piece of code was developed to extract the data from each InfoXML.xml file in each of the order folders. There were some issues with the format of the XML files that corrupted the automatic import. These findings are outlined below.

Multiple Tables and Records in the XML File
The online XML files include a second PRODUCT table. The second table has all of the same variables as the first, but the values that are entered are different. In the second PRODUCT table, the variables PROD_NO and PROD_NO8 both have a value of “DELIVERY”. This means that when the online InfoXML.xml files are imported to MS Access, the first PRODUCT table of the XML file imports as one record and the second table imports as a separate record. For this table only, it was found that the second record was not necessary. The values that were entered were either not required or they could be found in the other tables.
The next finding showed that the format of these tables could not be changed. This was discovered by trial and error. The data was imported a number of times into MS Access. The first trial involved removing a table altogether. When the import was run, the file would not import and produced an error. The second trial involved removing fields from a table and again importing the data. Once again, the file would not import and generated an error. The third attempt was to delete records from the tables. This was successful and did not produce an error. The result of this trial and error meant that if other files were to be imported, the format that MS Access had set for the data had to remain the same. If the format of the tables was amended (i.e. a variable was removed from a table, or a table was removed), an error would appear and the data would not import. The second conclusion found that individual records could be deleted from tables without causing trouble to further imports.

A similar difficulty occurred in the RECORD table. In both online and kiosk orders, there were multiple RECORD tables in the XML file. This resulted in multiple records being imported for one order. This meant that there were duplicate values and unnecessary variables.

As an attempt to overcome this, a new set of tables were built. For the sake of explanation, the original tables that the data imports to, will be called the RAW tables and the custom built copy shall be called the LIVE tables. The LIVE tables were formatted to only include the variables that were required. This would make the tables easier to read and eliminate the extra records that were described above. The tables used the same variable names and table names to ensure that the data moved to the correct place. SQL was used to move the data. An example of the SQL code can be seen in Appendix G.1, p. G.2. Copying the tables to the LIVE set was successful. These tables could be formatted as required and unnecessary variables and records could be eliminated.

**Picture and Product Tables are Unlinked**

The most problematic aspect of importing the XML files lay in the PICTURE and PRODUCT tables. It was identified that the only unique identifier for an order is the variable JOBID. For both an online and a kiosk order, the XML files do not include this variable in the PICTURE and PRODUCT tables. This means that MS Access cannot link the records in these tables with each other, or with the other tables. The result is that there is no way of knowing the JOBID that a record in the PICTURE or PRODUCT tables corresponds to. Without this variable in these tables, they information is useless. The variable JOBID must be in every table because it is the only variable that uniquely identifies an order. The next step involved amending the format of the tables so that JOBID would be present in all tables.

When these tables were looked at in more detail, it was decided that the PRODUCT table was not required. All the variables that were needed could be found in the three remaining tables: JOB, PICTURE and RECORD. After discovering this, the dilemma remained that the PICTURE table had no link to the other two.
On studying the PICTURE table, it was then noted that the PICTURE table included a field called FILENAME. This variable holds the name of the file of the picture that needed to be used when making the product. For an online order, the value of this field was “img_0.jpg”. For kiosk orders however, the filename included the JOBID. There is a possibility that this variable could be used to link this table to the others, but this would only work for kiosk orders.

The next step taken was to investigate as to whether the format of InfoXML.xml could be changed. The issue was explained to the client. The client replied that the format could not be changed. The format of this file was programmed by the Australian company Whitech and a change would require Fujifilm to amend their whole system, which spans over all of Ireland. Since changing the format of InfoXML.xml was not an option, the other files within an order folder were examined to see if they could be used in exchange for InfoXML.xml. Although the second XML file, Summary.xml only existed for kiosk orders, it was investigated as an option. This file had a similar issue to InfoXML.xml. When this file was imported, two tables were built in MS Access: PRINT and SUMMARY. In this case the table PRINT did not have JOBID as a variable. It also didn’t have any other unique identifier and so it could not be connected to the other table. This meant that this file was not useful.

The next option was to use try use the LIVE tables to connect the data together. A new variable called PIC_ID, set to be an auto number was formatted into tbl_LIVE_PICTURE. Tbl_LIVE_JOB was also given an auto number variable called J_ID. The idea was that the auto numbers would correspond to each other and link tbl_LIVE_PICTURE and tbl_LIVE_JOB together. After attempting this, it was revealed that an order may have multiple products ordered in it. This means that there may be multiple records in tbl_LIVE_PICTURE that correspond to one record in the tbl_LIVE_JOB. This technique could therefore not be used.

The forth option was to look at the .txt file, Info.txt. This file existed for both kiosk and online orders. Upon analysing this file, it was found that there was a standard format for the majority of the files. The first issue with this file was that some orders strayed from this format by including extra spaces, which meant that this file could not be relied upon. This was found by manually importing Info.txt files into Excel. Some files skipped an extra line which would cause problems if this file was used for the database. This format can be seen in Appendix C.6, p. C.11. This format was common for the majority of files, but could not be used as a data source for the database.

The second issue which had to be addressed with this format was that it did not follow a tab delimited textfile. A tab delimited textfile is a file where the values are separated by tabs only. This means that when a textfile is imported, MS Access reads the text until it reads a tab and sets this value to be entered into the first field. It then reads the next set of text until it hits another tab and enters this value into the second field. It continues this until there is no text left in the file. The issue with the textfile in the order folders is that the values are separated by tabs and also by new lines. Looking at Appendix C.6, p. C.11, the first value lies on the first line of the textfile. However, the second value is not separated by a tab;
rather it is separated by a new line. As you move down the textfile, some values are separated by a tab as well. Because this file is not explicitly tab delimited, or separated by tabs only, it makes it very difficult for MS Access to read. This renders the Info.txt file unusable in this case also.

The fifth and final option that was attempted to link the PICTURE table to the JOB and RECORD table was to write VBA code to read the InfoXML.xml file. The code used can be found in Appendix G.1, p. G.1. This code imports the orders one at a time. It opens each order into temporary tables (JOB, PICTURE, PRODUCT and RECORD). It then reads the JOBID and puts it into the PICTURE table. Then the data is moved to the corresponding LIVE tables (tbl_LIVE_JOB, tbl_LIVE_PICTURE, tbl_LIVE_CUSTOMER and tbl_LIVE_RECORD). tbl_LIVE_CUSTOMERS was created in order to separate the customer details from store details. When the data is moved to the LIVE tables, the record is then deleted from the temporary tables.

This solution allows for the details in the PICTURE table to be connected to the rest of the data. It avoids importing duplicate and also irrelevant data. Once the data was formatted to the design specifications, the design of the system could then be established.

### 3.3 Design

The design of this system can be divided into two categories. The first category will be the design of the database. This design was built for ease of use and simplicity. The second category will be the design of the user interface. The interface was designed so that users did not need any technical background in order to use this system. Both categories are discussed below.

**Database Design**

This system was designed following the Structured Systems Analysis and Design Method (SSADM). This method was built for use by the Office of Government Commerce, formerly known as the Central Communications and Telecommunications Agency in the 1980’s [7]. This method includes a number of steps such as feasibility study, investigation of current environment, business system options, requirements and specifications, technical system options, logical design and physical design. The entity relationship diagram can be found in Appendix E.4, p. E.3. This method is suited to the waterfall model for system design which can be found in Appendix E.2, p. E.2.

After overcoming issues with the data, this system aimed to provide a simple database design. As has been mentioned in Chapter 1, it was discovered that two sets of tables were required. One set of tables, known as the RAW tables, are required for the importation of the data. Their format and names cannot be changed and they are only used temporarily, for importing. These tables have not been amended from the format that MS Access produced. The second set of tables, known as the LIVE tables were designed to avoid duplicate data and for ease of understanding. Each table contains only the variables that are required by the client. Appendix E.3, p. E.2, shows a list of the LIVE tables and a brief description of
what they contain. There are 7 LIVE tables. Four of these tables transfer data directly from the RAW tables (tbl_LIVE_JOB, tbl_LIVE_PICTURE, tbl_LIVE_RECORD and tbl_LIVE_CUSTOMER). One table, tbl_LIVE_PRODUCT was designed to keep track of the products that Frame Line produce. The remaining two tables were designed to keep track of the files being imported and also any duplicates that were being imported also. These two tables were designed to aid the client in tracing what files are being imported to the system and when.

The tables also follow a commonly used naming convention which can be found in Appendix E.5, p. E.3. The naming convention is based upon Leszynski/Reddick naming convention [8]. The names of the RAW tables could not follow this naming convention as their format cannot be changed.

**User Interface Design**

The forms were designed with the aim of making the system easy to use. This will allow any user, whether they are computer literate or computer illiterate to work through the system easily and efficiently. The forms are designed to be consistent with each other. Each form is laid out with the Fujifilm Logo and title to describe what the form does along the top and the operations in the main body. The forms are navigated through using the navigation pane at the base of the form or using the buttons that are present. The forms also adhere to naming conventions which can be found in Appendix E.5, p. E.3. The forms have been designed so that any error that might occur when entering data will be identified before any action is performed.

The homepage allows the user to navigate to any part of the system. It also remains open throughout the time the system is in use. This allows the user to easily and quickly access invoices, reports and training program all at the same time if needs be. Once MS Access is opened, this form is automatically opened. This avoids the user getting confused by the objects that are listed and ensures that the user can easily find what they are looking for.

### 3.4 Database Development

The way in which the database was developed can be seen in the entity-relationship diagram found in Appendix E.4, p. E.3. From this image, it is clear that tbl_LIVE_JOB is the main table. This table connects to the majority of the other LIVE tables. Each table has a primary key which uniquely identifies a record within that table. If tbl_LIVE_PICTURE is taken as an example, there may be many products ordered by a store and so there may be many records with the same JOBID in this table. The primary key, PIC_ID allows each record to have its own unique key.

The raw tables are not part of the entity-relationship diagram. The reason for this is that these tables are only used temporarily. When a file is imported into the database, it first arrives in these RAW tables. The order is then amended so that JOBID occurs in each table in the database. Once this is performed, the order is transferred to the LIVE tables and deleted from the RAW tables. Then the next order can be imported. The system was set up
like this to ensure that all the tables in the database were connected. As was mentioned in the Data Analysis section in Chapter 1, there were issues when importing the data at first. This method of dealing with an order one at a time resolves this issue and allows the data to be formatted correctly.

The client did not feel that user level security was necessary for this system. To avoid changes being made to the source code of this system, a unique password has been assigned to the Visual Basic Application of this system. Changing the password is described in the User Manual that accompanies this report in Appendix D, p. D.1.

3.5 Application Development

This section will describe the development of the applications in detail. It will discuss how the data is imported and how each form connects to another and also to queries and reports where necessary. There are 9 forms in total, including the Training Program. This system relies heavily upon Visual Basic (VBA) coding for functionality. Screen shots of these forms and reports may be found in Appendix F, p. F.1.

Frame Line Home Page

The homepage offers the user four options, showing four command buttons. The user has the choice at this stage to go to:

1. Training Program
2. Invoices
3. Reports
4. Import Data

Each button operates an event procedure or an embedded macro. An event procedure means that once the button is clicked, the system will run a section of code that has been specified. An embedded macro is a tool which MS Access offers, to perform tasks such as opening the next form or report. The system overview diagram in Chapter 2.3 shows how the system networks. Each form has a command button that will either take the user back to the previous form, or to the homepage. The next few sections will discuss each of these four options.

1. Training Program

Upon clicking the Training Program button, another form appears called frm_Training_Options. This section of the system is for the use of staff to learn how to produce a certain product. From this page, the user may select to learn about a product, the change a product’s details, to delete a product or to add a new product to the list. An option is also provided to return to the homepage. The first and second options open another form that requests the user to select a product from a drop-down list. This list shows the product code and description. They may not choose anything other than what is provided by the list. If no product has been chosen from the drop-down list, then the system will prompt the user to select a product before hitting a command button. Upon selecting a product, and clicking
the command button, a form then appears with the information about the product they selected.

If the user clicks the command “Learn About This Product”, then the form opens in read-only format. This means that the user cannot change any details on the form. If, however the user clicks the command “Change This Product”, then the form opens in edit format, allowing changes to be made to the details. On this form, there is the option to close the form without saving the changes that have been made. This allows a user that might have mistakenly clicked “Change This Product” to undo their mistake.

The other options on the Training Options form allow the user to delete or add a record. Upon clicking the delete command, the user is taken to the first product. The user can then click the command button to delete the product. A prompt then appears to check if the user wishes to continue with the deletion. If the user selects “No” on this prompt, then the product will remain in the database. The user may also navigate through the products using the navigation pane at the bottom of the form.

The final command on the Training Options form allows the user to add a product to the list. This opens a blank form for the user to enter details into. Again on this form there is the option to close the form without saving the new product. When filling in the details of this form, the user must enter details into the “Product Code” and “Product Description” boxes. If these boxes are not filled and the user tries to save the new product, a prompt will request for these fields to be entered.

2. Invoices
The client requested that invoices may be generated at the click of a button. He also asked that the invoices may be generated between two dates that the user selects. The “Invoices” command button does exactly this. When the user selects “Invoices”, a form appears requesting two dates to be entered, a start date and an end date. These dates correspond to the date that the user wishes to invoice from and to. The dates should be in the format dd/mm/yyyy. An input mask stops the user from entering anything other than a date value into the system. If no dates are selected, and the “Generate Invoices” command is clicked, a prompt will request the user to enter a start date. The end date will automatically enter today’s date, if no end date is selected. When “Generate Invoices” is clicked, the following procedures take place:

1. Check Start Date is before End Date
2. Opens qry_INVOICES filtered between the two dates
3. Opens rpt_INVOICES based upon qry_INVOICES
4. Inserts “Yes” into INVOICED field in tbl_LIVE_JOB table
5. Moves the relevant files to ARCHIVED folder
6. Opens rpt_Jobs_Not_Invoiced

Once the two dates are selected, and the “Generate Invoices” command is clicked, the system will check that the Start Date is before the End date. If this is correct, then the
system will generate a query called qry_INVOICES which will filter the relevant data needed for the report between the two dates entered. Then the system will open a report called rpt_INVOICES which shows all of the invoices between the two dates entered. The invoices are grouped by the store name that the order came from. Each invoice starts on a new page. This report opens in print preview mode. This means that not all of the invoices can be seen at once. These invoices can then be printed by clicking the print icon at the top left corner of the screen as seen in the screen shot in Appendix D, p. D.11.

Once the invoices have been generated, the system is required to keep track of which jobs have been invoiced and which have not. There is a field called INVOICED in tbl_LIVE_JOB which indicates whether a job has been invoiced or not. When the “Generate Invoices” command is clicked, the system updates this field to “Yes” only for the jobs that lie between the two dates entered. After these jobs have been marked as INVOICED, the system then moves these folders from the COMPLETED folder to the ARCHIVED folder. This means that once an invoice is generated, only then does the file move to the ARCHIVED folder. The system will then run another query called qry_Jobs_Not_Invoiced which only shows the jobs that are yet to be invoiced, and opens a report based on this. This report, frm_Jobs_Not_Invoiced_Yet shows the user what jobs have not yet been invoiced in case a mistake has been made.

3. Reports
Like the invoices, the client specifically asked that two reports be produced: sales per product and sales per store. Head Office requires that Frame Line produce these reports monthly. When the user clicks the command “Reports” on the homepage, another form opens. This form requests that the user select a report from a drop down box. Once the report is selected, then the user may enter a date range for that report. If a report is selected that does not need a date range, the text boxes are disabled. If the user does not select anything and hits the command “Generate Report”, then the system prompts the user to select a report. If the user selects a report but does not enter dates that are required, then the system again will prompt the user to enter the dates. Once the relevant fields are entered and the user hits the command “Generate Report”, a query relating to the report selected is run, selecting only the relevant data required for the report. Then the report is opened in print preview mode. Again the user may print the report by clicking on the print icon at the top left corner of the screen. The user may study the records that were selected after the inputs were processed by looking at the query that corresponds to the report generated.

4. Import Data
Initially, the client did not indicate whether the data should be imported into the system automatically or manually. On further questioning, the client was impartial to either option. As a result of this, the system was designed to incorporate both features. Currently, the system has been set to run a custom built import function using a scheduled task on the main computer. This means that this set of code will run at a set time every day and the client does not need to import orders manually, unless there is an urgent task that needs to be executed. For example, the system has been built so that a custom built function (block of code) called ImportAndChange() is run at 5pm every day. This code, checks the
COMPLETED folder for any new order folders. If it finds any new orders, it checks if it is in the database already. If it is not currently in the database, it imports the data temporarily into the RAW tables, formats the record and moves it to the LIVE tables. Then the code inserts a record into tbl_LIVE_IMPORTS including the JOBID, the path that the folder was imported from, the folder name and the date and time it was imported. If the record was already in the database, the system will not import the order. It will enter a record into tbl_LIVE_DUPLICATES, indicating when the file was being duplicated. The system does this automatically every day according to the scheduled task.

The system has an option built in to import the data by hand. The client may do so by clicking the command “Import Data” on the homepage and importing files one by one. Once this command is clicked, a dialog box appears requesting the user to select the folder he wishes to import. Once the “Open” command is clicked on the file dialog box, the code checks if this file has already been imported into the database. If the job already exists in the database, then the system does not import it again. Instead, a record is inserted into tbl_LIVE_DUPLICATES to indicate which was being duplicated and when. If the file does not exist in the database, then the file is imported as it would have been, were it selected automatically.

3.6 Testing

Throughout the development stage, the code was constantly being checked for logic, syntax errors and bugs. The system was also being tested for navigation and accuracy. The whole system was tested by inputting correct data as well as incorrect data, for ease of use and for logic. A test procedure that was performed can be found in Appendix H.2, p. H.3. When the system was being implemented, a number of changes and additions were made. These are outlined in the implementation document in Appendix H.1, p. H.1, along with some findings that were noted.
4. CONCLUSIONS AND RECOMMENDATIONS

This project had clear objectives which have been met to the client’s satisfaction. The first section in this chapter outlines the conclusions that were made upon completion of the project. Following on from this, section 4.2 will discuss recommendations that should improve the system further for Frame Line.

4.1 Conclusions

This aim of the project was to develop a system that automates the generation of invoices and reports for Frame Line. The following achievements were made by the system:

- Automatic generation of individual invoices
- Automatic generation of management reports, such as sales per product and sales per store
- Automatic importation of data
- A standardised training program for staff
- An easy to use interface to navigate between the training program, invoices and reports
- Tables dedicated to customer and store details
- A database that holds all details on orders made to Frame Line

The automation of invoicing and reporting reduces the time spent and cost of manually generating these documents by Frame Line. The reports that are produced enable Frame Line to track their sales and perform trend analysis on the data. Because the data is imported automatically, Frame Line will save the time currently spent in gathering the information they need. All the information they require is included in the database.

The database will provide Frame Line with information about their customers and stores. This information can be used for marketing techniques such as customising promotions to suit customers’ purchasing habits. Frame Line will be able to contact their clients’ directly using the information provided.

A standardised training program ensures that all of Frame Line’s employees have a resource they can rely on for reliable and up to date information on products. This also allows new employees to quickly adapt to the production process that occurs at Frame Line by providing the information they need at the click of a button. The user interface will facilitate all users to access relevant data with ease.

A number of additional recommendations were identified during the course of this project, which are outlined in section 4.2. These recommendations if implemented, will aid Frame Line by improving the operation of the processes currently in place.
4.2 **Recommendations**

The first recommendation would be to reassess how stores are numbered. As was noted in the implementation document in Appendix H.1, p. H.1, there are two indexes for identifying a store. These indexes do not correspond with each other and cause difficulty when trying to differentiate the stores. A similar situation occurs when looking at the description of products. Products with the same product code have a variety of descriptions associated with them. Reassessing how stores and products are defined will eliminate confusion and mistakes that may be made when inputting data and generating invoices. This will require setting a standard with Fujifilm and is advised.

The second recommendation is the issue of security with the new system. Although Frame Line employs fewer than 20 staff, currently they all have access to the data and reports that the system generates. It is advised that Frame Line assess which staff should have access to which information. The client should enhance the system by introducing restricted access to sensitive data in order to avoid the risk of data being changed, stolen or manipulated inappropriately. Frame Line should also ensure that only personnel with experience in Visual Basic coding should be allowed to access the VBA code and that the password should be kept for maintenance purposes only.

The third and final recommendation to Frame Line would be to import the data that is held on the external hard drive at Shankill photo centre, to the database. By importing this data, Frame Line will be able to perform trend analysis and assess any sales patterns that might have occurred. This will assist the client in predicting sales figures which may be useful at peak periods for example the Christmas season, when sales increase dramatically.
FRAME LINE LTD.
Development of an Invoicing and Reporting System

APPENDICES
## APPENDICES

<table>
<thead>
<tr>
<th>NO.</th>
<th>CONTENT</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.</td>
<td>Original Project Outline</td>
<td></td>
</tr>
<tr>
<td>B.</td>
<td>Interim Report</td>
<td></td>
</tr>
<tr>
<td>C.</td>
<td>Technical Commentary</td>
<td></td>
</tr>
<tr>
<td>D.</td>
<td>User Manual</td>
<td></td>
</tr>
<tr>
<td>E.</td>
<td>Design Documentation</td>
<td></td>
</tr>
<tr>
<td>F.</td>
<td>Sample Reports and Screen shots</td>
<td></td>
</tr>
<tr>
<td>G.</td>
<td>Source Code</td>
<td></td>
</tr>
<tr>
<td>H.</td>
<td>Implementation and Test Documentation</td>
<td></td>
</tr>
</tbody>
</table>

REFERENCES
A. ORIGINAL PROJECT OUTLINE

Project No. 5

Management Science and Information Systems Studies

Client: Frameline Ltd. – Fujifilm Ireland
Project: Development of an invoicing and reporting system
Location: Shankill Photo Centre, Unit 2, Ashwood House, Mainstreet, Shankill, Co. Dublin.

Client Contact: Ian Mac Mahon

Client Background

Frameline Ltd. is a professional photography company that owns Fujifilm photography stores. We provide a large range of products and services including film and digital processing, photo restoration, poster printing and personalised gifts. The company also processes orders from www.fujipix.ie, Fujifilm's online website, and delivers products all over Ireland.

Orders made online are manually entered into an invoicing system, which is very laborious and can take up to 2 days each month to complete. Currently when an online order is made, an order sheet is printed out detailing the price, quantity, shop location and some other information. Each one of these invoices must be entered into the system manually at the end of the month, which generates an invoice.

Client Requirement

The current process takes up too much time and mistakes can happen easily. The accountant would like a new software package that generates the invoices without having any manual input. Every order that is made has an xml file automatically generated and stored in a folder. This file has all the data necessary for generating invoices. The system must read in these files from multiple folders and produce invoices. There must also be some functionality for reporting and data analysis of the orders (per client and per product at least). This system must be easy to use and fully automated.

What is involved for the student?

- Review of current invoicing system
- Design of new system for automated invoicing
- Installation of this system on the company computers
- Production of a user manual for the system
- Training the accountant to use the system
Management Science and Information Systems Studies

Project: Development of an invoicing and reporting system
Client: Ian Mac Mahon, Frame Line Ltd. (Fujifilm Ireland)
Student: Gillian Lane
Supervisor: Denise Leahy

Background and Work to Date
Frame Line Ltd. is a professional photography company that runs three photography stores, a professional photography service and an online processing facility linked to the three stores. Frame Line Ltd provides Fujifilm Ireland with a wholesale photo-gift processing facility. Fujifilm has an online ordering system which sends gift orders directly to a main computer at Shankill Photo Centre, one of three stores of Frame Line Ltd. All gift orders that Frame Line Ltd. produce for other Fujifilm stores are invoiced to Head Office, Fujifilm Ireland.

The client has two main problems. Firstly, at the end of each month Frame Line Ltd manually produces invoices for Fujifilm Ireland. This process takes approximately 2 working days to complete. This system is inefficient and costly. The second issue occurs when the client needs to process reports for Fujifilm Ireland. These reports are also produced manually at the end of each month. Frame Line Ltd. would like a fully automated system. This system is needed to improve efficiency when writing reports and producing invoices.

The first step taken was to meet the client and discuss the project in detail. This meeting was set up to ensure that any possible issues with the original project description were resolved and that possible additions could be made to the first outline. A background report, outlining the client background along with the project background was then drawn up. This report also includes specified requirements and suggested solutions to the problem. When it was complete, a second discussion took place with the client to ensure the report was meeting the requirements set out. The client was presented with the report and was offered to provide any feedback they felt necessary.

Terms of Reference

Primary
- To analyse the current system, including reporting and invoicing methods
- To create an automated system for reporting and invoicing
- To implement the system with Frame Line Ltd
- To prepare supporting documentation for the new system
- To train staff in the use of the new system

Secondary
- To create a customer database from previous orders
- To develop a training system for current and future staff

Further Work
Christmas Break:
- Produce Gantt chart for the remainder of the project
- Design the lifecycle model to use
- Research possible software packages to use (because of the special requirements, it is not expected that a standard system will meet the needs)

**Week 1-5:**
- Design the system (or configure the system if a standard system is available)
- Import data
- Develop the system using the chosen software package

**Week 5-7:**
- Test system
- Adjust system, ensuring it meets requirements
- Prepare supporting documentation

**Week 7-8:**
- Demonstration of the system and training of staff

**Week 9-10:**
- Prepare the Final Year Project Report

**Conclusions**
Frame Line Ltd spends two days a month preparing invoices and report. This is an inefficient process. The development of a fully automated system is needed to meet the client’s requirements. Once Frame Line Ltd has solved this problem, there is the possibility to extend the project further into Fujifilm Ireland as a whole. These options need to be looked at in more detail once the original project is complete. The additional requirements will also aid Frame Line Ltd to becoming a more efficient company as a whole, reducing time spent training staff and also improving client – customer relationships.
C. TECHNICAL COMMENTARY

C.1 Fujifilm Organisational Hierarchy
C.2 Requirements Document

**Fujifilm Ireland:**
Fujifilm are known as “the world’s largest photographic and imaging company” (Fujifilm.com). The company is renowned for innovative work in areas such as motion picture film, computer media and medical imaging. Their products include photo development, cameras, camera accessories and business systems. Fuji have 169 stores in Ireland, operated by individuals and multiple chains, situated in 24 of the 32 counties of Ireland. Each location produces photos and sells Fujifilm products.

Fujifilm have approximately 1500 kiosks situated around Ireland, located in chain stores such as Harvey Norman, McCabes Pharmacy Group and Health Express Group and other owner operated stores. These kiosks act as an automated sale attendant. The consumer can order Fujifilm prints and gifts directly from these free-standing units in stores. The kiosks use software that was developed by an Australian company called Whiteck. There software is used for the up loader and the downloader that Fujifilm Ireland uses to operate the kiosks. These kiosks currently offer 16 products that Frame Line’s Shankill Photo Centre produces. Frame Line agrees the prices of these products with Fujifilm Ireland. Once the prices are agreed, they are inputted into all of the kiosks around Ireland. There are kiosks in the North of Ireland and so the current systems account for Sterling as well as Euro.

**Frame Line Ltd:**
Frame Line Ltd. is a professional photography company that runs three photography stores, a professional photography service and an online processing facility linked to the three stores. It is made up of Bray, Shankill and Dalkey Photo Centres, StudioLab and Print Digital 4 U. Bray, Shankill and Dalkey Photo Centres are stores in which Fujifilm products are sold. They produce a range of products from photographs, cameras and personalised gifts. Studio Lab is aimed at corporate customers. It provides a professional photography service (E.g. “Photographing brochure and product images for small-medium enterprise companies” : www.studiolab.ie). Print Digital 4 U is the website that allows full time access to both Studio Lab and Bray, Shankill and Dalkey Photo Centres.

Frame Line Ltd provides Fujifilm Ireland with a wholesale photo-gift processing facility. Fujifilm has an online ordering system which sends gift orders directly to a main computer at Shankill Photo Centre, one of the three stores of Frame Line Ltd. All gift orders that Frame Line Ltd. produce for other Fujifilm stores are invoiced to Head Office, Fujifilm Ireland. Shankill Photo Centre is a wholesaler for Fujifilm Ireland Ltd. Because of this, Shankill Photo Centre receives all gift orders from kiosks all over Ireland. The Fujifilm hierarchy can be seen in Fig. 1.
Frame Line’s Current System:

There are two categories of products that can be ordered from Fujifilm around Ireland: photos and gifts. Orders are generated in two ways: online and at a Fujifilm kiosk in store. Orders of photographs are printed locally in the shops around the country, gifts ordered in store are sent via a VPN (virtual private network) to Frame Line in Shankill Photo Centre.

- **Online orders**
  - Photos for Frame Line shops only
  - Gifts for all Fujifilm shops in Ireland
- **Kiosk orders**
  - Photos for Frame Line shops only
  - Gifts for all Fujifilm shops in Ireland

The following diagram in Fig. 2 describes the system as it is today at Shankill Photo Centre.

When an online or kiosk order comes through, Shankill Photo Centre’s main computer receives a computer file (i.e. folder) from Fujifilm Ireland. This file is automatically generated by the online system at Fujifilm Ireland and
contains details of the order that was placed. As soon as the order comes to Shankill Photo Centre, an order form is automatically printed by the main computer. It contains information such as price, quantity, shop location etc. The main computer is situated at Shankill Photo Centre. It is the only computer in Shankill Photo Centre that is connected to Fujifilm Ireland online ordering system. Staff can access this computer. The files are stored on this system for two months. After this time, the files are transferred to an external hard drive within Shankill Photo Centre, where they are saved for a further six months.

Once the orders are completed by the Shankill branch, the finished products are couriered out to the store in which they were ordered from. The address of the kiosk that was used to generate the order is printed on the order form. Shankill Photo Centre then uses the order form and the computer files to generate an invoice. All the invoices are typed up manually at the end of each month by a staff member. This process takes approximately 2 days of manual labour. Once the invoices are completed, they are sent to Fujifilm Head Office and a copy is saved by Frame Line. An order invoice includes the following details:

- Job Number
- Name of Customer
- Store ID
- Store Name
- Store Address
- Store Phone Number
- Date order made
- Date order received by Shankill Photo Centre
- Date order due
- Date order fulfilled (completed and sent)
- Product codes
- Product Details
- Quantity of Products
- Price per Product
- Total Price

Fujifilm Ireland currently has an SAP (Systems, Applications and Products) system in place. This type of system is used to integrate all fields of a business, resulting in one database containing all of the information. Frame Line invoice Fujifilm Ireland for all gift orders they receive from the kiosks around Ireland. When Fujifilm Ireland receives these invoices, they currently manually input them into their system. This currently takes up a lot of time and is inefficient.

Frame Line has an accounting system called TAS Books in place. This system is used to archive any data which is not in use. This system is also used to forecast stocks for the stores. In each of the three stores there is an EPOS (electronic point of sale) system. The EPOS systems in each of the stores do not interact with each other at the moment. This means that the three stores of Frame Line are not connected together. Their stocks are separate and there is no interaction between the three photo centres. Along with this, the EPOS and TAS Books systems do not link together either.

Frame Line promise to complete an order within 10 days of receiving it. The folder that they receive, when an online order is made includes a .txt file and an xml file. A .txt file, otherwise known as a textfile, is a computer file containing letters and number. XML stands for extensible mark-up language. Both files contain the same information. Currently Frame Line produces an order form using the information on these files.

**Requirements:**

**Aim:** To automate invoicing, cutting out manual processing of orders for Frame Line
Frame Line currently spends two working days generating invoices at the end of each month. The client believes that this process takes too long. Along with an invoicing functionality, Frame Line would like the system to be used to produce reports (outlined below) that they can also send on the Head Office. These reports that the client would like are as follows:

1. **Goods sold per store**
   - For Fujifilm Ireland Head Office
   - Breakdown of how many of each product was ordered/produced in a month at Shankill Photo Centre
     - Details to include:
       - Job Number
       - Store ID
       - Store Name
       - Store Address
       - Store Phone Number
       - Product Number
       - Product Details
       - Quantity of Products
     - Price per Product
     - Date order received by Shankill Photo Centre
     - Date order due
     - Date order fulfilled
     - GLS delivery number

2. **Sales per Product**
   - For Fujifilm Ireland Head Office
   - Search by Store Number
   - Search by Date range
     - Details to include:
       - Product Number
       - Product Details
       - Sum of Products Sold

Frame Line currently does not have any way of keeping track of the transaction history of new and existing customers. This includes names, addresses and previous orders. The client would like to be able to build and maintain a customer database. By doing this, Frame Line will be able to target loyal customers. They will also be able to conduct analysis to discover what types of customer are ordering their products. By doing this, Frame Line will be able to develop a loyal customer base while also targeting potentially profitable customers.

Finally Frame Line would like a training system developed. This would aid the company in the training of new staff and keeping a record of the products and processes that they use. It would cut down on costs as it would be easier and more efficient for a new staff member to learn how the company works through a customised application.

Frame Line would like this system to be easy to manage and maintain. They would like a user manual to be produced to aid this. The relevant staff will also need to be trained in the use of the system. Once the data has been on the system for two months, Frame Line Ltd requires the data to be moved to an external hard drive. This archived data must still be accessible.

Once the requirements are agreed upon, the next step will be to design a new system for Frame Line Ltd. The design of this system must take into account all of the client's requirements while also providing the best solution to the problem. The requirements are outlined below:

- Review of current invoicing system
Design of new system for automated invoicing
Design and implement a software package to do the following:
1. Generate Invoices
2. Generate Reports
3. Training Staff Programme

System must
- be fully automated
- be easy to use
- be easily maintained
- read in xml or txt file from multiple folders
- store the products and processes that Frame Line Ltd. use
- allow for new records to be added
- allow for specified records to be changed
  - Product records
  - Process records
  - Customer records
- protect against specified records being changed
  - Order records
- allow for specified records to be deleted
  - Product records
  - Process records
- protect against specified records being deleted
  - Customer records
  - Order records
- allow for archiving of out of date data (2 months old)
- allow archived data to remain accessible and searchable
- keep track of a customer database

Installation of this system on the company computers
Production of a user manual for the system
Training the accountant to use the system

Proposals for Design:
The following suggestions are being considered for the design of Frame Lines’ system:

Research current accounting packages available
- What is available
- How the packages match the requirements
- Analysis of potential costs to implement
- Consideration of the ease of use for the staff

Create a customised system using Microsoft Excel
- Custom built system to generate invoices and reports
- Fully automated system
- Require writing Visual Basic code
- Potential security risks
- Consideration of maintenance issues

Create a database system using Microsoft Access
- Create a database
- Import data from selected files
- Generate invoices automatically
- Generate reports
- Potentially include Visual Basic coding
- Potential security risks
- Consideration of maintenance
C.3 Software Selection Document

This section aims to compare accounting software packages that are available on the market today. Many of these packages include the functionality to produce invoices, but the question is whether to adopt a new package or to build a customised program? A number of software packages will be assessed on their suitability to use for this project. Custom building a program will also be used as an option in the comparison. In order to assess the suitability of software, the requirements need to be examined. The requirements that need to be specifically looked at are those that relate to the building of the system. The relevant requirements have been transferred into a table that will be used to assess the suitability of the software packages in question. Each requirement is equally important and so the resulting system must meet all of the needs. This table can be seen in figure 1.

There are a few important factors to take into account when deciding what type of software to use for this system. The most important factors are cost, ease of use, support/maintainability and customisation. Other factors would include a CRM functionality, ability to archive out of date data and ability to produce training program. The following rating system was developed in order to assess the products fairly. A rating of ‘easy’ to ‘difficult’ is chosen for each factor based upon research done. This rating then converts to a score between one and four. The scores are added up for each product and summed on the bottom line of the table.

The first package, TAS Books is aimed at small businesses. They provide software for accounting, payroll and customer relationship management purposes. They offer a free download, TAS Books Basic. This package provides no support when setting up or using the system. It also does not have any mechanism for customisation. The next level up for TAS Books cost $157, that’s the price of the support for this package. The next package QuickBooks is an Intuit product. Again these packages range from free to very expensive. The free download is easy to set up and easy to use. Again there is no support with the free package. You are able to customise invoices but producing a training program and CRM will prove more difficult. The third package, Peachtree is provided by Sage. Again the main issue with this package is that a customer has to pay for support. Other reviews explained that Peachtree was not as easy to use as other packages such as QuickBooks. These three packages were chosen for this comparison because they are the most popular and well known packages that small businesses use.

It is clear from the table that TAS Books is a good option for this system. It is also important to note that the client’s requirements all need to be met. The two major flaws this product has are archiving used data and developing a training program. If this software package was chosen to develop the system, the training program would need to be developed using a separate program. This is not an ideal option when aiming to meet all the client requirements. Archiving the data is also a very important aspect of this project. The client requires the data to be on file for a further 6-8 months after the orders have been made. It is for these reasons that a customised software program will be selected for the development of this system.
There is an issue of maintainability when a custom program is being built. Both the CRM functionality and the archiving will be difficult when developing a system from scratch, having said that it is not impossible to do. All of these aspects will be dealt with and the client requirements will be met.

Figure 1

<table>
<thead>
<tr>
<th>Factors to Consider</th>
<th>TAS Books Basic</th>
<th>QuickBooks Simple Start (Intuit)</th>
<th>Peachtree First (Sage)</th>
<th>Custom Built</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>Free</td>
<td>Free</td>
<td>Very Expensive</td>
<td>Free</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
</tr>
<tr>
<td>Maintainability</td>
<td>Difficult</td>
<td>Difficult</td>
<td>Hard</td>
<td>Difficult</td>
</tr>
<tr>
<td>Customisation</td>
<td>Hard</td>
<td>Possible</td>
<td>Difficult</td>
<td>Easy</td>
</tr>
<tr>
<td>CRM</td>
<td>Easy</td>
<td>Possible</td>
<td>Possible</td>
<td>Hard</td>
</tr>
<tr>
<td>Training Program</td>
<td>Difficult</td>
<td>Difficult</td>
<td>Difficult</td>
<td>Easy</td>
</tr>
<tr>
<td>Archiving</td>
<td>Hard</td>
<td>Hard</td>
<td>Hard</td>
<td>Hard</td>
</tr>
</tbody>
</table>

Scores: 17 17 21 14

Figure 2

<table>
<thead>
<tr>
<th>Score</th>
<th>Rating</th>
<th>Affordability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Easy</td>
<td>Free</td>
</tr>
<tr>
<td>2</td>
<td>Possible</td>
<td>Affordable</td>
</tr>
<tr>
<td>3</td>
<td>Hard</td>
<td>Expensive</td>
</tr>
<tr>
<td>4</td>
<td>Difficult</td>
<td>Very Expensive</td>
</tr>
</tbody>
</table>
Below is a list of variables that appear in the kiosk Info.XML file. The variables in red are the variables that the client required to be present in the database. There are 280 variables in these files, and the client requested 76 of them to be included in the database. Some of these variables are duplicated in the file and so were only imported once into the database.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APP_VERSION</td>
<td>PICKUPSTORE</td>
</tr>
<tr>
<td>STORE_NO</td>
<td>RETAIL_PRC</td>
</tr>
<tr>
<td>STORE_NAM1</td>
<td>ECONFIRMED</td>
</tr>
<tr>
<td>STORE_ADD1</td>
<td>SSIZE</td>
</tr>
<tr>
<td>STORE_ADD2</td>
<td>ADDRESS1</td>
</tr>
<tr>
<td>STORE_PCODE</td>
<td>CLR_DISC</td>
</tr>
<tr>
<td>STORE_STATE</td>
<td>GP</td>
</tr>
<tr>
<td>STORE_COUNTRY</td>
<td>DURATION</td>
</tr>
<tr>
<td>STORE_GUID</td>
<td>AUTHORISED</td>
</tr>
<tr>
<td>PHONE</td>
<td>SITE</td>
</tr>
<tr>
<td>MACHINE_NAME</td>
<td>AVG_COST</td>
</tr>
<tr>
<td>JOBID</td>
<td>STAFF_NO</td>
</tr>
<tr>
<td>IPRINTS</td>
<td>DISCRIPT</td>
</tr>
<tr>
<td>CUST_NO</td>
<td>CREAT_DATE</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>ROYALTY</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>UPC_CODE</td>
</tr>
<tr>
<td>PHONE_NO</td>
<td>PENALTY</td>
</tr>
<tr>
<td>CUSTOMMAIL</td>
<td>STAFF_NO</td>
</tr>
<tr>
<td>ADDRESS1</td>
<td>KIST.NO</td>
</tr>
<tr>
<td>ADDRESS2</td>
<td>SY.S_INFO</td>
</tr>
<tr>
<td>ADDRESS3</td>
<td>GROUP1</td>
</tr>
<tr>
<td>STATE</td>
<td>GROUP2</td>
</tr>
<tr>
<td>POSTCODE</td>
<td>GROUP3</td>
</tr>
<tr>
<td>PRICE</td>
<td>GROUP4</td>
</tr>
<tr>
<td>PRICE</td>
<td>GROUP5</td>
</tr>
<tr>
<td>REQUEST1</td>
<td>GROUP6</td>
</tr>
<tr>
<td>REQUEST2</td>
<td>GROUP7</td>
</tr>
<tr>
<td>REQUEST3</td>
<td>GROUP8</td>
</tr>
<tr>
<td>DDATE</td>
<td>GROUP9</td>
</tr>
<tr>
<td>TTIME</td>
<td>ITEMCOUNT</td>
</tr>
<tr>
<td>PICK_DATE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>PICK_TIME</td>
<td>ORDER_NO</td>
</tr>
<tr>
<td>STORE_ID</td>
<td>TAX_RATE</td>
</tr>
</tbody>
</table>

C.4 Kiosk Info.XML Layout
C.5  Online Info.XML Layout

<table>
<thead>
<tr>
<th>APP_VERSION</th>
<th>US_TAX</th>
<th>&lt;/PRODUCT&gt;</th>
<th>STATUS</th>
<th>AVG_COST</th>
<th>FAX_NO</th>
<th>NOTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>STORE_NO</td>
<td>TAX_RATE</td>
<td>- &lt;/PRODUCT&gt;</td>
<td>TOTAL</td>
<td>UNIT_NO</td>
<td>MOBILE_NO</td>
<td>VOUCHERS</td>
</tr>
<tr>
<td>STORE_NAME1</td>
<td>PST</td>
<td>PROD_NO</td>
<td>DEPOSITS</td>
<td>CATEGORY</td>
<td>EMAIL</td>
<td>TILL_NO</td>
</tr>
<tr>
<td>STORE_ADD1</td>
<td>SIMPLE_TAX</td>
<td>PROD_NO8</td>
<td>TAX</td>
<td>SUB_UNIT</td>
<td>ABN_ACN_NO</td>
<td>&lt;/RECORD&gt;</td>
</tr>
<tr>
<td>STORE_ADD2</td>
<td>EMB_ITEMS</td>
<td>DESCRIPT</td>
<td>UCDIE</td>
<td>PENALTY</td>
<td>NO_CUST_NO</td>
<td>&lt;/DEPOSITS&gt;</td>
</tr>
<tr>
<td>STORE_PCODE</td>
<td>EMB_PRICE</td>
<td>MAIN_CAT</td>
<td>JOB_TYPE</td>
<td>DISCOUNTBL</td>
<td>GST_EXEMPT</td>
<td>CDATA</td>
</tr>
<tr>
<td>STORE_STATE</td>
<td>EMB_CHARGE</td>
<td>SUB_CAT</td>
<td>CREDITS</td>
<td>DISCOUNT</td>
<td>PST_EXEMPT</td>
<td>&lt;/DATA&gt;</td>
</tr>
<tr>
<td>STORE_COUNTRY</td>
<td>GAR_ITEMS</td>
<td>QUANTITY</td>
<td>CRED_USED</td>
<td>STD_TAX</td>
<td>LAST_VISIT</td>
<td>&lt;/JOB&gt;</td>
</tr>
<tr>
<td>STORE_GUID</td>
<td>GAR_PRICE</td>
<td>NEW_UPRICE</td>
<td>UPC_CODE</td>
<td>NEG_NO</td>
<td>&lt;/RECORD&gt;</td>
<td></td>
</tr>
<tr>
<td>PHONE</td>
<td>GAR_CHARGE</td>
<td>TOTAL</td>
<td>ORDER_NO</td>
<td>COMMENTS</td>
<td>&lt;/CUSTOMER&gt;</td>
<td></td>
</tr>
<tr>
<td>EMAIL</td>
<td>PROD_NO</td>
<td>TAX_RATE</td>
<td>&lt;/RECORD&gt;</td>
<td>SERVICE</td>
<td>- &lt;/RECORD&gt;</td>
<td></td>
</tr>
<tr>
<td>USER_NAME</td>
<td>PROD_NO</td>
<td>GP</td>
<td>&lt;/PICKUP&gt;</td>
<td>SEQUENCE</td>
<td>INDATE</td>
<td></td>
</tr>
<tr>
<td>MACHINE_NAME</td>
<td>DESCRIPT</td>
<td>AVG_COST</td>
<td>- &lt;/JOBS&gt;</td>
<td>SYSINFO</td>
<td>TMIME</td>
<td></td>
</tr>
<tr>
<td>JOBID</td>
<td>QUANTITY</td>
<td>STD_TAX</td>
<td>&lt;/RECORD&gt;</td>
<td>&lt;/RECORD&gt;</td>
<td>REC_NO</td>
<td></td>
</tr>
<tr>
<td>IPRINTS</td>
<td>REPRINTQUANTITY</td>
<td>&lt;/PRODUCT&gt;</td>
<td>JOB_ID</td>
<td>&lt;/RECORD&gt;</td>
<td>BREC_NO</td>
<td></td>
</tr>
<tr>
<td>LOADED</td>
<td>NEW_UPRICE</td>
<td>- &lt;/DATA&gt;</td>
<td>CATEGORY</td>
<td>PROD_NO</td>
<td>CUST_NO</td>
<td></td>
</tr>
<tr>
<td>IUSED</td>
<td>FILENAME</td>
<td>- &lt;/PICKUP&gt;</td>
<td>CUST_NO</td>
<td>PROD_NO8</td>
<td>STAFF_NO</td>
<td></td>
</tr>
<tr>
<td>I2EENHANCED</td>
<td>PRINTWIDTH</td>
<td>- &lt;/RECORD&gt;</td>
<td>DESCRIPT</td>
<td>MAIN_CAT</td>
<td>MACHINE_NO</td>
<td></td>
</tr>
<tr>
<td>CUST_NO</td>
<td>PRINTHEIGHT</td>
<td>PAID</td>
<td>SOURCE</td>
<td>SUB_CAT</td>
<td>TOTAL</td>
<td></td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>IMAGESIZE</td>
<td>RED_SAMED</td>
<td>DUE_DATE</td>
<td>QUANTITY</td>
<td>ROUND_OFF</td>
<td></td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>IMAGESIZE</td>
<td>COST</td>
<td>DUE_TIME</td>
<td>NEW_UPRICE</td>
<td>CASH_OUT</td>
<td></td>
</tr>
<tr>
<td>PHONE_NO</td>
<td>ORIGINAL</td>
<td>MAIN_CAT</td>
<td>STAFF_NO</td>
<td>TOTAL</td>
<td>CHANGE</td>
<td></td>
</tr>
<tr>
<td>CUSTEMAIL</td>
<td>SLEEVE</td>
<td>SUB_CAT</td>
<td>&lt;/RECORD&gt;</td>
<td>RETAIL_PRC</td>
<td>TOT_DISC</td>
<td></td>
</tr>
<tr>
<td>ADDRESS1</td>
<td>BACKPRINT</td>
<td>POINTS</td>
<td>&lt;/JOBS&gt;</td>
<td>TAX_RATE</td>
<td>ORDER_NO</td>
<td></td>
</tr>
<tr>
<td>ADDRESS2</td>
<td>I2EENHANCED</td>
<td>FILMTYPE</td>
<td>- &lt;/JOBSTOCK&gt;</td>
<td>GP</td>
<td>TAX_TYPE</td>
<td></td>
</tr>
<tr>
<td>ADDRESS3</td>
<td>MATTE</td>
<td>INSTORE</td>
<td>&lt;/RECORD&gt;</td>
<td>AVG_COST</td>
<td>PAY_CASH</td>
<td></td>
</tr>
<tr>
<td>STATE</td>
<td>FWJ_PROD_NO</td>
<td>PROMOT</td>
<td>JOB_ID</td>
<td>STD_TAX</td>
<td>PAY_TYPE1</td>
<td></td>
</tr>
<tr>
<td>POSTCODE</td>
<td>LEAD_TIME</td>
<td>ILOADED</td>
<td>PROD_NO</td>
<td>SEQUENCE</td>
<td>REF_TYPE1</td>
<td></td>
</tr>
<tr>
<td>PRICE</td>
<td>HIGH_GLOSS</td>
<td>IUSED</td>
<td>PROD_NO</td>
<td>&lt;/RECORD&gt;</td>
<td>PAY_TYPE2</td>
<td></td>
</tr>
<tr>
<td>REQUEST1</td>
<td>- &lt;/PICTURE&gt;</td>
<td>IPRINTS</td>
<td>DESCRIPT</td>
<td>&lt;/JOBSTOCK&gt;</td>
<td>PAY_TYPE2</td>
<td></td>
</tr>
<tr>
<td>REQUEST2</td>
<td>- &lt;/PRODUCT&gt;</td>
<td>UPLOADED</td>
<td>QUANTITY</td>
<td>&lt;/CUSTOMER&gt;</td>
<td>PAY_TYPE3</td>
<td></td>
</tr>
<tr>
<td>REQUEST3</td>
<td>PROD_NO</td>
<td>DURATION</td>
<td>UNIT_NAME</td>
<td>&lt;/RECORD&gt;</td>
<td>REF_TYPE3</td>
<td></td>
</tr>
<tr>
<td>DDATE</td>
<td>PROD_NO</td>
<td>FILE_SIZE</td>
<td>NEW_UPRICE</td>
<td>ONLINE_ID7</td>
<td>PAY_TYPE4</td>
<td></td>
</tr>
<tr>
<td>TTIME</td>
<td>DESCRIPT</td>
<td>JOB_ID</td>
<td>TOTAL</td>
<td>ONLINE_ID8</td>
<td>REF_TYPE4</td>
<td></td>
</tr>
<tr>
<td>PICK_DATE</td>
<td>MAIN_CAT</td>
<td>JOB_ID</td>
<td>SERIAL_NO</td>
<td>TITLE</td>
<td>PAY_TYPE5</td>
<td></td>
</tr>
<tr>
<td>PICK_TIME</td>
<td>SUB_CAT</td>
<td>JOB_TYPE_PART</td>
<td>RETAIL_PRC</td>
<td>FIRST_NAME</td>
<td>REF_TYPE5</td>
<td></td>
</tr>
<tr>
<td>ONLINE</td>
<td>QUANTITY</td>
<td>DESCRIPT</td>
<td>TAX_RATE</td>
<td>LAST_NAME</td>
<td>PAY_TYPE6</td>
<td></td>
</tr>
<tr>
<td>STORE_ID</td>
<td>NEW_UPRICE</td>
<td>COMMENTS</td>
<td>SIZE</td>
<td>BUSINESS</td>
<td>REF_TYPE6</td>
<td></td>
</tr>
<tr>
<td>J0RTYPE</td>
<td>TOTAL</td>
<td>CREA_DATE</td>
<td>COLOUR</td>
<td>ADDRESS1</td>
<td>PAY_TYPE7</td>
<td></td>
</tr>
<tr>
<td>ITEMCOUNT</td>
<td>RETAIL_PRC</td>
<td>CREA_TIME</td>
<td>AUTHORISED</td>
<td>ADDRESS2</td>
<td>REF_TYPE7</td>
<td></td>
</tr>
<tr>
<td>PRICES_ADJUSTED</td>
<td>TAX_RATE</td>
<td>LAST_DATE</td>
<td>KIT_NO</td>
<td>ADDRESS3</td>
<td>PAY_TYPE8</td>
<td></td>
</tr>
<tr>
<td>HOMEDELIVERY</td>
<td>GP</td>
<td>LAST_TIME</td>
<td>DISC_TYPE</td>
<td>STATE</td>
<td>REF_TYPE9</td>
<td></td>
</tr>
<tr>
<td>PICKUPSTORE</td>
<td>AVG_COST</td>
<td>CUST_NO</td>
<td>TIME</td>
<td>POSTCODE</td>
<td>PAY_TYPE9</td>
<td></td>
</tr>
</tbody>
</table>
### C.6 Structure of Info.txt File

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STORE_NO</td>
<td></td>
</tr>
<tr>
<td>STORE_NAME</td>
<td></td>
</tr>
<tr>
<td>STORE_ADD1</td>
<td></td>
</tr>
<tr>
<td>STORE_ADD2</td>
<td></td>
</tr>
<tr>
<td>PHONE</td>
<td></td>
</tr>
<tr>
<td>EMAIL</td>
<td></td>
</tr>
<tr>
<td>USER_NAME</td>
<td></td>
</tr>
<tr>
<td>MACHINE_NAME</td>
<td></td>
</tr>
<tr>
<td>JOBID</td>
<td></td>
</tr>
<tr>
<td>IPRINTS</td>
<td></td>
</tr>
<tr>
<td>ILOADED</td>
<td></td>
</tr>
<tr>
<td>IUSED</td>
<td></td>
</tr>
<tr>
<td>I2EENHANCED</td>
<td></td>
</tr>
<tr>
<td>CUST_NO</td>
<td></td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td></td>
</tr>
<tr>
<td>LAST_NAME</td>
<td></td>
</tr>
<tr>
<td>PHONE_NO</td>
<td></td>
</tr>
<tr>
<td>PRICE</td>
<td></td>
</tr>
<tr>
<td>DDate</td>
<td></td>
</tr>
<tr>
<td>TTIME</td>
<td></td>
</tr>
<tr>
<td>PROD_NO</td>
<td></td>
</tr>
<tr>
<td>DESCRIPT</td>
<td></td>
</tr>
<tr>
<td>QUANTITY</td>
<td></td>
</tr>
<tr>
<td>REPRINTQUANTITY</td>
<td></td>
</tr>
<tr>
<td>NEW_UPPRICE</td>
<td></td>
</tr>
<tr>
<td>FILENAME</td>
<td></td>
</tr>
<tr>
<td>PRINTWIDTH</td>
<td></td>
</tr>
<tr>
<td>PRINTHEIGHT</td>
<td></td>
</tr>
<tr>
<td>IMAGESIZEX</td>
<td></td>
</tr>
<tr>
<td>IMAGESIZEY</td>
<td></td>
</tr>
<tr>
<td>ORIGINAL</td>
<td></td>
</tr>
<tr>
<td>BACKPRINT</td>
<td></td>
</tr>
<tr>
<td>I2EENHANCED</td>
<td></td>
</tr>
<tr>
<td>MATTE</td>
<td></td>
</tr>
<tr>
<td>HIGH_GLOSS</td>
<td></td>
</tr>
</tbody>
</table>
C.7 Permission to use TheMagicTouch.ie

Begin forwarded message:

From: Nathan Newbury <technical@themagictouch.co.uk>
Date: 16 February 2011 17:39:56 GMT
To: Ian MacMahon <ian.macmahan@studiolab.ie>
Subject: Welcome to Customer Zone

Dear Ian

Thank you for registering for our new Customer Zone.

Your email address and password will be your unique login.

Should someone else wish to use the Customer Zone please ask them to create their own login.

Marketing and Technical support is now available, please download any images or templates that you require.

We look forward to your feedback on how we can support and serve you better in the future.

Enjoy

Nathan Newbury
Head of Technical
www.themagictouch.co.uk

Skype Me™! Chat with me
Get Skype and call me for free.
D. User Manual

The User Manual for this system has been attached as a separate document.
E. Design Documentation

E.1 Data Flow Diagram
E.2 System Design Model – Waterfall Model

E.3 List of Tables and Descriptions

<table>
<thead>
<tr>
<th>Name:</th>
<th>Description:</th>
</tr>
</thead>
<tbody>
<tr>
<td>tbl_LIVE_JOB</td>
<td>Order details</td>
</tr>
<tr>
<td>tbl_LIVE_PICTURE</td>
<td>Product ordered details</td>
</tr>
<tr>
<td>tbl_LIVE_RECORD</td>
<td>Further order details</td>
</tr>
<tr>
<td>tbl_LIVE_PRODUCT</td>
<td>Product details, used for the Training Program</td>
</tr>
<tr>
<td>tbl_LIVE_CUSTOMER</td>
<td>Customer details</td>
</tr>
<tr>
<td>tbl_LIVE_IMPORT</td>
<td>Details of files imported</td>
</tr>
<tr>
<td>tbl_LIVE_DUPLICATES</td>
<td>Details of files that were duplicated, but only imported once</td>
</tr>
<tr>
<td>DATA</td>
<td>RAW table, temporarily contains details on orders</td>
</tr>
<tr>
<td>JOB</td>
<td>RAW table, temporarily contains all the details associated with the order.</td>
</tr>
<tr>
<td>PICTURE</td>
<td>RAW table, temporarily contains details on the product being ordered</td>
</tr>
<tr>
<td>PRODUCT</td>
<td>RAW table, temporarily contains details on the product being ordered</td>
</tr>
<tr>
<td>RECORD</td>
<td>RAW table, temporarily contains details on the order</td>
</tr>
</tbody>
</table>
E.4 Entity Relationship Diagram

E.5 Naming Conventions and Coding Standards

Each function or sub in the code will begin with a comment to describe what the set of code does. Comments are indicated in Visual Basic code by an apostrophe. Individual lines will also be commented where necessary.

The objects that were created for the database follow the Leszynski/Reddick Guidelines for naming database objects. These guidelines advise to indicate an object by a prefix of what the object is, in its name. Table E.4 below outlines the prefixes used for this database.

Table E.4

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>tbl</td>
</tr>
<tr>
<td>Query</td>
<td>qry</td>
</tr>
<tr>
<td>Form</td>
<td>frm</td>
</tr>
<tr>
<td>Report</td>
<td>rpt</td>
</tr>
<tr>
<td>Text Box</td>
<td>txt</td>
</tr>
<tr>
<td>Combo Box</td>
<td>cmb</td>
</tr>
<tr>
<td>Command Button</td>
<td>cmd</td>
</tr>
<tr>
<td>Check Box</td>
<td>chk</td>
</tr>
</tbody>
</table>
F. Sample Reports and Screen Shots

Figure F.1.1 Homepage

Figure F.1.2 Training Program Options
Figure F.1.3 Learn About a Product

Figure F.1.4 Example Invoice
### Total Products Sold

<table>
<thead>
<tr>
<th>PROD_NO</th>
<th>Description</th>
<th>Total Sold</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>G54R</td>
<td>Standard Mug - Right Hand</td>
<td>101</td>
<td>15.99</td>
</tr>
<tr>
<td>G34</td>
<td>Keyring Rectangle</td>
<td>52</td>
<td>5.99</td>
</tr>
<tr>
<td>G345</td>
<td>Small Heart Keyring</td>
<td>49</td>
<td>6.99</td>
</tr>
<tr>
<td>G34</td>
<td>Small Rectangular Keyring</td>
<td>33</td>
<td>8.99</td>
</tr>
<tr>
<td>S7</td>
<td>5” x 7” (standard size)</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>G34</td>
<td>Small Rectangular Keyring</td>
<td>16</td>
<td>3.99</td>
</tr>
<tr>
<td>G34L</td>
<td>Large Heart Keyring</td>
<td>14</td>
<td>8.99</td>
</tr>
<tr>
<td>G54R</td>
<td>Mug</td>
<td>13</td>
<td>13.99</td>
</tr>
<tr>
<td>G108</td>
<td>Jigsaw 16x16 36pcs</td>
<td>11</td>
<td>12.99</td>
</tr>
<tr>
<td>G119</td>
<td>Mouse Mat</td>
<td>11</td>
<td>14.99</td>
</tr>
<tr>
<td>G119</td>
<td>Mouse Mat - Driver</td>
<td>10</td>
<td>16.99</td>
</tr>
</tbody>
</table>

**Figure F.1.5 Example Report Sales Per Product (All Records)**

### Total Products Sold Between The Dates

<table>
<thead>
<tr>
<th>PROD_NO</th>
<th>DISCRPT</th>
<th>SUMQUANTITY</th>
<th>NSAV_UPRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>G100</td>
<td>Jigsaw (Small) 36pcs</td>
<td>1</td>
<td>16.99</td>
</tr>
<tr>
<td></td>
<td>Jigsaw 16x16 36pcs</td>
<td>1</td>
<td>15.99</td>
</tr>
<tr>
<td></td>
<td>Jigsaw Small (36 pcs)</td>
<td>1</td>
<td>12.99</td>
</tr>
<tr>
<td></td>
<td>Jigsaw Small (30 pcs)</td>
<td>1</td>
<td>12.99</td>
</tr>
<tr>
<td></td>
<td>Jigsaw Small 16x7 (30 pcs)</td>
<td>1</td>
<td>18.99</td>
</tr>
<tr>
<td></td>
<td>Mouse Mat</td>
<td>2</td>
<td>18.99</td>
</tr>
</tbody>
</table>

**Figure F.1.6 Example Report Sales Per Product Between Date Range**
Figure F.1.7 Example Report Sales Per Store Between Date Range
This appendix provides the visual basic coding for this system. The code was developed using Visual Basic Editor for Microsoft Access. The full code is not provided as it is too extensive.

### G.1 ImportAndChange Function

```vba
Function ImportAndChange()
    ' This function imports and checks if the record already exists in the database.
    ' The data is then imported to the temporary RAW tables, formatted and moved to
    ' the LIVE tables. A record is then inserted into tbl_LIVE_IMPORT to track the import.
    ' If the file already existed in the database, then the function inserts a record into
    ' tbl_LIVE_DUPLICATES. This keeps track of any duplicates that were attempted.

    Dim FilePath As String
    Dim XmlFile As String
    Dim sFoldername As String
    Dim tDATE As Date
    Dim sJobFile As String
    Dim JOBID As String
    Dim rs As DAO.Recordset

    FilePath = "N:\COMPLETED_JOBS\""  
    XmlFile = "\InfoXML.XML"
    sFoldername = Dir$(FilePath, vbDirectory)
    tDATE = Now

    Do While Len(sFoldername) > 0
        If sFoldername <> "." And sFoldername <> ".." And sFoldername <> "ARCHIVED_JOBS" Then
            DoCmd.SetWarnings False
            sJobFile = FilePath & sFoldername & XmlFile
            ' Find the JobID.
            JOBID = GetJobID(sJobFile)

            ' check if job id already exists in database
            Set rs = CurrentDb.OpenRecordset("SELECT JOBID FROM tbl_LIVE_JOB WHERE JOBID=" & JOBID & ",","")
            If rs.RecordCount = 0 Then
                ' Clear down temporary tables
                DoCmd.RunSQL "DELETE * FROM JOB;"
                DoCmd.RunSQL "DELETE * FROM PICTURE;"
                DoCmd.RunSQL "DELETE * FROM PRODUCT;"
                DoCmd.RunSQL "DELETE * FROM RECORD;"

                ' import from xml file to database
```

G.1
Application.ImportXML DataSource:=sJobFile, ImportOptions:=acAppendData

' change the date of DDATE, PICK_DATE and CREA_DATE before it is moved to tbl_LIVE tables
Call ChangeDateFormat

' move records to tbl_LIVE tables
DoCmd.RunSQL "UPDATE PICTURE SET JOBID = " & JOBID & ";
DoCmd.RunSQL "UPDATE JOB SET INVOICED = 'No';"
DoCmd.RunSQL "UPDATE JOB SET INVOICED = 'No';"
DoCmd.RunSQL "INSERT INTO tbl_LIVE_JOB ( STORE_NO, STORE_NAM1, STORE_ADD1, STORE_ADD2, STORE_STATE, STORE_COUNTRY, PHONE, EMAIL, MACHINE_NAME, JOBID, JOBTYPE, CUST_NO, PRICE, DDATE, PICK_DATE, ONLINE, STORE_ID, ITEMCOUNT, HOMEDELIVERY, GROUPSTORE, JOB.INVOICED FROM JOB;"
DoCmd.RunSQL "INSERT INTO tbl_LIVE_CUSTOMERS ( CUST_NO, FIRST_NAME, LAST_NAME, PHONE_NO, CUSTEMAIL, ADDRESS1, ADDRESS2, ADDRESS3, STATE ) SELECT JOB.CUST_NO, JOB.FIRST_NAME, JOB.LAST_NAME, JOB.PHONE_NO, JOB.CUSTEMAIL, JOB.ADDRESS1, JOB.ADDRESS2, JOB.ADDRESS3, JOB. STATE FROM JOB;"
DoCmd.RunSQL "INSERT INTO tbl_LIVE_STORES (STORE_NO, STORE_ID, STORE_NAM1, STORE_ADD1, STORE_ADD2, STORE_STATE, STORE_COUNTRY, PHONE, EMAIL) SELECT JOB.STORE_NO, JOB.STORE_ID, JOB.STORE_NAM1, JOB.STORE_ADD1, JOB.STORE_ADD2, JOB.STORE_STATE, JOB.STORE_COUNTRY, JOB.PHONE, JOB.EMAIL FROM JOB;"
DoCmd.RunSQL "INSERT INTO tbl_LIVE_PICTURE ( JOBID, PROD_NO, PROD_NO8, DESCRIPT, QUANTITY, REPRINTQUANTITY, NEW_UPRICE, FILENAME, PRINTWIDTH, PRINTHEIGHT, IMAGESIZEX, IMAGESIZEY) SELECT PICTURE.JOBID,PICTURE.PROD_NO, PICTURE.PROD_NO8, PICTURE.DESCRIPT, PICTURE.QUANTITY, PICTURE.REPRINTQUANTITY, PICTURE.NEW_UPRICE, PICTURE.FILENAME, PICTURE.PRINTWIDTH, PICTURE.PRINTHEIGHT, PICTURE.IMAGESIZEX, PICTURE.IMAGESIZEY FROM PICTURE;"
DoCmd.RunSQL "INSERT INTO tbl_LIVE_RECORD ( JOB_ID, JOB_ID8, JOB_TYPE, ORDER_NO, MAIN_CAT ) SELECT RECORD.JOB_ID, RECORD.JOB_ID8, RECORD.JOB_TYPE, RECORD.JOB_TYPE_PART, RECORD.DESCRIPT, RECORD.COMMENTS, RECORD.CREA_DATE, RECORD.CUST_NO, RECORD.STAFF_NO, RECORD.TOTAL, RECORD.DEPOSITS, RECORD.ORDER_NO, RECORD.MAIN_CAT FROM RECORD WHERE (((RECORD.Job_TYPE)<>''));"

' keep track of what was imported and when
DoCmd.RunSQL "INSERT INTO tbl_LIVE_IMPORT (PATH, FOLDER, JOBID, DATE_IMPORT) VALUES (" & FilePath & "," & sFoldername & "," & JOBID & "," & tDATE & ");"
Else
' got a duplicate so tell the user and increment duplicate table
DoCmd.RunSQL "INSERT INTO tbl_LIVE_DUPLICATES (PATH, JOBID, DATE_DUPLICATE) VALUES (" & FilePath & sFoldername & "," & JOBID & "," & tDATE & ");"
End If
DoCmd.SetWarnings True
End If
' reset sFoldername
sFoldername = Dir
Loop
End Function

G.2 GetJobID Function

Function GetJobID(ByVal FilePath As String) As String

' This function finds the JOBID in each XML file from the path passed through.

Dim XmlFile As MSXML2.DOMDocument
Set XmlFile = New MSXML2.DOMDocument

XmlFile.async = False
XmlFile.validateOnParse = False

' Read in file from address given.
XmlFile.Load (FilePath)

' Exit if the file is nothing.
If XmlFile Is Nothing Then
    GetJobID = vbNullString
    Exit Function
End If

Dim XmlNode As IXMLDOMNode

' Search for JOBID node.
Set XmlNode = XmlFile.SelectSingleNode("JOB//JOBID")

If XmlNode Is Nothing Then
    GetJobID = vbNullString
Else
    GetJobID = XmlNode.Text
End If
End Function
G.3 **Change the Format of the Date Fields**

Sub ChangeDateFormat()
    ' This sub changes numbers to dates for the date fields

    Dim db As DAO.Database
    Dim rs As DAO.Recordset
    Dim ChangeDate, sqlState As String

    ' Open a recordset and select the values for the recordset
    Set db = CurrentDb()
    sqlState = "SELECT DDATE, PICK_DATE FROM JOB"
    Set rs = db.OpenRecordset(sqlState)
    rs.MoveFirst

    ' While recordset is not empty do the following
    With rs
        Do Until rs.EOF
            ' Call function NumberToDate and update the field
            If rs!DDATE <> "" Then
                .Edit
                ChangeDate = NumberToDate(rs!DDATE)
                rs!DDATE = ChangeDate
                .Update
            End If
        End If
        rs.MoveNext
    Loop
    End With
End Sub

Private Function NumberToDate(ByVal Number As Long) As Date
    ' Changes a number to a date using the DateAdd method

    Dim BaseDate As Date

    BaseDate = CDate("30/12/1899")
    NumberToDate = DateAdd("d", Number, BaseDate)
End Function

G.4 **Checking Dates are Valid for frm_Select_Report**

Function CheckDatesReport()

    ' This function checks that the dates entered in frm_Select_Report
    ' are valid.
If Forms!frm_Select_Report!cbo_Select_Report.ListIndex > 0 Then ' if report selected has index > 0
    If Forms!frm_Select_Report!cbo_Select_Report.ListIndex <> 0 Then ' combo doesn't have all database selected
        Forms!frm_Select_Report!txt_StartDate.SetFocus
    End If
ElseIf Forms!frm_Select_Report!cbo_Select_Report.ListIndex = 0 Then ' if report selected has index = 0
    DoCmd.OpenReport "rpt_Sales_Per_Product", acViewPreview
End If
Else ' no report is selected
    MsgBox ("Please select a report.")
End If
End Function

G.5 Generating Invoices
Private Sub cmd_Generate_Invoices_Click()
    ' This sub occurs when the Generate Invoices command is clicked on frm_Select_Invoice_Dates
    ' First it checks if the dates have valid fields.
    ' If the fields are not valid and msgbox is opened to notify the user.
    ' When the fields are entered correctly, queries and report are opened.
    ' Then a query is run to change the INVOICED field in tbl_LIVE_JOB to Yes only
    ' for the records that were selected by the query. Then the files associated with
    ' those orders are moved to the ARCHIVED folder.
    ' Then a report is opened to show the user the orders that have not yet been invoiced.

    txt_StartDate.SetFocus

    If txt_StartDate.Text = "" Then
        MsgBox ("Please enter Start Date.")
        txt_EndDate.SetFocus
    ElseIf txt_StartDate.Text <> "" Then
        DoCmd.OpenQuery "qry_INVOICES"
        DoCmd.OpenReport "rpt_INVOICES", acViewPreview
        DoCmd.SetWarnings False
        Dim rs1 As DAO.Recordset
Dim sqlScript As String
Dim sFilePath As String
Dim sFolder As String
Dim FSO As Object

sFilePath = "N:\COMPLETED_JOBS\"
sqlScript = "SELECT tbl_LIVE_JOB.JOBID, tbl_LIVEJOB.DDATE, tbl_LIVE_JOB.DDATE, tbl_LIVE_IMPORT.FOLDER " & _
"FROM tbl_LIVE_JOB INNER JOIN tbl_LIVE_IMPORT ON tbl_LIVE_JOB.JOBID = tbl_LIVE_IMPORT.JOBID " & _
"WHERE (((tbl_LIVE_JOB.DDATE)>=#" & [Forms]![frm_Select_Invoice_Dates]![txt_StartDate] & "] #) AND 
((tbl_LIVE_JOB.DDATE)<=#" & [Forms]![frm_Select_Invoice_Dates]![txt_EndDate] & "] #));"

Set FSO = CreateObject("scripting.filesystemobject")
Set rs1 = CurrentDb.OpenRecordset(sqlScript, dbOpenDynaset)
rs1.MoveFirst

With rs1
  Do Until .EOF
    ' update the INVOICED field to Yes
    DoCmd.RunSQL "UPDATE tbl_LIVE_JOB SET tbl_LIVE_JOB.INVOICED = 'Yes' WHERE 
tbl_LIVE_JOB.JOBID=" & rs1!JOBID & ";"
    sFolder = rs1!FOLDER
    If FSO.FolderExists(sFilePath & sFolder) Then
      FSO.MoveFolder Source:=sFilePath & sFolder, Destination:=sFilePath & "ARCHIVED_JOBS 
    Else
      MsgBox sFolder & " has already been invoiced and archived!", vbExclamation, "Order Already Invoiced and Archived"
    End If
    rs1.MoveNext
  Loop
End With
If rs1.RecordCount = 0 Then
  MsgBox ("There were no orders made between these two dates.")
End If
rs1.Close
Set rs1 = Nothing
DoCmd.OpenReport "rpt_Jobs_Not_Invoiced", acViewPreview
End If
End Sub
H. Implementation and Test Documentation

This section of the appendix gives a description of the implementation stage and the changes that were made during this step of the process. Explanations of the tests carried out on the system are also outlined.

H.1 Implementation Document

The implementation started by moving a copy of the database to the desktop of Shankill photo centre’s main computer. Once this was done, the COMPLETED and ARCHIVED folders had to be set up. The ARCHIVED folder was placed inside the COMPLETED folder to allow the order files to be moved easily. Once these folders were in place, the path to these folders was then inserted into the VBA code. After this was completed, a large set of orders which amounted to 322 was imported into the database. The database imported all the data as expected.

Once this data was imported, the staff of Frame Line were given a demonstration on the use of the system. This demonstration included use of the training program, generating invoices, generating reports and also importing data manually. This demonstration included assessing the ability of the system. The system was used to generate invoices for all 322 orders. The system executed this task very effectively and produced the invoices accurately.

Once the demonstration was complete, the client wished to make some changes to the system. The invoices and reports were re-formatted to comply with the client’s preference. Some variables were removed from the invoices such as CREA_DATE and DDATE. The client did not deem these variables to be important. The client wished to group the invoices by STORE_NO. This would mean that all online orders would be in the same invoice. This amendment was made successfully.

The client requested that two additional variables were to be added to the invoices: “Net Amount” and “VAT Amount”. These variables were set as calculated fields on the invoices. The two variables were built and included in the invoices effectively. The client also wished two variables to be entered into tbl_LIVE_JOB. These variables were Delivery_Method and GLS_Delivery.

The next modification that the client made to the database was to include a table containing all details related to individual stores. This table was named tbl_LIVESTORES. The primary key in this table is STORE_NAM1. This variable had to be used since STORE_ID and STORE_NO were not consistent with each other. This table would be used to send special offers and promotions to individual stores.

The final adjustment that was made to the database included adding a search function. A form called frm_Search_For_Job was designed. This form would allow the user to search tbl_LIVE_JOB for a particular Job ID. This would allow staff to manually check the details of an order, including whether it has been invoiced or not. This form requires the user to insert a Job ID in the form of JOB12345. An additional command button was added to the
homepage to allow access to this form. During the implementation stage, a number of findings were noted. The first of these was the way in which the stores are identified. For an online store, STORE_NO is entered as 0000, 9999 or 1111. There is then another numbering system used called STORE_ID that is used for kiosk orders. These numbers do not correspond to each other and cause difficulty when trying to differentiate the store. The database must use the variable STORE_NAM1 to uniquely identify a store rather than one of these indexes.

Another issue that was recorded was inaccuracies in the descriptions of the products being ordered. Fujifilm have created an index that differentiates each product. Currently, taking product G108 as an example, this product is a 30 piece jigsaw. When a large import of 322 orders was entered into the system, it was found that this item had five different descriptions. This caused issues in the reports and invoices. If a product is given the wrong product code, then it is potentially given the wrong price also. This issue was noted in the implementation stage of this project. Other products have similar inaccuracies.

After the implementation was complete a number of final tests were run on the system. These tests are outlined below in Appendix H.2, p. H.3.
H.2 Test Document

Logic / Desk Checking
This was a manual task that was performed by the developer. This involved working through the system one step at a time, ensuring that the logic of the system made intuitive sense and that the navigation functions were performing correctly. This test had to be performed before the system was implemented. The result of this test was that the system was logically sound.

Syntax
This stage of the testing involved the developer working through any code that was written during the development stage. It required the tester to check the syntax of the code by running or compiling sections individually, ensuring there were no bugs in the code and no syntax errors. This process was performed during the whole development stage and again before implementation. Some syntax errors were found and rectified immediately.

Unit
This type of testing requires the tester to use incorrect data for inputs and perform the unexpected on the system. Incorrect data was attempted for the date range on the forms and also for the training program. This stage of the testing involved the tester assuming the role of a user with very low level of computer skills. It resulted in extra labels being introduced to the date entry forms and also amendments to the report layouts. This part of the testing phase included the checking of functionality of the forms. If a record was to be deleted after clicking a command button, the tester would check that a record was deleted and also that it was the correct record to delete. This stage also required ensuring that the data outputted to the reports was accurate according to the inputs.

Acceptance
In order to perform an acceptance test, it is required to simulate the environment that the system will be used in. An employee of Frame Line was performing this test along with the observation of the system developer. Any issues or bugs were noted and changed immediately. This stage also involved introducing a larger dataset to ensure the system could cope with a larger quantity of records. The result of this test, after the bugs were fixed, was that this system performed according to the requirements outlined by the client.
REFERENCES

[1] Print Digital 4 U
Available at: <URL: www.printdigital4u.com>

[2] Studio Lab
Available at: <URL: www.studiolab.ie>

[3] Fujifilm Glodal Website
Available at: <URL: www.fujifilm.com>

Available at: <URL: www.themagictouch.ie>

Available at: <URL: http://www.microsoft.com/windows/early-versions.aspx>

Available at: <URL: www.toptenreviews.com>

[7] SSADM Diagrams, Smart Draw
Available at: <URL: http://www.smartdraw.com/resources/tutorials/ssadm-diagrams/>

Available at: <URL: http://www.mvps.org/access/general/gen0012.htm>

Available at: <URL: http://msdn.microsoft.com/en-us/library/aa221653%28v=office.11%29.aspx>

[10] Gliffy: Online Diagram Software
Available at: <URL: www.gliffy.com>

Available at: <URL: http://www.xml.com/pub/a/98/10/guide0.html?page=2>
FRAME LINE LTD.
Invoicing and Reporting System

28th March 2011

Prepared by: Gillian Lane
Supervisor: Denise Leahy
## Development of an Invoicing and Reporting System

### March 2011

---

### TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>NO.</th>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>D.1</td>
<td>Frame Line Homepage</td>
<td>D.1</td>
</tr>
<tr>
<td>D.2</td>
<td>Training Program</td>
<td>D.3</td>
</tr>
<tr>
<td>D.2.1</td>
<td>Learn How to Produce Frame Line Products</td>
<td>D.3</td>
</tr>
<tr>
<td>D.2.2</td>
<td>Change Product Info</td>
<td>D.5</td>
</tr>
<tr>
<td>D.2.3</td>
<td>Delete Product from Database</td>
<td>D.6</td>
</tr>
<tr>
<td>D.2.4</td>
<td>Add Product to Database</td>
<td>D.7</td>
</tr>
<tr>
<td>D.3</td>
<td>Invoices</td>
<td>D.9</td>
</tr>
<tr>
<td>D.4</td>
<td>Reports</td>
<td>D.12</td>
</tr>
<tr>
<td>D.5</td>
<td>Import Data</td>
<td>D.14</td>
</tr>
<tr>
<td>D.6</td>
<td>Search for Job</td>
<td>D.16</td>
</tr>
<tr>
<td>D.7</td>
<td>Logging Out</td>
<td>D.18</td>
</tr>
<tr>
<td>D.8</td>
<td>Entering Source Code to Change Password</td>
<td>D.19</td>
</tr>
</tbody>
</table>
D. User Manual

This document accompanies an invoicing and reporting system developed for Frame Line Ltd. This manual has been constructed from the perspective of the user. This document explains how the system may be used.

D.1 Welcome to Frame Line Ltd Homepage

Step 1: Open up the system
This is achieved by navigating to FrameLine.accdb. This may be done by double-clicking the shortcut on the Desktop.

Step 2: Enter password
MS Access then opens automatically and a prompt requests the user to enter a password to enter the database. Enter the password and select “Ok” as seen below in figure D.1.2.

Step 3: The system will then open the system and the user will see the screen below. The homepage is automatically opened as the first page. Before continuing, the user must select the “Options” button on the toolbar. When this is clicked, a screen will pop as can be seen below in figure D.1.3. Select “Enable this content” and click “Ok”.

Figure D.1.2 Enter password
Step 4: Now the homepage is displayed. There are four options on this page. These options will be discussed individually in the following sections.
D.2 Training Program

This section outlines the steps a user may take when navigating through the training program.

**Step 1:** Select the “Training Program” option on the homepage. This button will bring the user to a form called “Training Program Options”. This screen can be seen below in figure D.2. There are five options on this page.

![Figure D.2 Training Options](image)

**D.2.1 Learn How to Produce Frame Line Products**

**Step 1:** Select the button “Learn How To Produce Frame Line Products”. Clicking this button will bring the user to the page below in figure D.2.1.1. This page requests that the user selects a product from the drop-down list. Once the user selects a product, they then have to click the button named “Learn About This Product”.

Step 2: After clicking “Learn About This Product”, the user is then brought to a form showing the product that was selected. The details on this form are instructions for how the selected product is to be made, including a working hyperlink to the website theMagicTouch.co.uk. An example of this form can be seen below in figure D.2.1.2. The user may not change any of the details on this form. The user may return to the Training Options, to the homepage or select a different product by clicking on the buttons at the bottom of the page.
D.2.2 Change Product Info

**Step 1:** After clicking this button, the user is brought to the same page as is seen in figure D.2.1.1. The user is asked to select a product. Then user may click the button “Change This Product”, if they would like to change any of the information for the product selected.

**Step 2:** After selecting the product and clicking the “Change This Product” command, the user is brought to a screen showing all the product details. This screen can be seen below in figure D.2.2.1. The user **may change the details of the product.** If the user wishes to change the details, then they must hit the “Save Changes” button.

![Figure D.2.2.1 Details of product to change](image)

If the user makes a change and hits the “Save Changes” button, a prompt will appear, as seen in figure D.2.2.2, which warns the user that the product is being changed. If the user does not want to proceed with the change, they can hit “Cancel”. Otherwise, in order to save the change, the user must hit “Ok”.

If the user makes a change and hits the “Save Changes” button, a prompt will appear, as seen in figure D.2.2.2, which warns the user that the product is being changed. If the user does not want to proceed with the change, they can hit “Cancel”. Otherwise, in order to save the change, the user must hit “Ok”.
The user then has the option to close this form without saving anything, by selecting the button “Cancel Without Saving”.

D.2.3 Delete Product from Database

**Step 1:** After clicking this button, the user is brought to the form below, showing the product details. The user may navigate through the products using the navigation pane at the bottom of the form. This can be seen in figure D.2.3.1 below.
Step 2: If the user wishes to delete a product, they can navigate to the product, and hit the “Delete Record” button. When this button is hit, a prompt warns the user that they are about to delete a record. If the user wishes delete the product, then they press “Yes”, otherwise they should select “No” and the record will remain in the database.

D.2.4 Add Product to Database

Step 1: Upon clicking this command, the user is brought to a blank form. This form can be seen below in figure D.2.4.1. The user can then enter the details of the product they wish to add.
If the user enters the details and hits "Add Product to Database", the system will add the product to the database. The user **must enter a Product Code and Product Description** for the record to be saved. If either or both of these boxes are empty, the system will prompt the user to enter them. This can be seen below in figure D.2.4.2. The user can then close this form without saving the product details they entered by selecting the button "Close Without Saving".

![Figure D.2.4.1 Add new product to database](image1)

![Figure D.2.4.2 Must enter Product Code and Description](image2)
D.3 Invoices

This section outlines the steps required by the user to generate invoices from this system.

**NOTE:** Before the invoices are generated, close all other queries and reports.

**Step 1:** When the “Invoices” button is selected on the homepage, a new form appears requesting that the user enters a date range. This form can be seen in figure D.3.1. If the user does not enter a start date, and they try to press “Generate Invoices”, then the system prompts them to enter the missing value.

![Image of the form for entering date range](image_url)

**Figure D.3.1 Please enter Start Date**

**Step 2:** If the user attempts to enter a start date that is above the end date, then the system will notify the user that this value is not valid.

**Step 3:** When the user enters two dates that meet the requirements, and clicks “Generate Invoices”, the system opens three pages: qry_INVOICES, rpt_INVOICES and rpt_Jobs_Not_Invoiced_Yet. The first page, qry_INVOICES shows the jobs that have been selected between the two dates. The second page, rpt_INVOICES shows the invoices for the stores being invoiced. Each store starts on a separate page. The third page,
rpt_Jobs_Not_Invoiced_Yet shows a list of the orders that are yet to be invoiced. If the user has left any orders out, this report will indicate so.

**Step 4:** The system may prompt the user with the screen seen below in D.3.2. This warning tells the user that they have already invoiced the order specified in the warning box. However, the invoice is still generated with the others.

![D.3.2 Order already invoiced!](image)

**Step 5:** The invoices are currently in “Print Preview”. If the user wishes to look at the invoices generated before printing, they may do so by choosing one of the following options.

**Option 1:** Click the icon along the top toolbar, in the “Zoom” section. Select “More Pages” and choose from the options. Then the user can zoom in to look at the invoices using the toolbar running along the bottom of the screen. Both of these functions are highlighted in figure D.3.3 below.
Option 2: Right-click on the rpt_INVOICES tab and select “Report View”. This can be seen in figure D.3.4. This allows the user to view all of the invoices that were generated.

Step 6: In order to print the invoices, the user just has to click on the print icon at the top left hand side of the screen. This can also be seen in figure D.3.4
This section outlines the steps required to generate reports using this system.

**NOTE:** Before the reports are generated, close all other queries and reports.

**Step 1:** When the “Reports” button is clicked on the homepage, the user is brought to form called “Choose a Report”. This page requests that the user selects a report to view and then enters the date range they require. This page may be seen in figure D.4.1 below.

![Choose a Report](image)

**Figure D.4.1 Choose a report and date range**

**Step 2:** The user must select a report. Once this is done, they may need to enter a date range. If the user does not enter a date range and it is required, the system will prompt the user to fill in the missing fields. An example of this is seen below in figure D.4.2.
Step 3: Once the relevant fields are filled, and the user clicks “Generate Report”, the system produces the report. An example of this can be seen below in figure D.4.3. This report shows the total number of products sold between two dates that were entered.

Step 4: If the user wishes to change the view of the report, they may do so by choosing one of two options outlined in Step 5 of section D.3 Invoices.
D.5 Import Data

This system imports data automatically using a scheduled task on the computer the system is installed on. If the user would like to import data by hand, they may do so using the following steps outlined below. This functionality allows the user to import orders one at a time only.

If the user would like to run the code that is automatically scheduled, they may do so by clicking on the object named “ImportData” in the Macros section. This will run the same code that is run when the scheduled task is performed.

**Step 1:** When the “Import Data” button is clicked on the homepage, a file dialog box appears. This dialog box requests that the user selects the folder that they wish to import from. The user must navigate to find the folder using this dialog box. This may be seen below in figure D.5.1. Once the folder name can be seen in the “Folder Name” textbox, then the user may hit “Ok”.

**Note:** The system requires a folder to be selected only. The folder name should correspond to the JOBID.

![Figure D.5.1 Select folder to import from](image)

**Step 2:** The user may cancel this procedure without importing any data by clicking “Cancel” on the file dialog box. Otherwise, once the user hits “Ok”, a message pops up to notify the
user that the import has been successful. It shows the user the folder name that was selected for import. This can be seen below in figure D.5.2. The user may check that the file has been imported by looking at the most recent record in tbl_LIVE_IMPORT.

**Figure D.5.2 Import successful**

**Step 3:** If the user has by accident tried to import an order that is already in the system, they will be notified as can be seen below in figure D.5.3. The file will not be imported again and a record will be entered into the tbl_LIVE_DUPLICATES to track this mistake.

**Figure D.5.3 Duplicated import**
D.6 Search for Job

If the user wishes to search for an order, they may do so by clicking this command on the homepage.

**Step 1:** Upon clicking the command “Search for Order” on the homepage, the system shows the user a form called frm_Search_For_Job. This form requests that the user enter the Job ID of the order they are searching for. This form can be seen below in figure D.6.1.

![Search by Job ID](image)

**Figure D.6.1 Search for Order**

**Step 2:** Once the user has entered the Job ID and clicked the command “Search for Order” on the form, tbl_LIVE_JOB is shown. If the Job ID is in the database, the record will be highlighted by the system. This can be seen below in figure D.6.2. If however the order number does not exist in the system, then the user will be shown the first record in tbl_LIVE_JOB.
**Step 3:** The user may look at the record by scrolling across the screen.
D.7 Logging Out

The system does not require a log out. If the system needs to be closed, then the user has to click the button in the top right hand corner. Figure D.5.3 has this button highlighted.
NOTE: In order to change the source code, the user must have an understanding of Visual Basic coding. Changes to this code could cause problems within the system, if not executed successfully.

Step 1: The user may enter the Visual Basic code (VBA) by clicking the Visual Basic icon in the “Database Tools” tab on the toolbar along the top of the screen. This can be seen below in figure D.8.1.

![Visual Basic Code](image)

Figure D.8.1 Visual Basic Code

Step 2: Once this icon is clicked, the VBA Editor will appear on the screen. On clicking the project named “FrameLine” in the left-hand pane, the user is asked to enter a password. This can be seen in figure D.8.2 below. Upon entering the password, the user can then see the code behind the database.
Step 3: Once the correct password is entered, the code may be viewed and changed where necessary. In order to change the password for the source code, the user must click the “Tools” tab on the top toolbar, and click on “FrameLine Properties…”, as seen below in figure D.8.3.
**Step 4:** The user is then shown the task pane below. By clicking on the “Protection” tab, the user may enter a new password for the VBA Editor. Upon clicking “Ok”, the new password will be saved. The task pane can be viewed below in figure D.8.4

![FrameLine - Project Properties](image)

Figure D.8.4 Enter new password