COMP4 Coursework

Joel Butcher

October 18, 2014

Contents

Chapter 1

Analysis

1.1 Introduction

1.1.1 Client Identification

My client is Josh Campbell, he is 24 years old. He uses computers regularly for deisgn work, so has experience of computer systems. He uses his computer to design flyers, handouts, banners and visual graphics for projection, as well as surfing the web, email and various social media networks.He rarely uses hard copies other than to preview hes work before sending it off to print. Josh uses a 2012 Mac Pro with the latest version of Apple's operating system, OS X (10.9).

Josh is the head of the media department for Cambridge Community Church. This involves being responcible for the large amount of Audio and Visual equipment used on the churches Sunday services. This currently invloves spreadsheet with limited info on each item.

Josh would like to have a database management system to be able to hold information about each item and their various attributes. He would likke this database to be lovated on the churches central server so that it can be accessed by all staff if it it deemed necessary. He would use this database to store location, value and insurance details incase of damage or theft.he would like all of the information kept as a virtual copy as well as a hard copy to kept as a visual backup in case of harddrive failure or corruption.

He would also like to keep the location of each item as up to date as possible and if the location changes, he would like to be notified by email when it is entered/updated in the system.

1.1.2   Define the current system

The current system consists of multiple excel spreed sheets. There is one spread sheet for each of three locations; main office, main church building, and storage. Each spreedsheet consists of items located there as well as information on the value of each item, the quantity and the total value for the items with multiple entries. Each spreedsheet is divided up into equipment type (i.e Cableing, lighting, audio, visual/camera's)

1.1.3   Describe the problems

There are a number of problems with the current system. One of the problems is that there is no notification system to tell you when information is getting outdated or something is changed. For example, if an item is bought or sold, the total costings for that item will be updated and no-one will be notified. Another problem is that the current system doesn't show the PAT testings for all the items, these tests go out of date every 6 months and there is no way of being notified when a new PAT test is needed on an item.

1.1.4   Section appendix

Below are the questions that I asked my client at the interview and the answers he gave to me. I have typed up the questions and answers in markdown format then imported it as a pdf document so that it is easier to read.

Figure 1.1: Interview Questions (pg 1)

Interview Questions

1. What does the current system do?

   - Multiple excel spreadsheets that list all the AV equipment

2. What are the problems or drawbacks of the current system?

   - There is no notification system
   - Data is easily out of date.

3. How much data is currently recorded?

   - Current data stored is the item name, its location, the quantity and it's value.

4. What extra data will need to be included?

   - PAT testing's
   - Current location
   - The item's usable state (working, in need of repair, being repaired etc)

5. How frequently will the data need to be updated?

   - The data will need to be updated a few times a month or so
   - Whenever the location changes.

6. Will new records need to be added or deleted? If so, how often?

   - New records will need to be entered, or some deleted every couple of months.
   - Whenever new equipment is bought or if an item is sold

7. How important is the data or information that is to be recorded?

   - Data is of high importance as it will be kept as a record for insurance in case of theft or damage

8. Are there any algorithms that are going to need to be implemented?

   - The number of a single item there is at a particular location
   - The total number of that item altogether
   - The sum of the values those individual items (value per unit * quantity)

9. When are the algorithms going to be run?

   - These will need to be run when there are new items added/removed to a group of the same item
   - If the value of an item changes

Figure 1.2: Interview Questions (pg 2)

10. What inputs are required for the proposed system?

    ◦ Inputs are likely to be text, numbers and currency

11. What outputs are required for the proposed system?

    ◦ Outputs are likely to be the same as the inputs
    ◦ Notifications of when PAT tests are in need or reissue
    ◦ Notifications when an items location or quantity is changed
    ◦ A print function would be necessary

12. Are hard copies required?

    ◦ Yes, hard copies would be required a visual backup.

13. Are back dated records required?

    ◦ Yes, for insurance purposes

14. How long are these records going to be kept?

    ◦ We will keep back dated records for a year

15. How are these records going to be stored?

    ◦ We will store them electronically on the file server

16. How often will outputs be required?

    ◦ Outputs will be required whenever possible

17. What computing resources do you currently possess to aid the new system's operation?

    ◦ We currently have a Mac Pro that we use as a file server. This is where the database system will be placed.

18. Is security an issue?

    ◦ No, security is not an issue, although the data would need to be backed up.

19. Should there be restricted access to certain areas?

    ◦ No, restricted access is not needed.

20. What errors and exceptions will need to be reported in the new system?

    ◦ I'm not 100% until we start testing the system.

21. How should these errors and exceptions be reported?

    ◦ Errors should be reported to you either via email or another notification

Figure 1.3: Interview Questions (pg 3)

method.

22. Are there any constraints on hardware, software, data, cost or time?

  ○ No budget, time deadline is flexible and we'll adapt to whatever software/hardware resource available.

1.1.5   The current system

Data sources and destinations

In the current system, there are multiple data sources. The client and his colleagues as well as members of the AV crew for the church can enter data into the spreadsheet by using a computer in the office and accessing the on the server.

Algorithms

In the current system, there are only a few algorithms in place.

Algorithm 1 Algorithm 1, When new item is bought:

1: **IF** $Item = NewItem$ **THEN**
2:     **SET** $Action$ **TO** $EnterNewItem$
3: **ELSE IF** $Item = ItemMatch$ **THEN**
4:     **SET** $Action$ **TO** $UpdateItem$
5: **END IF**

Algorithm 2 Algorithm 2, When an item is sold or replaced:

1: **IF** $Item = Sold$ **THEN**
2:     **SET** $Action$ **TO** $UpdateQuantity$
3: **ELSE IF** $Item = Damaged$ **THEN**
4:     **SET** $Action$ **TO** $UpdateQuantity$
5:     **SET** $Action$ **TO** $FileInsuranceClaim$
6: **ELSE IF** $Item = Stolen$ **THEN**
7:     **SET** $Action$ **TO** $FileInsuranceClaim$
8: **END IF**

Data flow diagrams

Figure 1.4: Flow Diagram Key.
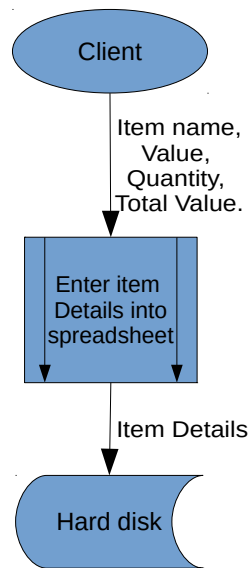
Figure 1.5: Entering a new item.
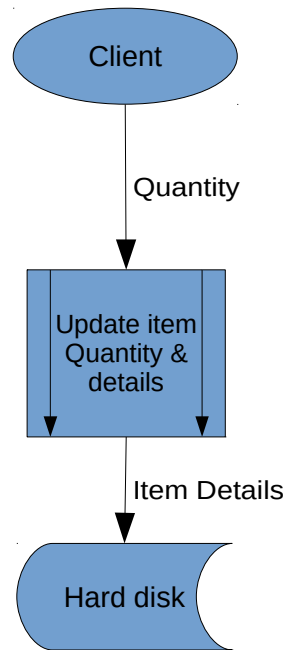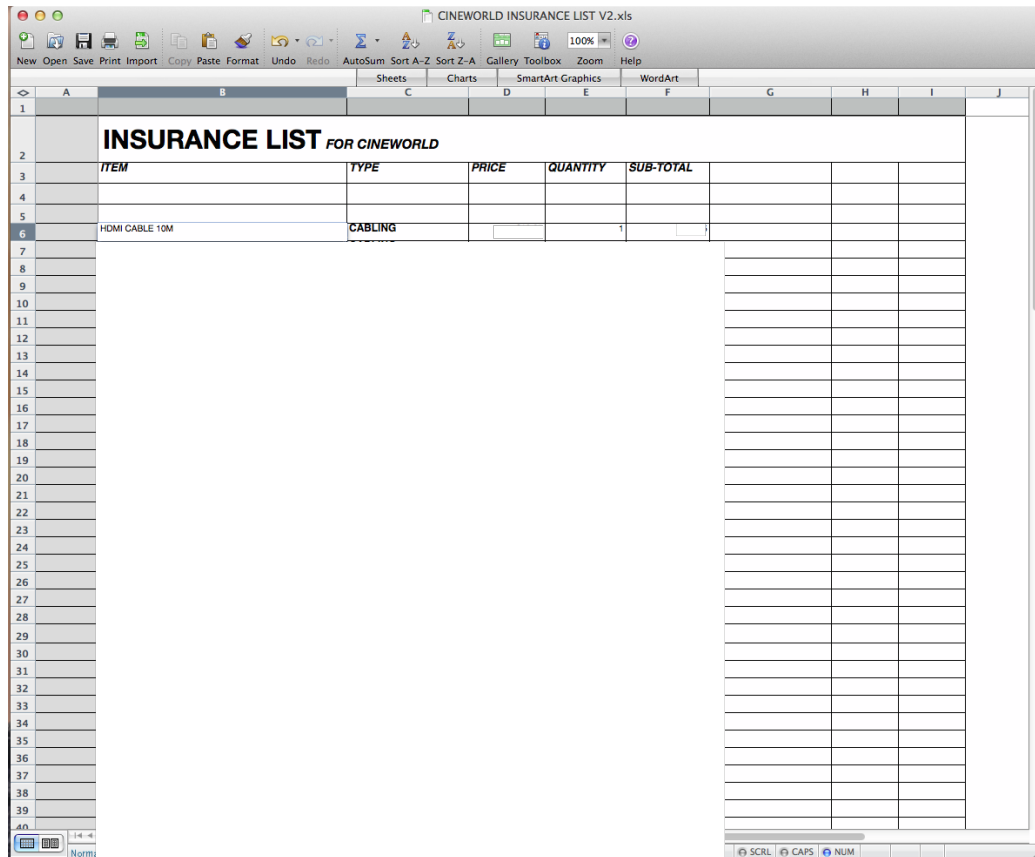
Figure 1.6: Updating an item that already exists in the table.

Input Forms, Output Forms, Report Formats

Josh has provided me with a screenshot of him entering some data into his current system. I have boxed out confidential information such as item values and their respective sub-total values:

Figure 1.7: Josh Entering Item Name.

Here is an screen shot showing the calculation used to get the Sub-Total Value:

Figure 1.8: Sub-Total Calculation.

1.1.6   The proposed system

Data sources and destinations

The Following table shows the proposed data and their respective sources and destinations.

Source	Data	Data Type	Destination
Josh	ItemID	Integer	Database - Item Records
Josh	ItemName	String	Database - Item Records
Josh	ItemType	String	Database - Item Records
Josh	Value	Real	Database - Item Records
Josh	Quantity	Integer	Database - Item Records
Josh	SubTotal	Real	Database - Item Records
Josh	OnLoan	Text	Database - Item Records
Josh	LoanID	Integer	Database - Loan Records
Database - Item Records	ItemID	Integer	Database - Loan Records
Database - Item Records	ItemName	Text	Database - Loan Records
Josh	LoanRate	Real	Database - Loan Records
Josh	LoanStart Date	Date	Database - Loan Records
Josh	LoanEnd Date	Date	Database - Loan Records
Josh	LoanTime	Time	Database - Loan Records
Josh	LoanCost	Real	Database - Loan Records
Josh	TestID	Integer	Database - PAT test records
Database - Item Records	ItemID	Integer	Database - PAT test Records
Database - Item Records	ItemName	Text	Database - PAT test Records
Database - Item Records	LastTest	Date	Next PAT test Calculation, Database PAT test Records
Next PAT Test Calculation	NextTest	Date	Database - PAT test Records
Josh	TestResult	Text	Database - PAT test Records

Data flow diagram
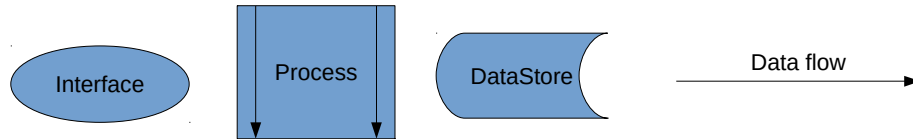
Figure 1.9: Flow Diagram Key.

Figure 1.10: Enter New Item.
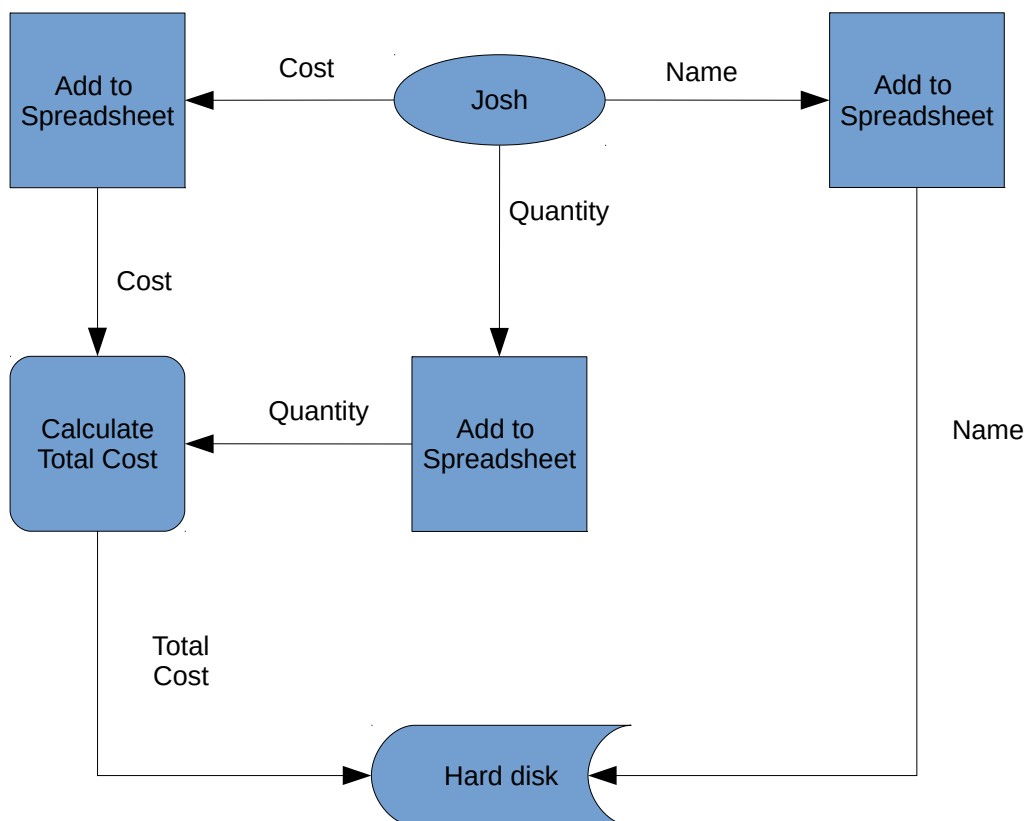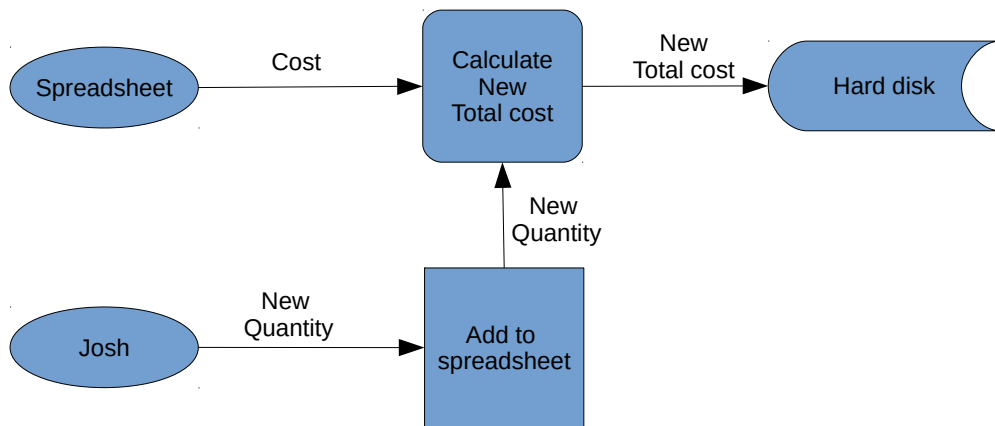
Figure 1.11: Enter New Item.

Data dictionary

Name	Data Type	Length	Validation	Example Data	Comment
ItemID	Integer	1-435	Range	184	-
ItemName	Text	5 - 40 Characters	Length	Mac Pro	-
Value	Real	2-5 figures	Range	1,300	-
Quantity	Integer	1-150	Range	8	-
Total Value	Real	2-8 figures	Range	133,204.86	
OnLoan	Boolean	-	Status Check	True	If an item is on loan or not
LoanRate	Real	1-3 figures	Range	75	
LoanStartDate	Date	-	Format	25/09/2014	
LoanEndDate	Date	-	Format	27/09/2014	
LoanTime	Integer	-	Date Range	7 Days	
LoanCost	Real	1-4 Integers	Range	250	
LastTest	Date	-	Format	01/10/2014	
NextTest	Date	-	Format	20/10/2014	Calculated 12 months from LastTest date

Figure 1.12: Data Dictionary.

Volumetrics

I have chosen to start off with only 20 Item Records along with 20 Loan Records and 20 PAT Test Records. In total there will be 60 Records. I have chosen this number of records as my Client and I had previously agreed that this would be a suitable number of records to start with in order for him to get used to the system and train up other colleagues to know how to use it also. This can be increased as time goes by.

The Item Records Database, Loan Records Database and the PAT Test Records Database will store 18 fields of combined data. Each field should take up 1KB of hard disk space. With this the required initial storage space will be:

18KB * 60 = 1080KB

1080KB / 1024 = 1.05MB

If the rest of database managemnent system took up 28MB, the client would need 19.05MB of space for 60 records, with 18 fields of data

1.2   Objectives

1.2.1   General Objectives

- Easily understandable layout and structure for records.
- Easy structure for input and outputs.
- Easy viewing of records

1.2.2   Specific Objectives

Record viewing:

- Clear labels for data attributes.
- Next and Preious record buttons.
- Edit button so data cannot be changed accidentally.
- Submit button to save data changes (if any) to the current record.

- First and Last record buttons to jump to respective record.

Data input:

- Data fields become editable

- Drop down selection for location selection

- Changes saved immediately after editng has finished (ie submit button pressed)

Data output:

- Print button and functionality

- Export records to PDF

- Print/Export a batch of records to PDF

- Email notifications when new item is entered into database or an item is updated, the details and who entered/updated.

1.2.3   Core Objectives

- Viewing of Item/Loan/PAT-test Records

- Item/Loan/PAT-test data input

- Item/Loan/PAT-test data editing

- Sending of Loan Invoices

1.2.4   Other Objectives

- Printing records/invoices
- Exporting records/invoices to PDF

1.3 ER Diagrams and Descriptions

1.3.1 ER Diagram
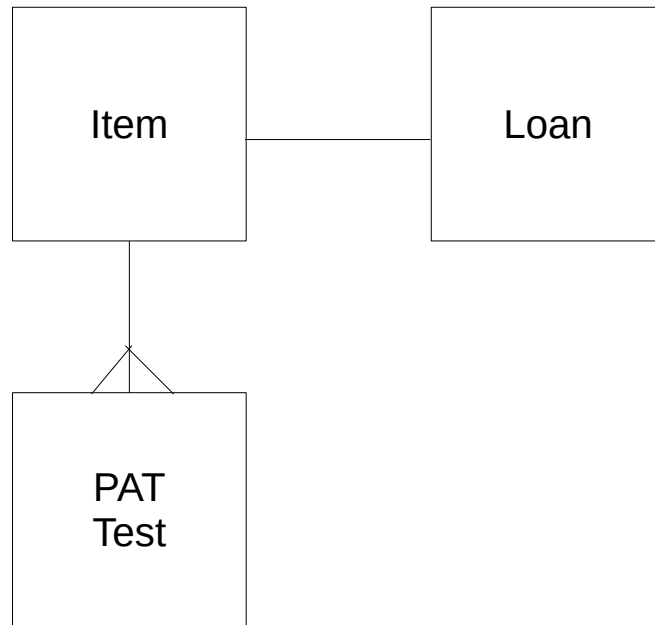
Figure 1.13: ER Diagrams.

1.3.2   Entity Descriptions

Item(<u>ItemID</u>, Name, Location, Value, Quantity, SubTotal, OnLoan,
PATNeeded)

Loan(<u>LoanID</u>, *ItemID*, *ItemName*, LoanRate, LoanStartDate, LoanEndDate,
LoanTime, LoanCost)

PATTest(<u>TestID</u>, *ItemID*, *ItemName*, LastTest, NextTest)

1.4   Object Analysis

1.4.1   Object Listing

- Client
- Item
- Loan
- PAT test

1.4.2   Relationship diagrams
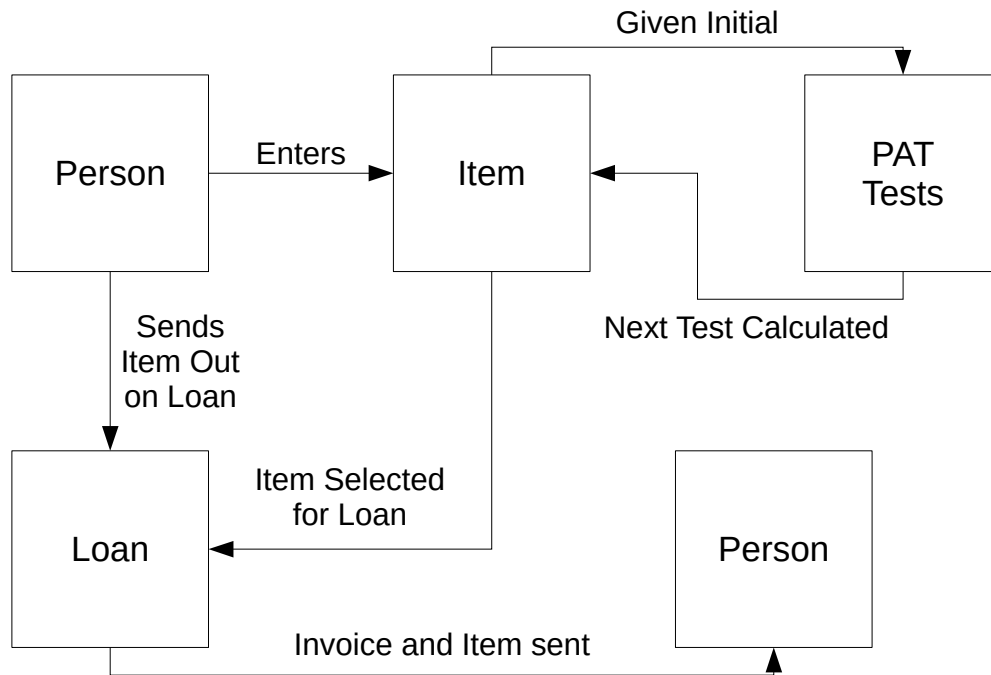
Figure 1.14: Relatioship Diagram.

1.4.3 Class definitions
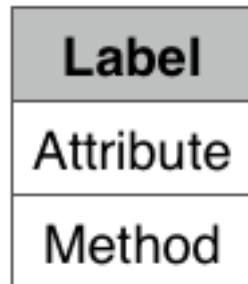
Figure 1.15: Class Diagram Key.
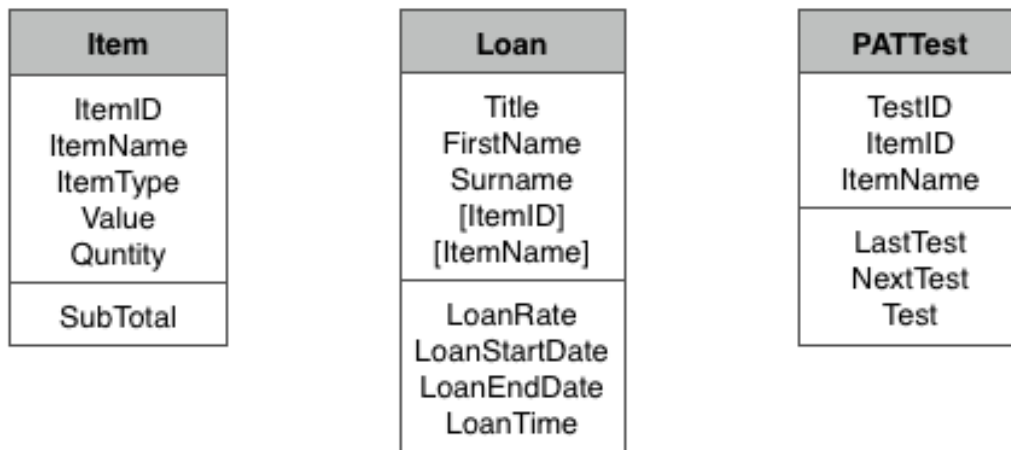
Figure 1.16: Class Diagrams.

1.5 Other Abstractions and Graphs

1.6 Constraints

1.6.1 Hardware

Presently, Josh uses a custom built, 2008 MacPro Desktop Computer. This is primarily used as a file server for images, audio and video files a well as a backup for his current work desktop. My system will need to be compatible with this system.

Computer Specifications:

- 2x 2.8 GHz Quad-Core Intel®Xeon™Processor
- ATI Radeon HD 2600 XT 256MB Graphics Card
- 661-4449 Apple Mac Pro A1186 Motherboard
- 16.00GB DDR3 RAM
- 1TB SATA Disk-Drive
- 6TB RAID Storage
- Apple SuperDrive
- 15" LG E1942 LCD Display. 1280 x 720 pixels

The proposed system should have little to no impact on this machine as the processing power and memory that can be disipated by the computer, greatly excedes the requirements for the proposed system.

There are, however, a few hard constraints that will have to be considered. One of which is the resolution of the display. The proposed system will have to be designed and implemented by taking this into account, otherwise the system may not fit the screen size appropriately.

One other constraint of the computer to be used is that it is a desktop computer. This means that the system is only accessible where Josh chooses to have the computer based in his place of work, as the computer is not portable. In addition to this, the computer requires a constant supply of

power in order to opperate as there is not internal battery.

1.6.2   Software

Josh has told me that there little restriction as to what software can and can't be stalled. The only restriction in place is that I don't install pirated, illegal or damaging software on the machine as it is part of an inter network of systems and can be accessed by other computers. The current operating system in place is Apples OSX 10.8 (Mountain Lion). Josh wishes to update the software sometime in the near future to OSX 10.9 (Mavericks) and possibly update to OSX 10.10 (Yosemite) when a stable version has been released within the next 3-5 months.

1.6.3   Time

Josh has said that there is no deadline requirement for the proposed system to be in place and doesn't need it until I have finished implementing it. The only deadline I need to meet is the project deadline set by my Computing course leader. This is Friday 13 [th] February 2014.

1.6.4   User Knowledge

Josh posses a qualification in A level Media studies as well as 2 years use of comuters during his degree. He has substantial understanding of how to use computers as his job requires he uses one most of the time. Josh also has required knowledge of how to use many variaties of applications. He uses Adobe Creative Suite for most of his job as he designs various forms of media. He also has knowledge of Apple's Final Cut Pro application as well as many others.

When designing and implementing the proposed system, Josh's experience with computers will have to be considered. Josh tends to use the internet browser Google Chrome for all his web-browsing and research as well as a third party mail application called. By designing the system similarly to these applications, it should make it easier to understand how the system works and get used to using it a lot faster than it would if the system had a

primitive design.

There will also be a full manual included to aid Josh with learning and understaning the familiar interface, the functionality of the new system and how to use certain features.

1.6.5   Access restrictions

The proposed system is primarily to be accessed by Josh himself. However, he can see it being an advantage if other people had access to the system.

For this reason, we have agreed that having the database password protected is the best way fo Josh to control who can do what with the data. He will be able to distribute usernames and passwords to other colleagues who he feels should have different access levels (Admin or standard). This reduces the risk of records being deleted that shouldn't have been. It also allows Josh to monitor who has entered new data, or updated existing data.

Users should be able to change their passwords to a more memorable one when they log into the system for the first time.

1.7   Limitations

1.7.1   Areas which will not be included in computerisation

Initial buying of new items will not be included in the computerisation as this is still done either in person or over the world wide web. Similarly, intial sales of items will not be included in the computerisation, it will only be once the item has been bought/sold that the data will be updated to coincide with the quantity changes.

1.7.2   Areas considered for future computerisation

The process of an external client of Josh's buying and receiving an item could be included in the system as it would be easier to produce an invoice

and quote sheet from the data already in the system rather than query the database and hand draft an email to be sent. The system could be included if I have time at the end and have the relevant knowledge in order to accomplish the task

1.8 Solutions

1.8.1 Alternative solutions

Custom made database:

Advantages

- No need to install addidtional software, only a simple database management system such as "Microsoft Access" or "Filemaker".

Disadvantages

- Database managment systems often cost a substantial amount of money for a license.

Web based application:

Advantages

- Easily accessible by other users. Doesn't rely on one machine.

- Can have 'Cloud based' storage of files.

- More than one user can be logged on at a time.

Disadvantages

- Website or server hosting can be expensive.

- More advanced security methods will be required due to the system being constantly online and therefore vulnerable to attack.

- Better networking knowledge required to compensate for the security implications and risks.

Terminal or Command based application:

Advantages

- More power efficient as it isn't graphics heavy, much easier to desgin as the interface is just text.

- Fast efficient opperation provided the client has knowledge of terminal and shell commands.

Disadvantages

- Careful error handeling needed as the user could enter any known/valid command.

- Training is required so that the client knows what commands to use when.

- There are often commands that the client don't know about that could potentially corrupt his computer.

Python destop application with a GUI:

Advantages

- Designed and layout can be client specific.

- Minimal error with radio buttons and other widgets.

- Easy to understand layout as data can be formatted to fit the clients requirements.

- Easy to visualise what is happening with graphs and tables.

Disadvantages

- More time needed to build the interface and sql database compaired to a command based application.

- More resources needed from the computer for graphical visualistaion and database storage

- Programming the graphical interface could prove a difficult task

---

1.8.2   Justification of chosen solution

I have chosen to use the *'Python Desktop Application with a GUI'* solution.

These are my reasons:

- The application takes up no physical space apart from the computer it is installed on.

- I already have the required language knowledge needed to program a database and a GUI in Python

- Using a custom made desktop application is faster for Josh to manage his inventory than the current spreadsheet based system.

- Backup can be made and data can be restored easily in the event of corruption or unresolvable data loss

Chapter 2

Design

2.1 Overall System Design

2.1.1 Short description of the main parts of the system

2.1.2 System flowcharts showing an overview of the complete system

2.2 User Interface Designs

2.3 Program Structure

2.3.1 Top-down design structure charts

2.3.2 Algorithms in pseudo-code for each data transformation process

2.3.3 Object Diagrams

2.3.4 Class Definitions

2.4 Prototyping

2.5 Definition of Data Requirements

2.5.1 Identification of all data input items

2.5.2 Identification of all data output items

2.9.1 Outline Plan

Test Series	Purpose of Test Series	Testing Strategy	Strategy Rationale
Example	Example	Example	Example

2.9.2 Detailed Plan

Test Series	Purpose of Test	Test Description	Test Data	Test Data Type (Normal/ Erroneous/ Boundary)	Expected Result	Actual Result	Evidence
Example	Example	Example	Example	Example	Example	Example	Example

Chapter 3

Testing

3.1 Test Plan

3.1.1 Original Outline Plan

Test Series	Purpose of Test Series	Testing Strategy	Strategy Rationale
Example	Example	Example	Example

3.1.2 Changes to Outline Plan

Test Series	Purpose of Test Series	Testing Strategy	Strategy Rationale
Example	Example	Example	Example

3.1.3 Original Detailed Plan

Test Series	Purpose of Test	Test Description	Test Data	Test Data Type (Normal/ Erroneous/ Boundary)	Expected Result	Actual Result	Evidence
Example	Example	Example	Example	Example	Example	Example	Example

3.1.4 Changes to Detailed Plan

Test Series	Purpose of Test	Test Description	Test Data	Test Data Type (Normal/ Erroneous/ Boundary)	Expected Result	Actual Result	Evidence
Example	Example	Example	Example	Example	Example	Example	Example

3.2 Test Data

3.2.1 Original Test Data

3.2.2 Changes to Test Data

3.3 Annotated Samples

3.3.1 Actual Results

3.3.2 Evidence

3.4   Evaluation

3.4.1   Approach to Testing

3.4.2   Problems Encountered

3.4.3   Strengths of Testing

3.4.4   Weaknesses of Testing

3.4.5   Reliability of Application

3.4.6   Robustness of Application

Chapter 4

System Maintenance

4.1 Environment

4.1.1 Software

4.1.2 Usage Explanation

4.1.3 Features Used

4.2 System Overview

4.2.1 System Component

4.3 Code Structure

4.3.1 Particular Code Section

4.4 Variable Listing

4.5 System Evidence

4.5.1 User Interface

4.5.2 ER Diagram

4.5.3 Database Table Views

4.5.4 Database SQL

4.10.1   Module 1

Chapter 5

User Manual

5.1   Introduction

5.2   Installation

5.2.1   Prerequisite Installation

Installing Python

Installing PyQt

Etc.

5.2.2   System Installation

5.2.3   Running the System

5.3   Tutorial

5.3.1   Introduction

5.3.2   Assumptions

5.3.3   Tutorial Questions

Question 1

43

Question 2

5.3.4   Saving

Chapter 6

Evaluation

6.1   Customer Requirements

6.1.1   Objective Evaluation

6.2   Effectiveness

6.2.1   Objective Evaluation

6.3   Learnability

6.4   Usability

6.5   Maintainability

6.6   Suggestions for Improvement

6.7   End User Evidence

6.7.1   Questionnaires

6.7.2   Graphs

6.7.3   Written Statements