

Integrating non-ontological Resources in Semantic Based Investigative Systems

Paul Fairley

076239087

A Project Dissertation submitted in partial fulfilment of the regulations governing the award of the degree of M.Sc. in Software Engineering, University of Sunderland 2011.

Project Supervisor: Dr Albert Bokma.

Abstract

The amount of data that companies have to deal with on a daily basis has grown dramatically over the last few years. The Durham Constabulary is no different. When investigating a crime they must assess all documents that could prove or disprove whether or not a potential suspect is guilty or not.

This project is in relation to the Economic Crime Unit who deals with a lot of information sent to them by third party companies. Following receiving the documents the unit assess them, and they are held in isolation with no quick way to re-assess the data.

This is where this project comes in.

Acknowledgments

I would firstly like to thank Dr. Albert Bokma for he continued support and assistance he has given throughout the project.

I would also like to thank the client for the project, Dave Sampson and his team at the Durham Constabulary, who provided the project to work with.

And finally I would like to thank my family and friends who have given me support for the duration of the project.

Table of Contents

Abstract	2
Acknowledgments.....	3
Table Of Figures.....	6
1 Introduction	8
1.1 Project Overview.....	8
1.2 Project Aim.....	9
1.3 Research Focus	10
1.4 Project Objectives	11
1.5 Limitations.....	12
1.6 Structure of Chapters	13
2 Analysis of Current Problem	14
2.1 Introduction.....	14
2.2 The Existing Problem	15
2.3 Requirements.....	16
2.4 Requirements Analysis.....	17
2.5 Current System	18
2.6 Conclusion	21
3 Literature Review	23
3.1 Introduction.....	23
3.2 Web Scraping.....	24
3.3 XML	25
3.3.1 Querying XML	26
3.3.1.1 Raw Data.....	26
3.3.1.2 Separating Fields.....	27
3.3.1.3 Grouping Fields.....	28
3.3.1.4 Naming Fields	29
3.3.1.5 Structural Map Of Data	31
3.3.1.6 Meaning.....	34
3.3.2 XML Data	34
3.3.2.1 Structured Data.....	34
3.3.2.2 Unstructured Data	35
3.4 Semantic Web Technologies	35
3.4.1 RDF	36
3.4.1.1 Ontologies in RDF	39
3.4.2 OWL.....	41
3.4.2.1 Ontologies in OWL.....	41
3.4.2.2 OWL 2 Prefuse	42
3.4.3 Query Languages.	42
3.4.4 Jena Framework	44
3.4.5 Toolkits.....	45
3.5 Navigability	46
3.6 Conclusion	46
4 Project Process.....	48
4.1 Introduction.....	48
4.2 Processes	48
4.3 Conclusion	53
5 Proposed Solution.....	54
5.1 Introduction.....	54

5.2	Prototype Design.....	54
5.2.1	Introduction	54
5.2.2	Overview	55
5.2.3	Design	55
5.2.4	System Architecture	61
5.3	Implementation.....	62
5.4	Screen Scraping.....	62
5.4.1	XML Document.....	63
5.4.2	Database implementation	63
5.4.3	Populating the Database.....	65
5.5	Conclusion	65
6	Evaluation of Prototype	67
6.1	Introduction.....	67
6.2	Methods of Evaluation Used	67
6.3	Client Feedback	69
6.4	Evaluation of the Functionality of the System.....	70
6.5	Evaluation of Libraries used	71
6.6	Further Development.....	71
6.7	Conclusion	72
7	Project Evaluation.....	74
7.1	Introduction.....	74
7.2	Evaluation of Selected Project Process.....	74
7.3	Evaluation of Project Control and Progress.....	75
7.4	Evaluation of Literature Review	78
7.5	Personal Evaluation	79
7.6	Evaluation of Objectives Completed.....	81
7.7	Conclusion	82
8	Conclusion and Recommendations	83
8.1	Introduction.....	83
8.2	Overall Conclusions.....	83
8.3	Recommendations.....	84
8.4	Final conclusion	85
	References	86
	Appendices	90
	Appendix 1 - Project Proposal.....	90
	Appendix 2 - Project Preparation Diary.....	93
	Appendix 3 - Terms of Reference	94
	Appendix 4 - Project Schedule	100
	Appendix 5 - Gantt Chart	104
	Appendix 6 - (up to date) Gantt Chart.....	106
	Appendix 7 - Risk Analysis Documents	108
	Appendix 8 - System Use Cases.....	117
	Appendix 9 - System Class Diagram.....	118
	Appendix 10 - System Designs.....	119
	Appendix 11 - Test Plan	122
	Appendix 12 - Document 'A' - HTML code	123
	Appendix 13 - XML Document	124
	Appendix 14 - Document 'A' - Screenshots	125
	Appendix 15 - Learning Logs.....	126
	Appendix 16 - Formal Meetings with Supervisor.....	139

Table Of Figures

Figure 2.1 – Opening Current System.....	18
Figure 2.2 – Import PDF	19
Figure 2.3 – Current System in Use	21
Figure 2.4 – Adding Information	21
Figure 3.1 –Example of RAW Data	26
Figure 3.2 – Example of separated data	27
Figure 3.3 – Grouping Fields.....	28
Figure 3.4 – Fields grouped and named	30
Figure 3.5 –Example of DTD.....	31
Figure 3.6 –Example of XML Schema.....	33
Figure 3.7 – The Markup Language Pyramid	36
Figure 3.8 – A graphical RDF statement	38
Figure 3.9 – Example of RDQL query	43
Figure 3.10 - Serql query.....	44
Figure 4.1 – TDD Diagram.....	49
Figure 4.2 – Waterfall Method.....	51
Figure 4.3 – Prototyping Method.....	52
Figure 5.1 – Details of suspect	57
Figure 5.2 – SQL code for creating table	58
Figure 5.3 – table structure for ‘contact_details’ table.....	58
Figure 5.4 – Credit Record	59
Figure 5.5 – Relationship View.....	60
Figure 5.6 – Linking foreign keys.....	60

Figure 5.7 – options for foreign keys in database	60
Figure 5.8 – system architecture	61
Figure 5.9 – tables in database	64
Figure 5.10 – Table print view	64
Figure 5.11 – XML code for suspects name	67
Figure 7.1 – Schedule	76
Figure 7.2 – Gantt chart	77

1 Introduction

The amount of information that companies have to deal with on a daily basis has grown dramatically over the last number of years, which has meant that the storage capabilities that are needed within organisations have grown. With this rise it has meant that more data is now needed to be accessed as well as stored. The size of storage has grown considerably, yet there has been little advancement in the way that said data can be queried. It is believed that 80% of data is currently buried and not in facilities such as a data warehouses, "A warehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process" (Inmon, W.H., 2005) making it harder to query (Tseng and Chou, 2006). With this said, it is becoming harder for established businesses such as the police to query all of the data that they go through on a daily basis, which causes an increase in cost for both time and resources to sort through all of the data to find the information that is needed.

1.1 Project Overview

The project is in relation to the Economic Crime Department, which is responsible for fraud investigations within the Durham Constabulary. The Durham Constabulary is a regional police force that is responsible for the County of Durham. The department has to deal with a growing number of complex cases, which involve a large amount of evidence to be investigated, often running into tens of thousands of documents and records which considerably slow the process of preparing witness statements to be presented at court. The police have an obligation of going through all the evidence, not

just to find evidence for the alleged crime but also to ensure that any evidence that may disprove their case is also disclosed to the courts and the defence. As the data is often complex and can involve a number of offences and a number of individuals. The investigation can take more than a year to complete and has a high risk of collapsing in court if the court, and the jury cannot reliably comprehend the evidence laid before them.

The police have systems to help them manage investigations which adequately help in keeping track of individuals and vital data associated with them, as well as keeping a searchable inventory of documents and exhibits these tools do not help the investigative process or the sense-making of these collections. In financial investigations in particular it is often important to build a complete picture of individuals and their activity. The police use information sources for the financial involvement of individuals but have no way of linking these to the investigation and inspecting the totality of the results.

1.2 Project Aim

The project aims to help in integrating records on the financial involvement of individuals into an existing system that is used to conceptualise navigating of exhibits. The constabulary currently receives the data from external agencies in an electronic format and saved as an HTML (HyperText Markup Language) file. Once the HTML file is received, the constabulary saves it on an ad-hoc basis, which is in isolation and is not easily accessible for the unit to re-evaluate or cross-reference it in the future.

Further to this where multiple records exist there is currently no system to visualise the linkage between the relevant information.

A previous project developed a 'Visual Document Management System', which is currently in use with the constabulary; the client has informed the developer that the visualisation of documents should link to the visual system already in use.

1.3 Research Focus

To gain a better understanding of the current problem, as well as the best way to address it a literature review was carried out. This allowed research into various different aspects that could impact on the development of the planned prototype, which included the technologies that are currently available as well as the system that is currently in place, but more importantly the literature review will investigate the semantic web, this is due to the aspects that are used in the current system, which we will look into in a later chapter.

Research into the semantic web will begin with an investigation into the main technologies 'RDF', 'OWL' and 'SPARQL', following a brief introduction into the three technologies, the chapter will also discuss aspects that will be essential to the development of the prototype which included screen scraping. The investigation then lead into non-ontological resources, particularly looking into the Jena Framework as well as the NeOn toolkit, which is ontology based "Engineering Environment" which was developed as a part of the NeOn Project.

The final aspect that was investigated was an extension of work that has been completed on the MSc project and the developer will look into 'usability'. This investigation will look into the 'usability' of the current system as well aspect which can improve the 'usability'

Which leads to the question: **How important are Semantic Web Technologies in today's web based systems?**

1.4 Project Objectives

Throughout the duration of the project there were many different objectives, which the developer hoped to achieve. The first of which was to liaise with the client (Dave Sampson) at the Durham Constabulary to gain a better understanding of the current problem, as well as to gather information on how they want the problem addressing. This enabled the developer to gain a better understanding of the best way to incorporate the prototype system that has been developed into the current system that the Constabulary is working with.

Below shows the main objectives that the developer plans to get from the project as a whole.

- ✓ Research into Semantic Web Technologies.
 - ✚ Research into the different technologies
 - ✚ Research into the frameworks available
 - ✚ Compare and contrast the technologies and come to a decision on the best technology to use.
- ✓ Determine clients Requirements
 - ✚ Meet with client and discuss the problem
 - ✚ Following this meeting generate requirements, and the proposed solution.
 - ✚ Meet client as discuss proposed solution

- ✓ Produce a prototype system
 - ✚ Following the meeting with client and finalising solutions, it is essential to thoroughly plan out the proposed system to ensure all aspects that are needed will be covered.
 - ✚ Define a development life cycle to ensure then project will me a success
- ✓ Evaluate the success of the deliverables
 - ✚ Create test session with potential users to ensure the prototype meets all of the requirements stated.
- ✓ Critically evaluate the project as a whole with reference to terms of reference
 - ✚ Evaluate with use of the terms of reference weather the project was successful, what went well, what went not so well, and what would be changed... if anything
- ✓ Write dissertation
 - ✚ Write a flowing document that is a high standard

1.5 Limitations

The project has many limitations upon it, from the beginning of the project through to the completion. The main obstacle that had to be overcome was the disclosure of sensitive information that the developer would have to deal with during the gathering of requirements, development and finally the testing of the prototype.

Further to this with the Durham Constabulary stating there was no budget for the project, the developer was limited to the technologies, which were freeware/open source for the development of the prototype.

1.6 Structure of Chapters

This dissertation is arranged into eight chapters, which walk through the project from start to the completion of the project. This chapter is an introduction to the project, and provides information of the client. Chapter two gives an insight to the current problem that is faced by the constabulary. After analysing the problem that is currently faced, a literature review will be then carried out.

Chapter three, the literature review, will look into the chosen topic 'Semantic Web Technologies' as well as other aspects, such as the current system. Further to this an investigation into usability will be conducted, these topics would allow the developer to gain a better understanding of the best way to go about solving the problem that is currently faced by the constabulary. Following this investigation the developer will discuss how they believe the findings can be implemented on the prototype system they are going to develop. Following this investigation the document will then move on and talk about the various different project life cycles that can be used in the development of a project.

Chapter five goes into the developer's proposed solution to the problem the constabulary is currently faced with. This chapter includes how the developer gathered the requirements of the constabulary as well as how they constructed the prototype from design through to implementation.

Following the development of the prototype system the dissertation then leads into an evaluation of the prototype, following the evaluation of the prototype can evaluate their,

from the system, which includes how they worked as well as how the project control documents were used through the project lifecycle and how they aided progress.

The final chapter of the dissertation concludes the project, and allows the developer to make conclusions on how they felt the project went, as well as to comment on whether the aims of the project have been completed, and if not why not. Further to this the developer will make recommendations for future work to be carried out to improve the prototype.

2 Analysis of Current Problem

2.1 Introduction

It is becoming more and more vital to be able to query data from the thousands of documents that the Durham constabulary have to deal with on a daily basis, which can lead to many different problems in organisations that are trying to save both time and money. There are many different ways in which money can be saved in such organisations, and this is what the development team proposes.

This chapter is used to completely analyse the current problem that is faced by the Economic Crime Department, including the processes that were carried out to gain a better understanding of the problem. Once the client had conveyed the current problem they then gave a set of requirements that they needed, and a requirements specification was written.

Following the generation of the requirements specification the developer then defined aspects that must be incorporated into the system. The main aspect that was

investigated was how the users would navigate through the system. Finally, the developer considered the design of the prototype, the structure of the databases that was planned be used, as well as the frameworks and libraries that could be incorporated within the system.

2.2 The Existing Problem

When investigating a case the Economic Crime Department receive credit reports from third party sources. These reports contain sensitive data, which is received in an electronic format. When received the constabulary assess these reports and store them on an ad-hoc basis, in isolation with no easily accessible option for re-evaluation or cross-referencing. The constabulary currently have no way to assess multiple records within a field; there is currently no system in place for visualisation and linkage of relevant data.

This is where the development team are planning to aid the constabulary. We have seen that the current system does not allow for multiple records to be incorporated into the system, and when these reports are received from the external agencies there is a chance that suspects will have more than one entry such as, more than one bank account, or more than one previous address. This information could be vital so an investigation, as it may show where money could have changed hands. Further to this suspects may have multiple accounts, which will aid an investigation, and allow the Economic Crime Department to track money between accounts.

2.3 Requirements

The system requirements were gathered during an initial meeting with the client for the project in March where the developer and client spoke about the various different aspects of the system as well as how the current system was being used. Following this initial meeting the developer created draft version of the Terms of Reference (Appendix 3), which allowed the developer to document what the proposed prototype would incorporate into the current system, as well as an overview of the product that will be delivered at the end of the project.

The main requirements that were specified by the client for the prototype system were as follows:

- ✓ The development of a 'low-cost' solution to the problem, so that the Durham Constabulary can reduce the amount of time used in processing documents and access stored documents in a proactive systematic way.
- ✓ That the technologies used to run the system be either open source or ones, which are already present within the organisation.
- ✓ Visualisation of data may be built upon the platform designed by A. Spencer

Further to this the developer also had to investigate functional requirements for the prototype system. The proposed functional requirements that the client thinks should be added to the current system that will allow for the aspects in which the constabulary are looking for in the proposed system. The first function that must be incorporated into the current system is to allow for the data from the documents that are received from the third-party sites to be read into the system, which will allow for the data to be shown in the current system. For this to happen within the system there are various different steps that must be completed, the most important of which is to ensure that

data is held in a database so that the data from the reports can be re-assessed at a later date.

2.4 Requirements Analysis

In order for the developer to continue onto the design stage for the prototype, it was essential for the developer to analyse the information given to them by the client. Following the analysis of the requirements, the developer documented what was expected via a UML (Unified Modelling Language(Kimmel, P. 2005)) diagram. The developer came to the conclusion that using UML diagrams would benefit both themselves and the client, as this meant that in further meetings the developer could show the client how the system would work for the major aspects of the system.

UML is a language that unifies the best engineering practices together for a modelling system . (Alhir, S,S.1998). The diagrams that are generated via the UML model can aid both the developer as well as the client, however not everyone that has researched into the field shares the same views. These views extend to various different aspects, the main of which is the extra time that is required throughout the development of a project.

However of recent years authors such as (Fowler, M.2000), have challenged the idea whether design tools such as UML maybe 'dead', however in his paper, is design dead?, he comes to the conclusion of 'NO', he raises several valid points, the most important of which being that the primary use for design documents such as UML is used primarily for communication, using UML diagrams also allows a development team to map

aspects of a system that are known, this is significant as it allows for classes to be labelled, even if all methods are not known.

2.5 Current System

The constabulary has stipulated that the current system should be used as a basis for the system. The current system was developed by a student doing a MSC a few years ago, and was developed in the Java programming language. The current system is used to take 'non-structured' data such as information from documents such as letters and emails to extract information from and use to link 'suspects'.

When opening the current system the user is asked which file they would like to query in the system. And is greeted with the following screen, this allows the user to import a file, the developer used PDF (Portable Document Format) documents as the primary document used.

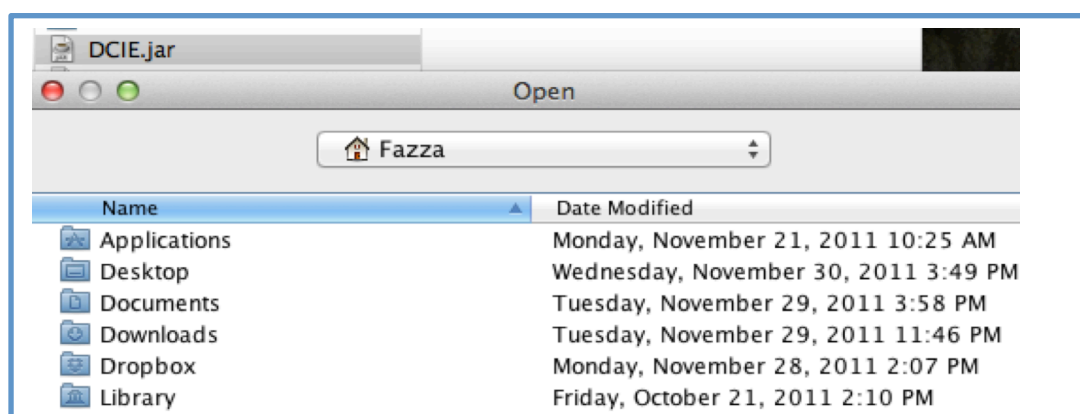


Figure 2.1 – opening the current system

After opening the current system it is still possible to import a PDF, by clicking the “import PDF” which shows an imported PDF with the information that is contained in the PDF file.

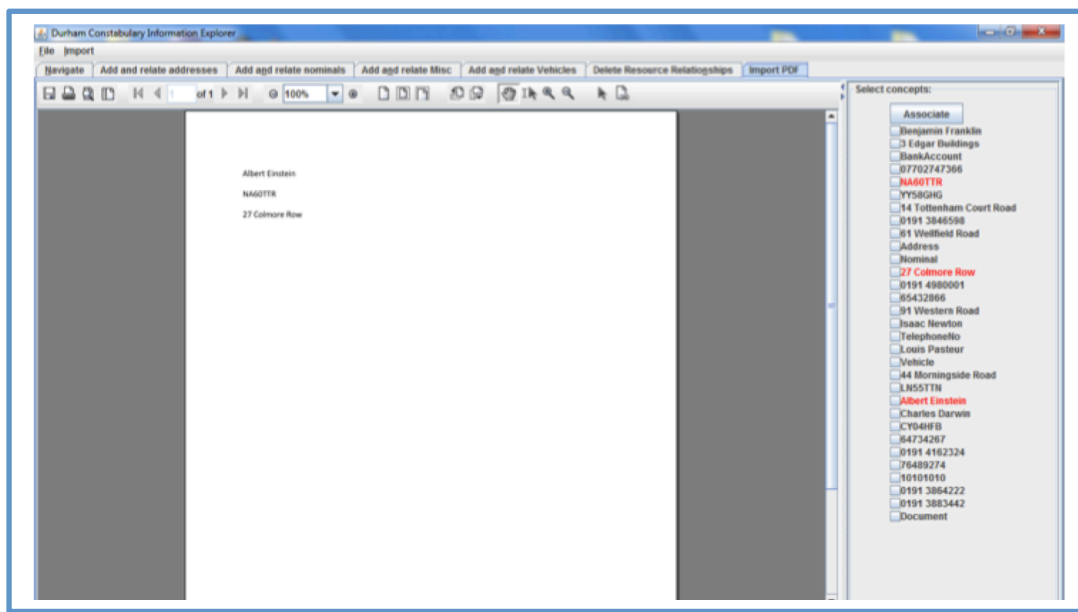


Figure 2.2 – Import PDF

The current system allows for links to be made within investigations, this allows for the constabulary to link suspects within a crime investigation and also allows them to find out which other people are associated with the suspect, meaning that people that the police did not first think were associated with the crime they are investigating may become potential suspects. Another feature that is key within the current system is the option to remove suspects that they are no longer interested in investigating as part of the investigation.

Figure 2.3 shows the current system incorporates a navigation tab, which allows the constabulary to the links between suspects and resources, furthermore provides a small description of who they are related when the user hovers over the link.

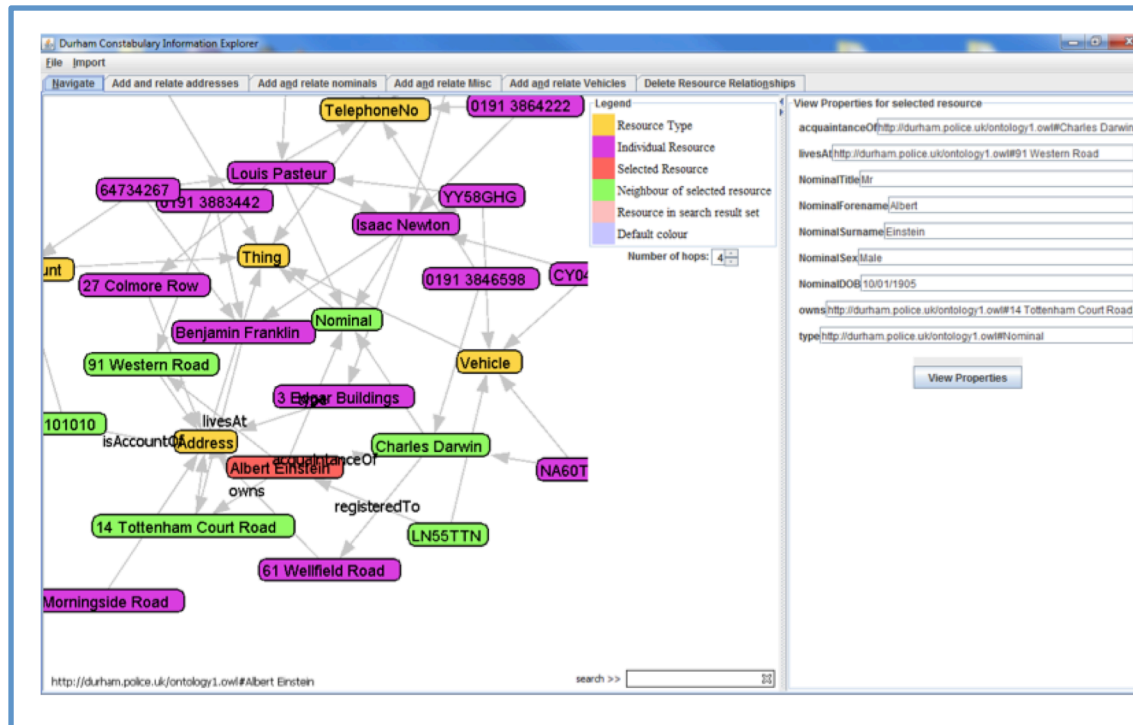


Figure 2.3 - Current system in use.

As Figure 2.3 shows the link between people, vehicles, addresses and phone numbers. The current system also shows the difference between each of the modules within the system is colour coded, and shown in the legend on the right hand side of the main window. The example was taken from the documentation that was provided with the current system, the reason that the developer chose to you this image is that the example suspects, as this meant that sensitive information would not be exposed.

The constabulary can also add further information into the current system by clicking on the “add and relate misc” tab. This allows further information to be added to the main content of the system. Figure 2.4 shows the tab described;

The screenshot displays the 'Durham Constabulary Information Explorer' web application. The interface features a top navigation bar with tabs: 'Navigate', 'Add and relate addresses', 'Add and relate nominals', 'Add and relate misc' (which is the active tab), 'Add and relate Vehicles', and 'Delete Resource Relationships'. The main content area is divided into two panels. The left panel contains two sections: 'Add a Telephone No' with a text input for 'Enter Telephone No:' and an 'Add Telephone No' button; and 'Add a Bank Account' with text inputs for 'Enter Bank Account No:', 'Enter Bank Sort No:', and 'Enter Bank Name:', along with an 'Add Bank Account' button. The right panel contains two sections: 'Relate Misc' with dropdowns for 'Select Telephone No:' (showing '0191 4980001'), 'Select Nominal:' (showing 'Isaac Newton'), 'Select Address:' (showing '91 Western Road'), and 'Select Document:', each with a corresponding 'Add' relationship button; and 'Relate Bank Account' with similar dropdowns for 'Select Bank Account:' (showing '10101010'), 'Select Nominal:', 'Select Address:', and 'Select Document:', each with an 'Add' relationship button.

Figure 2.4 – adding Information

Yet the current system does not allow for information from the third-party companies to be incorporated into the system, this is where the development team try to aid the Durham constabulary.

2.6 Conclusion

In this chapter we looked at what the constabulary need from a new prototype system. The system must incorporate aspects that have been covered by the client in the gathering of requirements. This chapter allowed the developer to set out exact requests from the constabulary with regards to the prototype system. We also looked into UML diagrams, which will allow the developer to provide visual representations of aspects of the system, and how they will be incorporated in the system. The developer felt that this

was the best way to go about designing the system properly as it allows for classes to be added into the diagrams without all necessary methods needing to be in place.

This chapter also looked into the aspects of the current system that allow the constabulary to import documents such as PDF's, with the aid of various different API's (Application programming interface), that aid the visualisation of data. Yet, the current system only allows for the import of PDF files, and when the constabulary receive data in the form of credit reports from the third-party sites is not in the form of a PDF file so the current system must be adapted to enable the constabulary to incorporate the data that is held in the credit reports. The current system must be edited to allow for aspects to search the information that is added to the current system, such as being able to search investigation numbers, and well as associates of a potential criminal.

3 Literature Review

3.1 Introduction

Semantic web technologies have boomed over the last 3 to 4 years and have become very important to the development of all aspects of the world wide web. With so many new technologies emerging it is becoming more and more important when choosing a development language to choose a language that is going to stand the tests of time. We currently have XML (Extensible Markup Language (Goldfarb,C.F, & Prescod, P. 2000)) which is used in most aspects of world wide web, however with the emergence of these semantic technologies it is important to move with the times to ensure that the software that is developed is with the times and has longevity.

In this chapter an investigation will look into the existing technologies, starting with the HTML file that has been given to the developer, 'scrape' the data from the said document and then generate a versatile XML file so that the information can be more easily stored, as well as how to query data that is contained within an HTML document more thoroughly. Following this the investigation will extend into the semantic web technologies that are at the forefront of the semantic movement, as well as how far technology has moved from Web 2.0 (XML) to Web 3.0 (semantic technologies). Further to this it is essential to understand the best way to check the data that is contained in the field so the data can then be used to populate data within a system.

3.2 Web Scraping

In a perfect world, data would come in a structured form, which would make it easier for programmers to easily extract data, however this is not the case. When working with documents that are received from third-party companies by the constabulary which are in the form of a HTML documents. The main problem with HTML documents is that the data is not in a structured manner, and to get the best out of the documents that have been provided it will be essential to extract the data to a database. To accurately get the data from the HTML document to a database which will hold the data and allow of the constabulary to re-evaluate it is to follow several steps. The first of which is web scraping, which is also known as Web data extraction or web harvesting. Web scraping is a technique in which to extract data from a webpage. (Myllymaki 2002).

When researching into web scraping it very quickly became apparent that Hendler, J is at the forefront of the movement, however it wouldn't be advised to just look at the information that Hendler provides, further investigation led into several different authors. Scraping is become increasingly more useful within the world of data extraction as it has many advantages, the most significant of which for the development of the proposed prototype is that it allows unstructured data to be used, and analysed in a database.

When investigating web scraping it was discovered that there is a scraping 'tool' in almost every language, including various different 'tools' in the Java framework, which the current system is developed in. It seems that it wouldn't be practical not to incorporate one of these Java scraping tools into the current system, which will allow

for the data extraction. The first example that was investigated was (Goetz, B 2005) investigated how to incorporate “screen-scraping with XQuery”, and the first point that is raised of note with relation to the problem the developer is currently faced is that when using web scraping on such documents such as web pages is that they “no self-identifying structure” as well as the structure is liable to change as content is changed, which means it is often a guess as to which aspects of the webpage you want to extract data from.

(Goetz, B 2005) also raises the point of how simple (as in lines of code) it is to extract data from a webpage with the aid of the libraries (Xquery and Jtidy). The point is also raised that Xquery is mainly used to query large amounts of data, it also serves very well to query “Simple” data as well. Following this investigation the developer then looked into “Jsoup” which incorporates DOM(Document Object Model), which is a Java library for working with real-world HTML. However Jsoup would not give the type of accuracy that the developer wanted, so the developer looked into other languages which included screen scrapers and the superior parser came in the form of a PHP script by the name of “SimpleHTMLDom” which gave a better rate of extraction and better accuracy than the Java alternative.

3.3 XML

A staple point of web development for many years, although not a semantic web technology is XML (Extensible Markup Language), (Goldfarb, C. and Prescod, P. 2000) which is a Subset of the SGML(Standard Generalized Markup Language). A very versatile language that has been used extensively for the last 10 years or so it has grown stronger. The main reason to focus on XML is that documents that are sent in a digital

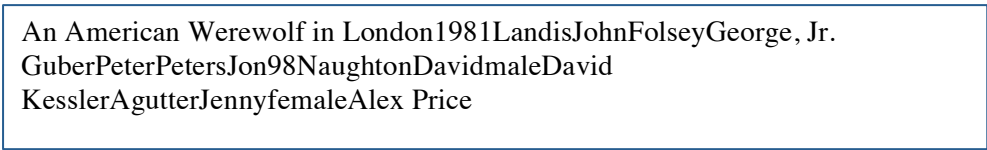
format in the HTML which has very little structure to it and the developer plans to convert the HTML to XML. Following the conversion the developer will then need to query the data that is contained in the XML files. This will allow the data to have more of a structure and enable data to be extracted accurately which will mean that the tables of data to be used will have more meaningful data.

3.3.1 Querying XML

XML is used to add a set of rules to the 'markup' of data. Markup is what creates the structure of data, which also gives a way of discussing the meaning of data, which allows for a standardisation of data. This means that data can be processed with standard applications as well as transferring data from one source to another. We will look at how XML adds 'Markup' to the data below. Examples are taken from (Melton, J. *et al* 2006). Other examples exist, however the developer believed that their examples were the best to work with from the starting point through to a cohesive XML document.

3.3.1.1 Raw Data

Figure 3.1 shows how data maybe represented if it was in its raw form. This examples shows a single record for a movie, and in its raw form tells the user very little about the data contained with in it.



```
An American Werewolf in London1981LandisJohnFolseyGeorge, Jr.  
GuberPeterPetersJon98NaughtonDavidmaleDavid  
KesslerAgutterJennyfemaleAlex Price
```

Figure 3.1 – Example of Raw data.

The data above shows information for the movie “An American Werewolf in London”, however this information can only be obtained if the user has any familiarity of the movie, show therefore a computer would not be able to gain any useful information from the raw data.

3.3.1.2 Separating Fields

The next step is to add structure to the data, this is achieved by the use of a comma to define each aspect of data, or each field.

```
An American Werewolf in London, 1981, Landis, John, Folsey, George\, Jr.,  
Guber, Peter, Peters, Jon, 98, Naughton, David, male, David  
Kessler, Agutter, Jenny, female, Alex Price
```

Figure 3.2 – Example separated data.

The example in fig 3.2 shows the same film as shown in fig 3.1, however the fields are separated with commas. Further to this we have a ‘\’ (backslash) which is used as an escape from the current comma. Another way that is used to distinguish which fields are to be accessed is via fixed length fields, of which each field is 8 bytes, this means if you require the information in the 3rd field you can simply query the 17th byte and this will give you the information required. However even with the above separated data list we still have no way of knowing which aspects of data go together, this means that we now need to find a way to group fields together.

3.3.1.3 Grouping Fields

As discussed thus far the data that has been used is not very useful to either a human or a computer, however by grouping the data we can enable the data to be read by, and understood by a computer. The way that we do this is via further use of ','(at the beginning) as well as the use of a new symbol '\$' (at the end of a field), figure 3.3 shows how this works with the film example we have used thus far.

```
,
    ,An American werewolf in London$,
    ,1981$,
    ,
    ,Landis$,
    ,John$,
    $,
    ,
    ,Foley$,
    ,George, Jr.$,
    $,
    ,
    ,Guber$,
    ,Peter$,
    $,
    ,
    ,Peters$,
    ,Jon$,
    $,
    ,98$,
    ,
    ,Agutter$,
    ,Jenny$,
    ,female$,
    ,Alex Price$,
    $,
    $,
```

Figure 3.3 – Grouping Fields

Figure 3.3 shows the extra white space that is used when grouping field, however the white space is only used so that humans can easily read what they need to form the data.

The data shows that information such as “Agutter, Jenny, female, Alex Price” all belong together and is associated with the film “An American Werewolf in London”

However the above example has a major flaw when it comes to extracting the data, there is no way to indicate that ‘1981’ is the year of the films release, but further to this there is no way to ensure that then extracting data the the year of release will not be outputted as “LandisJohn”. So with this our next step is to name the fields so that data can be extracted more easily, and with less data redundancy.

3.3.1.4 Naming Fields

From the data shown in Fig 3.3 we can work out that the film is called “An American Werewolf in London” however we have do way of knowing what ‘98’ means within the data. Further to this we don’t know who “George Folsey” is, or what his role is within the film. This means we need to name the fields, to give the computer more meaningful data to work with.

Figure 3.4 below shows how we have gone from ‘,’ and ‘\$’ in the last step, and replaced them with “<tagname>” at the beginning of the tag, and “</tagname>” at the end of the field, which in terms of the XML each field is an portion within the document we are looking at, and specifies it’s name within the document. This means that we can now query aspects of the document by the titles of the fields.

Now that we can query the fields within an XML document, if it is possible to structurally map the data then it will be more beneficial to the developer as they can pull out aspects of data within the document with ease.

```

<movie>

  <title>An American Werewolf in London</title>

  <yearReleased>1981</yearReleased>

  <director>

    <familyName>Landis</familyName>

    <givenName>John</givenName>

  </director>

  <producer>

    <familyName>Folsey</familyName>

    <givenName>George, Jr.</givenName>

    <otherNames></otherNames>

  </producer>

  <producer>

    <familyName>Guber</familyName>

    <givenName>Peter</givenName>

    <otherNames></otherNames>

  </producer>

  <producer>

    <familyName>Peters</familyName>

    <givenName>Jon</givenName>

    <otherNames></otherNames>

  </producer>

  <runningTime>98</runningTime>

  <cast>

    <familyName>Agutter</familyName>

    <givenName>Jenny</givenName>

    <maleOrFemale>female</maleOrFemale>

    <character>Alex Price</character>

  </cast>

</movie>

```

Figure 3.4 – Fields Grouped and Named.

3.3.1.5 Structural Map Of Data

As mentioned in chapter 3.2.1.4 we introduced adding 'tagname' to a field to make it easier to recognise, now we will look into how the data can be mapped structurally. Mapped data allows for data to be manipulated so that data can be retrieved from a document such as the 'given name' and 'family name'. The way that this can be achieved is either via DTD's (Document Type Definition) or via an XML Schema.

3.3.1.5.1 Document Type Definition

Document Type definition defines the elements of a document that are allowed to be used, as well as what order they are used in and where they are used. A DTD can also be used to enumerate values that are allowed from each of the different attributes within the document. A problem with using DTD is that it may have recognised aspects within the document and class them as a type ID, which means they must be unique within the XML document.

```
<!ELEMENT movie (title, yearReleased, director, producer+,  
                runningTime, cast+).>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT yearReleased (#PCDATA)>  
<!ELEMENT director (familyName, givenName, otherNames?)>  
<!ELEMENT producer (familyName, givenName, otherNames?)>  
<!ELEMENT runningTime (#PCDATA)>  
<!ELEMENT cast (familyName, givenName, otherNames?,  
                nameorFemale, character)>  
<!ELEMENT familyName (#PCDATA)>  
<!ELEMENT givenName(#PCDATA)>  
<!ELEMENT otherNames(#PCDATA)>  
<!ELEMENT maleOrFemale (#PCDATA)>  
<!ELEMENT character (#PCDATA)>
```

Figure 3.5 – example DTD

The DTD ensures that a movie must contain all of the following (title, yearReleased, director, producer+, runningTime) as well as at least one cast member. The lines that follow show the shape of the elements that are needed within the XML. Another problem with using DTD is that it contains no information on what data types will be used, for examples it doesn't specify that running time should be an integer, and title should be a sting.

3.3.1.5.2 XML Schema

As mentioned above DTD has drawbacks, which could hinder the entry of correct data into a document, or even result in incorrect information being extracted. However there is an alternative to using DTD and this comes in the form of an XML schema. The main advantage to incorporating an XML schema over a DTD is that the schema solves both of the major problems that come with a DTD, they hold data information, and more importantly they are XML documents. An XML schema has all of the aspects that a DTD uses with the advantages discussed. An example schema document for the example movie we have used thus far can be seen below.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
            elementFormDefault="qualified">
<xs:element name="familyName" type="xs:string"/>
<xs:element name="givenName" type="xs:string"/>
<xs:element name="movie">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="title" type="xs:string"/>
      <xs:element name="yearReleased">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:minInclusive value="1900"/>
            <xs:maxInclusive value="2100"/>
          </xs:restriction>
        </simpleType>
      </xs:element>
      <xs:element name="director">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="familyName"/>
            <xs:element ref="givenName"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="producer">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="familyName"/>
            <xs:element ref="givenName"/>
            <xs:element name="otherNames" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="runningTime" type="xs:integer"/>
      <xs:element name="cast" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="familyName"/>
            <xs:element ref="givenName"/>
            <xs:element name="maleOrFemale">
              <xs:simpleType>
                <xs:restriction
                  base="xs:string">

```

Figure 3.6 – example XML schema

3.3.1.6 Meaning

With the XML examples we have used in figures 3.5 and 3.6 we can learn a lot from the data that has been used. We know how to break the data into meaningful elements, each element has a meaningful name, which improves readability and makes it a lot easier to refer to the element, and we also have a schema that describes all of the rules that are necessary for the data. Which means we have achieved one goal of adding 'Meaning' to the data we are going to use.

From the movie example we have been looking at thus far we can see where aspects may be extended, an example of this could be that the element we have for "yearReleased" could be divided into sub attributes. Further to this we can also use a semantic web technology in the form of RDF which will help to define which person was the producer, as well as defining relationships.

3.3.2 XML Data

Thus far we have looked into how to query data within an XML document, however there are different types of data that we must consider, we must look into how structured data and unstructured data vary within a XML document. As discussed the current system looks at unstructured data, and gathers the information needed and adds it to the system.

3.3.2.1 Structured Data

The above examples can be classed as structured data; this is due to the data being in clearly defined chunks. Further to this the example we have looked at thus far has an obvious tree structure, and has both parent and child relationships. Other examples of

structured data include library records and patrol records. Structured data is often stored in a persistent data store, such as a database.

3.3.2.2 Unstructured Data

As we have seen in chapter 2.5 the current system that the constabulary have in place deals with unstructured data. Unstructured data is a deceptive term, as the data within any type of document has some sort of structure, this is true of documents such as letters having punctuation. However the use of XML allows for structure or formalizes the existing structure. Which makes it increasingly useful to incorporate XML into a document to ensure correct data can be pulled out of the document.

3.4 Semantic Web Technologies

As we rapidly move through Web 3.0 more semantic web technologies are immerging, however the three main technologies within the semantic web movement are RDF (Resource Description Framework), OWL (Web Ontology Language), and SPARQL, which is a query language. The investigation will primarily look at how ontologies are built within these technologies; from this it will then be possible to see how structured data can be incorporated into a semantic based system.

In the previous sections of this chapter we looked into documents in XML both structured and unstructured, as well as XML schema, however this is just the bottom of the pyramid when it comes to semantic web technologies. Below shows a markup language pyramid and shows how the technologies in XML is used together with RDF and OWL.

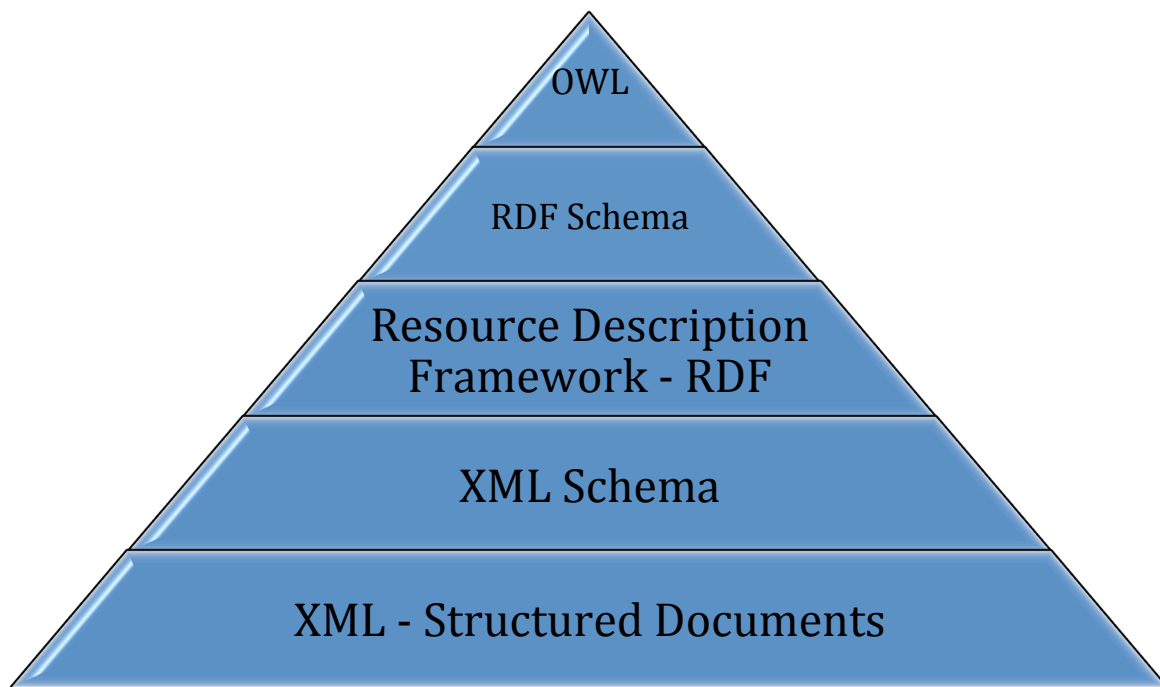


Figure 3.7 – the Markup Language Pyramid. (Alessio *et al* 2005)

3.4.1 RDF

The original idea of RDF dates back to 1990, when Tim Berners-Lee wrote the original proposal, which then led to the development of what we now know as the WWW (World Wide Web). The original proposal had “different types of links between documents” (Segaran, T. *et al* 2009), which meant that the hypertext was easier for computers to understand. However, the first official specification was published in 1999 by W3C. (Hitzler, P. 2010), this version was based entirely on representing *metadata* about Web resources. The term metadata normally refers to data, which specifies information which is given about data sets, or more simply data about data.

As the technologies have grown so has the number of tools available to deal with RDF, nearly every language has a library that is compatible with both reading and writing RDF documents nowadays. RDF has various different stores, named *triple stores*, named so due to the process that is undertaken to store and retrieve data.

The RDF model/syntax specifications as well as the RDF schema specification both extend the web standards that already exist for XML and XML schema. RDF schema are used to describe how using RDF can build upon RDF vocabularies (a collection of resources that are used as “predicates” within an RDF statement) (Alessio *et al* 2005). RDF data can be used by an XML schema which means it can be passed over the internet as a document and then parsed by an existing XML based system.

The RDF model uses declarations that are made regarding resources that are used by any URI (Uniform Resource Identifier), which can be anything that is associated with a document. The RDF model generates a triple, which links a resource to a labelled property (the predicate) to a value (the object). Below shows how this is achieved:

We can start by saying:

A thing [subject] has a property [predicate] with a specific value [object].

Or, more concisely,

[Subject] has [predicate] [object]

Or as a triple

(subject, predicate, object).

However we can also look at this from a different point of view :

A property [predicate] of a thing [subject] is a specific value [object]

Or

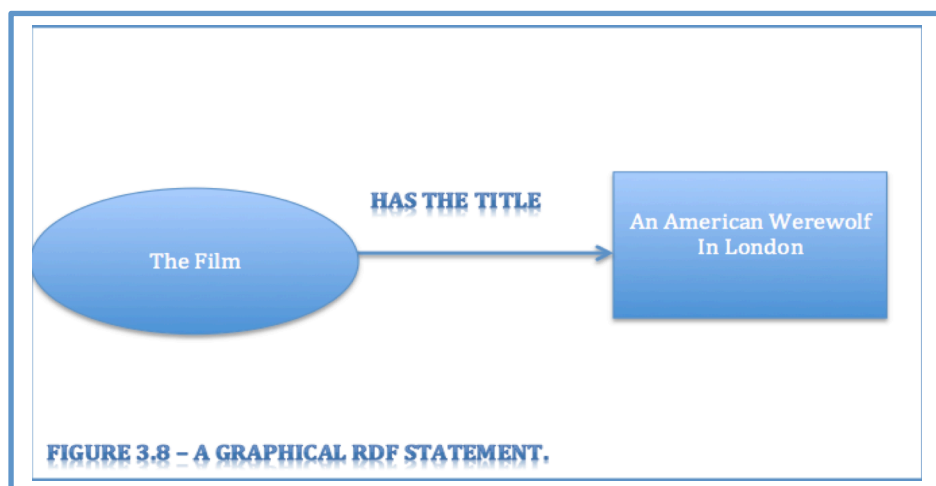
[predicate] of [subject] is [object]

Even with the changed order, the triple stays the same, as (subject, predicate, object).

We have looked at the three titles, subject, predicate and object, but what do these titles really mean in the bigger picture of an RDF primer.

- Subject – refers to a place, person or thing that a statement describes. Within a RDF resource can be anything, a document, user or product. A subject is uniquely identified by a URI.
- Predicate – is the property, a name, a city, a title etc of the *subject*. A RDF property is identified via a URI
- Object – is the value ()for the *property* of the *subject*. The value should a valid RDF data type. And as already discussed RDF supports all aspects of XML data types.

This can be represented by the example below, which links back to the example we looked at in 3.2.1 :



However there are four very important rules that need to be used when working with RDF triplets:

- ✚ Each triplets consisted of a subject, predicate and a object
- ✚ Each triple is a unique and complete fact
- ✚ Each triple is a 3-tuple that the subject that is either a uriref or a bnode.
The predicate is a uriref and the object is a uriref, bnode or literal
- ✚ Each triple can be joined to any other RDF triple, however the 2 triples still hold their own unique meaning, regardless of how complex it is.

However, once the data has been populated in these triples it is important to firstly query the data and then use the data. This is where another of the technologies come into play, as it is essential to query the data that we have. There are various different query languages that can be used, the most common of which is SPARQL, chapter 3.3 will look into SPARQL and its competitors.

3.4.1.1 Ontologies in RDF
















Now we have an understanding of the workings of RDF we will then look how to create ontologies within RDF. The first question that must be answered is ‘what is an ontology?’ Then we have to look into why using ontologies is important within semantic web systems.

What is an Ontology? – in computer science, an ontology is considered the backbone of semantic web. (De Nicola, *et al* 2009). Building an ontology displays structure and a logical complexity that is similar to the construction of a software artifact. However, this

is not the only definition of an ontology, (Gruber, T. 1993) describes an ontology as “a specification of a conceptualization”, which is the definition most authors cite.

There are many different ontologies that can be used within the semantic web, ‘<http://semanticweb.org/wiki/>’ lists the most popular ontologies, by their “swoogle hits” rating. An ontologies “swoogle hits” defines the number of hits that ontology has on the search engine swoogle, which gives an exact number of occurrences of a ‘namespace-prefix’ within a database. The main three ontologies that are used in conjunction with RDF are “Dublin Core”, “TrackBack” and “MetaVocab”. However this does not give a fair reflection on how much each of the ontologies score on the swoogle hits.

Most recently revised in 2006, ‘Dublin Core’ is by far the most popular ontology that is used in the RDF framework. With 1,364,337 swoogle hits ‘Dublin Core’ more than doubles its nearest competitor in ‘TrackBack’. A product of the Dublin Core Metadata initiative which primarily refers to a standard of semantic vocabulary, concentrates more on documents, and focuses upon an element set of metadata, and these datasets are :

 contributor	 format	 rights
 coverage	 identifier	 source
 creator	 language	 subject
 date	 publisher	 title
 description	 relation	 type

‘Dublin Core’ is very popular, sites such as Wikipedia have incorporated aspects of the ontology, such as ‘creator’ and ‘contributor’ would automatically created for Wikipedia to keep track of the history of a document as well as these this Wikipedia also incorporated ‘publisher’.

‘TrackBack’ is the next most popular ontology however it has a mere (in comparison to ‘Dublin Core’) 502,401 swoogle hits. ‘TrackBack’ is an extension of RSS versions 1.0 and 2.0, which enables statements about “trackbacks (which is the request for a link to a document has been notified)” in blogs. Finally we have ‘MetaVocab’ which was last revised in early 2002, is very similar to ‘TrackBack’ as is conceived as an extension of RSS 1.0 yet unlike ‘TrackBack’ it has been used with the Non-RDF format of RSS 2.0. However ‘MetaVocab’ is still classed as a “proposed” vocabulary, yet it is used extensively already in many different contexts.

3.4.2 OWL

As we have discussed thus far the web technology RDF, however another very important technology within semantic web is OWL (Web Ontology Language). First started in 2002 OWL is rapidly becoming the most used language within semantic web. OWL, which is a knowledge representation language for authoring ontologies, further to this OWL is branded by “formal semantics” and RDF based language. In October 2007, the W3C group began work to extend OWL with a lot of changes that were originally proposed in the OWL 1.1 member submissions.

3.4.2.1 Ontologies in OWL

As with RDF there are many ontologies that can be used with the OWL environment. Similar to RDF the ontologies that can be used are rated by their swoogle hits. As we looked at the most popular ontologies in chapter 3.3.1.1, it became apparent that there is one ‘market leader’ for OWL ontologies, and this is ‘FOAF’, however there is also ‘WOT’ and ‘SIOC’. With 619,538 swoogle hits ‘FOAF’ has a lot more hits than the other

two ontologies, yet if we compare ‘FOAF’ to RDF’s ‘Dublin Core’ it has less than half of the hits, so it appears that even though OWL is immerging as a more and more accessible language RDF still has the edge over it when it comes to ontologies that can be used. Out of the 18 ontologies that are described on *semanticweb.org* only 5 ontologies are OWL DL (a species of the OWL language that is based upon description logics) and three ontologies that are used within OWL 2. Having said this out of the top ten ontologies only 2 are OWL DL’s, so it appears that even though OWL is an immerging language the ontologies that are used today are still mainly based around the rules of RDF environment.

3.4.2.2 OWL 2 Prefuse

As we discussed in chapter 2.5 regarding the current system, the developer integrated the API, OWL 2 prefuse, which is a Java based toolkit for visualisation that is used within the current system. Having analysed the current system the developer does not believe that there is a great need to change the visualisation of the document view that is currently in place.

3.4.3 Query Languages.

Once we have data sets to work with it is then essential to be able to “query” them as well as being able to make use of the data once queried. To query any aspect of the semantic web it is essential for the query language that is compatible with the aforementioned RDF. This means that when you query languages that are based around RDF

such as OWL from the perspective of a RDF format it doesn't require any special procedures of language features. (Hebeler, J.2009)

The main query language that is used to query RDF is SPARQL (pronounced 'sparkle'). SPARQL is the W3C standard when it comes to query languages, however there are several other languages that can be used, these include languages such as RDQL(RDF Data Query Language) and SeRQL (Sesame RDF Query Language (pronounced 'circle')). The main focus will be on SPARQL as it is the standard, although we will investigate why it is indeed the standard, and why it's a better query language than the other two named above.

We will initially look into RDQL, which as described is a query language that is used with RDF, however its main purpose is to extract data from RDF Model or as it is also known a graph. RDQL has very similar syntax to SPARQL which both have similar syntax to SQL, an example query within RDQL is shown below in figure 3.9

```
SELECT ?x,?y
FROM <http://example.com/sample.rdf>
WHERE (?x,<dc:name>,?y)
USING dc for <http://www.dc.com#>
```

Figure 3.9 – example RDQL query (RDF Tutorial)

The example in Figure 3.9 of a RDQL query which looks very similar to the queries that are performed in SeRQL which can be seen below in figure 3.10

```
SELECT DISTINCT c
FROM
{c} serql:directSubClassOf {pc} rdf:type {owl:Class}
WHERE pc = owl:Thing
```

Figure 3.10 – SeRQL query (AID website)

The SWI-Prolog Semantic Web Server implements SPARQL and SeRQL on top of the SWI-Prolog Semantic Web library. (W3C SeRQL wiki). Therefore, it appears that all of the query languages that can be used are very similar, and link in one way or another.

3.4.4 Jena Framework

Jena is a framework developed with Java, which is used for building “Semantic Web” applications. The Jena framework provides a programming environment for RDF, RDFS, OWL, SPARQL and it also includes a “rule-based reference engine”. The reason that we are concentrating on the Jena framework is because it was used to develop the current system that is used by the Durham Constabulary. Within the current system the framework is being used as it allows manipulation of datasets, including RDF, N3/N-Triples and OWL, as well as using SPARQL as a query language, (Guo *et al* 2009) which allowed the previous development team to better deal with the data that they were extracting from unstructured sources.

However there are many different Semantic Web Frameworks that can be used although not all are directly compatible with Java. The biggest competitor to Jena is SCOP, which is developed via a project named DWeb, which is an acronym of Dream Web, with the primary aim of contributing to the semantic web by “offering an environment to set such of communities in the web”(Calian, M. *et al*) DWeb attempted

to build a tool to build virtual communities over the web, which could be very helpful to organisations that need to share extensive amounts of data,

3.4.5 Toolkits

Finally there are many different toolkits that can be used within a semantic web base. The main focus of our investigation is the NeOn toolkit. NeOn which is an eclipse based, open source, OWL supported toolkit. Dating back to March 2006 the NeOn project aimed to advance the use of ontologies for a large-scale semantic applications in distributed organizations. Further to this, the NeOn toolkit has a built-in OWL editor, which is used for the maintenance of semantic models, or as we know them ontologies. Currently with 20 plug-ins for the latest version 2.4, which handle a range of internal activities such as Annotation and Documentation, Modularization and Customization, Reuse, Ontology Evolution and translation (NeOn Project 2010).

Other ontology editors that can be used in conjunction with OWL, the most popular of which is the free, open source Java based “Protégé”. The editor Protégé can be used with both OWL and the afore mentioned RDF as well in XML schema, meaning that it is very versatile within semantic web technologies. However within semantic web based systems Protégé is not the only ontology editor. “SWOOP” developed by mindswap (Maryland information and network dynamics lab semantic web agents project) is another Java based ontology editor that can be used within the OWL environment, however it has now ceased development.

3.5 Navigability

Navigability is a large aspect to any system, however, when the investigation of the current system (chapter 2.5) it seems that the navigation within the system works very well, and incorporating OWL 2 Prefuse and it seems that editing the navigation that exists will cause a lot more work than incorporating the development into the system that exists. This is due to the many different types of mappings that can be used in the navigation of a system. The two most popular of which are Hyperbolic trees and cluster maps, both have very similar aspects to them, and link data cohesively to each other, however it looks like cluster maps allow for a more concise search between aspects of a system.

3.6 Conclusion

In this chapter we have looked at the semantic technologies that can be used within the development of a system. We have found out that there are many different semantic technologies that can be used in conjunction with most languages, however we have only looked into the three main technologies. There are many different technologies that can be used that we have not investigated, the literature review only gave enough time to inspect the three main technologies. The current system uses the semantic technology “OWL” so research, naturally led into other technologies linked to it. This chapter also looked into API’s and Libraries that can be used in conjunction with these technologies, which shows how the three technologies can expand easily with the use of these API’s and libraries.

Further to this we looked at how XML can be used to build documents, with meaning, which will allow for the information that has been provided by the client is in the form

of a HTML document so it will be vital to get the information into a format in which the proposed prototype will allow for the data to be added to the database, however the data will have to be queried so the next aspect that was investigated was how easily it is query the data. With so many different languages it seems that it is almost impossible to choose the best language to query data with, however with SPARQL being the choice of a lot of different sources. (Tatarinov,I. *et al* 2002) states that the best way to store and query ordered XML is best achieved by using a Relational Database system.

Finally this chapter looked at the toolkits that are available that can be used in union with the semantic technologies that have been investigated. The NeOn toolkit is the most highly regarded toolkit in use today, it combines a lot of the features that are used within the current system, however its main features focus upon ontologies, and the prototype system that is to be developed does not incorporate these ontologies, however the toolkit could be used in further development of the system, as it would allow for a more robust system with the opportunity to expand features within the system.

4 Project Process

4.1 Introduction

In the last chapter we looked into various different semantic web technologies that can be used within the development of a prototype system. We initially looked into the XML, as well as the best way to query the data contained within the file, this will be essential in the development of the prototype system as it will allow the development team to extract data from the source given to the development team by the Durham Constabulary. Further to this, as discussed in chapter 2.5 the current system that is in place uses the semantic framework of OWL, as well as the API, OWL2prefuse.

Now our investigation will lead into the various different project life cycles that can be used within a project, and the developer will make a decision on which life cycle will suit the needed of the project best.

4.2 Processes

There are many different life cycles that can be followed to aid the development of a project, however of all of the life cycles that can be used generally only 2 or 3 life cycles fit the development of different project. Our investigation will take a look at ‘agile’ methods, we will also investigate other life cycles such as the ‘Waterfall’ method as well as the prototyping model. Following the investigation into these methods then the developer will make a decision on which of the methods will be followed, and back up their decision with reasons why this method is superior to the other methods, for this project.

Our investigation first studies 'agile' (Lethbridge, T. & Laganière, R 2005) methods of development. There are many different variations on an 'agile' method of developing a system. The first of which is when following an 'agile' method it is suggested that TDD (Test Driven Development) is used, TDD urges the developer to begin with a small aspect of the system and then test it, if the test is successful then the developer adds another little aspect to the system and tests it and so on, however if the test fails then the developer must rectify why the aspect of the system fails.

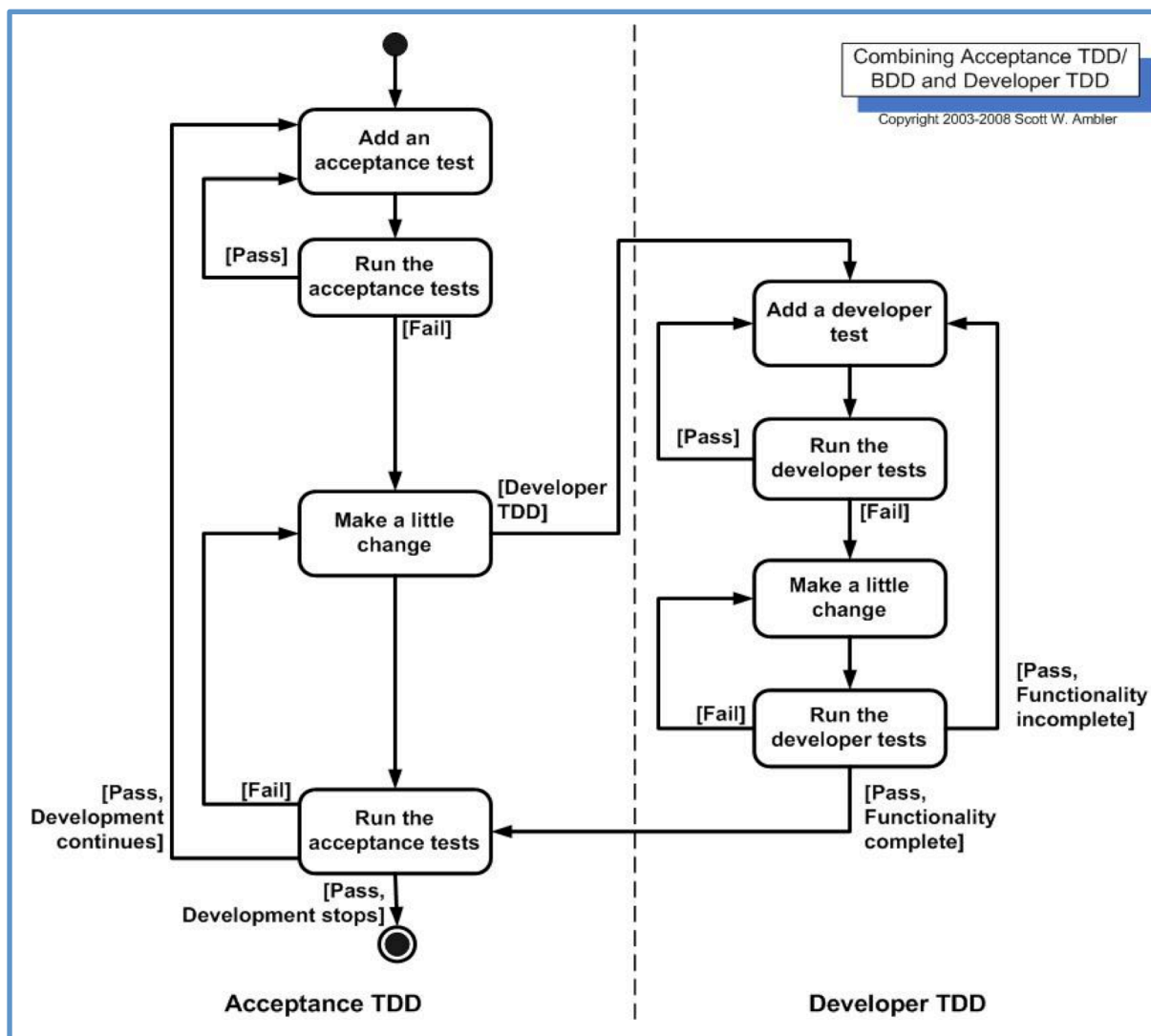


Figure : 4.1 – TDD diagram (Ambler, S.W. 2004)

Figure 4.1 shows how TDD is used in the development of a system, the left hand side of the figure shows how TDD is used for acceptance testing which is a set of tests that will determine whether or not a system will be accepted or not(Hall, P. & Fernández-Ramil, J. 2007), and the right hand side shows developer TDD, which involves writing a single test, normally referred to as a unit test. The main goal of developer TDD is to create a specified executable design to the planned solution based on a JIT (Just in Time) basis.

Following an 'agile' form of development also encourages regular meeting with the client, which links in to the '*manifesto for agile development*'. The first point in the manifesto is to "satisfy the customer through early and continues delivery of valuable software." Further to this the 'agile' process urges delivering working software frequently.

Another process that is used a lot in development is the waterfall process model. The waterfall model means that each stage must be completed before moving onto the next. However the biggest flaw with the waterfall method is if the client wants to change any aspect within the system then it is more costly in both time and money to edit the requirements, further to this if the developer encounters any problems and realises these it means that they have to go back and rectify them. Figure 4.2 shows a typical waterfall model, which shows the 'flow' of the waterfall from left to right, from top to bottom.

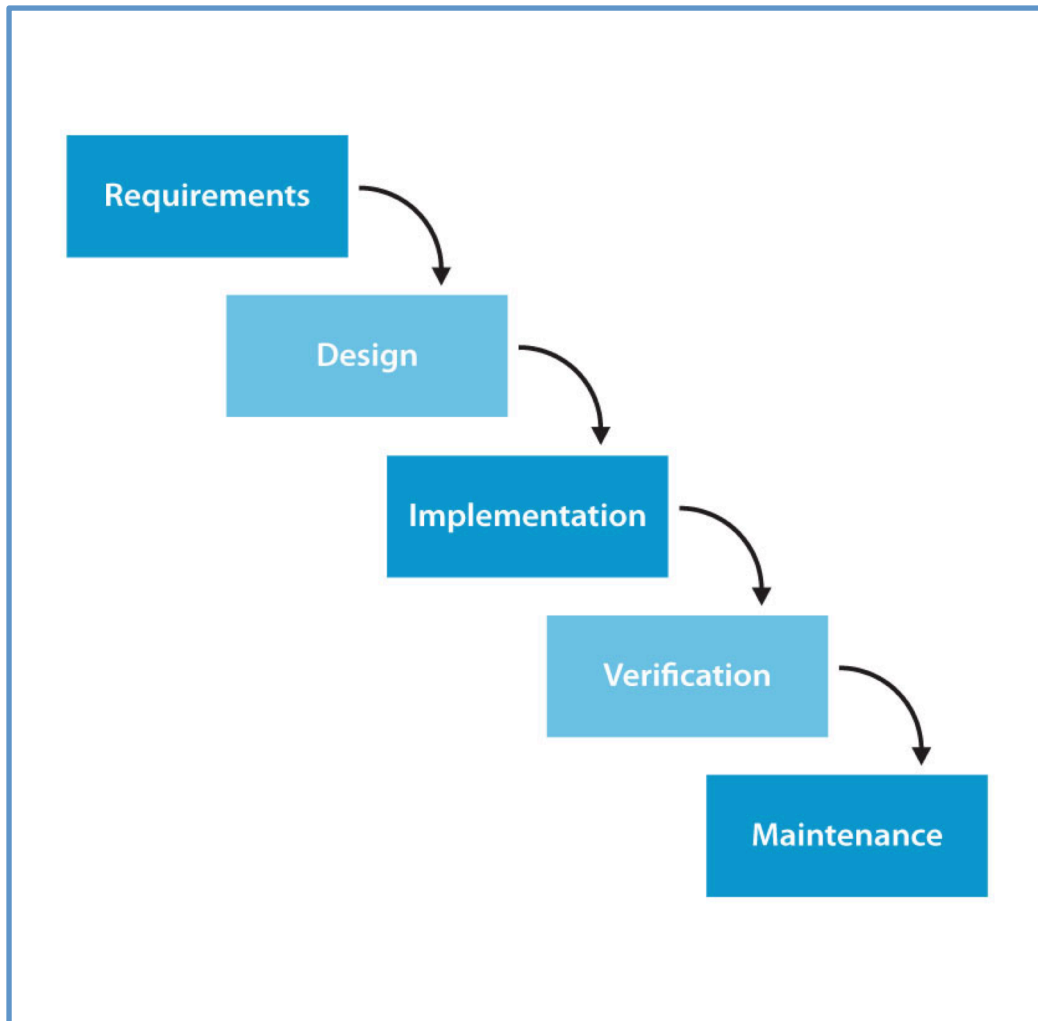


Figure 4.2 – waterfall method (Royce, W. 1970)

However with the waterfall method meaning if there are changes within the specification, they will be more costly the further through the process the project is through the waterfall method is not the best idea to follow this method.

Finally we will examine the prototyping method, which enables the developer to limit problems that can arise in the requirements stage. The main advantage of developing a prototype is that in reality only a few lines of code with a little functionality, to aid the client in what they want from a system. The biggest disadvantage is that when the client sees prototype systems they think that they are fully functional and ready for use

for the day to day running of the business, yet in reality there is a chance that the prototype will only be used once to show the client then disregarded and not used again. Figure 4.3 shows how the prototype methodology works.

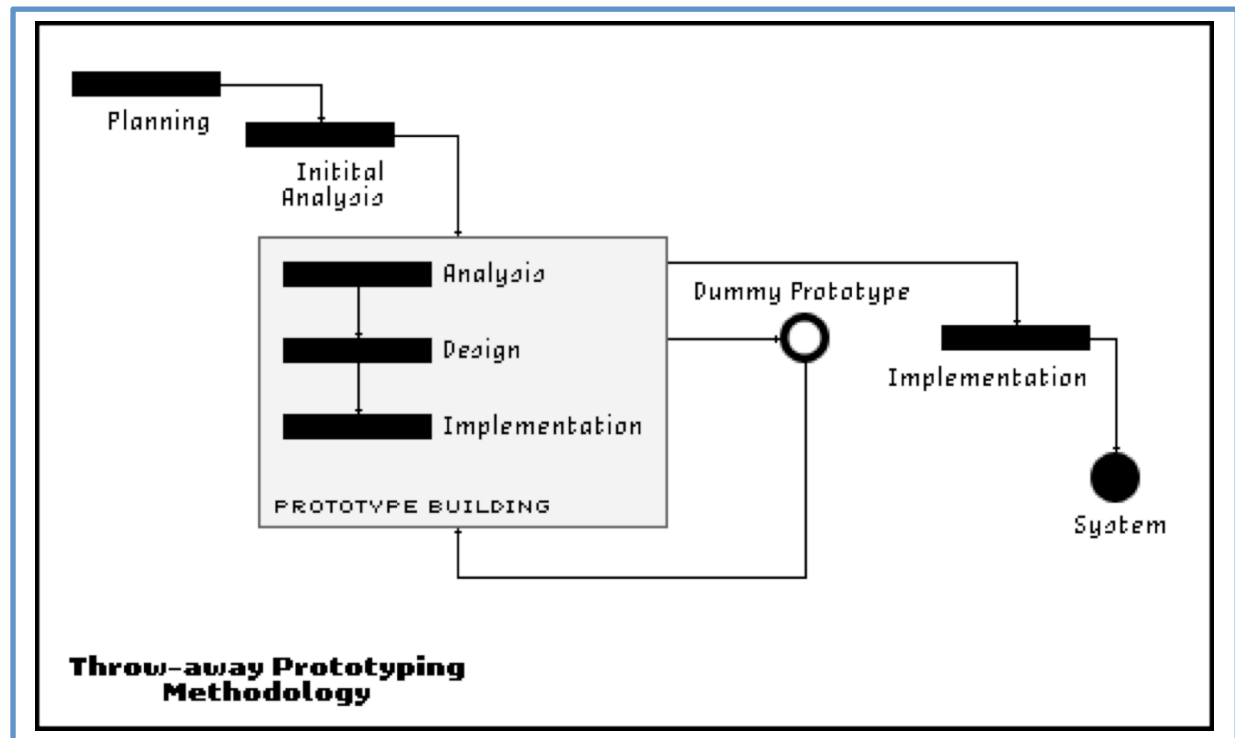


Figure 4.3 – prototyping methodology (Pressman, R.S. 2005)

As figure 4.3 shows that there could be many different prototypes before a 'Dummy Prototype' is even considered by the client.

4.3 Conclusion

We have looked in to three of the many different methodologies that exist for the development of a project. We have looked into 'agile' development, the prototyping methodology, and the waterfall model. Following the investigation in to the methodologies the developer decided that the best way to develop the project was by following a 'agile' style of development as it would allow the developer to engage with the client, to ensure that there requirements can be firstly noted properly, and secondly will mean that should the client want to change requirements at any time then this means that there wont have been a lot of work that would need to be re-done. This decision was made by the developer as it would have meant if the waterfall model was followed and the client changed requirements then the cost in time would be a lot more than if they were changed when following an agile methodology.

The developer also chose to follow an agile method over a prototyping method as it meant that several prototypes were not developed to be "thrown away". Further to this when gathering the original requirements from Dave Sampson it appeared that he and the Economic Crime Unit seemed very clear in what they wanted from the project and the system they wanted to be developed including the visualisation of the system, and following the prototyping methodology would have been a purposeless exercise as they knew what the system should look like.

5 Proposed Solution

5.1 Introduction

In the last chapter we looked in to project process life cycles, and we discovered that there are many different life cycles that can be used to aid with a project. After the developer chose the life cycle they were going to follow they could then move onto the best way to solve the problem that exists within the constabulary. We will also look into how aspects of the system will be designed, including database design. Following this we will look in to how the changes to the current system can be implemented without major problems.

5.2 Prototype Design

5.2.1 Introduction

The Durham constabulary currently have a system in place for the navigation of documents, which was developed in Java using OWL as well as API's and Frameworks. However the current process that the constabulary carry out does not allow for the data that is contained in the HTML documents, which are provided by third-party companies to the police. As discussed when investigating a crime the Durham Constabulary must deal with hundreds, even thousands of documents, which include these documents that are obtained from third party sources.

5.2.2 Overview

The current system allows the constabulary to read in documents that have an “unstructured” set of data, and this is where the developer comes in to help solve the problem the Constabulary is currently faced with. The planned development of the prototype system will enable the constabulary to read in the HTML documents provided, convert it to an XML document and save the data in a database which will then be able to navigate the information via the visualisation of the current system. Further to this the prototype system will allow the constabulary to revisit data that they have imported more easily and accurately.

The prototype must allow for the constabulary to load the HTML document into the current system and allow them to view the data. Further to this the Constabulary must be able to view links between the data in the document so they can find out if people they suspect are known to be linked to the investigation. Naturally the Constabulary will also require both the saving of an investigation as well as the loading of previous investigations. The navigation of the system will inherit from the current system, this was a decision from the developer, which was due to what was discussed in a meeting with the client, in which they stated that the users within the constabulary would be more beneficial if they didn't have to incorporate a new interface thus reducing costs.

5.2.3 Design

The design of the prototype system is very important, mainly for the end-user, so the developer decided to keep the visualisation within the system the same as it is currently

as this is what the client required, this will mean the constabulary will not need to spend hours training staff to operate the new system. However it is essential to get the design of other aspects of the prototype system correct so that the back-end of the system works without any complications.

The first aspect of design we will look into is the design of the database, which will store the data that is extracted from the source file that is received from the third party companies. However with there being sixteen different sections within the original form the development team felt that it would be more effective to create different tables and link them within a database rather than have one large table. There are many advantages to having multiple tables within a database, the most beneficial to the constabulary is that, it is easier to limit access within the database, which means they can limit which departments have access to what information, further to this it is easy to simplify the multiple table to make single virtual table.

We will begin with the most important table to be created, the 'contact_details' table, which includes information such as the suspect's name, as well as their address. This table will be essential to the running of the system as the majority of the fields within the document have the suspect's name, and address repeated throughout. The aim of this table is to try and eliminate redundant data, as well as holding information on the suspect. Figure 5.1 shows a screenshot of the details of the suspect that the police are investigating, this information will be essential to the system as it will allow for the suspect to be added to the database, from adding this information it will be possible to link all aspects in the form with the 'contact_details' table.

Summary	
Subject Details	
Name:	MR PETER JONES
Address:	2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH
Public Information	
Number	1
Total amount	£1,200
Latest	05/06/2010
Messages (click for details)	
CIFAS data present	
Reported Gone Away (GAIN)	
Previous Address	
Notices of Correction	

Figure 5.1 – details of the suspect.

From the above it shows that the table that will have to be created will need to incorporate twelve fields, to allow for the information to be used to its full potential. The reason that twelve fields will be required is there will be eleven fields for the information required, however the constabulary will also need to be able to distinguish between people, or even between cases. The first field that will be used will be a 'investigation_number' which will allow the investigating officer to add extra information such as '-1' so that each of the persons being investigated can be added to the same investigation, so for example if a person in investigation "12345" and is the 5th person the details are correct for the 'investigation_number' would be '12345-5'.

Following this investigation it was then onto designing the table, which including the information stated above the table was created with the aid of 'PHPmyadmin' with the use of SQL (Structured Query Language). Figure 5.2 shows the SQL code that was used to create the 'contact_details' table.

```

CREATE TABLE `bd79uq`.`contact_details` (
  `investigation_number` VARCHAR( 10 ) NOT NULL ,
  `title` VARCHAR( 5 ) NOT NULL ,
  `given_name` VARCHAR( 25 ) NOT NULL ,
  `surname` VARCHAR( 50 ) NOT NULL ,
  `house_number` VARCHAR( 5 ) NOT NULL ,
  `address_1` VARCHAR( 100 ) NOT NULL ,
  `address_2` VARCHAR( 100 ) NOT NULL ,
  `town/city` VARCHAR( 50 ) NOT NULL ,
  `postcode` VARCHAR( 8 ) NOT NULL ,
  `number` INT( 5 ) NOT NULL ,
  `last_amount` DECIMAL( 7 ) NOT NULL ,
  `latest` DATE NOT NULL ,
  PRIMARY KEY ( `investigation_number` )
) ENGINE = MYISAM ;

```

Figure 5.2 – SQL code for creating Table

When creating the table there were aspects where the developer had to decide whether the ‘type’ was the best to use for the information that could be input. The first of which was for the ‘house_number’, the developer originally planned to have this as an ‘int’ however they thought that using a ‘varchar’ would enable the use of flats, and apartments where the house number may “4-B” so therefore using an int would not be best to deal with this option.

	Field	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	<u>investigation_number</u>	varchar(10)	latin1_swedish_ci		No		
<input type="checkbox"/>	title	varchar(5)	latin1_swedish_ci		No		
<input type="checkbox"/>	given_name	varchar(25)	latin1_swedish_ci		No		
<input type="checkbox"/>	surname	varchar(50)	latin1_swedish_ci		No		
<input type="checkbox"/>	house_number	varchar(5)	latin1_swedish_ci		No		
<input type="checkbox"/>	address_1	varchar(100)	latin1_swedish_ci		No		
<input type="checkbox"/>	address_2	varchar(100)	latin1_swedish_ci		No		
<input type="checkbox"/>	town/city	varchar(50)	latin1_swedish_ci		No		
<input type="checkbox"/>	postcode	varchar(8)	latin1_swedish_ci		No		
<input type="checkbox"/>	number	int(5)			No		
<input type="checkbox"/>	last_amount	decimal(7,0)			No		
<input type="checkbox"/>	latest	date			No		

Figure 5.3 – table structure for ‘contact_details’ table

Following the creation of the 'contact_details' table it was then time to progress onto the next table, and find the best way to link the tables. The developer planned to link all tables via the 'investigation_number', this would allow multiple values that are contained in the form, such as multiple bank accounts and multiple loans.

Following this the next table that needs to be created is to the right hand side of the form, which contains information on the person being investigated such as the amount of searches that have been completed for the person as well as the amount of accounts that are currently held, including information on the current balance of these accounts as well as the clients worst credit score, coupled with the current credit score.

Below shows figure 5.4 which shows how the table is set out within the form. The table to contain the data for the credit report aspect will require 8 fields to contain the 7 field above, as well as a link to the 'contact details form'.

Previous Credit Searches	
0-3 months	0
4-6 months	0
7-12 months	2
Accounts Held	
Number	7
Total balance	£8,694
Worst current	8
Worst historical	8
Notices of Correction	
You are legally obliged to read all NOCs before making any assessment or decision regarding the applicant(s)	
Y2	

Table 5.4 – credit record

When creating the table for the above, table it was then essential to have a link to the original table with the suspect's contact details. This was achieved was via selecting the relation view as shown in figure 5.5.

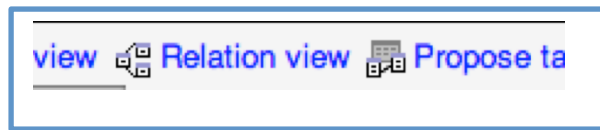


Figure 5.5 –relation view.

When clicking the relation view, the developer was greeted with the screen as in figure 5.6, and when the developer clicked the dropdown list, the developer can create relationships between tables, or as they are also ‘foreign keys’

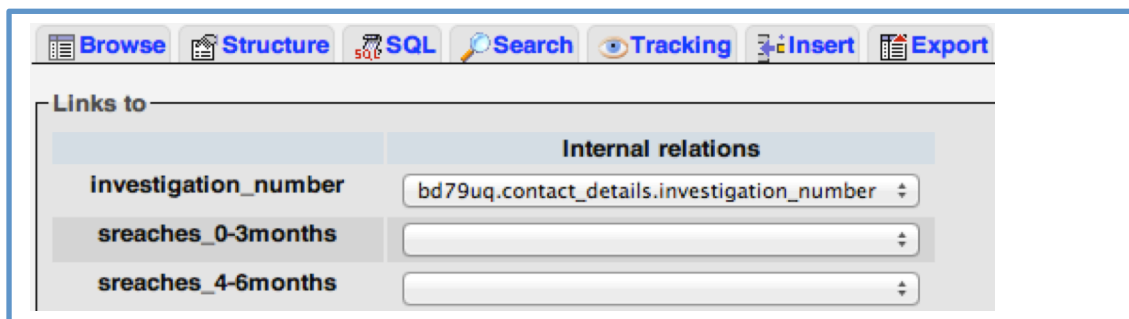


Figure 5.6 – linking foreign keys

As figure 5.6 shows the ‘foreign key’ that has been selected for the ‘investigation_number’ links to “bd79uq.contact_details.investigation_number”. This will allow for the investigation number being linked to the ‘contact details’ table which contains the most useful information to the investigation. As shown in figure 5.7 when the developer clicks the options they can select any of the fields within any table contained within the database.

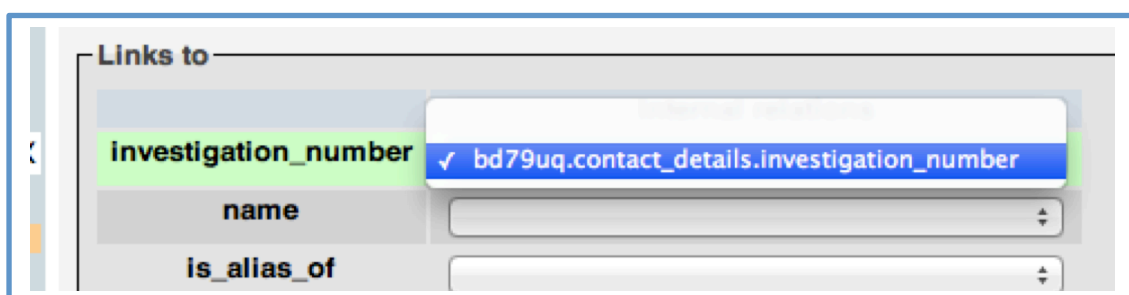


Figure 5.7 – options for foreign keys within the database.

While building the database the developer had to keep in mind that the Constabulary would need to be able to create links between known associates, so there will need to be a link these people, this is where the current visualisation system will allow the investigating officers to successfully map between associates, and will be able to navigate between them more easily.

5.2.4 System Architecture

As discussed previously in the design of the system the first major aspect of the system is the database, as it will allow data from the XML documents to be stored, but to get from the start point of the HTML document to viewing the data within the current system will take many different steps. The architecture of the system is shown below as to give a graphic representation of the proposed system.

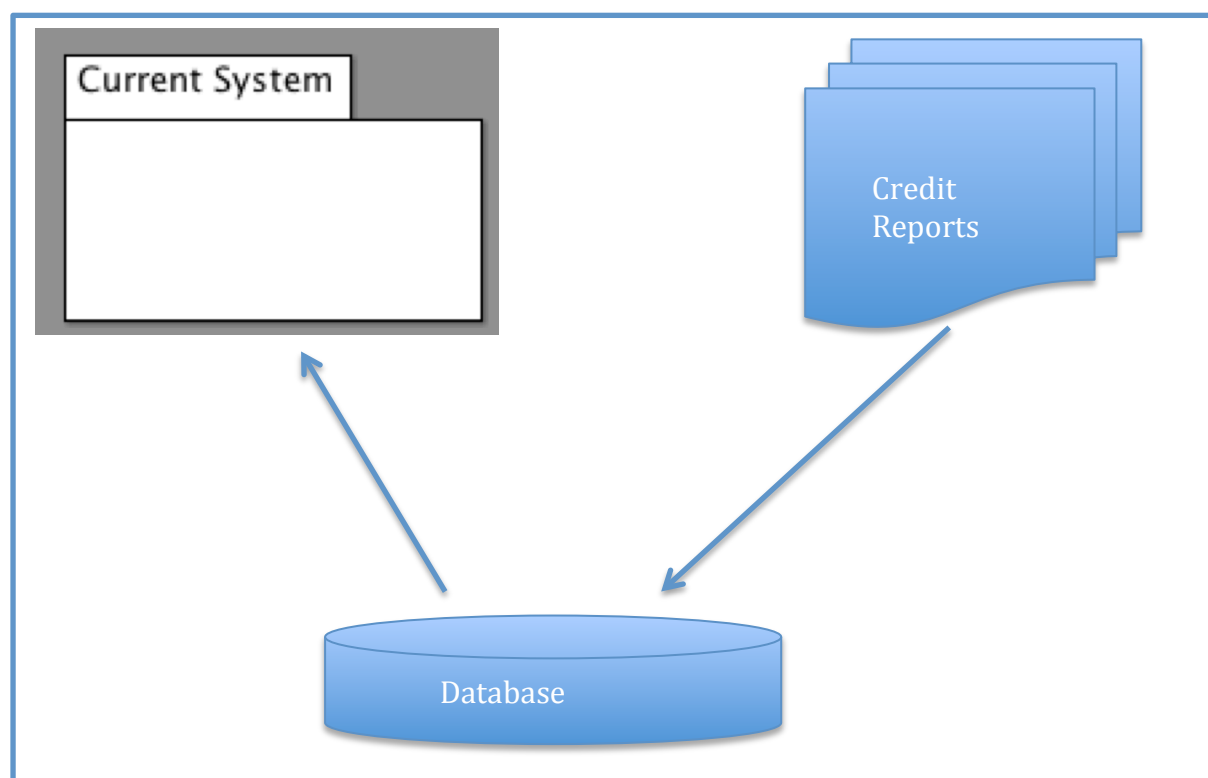


Figure 5.8 – system architecture

5.3 Implementation

We will now look into the implementation of the prototype system the implementation needed many different aspects to it to compile the 'finished' prototype system. The initial stages were to create the database; this was essential, as it would allow for the data from the XML document to be stored in a central location. However to get to this point the developer had to take the XML document that was created from the HTML to ensure that the data within the document can be used with the current system. We will now look at the steps that were taken to change the HTML document that was given by the client, to the incorporation of the data within the system.

5.4 Screen Scraping

The first aspect that must be considered was how to take the data from the unstructured HTML document, to a form in which it could be used, and it was decided that the best way to do this was to "scrape" the data from the HTML document so that the data could be extracted accurately, which would then allow for the data then to be entered into an XML document. As we seen in chapter 3.2 there many different API's and frameworks that can be used to scrape data. However it was originally thought that using a JAVA API to extract the data, however it seems that other programming languages offer a better range of features that can be used to successfully extract data. For the scraping within the prototype system it was decided that the Java tools that could be used were not accurate enough to extract the data that was needed, after further investigation it was decided that an 'SimpleHTMLDom' was the best way to extract the data. 'SimpleHTMLDom' is a HTML DOM parser written in PHP5+ let you manipulate HTML in a very easy way which allows for accurate extraction of data.

5.4.1 XML Document

As we described the Constabulary receive reports from third-party sources in the form of an HTML document. After converting to XML the constabulary need to assess the reports and link them to investigations. However as it stands the Constabulary have no way to store the data from their documents and re-assess them at a later date. The first set that had to be considered was the layout of the XML document, as well as how it looks in a 'browser'. When the Constabulary supplied the developer with the XML documents they also supplied a HTML (HyperText Markup Language) version (which can be found in appendix 13) of the document so that the developer could get a good idea of the layout of the document, what information is contained and where in the document it is held. This is essential to the developer as there may be aspects of the document that may not be in the sample documents that are given to the developer and could have information on another report that is obtained. While working with the XML document the developer also needed to construct the database so that all aspects of the XML document were accounted for. Following the data being converted into XML it was then time for the data to be transferred into a database.

5.4.2 Database implementation

The 1st aspect that had to be developed was the database so that the data contained in the XML document could be held in a central location. As shown in 5.2.3 the database was constructed before the developer extracted the data from the source file, the reason for this is that the developer felt that it was more important to have the right structure for the tables within the database. However the example that was provided by the constabulary, did not contain data in all fields, so it wasn't practical to add all tables that

could be incorporated, this is due to not knowing the data that could be needed, or the format of the data. Figure 5.9 shows the tables that were constructed within the database, all tables that were needed for the example that was supplied.

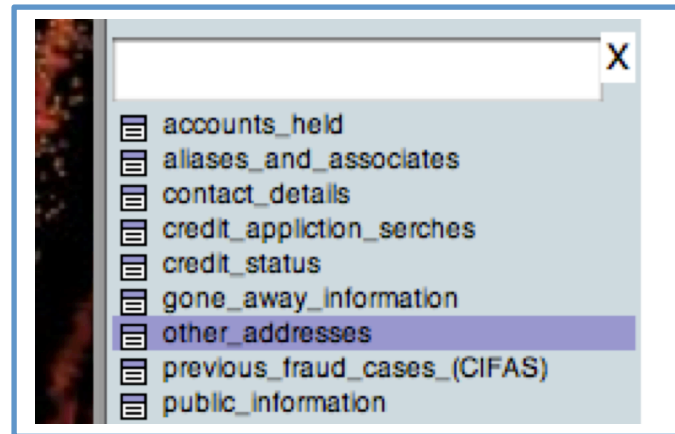


Figure 5.9 – tables within Database.

Figure 5.10 shows the print view of the “aliases_and_associates” table; this shows the structure of the table as well as links to other tables within the database.

aliases_and_associates						
Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(8)	No		contact_details -> investigation_number		
name	varchar(250)	No				
is_alias_of	varchar(250)	No				
house_number	varchar(6)	No				
address_1	varchar(50)	No				
address_2	varchar(50)	No				
town/city	varchar(50)	No				
postcode	varchar(8)	No				
on	date	No				
source	varchar(100)	No				
No index defined!						

Figure 5.10 – table print view.

All tables in the database relate back to the “contact_details” table as this is where the ‘investigation_number’ is held, this is used to ensure that all data for each of the different investigations are on running.

5.4.3 Populating the Database.

Following the generating the database and scraping the data from the HTML to XML it was then time to populate the database. This was essential to ensure that the data that was in the database was correct and accurate. The way that this was completed was by firstly focusing upon the name of the suspect, which is shown below, in Figure 5.11. following the data being converted to XML the developer felt it would be easier.

```
<table cellpadding="0" cellspacing="0" width="100%" border="0">
  <tbody>
    <tr class="rptSectionStripe">
      <td class="rptdatalabel">Name:</td>
      <td class="rptdata" colspan="6">
        <span class="c7">MR PETER JONES</span>
      </td>
      <td class="rptdatalabel"></td>
    </tr>
  </tbody>
</table>
```

Figure 5.11 – XML code for suspects name

5.5 Conclusion

This chapter has shown how aspects of the system have been constructed; however the developer did not fully develop what the client asked for. The developer’s time ran out in the development of the prototype system, and did not get chance to develop the current system to integrate the data into it. However the developer managed to get through the main obstacle, which was extracting data from the HTML so that it was properly formatted and so that it can be then converted into the structured form of XML. The development that took place allowed the developer to look into various

different ways to extract data from a HTML document, the developer decided on 'SimpleHTMLDom' as it allowed a stronger parse of the data that is needed from the document. It was very important to get the correct data from the source file as the information contained is vital to the system, as any information that is missed could lead to in the extreme a case failing. This is because the system links suspects and if the data is not correct then suspects could be missed within the system if there was spelling mistakes in the name or address.

The database that has been created has allowed for information to be stored, this will cover one of the objectives that was to allow the constabulary to re-assess data after it has been viewed. This is where this objective began, but with the developer not integrating the database into the current system means that the object could not be completed completely.

6 Evaluation of Prototype

6.1 Introduction

In the last chapter we looked at how the development of the system progressed, and how aspect of it were carried out and now we will critically evaluate the work that was carried out for the planned prototype system. With the focus being on getting the data from the initial form of a HTML document into the current system, there were many different steps to get from A to B. as we seen in the last chapter the developer didn't get fully from A to B, however there was a lot of work went in to getting the data into the database, and the next step was to incorporate it into the current system. In this chapter we will look at how the development of the prototype has gone, we will evaluate the prototype and note ways that it could be improved through further development.

6.2 Methods of Evaluation Used

There are many different ways to evaluate a the success or failure of a project, this can be done with a mix of objective and subjective methods. These methods allow the developer to gain a enhanced set of feedback results for the client. A subjective method that were planned to be used was getting the client to have a test session with the prototype system, which would allow for the developer to get a better idea of how the client interacted with the system, through feedback and observation. However these tests were edited with the system not incorporating data into the current system so the developer had to meet with the client to show them how far they had progressed with the system, so the tests were used to ensure that the data was being extracted from the

original HTML to the tables that were created and documented in chapter 5. The data was being successful pulled from one source to another, which meant that the developer had progressed well in the scope of the project, however as described the prototype was not a complete success, yet the constabulary were happy with what was developer and how far the developer progressed in the time frame.

6.3 Client Feedback

Masters Project: Client/Sponsor Evaluation of MSc Project Deliverable

Name of Sponsoring Organisation/Individual: DURHAM CONSTABULARY	
Name of Student : PAUL FAIRLEY	Programme: INTEGRATION OF NON-ONTOLOGICAL RESOURCES IN A SEMANTIC BASED INVESTIGATING SYSTEM.

Please give a score -5 to +5 to the following criteria (-5 = very bad, +5 = very good).

Please follow these up with written comments where you feel it is appropriate.

N.B. All these criteria refer to the practical product and the student undertaking the project.

Criterion	Score
Match between <u>your requirements</u> and the Terms of Reference negotiated by the student.	4
Satisfaction with <u>the product delivered</u> by the student.	4
Satisfaction with the manner in which the project was conducted by the student.	4
Professionalism of the student.	4
Preparedness of the student (for meetings, etc).	4
Level of knowledge displayed by the student.	4
Level of enthusiasm (commitment) displayed by the student.	5
Ability of student to work autonomously.	5
Ability of student to take direction (when appropriate).	4

Would you be interested in sponsoring MSc projects in the future? ☒ Yes ☐ No

If "yes" please briefly identify the areas of interest, and provide contact details for us to follow up.

**DURHAM CONSTABULARY - LAW ENFORCEMENT
WITH NUMEROUS PROJECTS AVAILABLE.**

If "no" please briefly provide let us know why.

Please add any further comments about your experience below (continue on other sheets if necessary).


761829

6.4 Evaluation of the Functionality of the System

The development that was completed by the developer meant that one of the major stages within the development of the prototype was not completed, however the stage that the developer progressed to covered the majority of the specifications that were asked for by the client. The test plan that was created and can be found in appendix 12, shows how the developer tested the aspects of the system that are currently completed.

The tests that were carried out were limited due to the developer not completing the prototype system. The tests that were carried out show how the aspects that were completed worked together to extract data from the HTML document and transfer the data into a database. The tests that were constructed at the beginning of the development of the system were not successful as the developer had to find which was the best technique of scraping the data from the HTML and the they had to decide which output worked best to transfer said data into the database. This means that the test suite was completed after what the developer believes to be the best way to extract the data had been chosen.

The greatest weakness within the prototype is that it does not allow for the data to be pulled into the current system and for the constabulary to visualise the data for more information refer to section 6.6 . However the prototype that was developed allowed for multiple variances within the system, for example with all links to tables within the database being linked by the “investigation_number” field then it allows for a suspect to have several bank accounts as well as several addresses.

6.5 Evaluation of Libraries used

Through the development the developer only ended up using one library, however many libraries were tried to try and the best results when extracting the data from the source file. The library that was decided upon was “SimpleHTMLDom” which the developer feels gave the best results when extracting data.

The developer looked primarily into the libraries that were available in Java, the libraries that were investigated were, XQuery, JTidy and HTMLCLient. However on further inspection of the libraries it appeared that the accuracy in retrieving data from the HTML document was not suitable as there were many different errors in the extraction. This is why “SimpleHTMLDom” was chosen, as it had a better accuracy for retrieving information from the source file.

6.6 Further Development

The developer has proposed to the constabulary that they would finish the project, which means integrating the database into the current system. This will ensure that the prototype is up and running to the standard the developer knows they can develop to. Further to this the developer must edit the current search functions to allow for the constabulary to search the additional features that will be added into the system.

Extending the current system to allow for log function would allow for the constabulary to see what reports were entered into the system as well as the date each report was added, this would allow them to keep track of data that may be out of date and in need of updating. With this said the developer also needs to look into a way to ensure that the

data is refreshed on a regular basis so that the constabulary know that they are working with up-to-date information.

The final recommendation the developer would make to the constabulary for further development is to allow for the system to automatically request credit reports for known associates to a 'suspect' and then automatically read in the report to the current system and create the links, this will save both time and effort for the police within the investigation of a crime.

6.7 Conclusion

In this chapter we have looked in to the development of the prototype system, and tried to gage if the project was a success or a failure. After analysing the client's requirements the developer deems the prototype as a partial success. The constabulary asked for a solution to the problem they face with the credit reports that are used by the police when investigation crime. The developer got as far as extracting the data at a very high level and moving said data into a database. However with the features not being added to the current system makes it a partial failure.

The client has given feedback on how they felt the development of the prototype went as a whole and have given positive feedback. Whoever the developer is always striving to be better within their specialised field, and they believe that the prototype could have gone better. With the requirements that were laid out in the terms of reference, the developer has covered 2 of the 3 requirements that were asked of them by the client. The prototype uses freeware/open source code which allows the data to be extracted and added to the data base which will save the constabulary time in both assessing the

data as well as allows the data to be re-assessed at a later date as it is stored in a central location which can be easily accessed.

The investigation into the existing semantic technologies that can be used also impacted upon the project, it is apparent that the semantic web technologies, which are being used as a part of the Web 3.0 movement impact upon all web based systems, and the prototype that was developed was no different. With so many different tasks that these technologies allow a developer to incorporate into a system makes the possibilities endless for the development of web-based systems.

7 Project Evaluation


7.1 Introduction


The last chapter evaluated the success of the development of the prototype program. Now will look into how the management of the finished project. This chapter will give a critical review of the processes that were completed, in order to successfully complete the project. This analysis will focus upon the techniques of project management including the planning and scheduling of the project, how the developer managed potential risks and the way that project design process was decided upon. We will also look into how the literature review, and how it impacted upon the development prototype. Finally the developer will evaluate their personal reflection on the project, and what they learnt from the process.

7.2 Evaluation of Selected Project Process

When planning the project it became apparent that the project would have to follow a development process, this would allow for a better knowledge of the task that needed to be completed, and ensure that what is being developed is what the client wants. With so many different methods that can be followed it was essential for the developer to select the right method to suit the needs of the project. Following a agile method meant that the developer could work closely with the client for the duration of the project, meaning that should there be any changes to requirements, which in the end there was the developer could incorporate these changes in to the system.

The agile methodology suited both the developer as well as the client; this was due to the client wanting to add features into the final prototype close to the end of the project. The additions that were asked of by the client were features that were to be incorporated into the system such as search functions that would allow for aspects of the credit report to be searched, these included:

 Account Number

 Phone Number

These two aspects are specific to the credit reports and the current system does not allow for the search of these topics.

Following an agile methodology also allowed for the client to keep a regular check on the progression of the project, and this also allowed the developer to keep track of progress via project control documents.

7.3 Evaluation of Project Control and Progress

For a project to be deemed a success it is essential to have the correct planning from the beginning or there was a large chance that the project would not be a success. Following the planning of the project it is vital to ensure that the plan is followed until the completion of the project. Thorough planning ensured that the objectives that were outlined could be completed both on time and on budget, as well as ensuring that the clients requirements are covered completely.

The first task that was undertaken was to record the initial meeting that were held with, the developer, the client and the supervisor for the project, which can be found in the work preparation diary (appendix 2). The diary was created to firstly document the

meetings that took place as well as to ensure that all aspects of topics that were discussed could be documented properly. Following the initial meeting with the Durham constabulary the developer was able to gain a better understanding of the prototype that was required by the client as well as the aspects that need to be included.

Following these initial meetings the developer was then able to generate a terms of reference (appendix 3). This document was used to clearly define the requirements of the client. Further to this the terms of reference outlined more clearly the objectives that need to be covered in the prototype system, as well as the area in which research was to be undertaken which would aid the development of the prototype system.

Following the terms of reference being generated and subsequently signed by both the client and the project supervisor, it was then time to generate a project schedule to manage the tasks that was needed to be completed. Generating a schedule would allow for an accurate mapping of the tasks, this would also allow for the developer to plan for the amount of time that they believed that each task would take. Further to this the schedule allowed the developer to plan their time properly, and allow to plan to allow for all tasks to complete by the completion date.

Task	Objective	Task Description	Hours	Start Date		End Date		Deliverables
				Planned	Actual	Planned	Actual	
Planning								
1.	1	Initial Client Meeting	3		11 th March 2011		11 th March 2011	Requirements
2.	1	Generate project Proposal	3	25 th March 2011	25 th March 2011	30 th March 2011	30 th March 2011	Project Proposal
3.	1,2	generate TOR(Terms of Reference)	5	30 th August 2011	31 st August 2011	31 st August 2011	31 st August 2011	Completed TOR
4.	1,2	Edits to TOR	2	31 st August 2011	1 st September 2011	10 th September 2011	9 th September 2011	Final TOR for
5.	5	Generate	5	1 st	1 st	5 th	5 th	Completed G

Figure 7.1 - Schedule

Figure 7.1 shows a small aspect of the schedule that was completed to track the project, which shows how it is formed. The schedule allowed the developer to link tasks to the objectives that must be completed to ensure a project is a success. Further to this the is both a start and end date which were planned at the beginning of the project, as well as actual start and end so that it was a lot easier to track tasks that were completed on time, ahead of time, or behind time, and unfortunately the project did have a couple of tasks that ran behind.

Along side the schedule a Gantt chart (appendix 4) was generated, and is used as a visual representation of the schedule, and allows for easier tracking of the project, the Gantt chart also allows for the visual concept of the timings for the project. As with the schedule that was generated, the Gantt chart was updated as the tasks were completed throughout the project. When progressing through the project tasks were coloured as they were completed, this allowed for the developer to more accurately keep track of where aspects over ran, and where time could be made up in the remaining tasks.

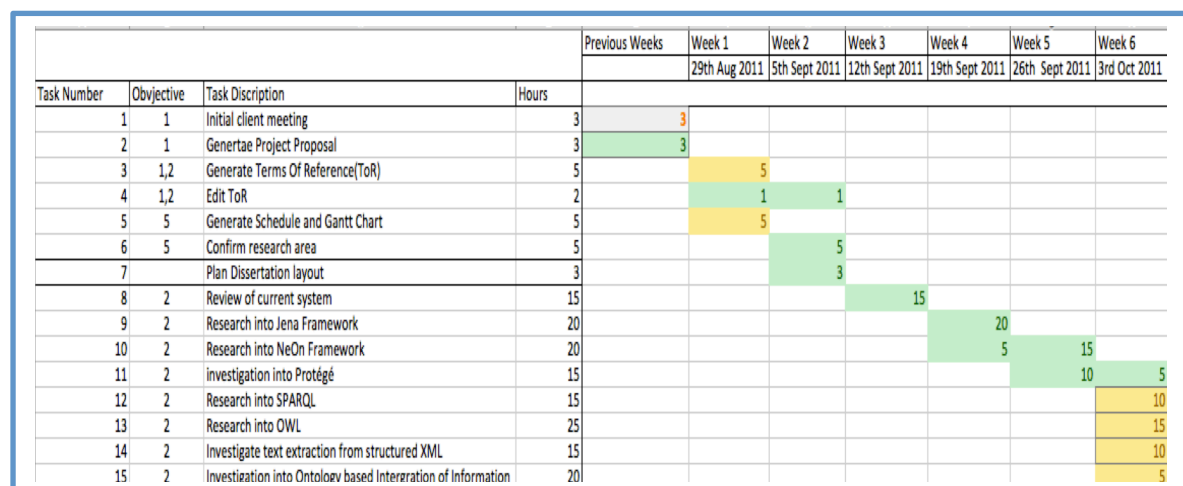


Figure 7.2 – Gantt chart

Figure 7.2 shows the Gantt chart, the colours that are shown in the Gantt chart show aspects that ran to time, and which aspects were completed before the time that was

initially factored into the schedule. The timings that are in green are tasks that were completed ahead of time, yellow the tasks that were completed on time and the timings in red, which are not showed in the above example are the tasks that ran over the time that was initially planned.

7.4 Evaluation of Literature Review

Following the investigation that was undertaken in the literature review it became very apparent that the aspects that were researched would impact highly on the prototype system that was to be produced. In the initial planning stages of the project the client informed the developer that the documents would be supplied as an XML document, however within the first couple of weeks it became apparent that the document format was in HTML. This impacted upon the research of the project, this was due to needing to structure the relatively unstructured HTML so the research started with screen scraping then led into generating XML documents. From the structured data it was then possible to store the data in a database, which gives it more structure, and makes it easier for the data to be re-assessed at a later date.

When beginning research for the project the main aspect that was to be investigated was semantic web technologies. As seen in the literature review chapter it these technologies are becoming more predominant in the development of web-based systems, and this advanced even more since the current system was built. However as the literature review also looked at XML is still very much a large part of development of systems that interact with the web.

7.5 Personal Evaluation

Overall the developer feels that the project went rather well as a whole, there were aspects, which they feel could have gone better. The project as a whole went well with the developer striking a good relationship with the client and worked well to the time scale that was set. The project has assisted the developer to improve their skills in time management, project management and organisational skills. The project has allowed the developer to exercise the skills that were taught in other MSc modules to aid with the development of both the project, and assisted them to understand the best way evaluate a project.

The developer feels that there were aspects of the project that he could have improved upon. The first of which was aspects of their problem solving, in one of the initial meetings with the client the developer was informed that the document would be supplied as an XML document, when it came to the development the file was in fact a HTML document. As we have seen in the investigation one is a structured document, and the other has no structure, and when the developer was given HTML over XML they panicked a little. However the developer managed to overcome this panic with the aid of the project supervisor who suggested different ways to overcome the problem.

The developer believes that the undertaking of requirements analysis was a successful task, resulted in a prosperous project overall. The literature review that was carried out aided with the development of the project, and linked well, and provided a respectable knowledge base for development. The choice to you third party frameworks and libraries could have backfired in the project as there was a chance that midway through

development the author would change aspects of them, thus meaning more development to ensure that they would still incorporate into the system.

The developer feels that the majority of the project went well, however the fact that the project was not completed to incorporate the work that was completed in to the current system was a little disappointing. The developer thought that they worked well as a whole for the majority of the project and not being able to present the constabulary with exactly what they wanted on the deadline of the project is unacceptable from the developer's point of view. However they have arranged with the client to ensure that the remainder of the project would complete following the deadline. Should the developer work on a similar project again the future then they would do aspects of the project differently, the first of which would be to manage their time a little more effectively, and plan to have more time towards the end of the project free so that should the project run over they have a better chance to complete the tasks that were set out in principle in time. Identifying the possible risks at the beginning of the project allowed the developer to plan for major factors that could impact upon the project and in the extreme case lead it to be a complete failure.

The last point that the developer has taken from the project is that there are so many different way to manage a project, these range from the way the a project can be tracked to the methodologies that can be used to ensure that the project can be completed to a high standard, on time, and most importantly work the way the developer planned it to.

7.6 Evaluation of Objectives Completed

At the beginning of the project the developer set out several different objectives that they hoped to be able to complete by the completion of the project. The table below shows the objectives and where evidence of their completion can be found.

Objective Number	Objective	Evidence of Completion
1	<u>Determine clients Requirements</u> Meet with client; generate requirements, and the proposed solution. Then finally meet with client and discuss proposed solution	Chapter 2
2	<u>Research into Semantic Web Technologies.</u> Including research into the different technologies, frameworks and compare and contrast.	Chapter 3
3	<u>Produce a prototype system</u> – following the completion of objective two, plan the prototype development thoroughly, and define a project life cycle.	Chapters 4 and 5
4	<u>Evaluate the success of the deliverables</u> Create test sessions with potential users.	Chapters 6
5	<u>Critically evaluate the project as a whole with reference to terms of reference</u> Evaluate with use of the terms of reference whether the project was successful, what went well, what went not so well, and what would be changed... if anything	Chapter 7
6	<u>Write dissertation, to a high standard.</u>	Entire document.

Table x – evidence of completion of tasks.

The above table shows that all objectives that were set out in the Terms of Reference (appendix 3), proves to the developer that the project as a whole was successful, with all objectives at least being covered, however as mentioned in 7.5 the prototype system

was not fully implemented so the argument could be was objective 3 was not fully completed as the credit reports were not incorporated in to the final system.

7.7 Conclusion

In this chapter the developer has analysed the project from their point of view with regard to the tools and approaches used for all aspects of the project as well how they felt the objectives were covered from the original specification drafted by the Durham constabulary. The developer used tools for the tracking of the project, which they had used in previous projects, which aided them in time management. The use of a schedule and Gantt chart enabled the developer to show the client what tasks would be completed by what date, and also give the project a structure.

The research that was conducted in to the chosen topic of “semantic web technologies” aided the development of the project and gave the developer a better understanding of how this topic would impact upon the development of a prototype system. The investigation into this topic also impacted on the design of the prototype. The development method that was chosen aided the developer throughout the project, however on reflection it may not have been the best method to use, and more research into more methodologies available there many have been a better alternative.

Section 7.6 shows how the tasks that were outlined in the terms of reference were completed. The only objective that the developer was disappointed about was objective 3, the reason for this is that following the extraction of data was not incorporated in to the current system.

8 Conclusion and Recommendations

8.1 Introduction

The dissertation that has been produced has shown the steps that were taken from the proposal of the project through the development to the evaluation of the prototype that the developer produced. As we have spoke about the amount of data that large companies go through on a daily basis is growing more and more by the day, and with crimes that the police must investigate are growing in size by the day.

The aim of the project was to reduce the time that the constabulary spend reduce the amount of time used in processing documents and access stored documents in a proactive systematic way, which has been partly completed through the project.

8.2 Overall Conclusions

With increasingly more data being analysed by the police in every investigation they must find alternatives to allow them to accurately assess data that is used in the investigation of crimes. The accuracy is the most important aspect when it comes to the investigation of possible suspects, as even a different spelling of a suspects name or a incorrect number in their date of birth of phone number could result in the wrong person being investigated.

Following the completion of the first task noted in the terms of reference, which was to liaise with the client, allowed the developer to get a better understanding of the current problem, as well as understand the best way to deal with the problem, research was completed. The research that was conducted into the current semantic web technologies meant that the developer could firstly get a better understanding of the

current system, but further to this it meant that they could also develop a prototype that would incorporate these features. The developer thinks that they should have researched more into the visualisation of the current system as it may have allowed for the features to process a little faster.

Following a project methodology gave the project definition and allowed the developer to keep track of the project. Another benefit that the developer seen from using an agile methodology, was that the methodology encourages regular meetings with the client, which meant that they could see how the developer was progressing with the project. Using project tracking techniques such as the schedule and Gantt chart allowed the developer to keep track of progress and tried to ensure that work didn't run too far behind schedule.

8.3 Recommendations

There are not many recommendations that the developer would make to the constabulary; this is due to 2 major factors, the current system, and the development of the prototype. The current system works very well with the reading of PDF files, and the prototype that has been developed will allow the credit reports to be displayed via the visualisation of the current system. However the current systems visualisation is very clutter when multiple items are added which can be seen in figure 2.3, the developer would recommend that the visualisation is tidied up so that the links are clearer and so that the main suspect is one colour, and then associates are added to the system they would be another colour.

This would aid the prototype system as when the information is added from the database and a suspect has multiple properties and multiple bank accounts the constabulary may get very confused very quickly due to the amount of links that will be on the screen at any one time.

8.4 Final conclusion

In conclusion the developer feels that the project was successful at extracting accurate data from the source file, and getting it into the database, however with the prototype not allowing for the data to be incorporated into the current system the project cannot be deemed a complete success. The objectives that were agreed between the client and the developer were all covered in the dissertation and proof can be found in the document. The deliverables of the project may have run behind time for a lot of the project but the client will soon have a fully working system the way that the developer planned for it to work, as the developer didn't want substandard programmes to be used by the police. The developer has made recommendations, which related to not only this project but to the current system which would improve the user's experience of using the system, as well as the speed that the system will run.

References

AID (Adaptive Information Disclosure) website Available online
(<http://adaptivedisclosure.org/aida/docs/workflows/bioaid-serql-query-examples/>)
accessed on 05th October 2011

Alesso, H.P. & Smith, C.F. Developeing Semantic Web Services. A K Peters. Natick, Massachusetts. 2009

Alhir, S.S. UML in a Nutshell. O'Reilly 1998

Balian, M. & de Carvalho, C.L. SCOP: a Java framework of Buliding Semanic Virtual Communities in the Web. Available online (<http://ceur-ws.org/Vol-427/paper6.pdf>)
accessed on 05th October 2011

Cardoso, J. Semantic Web Services – Theory, Tools and Applications. Information science reference 2007

De Nicola, A., Missikoff, M., & Navigli, R. A Software Engineering approach to ontology building. Available online (<http://www.mendeley.com/research/a-software-engineering-approach-to-ontology-building/#page-1>) accessed on 16th September 2011

Fowler, M. Is Design Dead? Available online
(<http://www.martinfowler.com/articles/designDead.html>) accessed on 20th September 2011

Goldfarb, C.F., Prescod, P. The XML Handbook – Second Edition. Prentice Hall, New Jersey. 2000.

Gruber, T.R. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199-220, 1993 Available online (<http://tomgruber.org/writing/ontolingua-kaj-1993.htm>) accessed on 01st October 2011

Guo, Q.L. & Zhang, M. Semantic Information integration and question answering based on pervasive agent ontology. *Expert Systems with applications*, 36, 10068-100777. 2009

Hall, P. & Fernández-Ramil, J. Managing the software enterprise, Software Engineering and Information systems in context. London: England Thompson 2007

Hebeler, J., Fisher, M., Blance, R., & Perez-Lopez, A. – Semantic Web Programming. Wiley Publishing. 2009

Hitzler, P., Krötzsch, M. & Rudolph, S. Foundations of Semantic Web Technologies. Chapman and Hall 2010.

Inmon, W.H. Building the Data Warehouse. Wiley : India 2005

Kimmel, P. UML Demystified. McGraw-Hill/Osbourne, New York 2005

Lethbridge, T & Lagnière, R. Object-Oriented Software Engineering: Practical Software Development using UML and Java Second Edition. Madenhead: England McGraw-Hill 2005

Melton, J. & Buxton, S. Querying XML – Xquery, XPath, and SQL/XML in context. Morgan Kaufmann 2006

Myllymaki, J. Effective web data extraction with standard XML technologies. Computer Networks, Volume 39, Issue 5, 5 August 2002, pages 635-644

NeOn Toolkit Documentation Available online (http://neon-toolkit.org/wiki/Documentation_and_Support) accessed on 05th October 2011

Pollock, J.T. Semantic Web for Dummies. Wiley Publishing. 2009

Pressman, R.S. Software Engineering - A practitioner's Approach - Sixth Edition 2005

RDF tutorial Available online (<http://phpxmlclasses.sourceforge.net/rdql.html>) accessed on 10th October 2011

Seabourne, A. RDQL – A Query Language for RDF Available online (<http://w3.org/Submission/2004/SUBM-RDQL-20040109/>) accessed on 10th September 2011

Tatarinov, I., Viglas, S.D, Beyer, K., Shanmugasundaram, J., Shekita, E. & Zhang, C. Storing and Querying Ordered XML using a Relational Database system Available online (http://delivery.acm.org/10.1145/570000/564715/p204-tatarinov.pdf?ip=157.228.90.213&acc=ACTIVE%20SERVICE&CFID=66315874&CFTOKEN=48920221&_acm_=1320061619_e5e73817c0a3ae935df6ad2d3b795ad9) accessed on 15th October 2011

W3C - OWL 1.1 specification - Available online (<http://www.w3.org/Submission/2006/10/>) accessed on 01st October 2011

Appendices

Appendix 1 – Project Proposal

Student Name: <i>Paul Fairley</i>	Student Number: 076239087	Programme: MSc Software Engineering
(Current) Correspondence Address: 1 Hartington Street Roker Sunderland SR6 0le	Correspondence Address During Project (if different):	
Tel: 01915141372	Fax: N/A	
Mobile : 07970377259	E-mail: paulfairley69@googlemail.com	

Name of Sponsoring Organisation/Individual: Durham Constabulary, Economic Crime Department
Postal address of Sponsoring Organisation/Individual: Durham Constabulary Police Headquarters Aykley Heads Durham DH1 5TT
Name of Contact (Client) at Sponsoring Organisation and contact details: <i>(where this is different from the overall sponsor)</i> Dave Sampson
Practical Project

Brief overview of the sponsoring organisation: *[one paragraph]*

Durham Constabulary is a regional police force that is responsible for the county of Durham. The project is in relation to the Economic Crime Department, which is responsible for fraud investigations. The department has to deal with a growing number of complex cases which involve a large amount of evidence to be inspected often running into tens of thousands of documents and records which considerably slow down the process of preparing witness statements to be presented at court. The police have an obligation of going through all the evidence and not just to find evidence for the alleged crime but also to ensure that any evidence that might disprove their case is also disclosed to the courts and the defence. As the data is often complex and can involve a number of offences and a number of individuals the investigation can take more than a year to complete and has a high risk of collapsing in court if the court and the jury cannot reliably comprehend the evidence laid before them.

Practical Outcome of the project: *[what will be produced and delivered to the sponsor, N.B. it must be possible to produce this deliverable within approximately 300 hours of work]*

Link to the current system to allow for the integration of the information from the 3rd party organisations.

Specific requirements within the project: *[provide a bullet point list of specific requirements the sponsor has: in terms of the final product, the process of developing it, any interim deliverables, etc]*

development of an 'low-cost' solution to the problem

technologies used to run the system be either open source or ones, which are already present within the organisation.

Visualisation of data maybe built upon the platform designed by A. Spencer.

Specific skills the student needs to do this project:

System will be developed in Java, with SQL link to a data base.

Specific constraints that will be imposed on the student's project by the sponsoring organisation:

Disclosure of documents that will be provided by the constabulary.
Limited meeting with constabulary with client working away
no funds from the constabulary to develop the project

Specific resources that will be provided for the student's project <u>by the sponsoring organisation</u> : Current System Credit Reports (which are received via 3 rd party companies)
Where will the student work? On site (at the company)/ <u>at the University</u> /to be negotiated. If on site at the company: will reasonable travelling expenses be re-imbursed? Yes/ <u>No</u>
Any other comments:
<i>If clarification is needed please contact the project tutors in the Department of Computing, Engineering & Technology, University of Sunderland, St Peter's Way, Sunderland, SR6 0DD. Tel: (0191) 515 2674 (I Potts), (0191) 515 2786 (HM Edwards/SM Young). Fax: (0191) 515 2781, e-mail: mscprojects@sunderland.ac.uk,</i>

Supervisor's assessment of the proposal

(this may be after consultation with the relevant programme leader/project advisor):

	Revise	Reconsider	Acceptable
Overview of the Sponsoring Organisation.			X
Practical Outcome of the project			X
Specific (Client) Requirements			X
Specific Skills Identified			X
Constraints Appropriate			X
Resources Appropriate			X

Any area identified as "Revise" MUST be revised and the proposal resubmitted for approval.

Any area identified as "Reconsider" MUST be carefully considered and discussed with the supervisor in scoping the subsequent Terms of Reference.

Signature _____ Date: _____

(this must be signed and dated by the project supervisor and a copy provided to the project tutor(s))

Appendix 2 – Project Preparation Diary

Date : 01 st May 2011	Description of Task(s) completed
Time : 09.00	✚ Spoke to Dr. Albert Bokma regarding a project that he had mentioned in conjunction with Durham Police Force.
Place: Sunderland University	
Tasks: 1	
Resulting Tasks : 1	✚ Further investigate what software could be used for the development of the project.

Date : 11 th May 2011	Description of Task(s) completed
Time : 09.00	✚ Met Dave Sampson for the first time and he told me of several project that were available.
Place: Durham Constabulary	
Tasks: 1	
Resulting Tasks : 1	✚ Further investigate what software could be used for the development of the project.

Date : 15 th May 2011	Description of Task(s) completed
Time : 09.00	✚ Met with Dave and we discussed two project which I was going to undertake development in
Place: Durham Constabulary	
Tasks: 1	
Resulting Tasks : 1	✚ Plan how do develop the planned system.

Appendix 3 – Terms of Reference

Integrating Non-Ontological resources in semantic based Investigative Systems.

Paul Fairley (076239087)

MSc Software Engineering

Overview

Guidance: provide the context to the work. Typically providing background about the sponsor and outlining reasons for undertaking the work

Durham Constabulary is a regional police force that is responsible for the county of Durham. The project is in relation to the Economic Crime Department, which is responsible for fraud investigations. The department has to deal with a growing number of complex cases which involve a large amount of evidence to be inspected often running into tens of thousands of documents and records which considerably slow down the process of preparing witness statements to be presented at court. The police have an obligation of going through all the evidence and not just to find evidence for the alleged crime but also to ensure that any evidence that might disprove their case is also disclosed to the courts and the defence. As the data is often complex and can involve a number of offences and a number of individuals the investigation can take more than a year to complete and has a high risk of collapsing in court if the court and the jury cannot reliably comprehend the evidence laid before them.

The police have systems to help them manage investigations which adequately help in keeping track of individuals and vital data associated with them as well as keeping a searchable inventory of documents and exhibits but these tools do not help the investigative process or the sense-making of these collections. In financial investigations in particular it is often important to build a complete picture of individuals and their activity. The police use information sources for the financial involvement of individuals but have no way of linking these to the investigation and inspecting the totality of the results. The present project aims to help in integrating records on the financial involvement of individuals into an exiting prototype system for conceptual navigating of exhibits.

Sensitive data is received in an electronic format from a number of external agencies that are used as intelligence and as part of the investigation of crime.

The data received is currently stored on an ad-hoc basis, in isolation with no easily accessible option for re-evaluation or cross-referencing.

Where multiple data records are available for an individual or a group there is no system in place for visualisation and linkage of relevant data.

A previous project developed a 'Visual Document Management system, which is currently in use with the constabulary, the client has informed the developer that the visualisation of documents should link to the visual system already in use.

Product to be delivered to client

Guidance: this should specify both the final deliverable which is to be produced for the client and also the manner in which this delivery will take place.

It is desired that a prototype system will be developed whereby users can readily store data in a system that is capable of navigation using an interface for cross referencing documents, further evaluation of documents when new documents are added and selected for visualisation.

Client requirements

Guidance: this outlines the list of the clients requirements in terms of product features required, and delivery mechanisms if appropriate (the "wish list"). Much of this initial material will exist in the previously completed project proposal.

The client has asked for the development of an 'low-cost' solution to the problem that has been outlined, above so that the Durham Constabulary can reduce the amount of time used in processing documents and access stored documents in a proactive systematic way.

It is also a requirement that the technologies used to run the system be either open source or ones, which are already present within the organisation.

Visualisation of data maybe built upon the platform designed by A. Spencer.

Constraints

Guidance These define the boundaries of the work for this specific project and typically include: Limitations on access to specific resources, budget, the research topic and standards.

Throughout the development of the prototype system there could be various different constraints that could affect the project. The first of which is the disclosure of documents that will be provided by the constabulary. This is a potential constraint as it could take time for the documents to be edited by the constabulary to ensure that all sensitive information is removed from the documents.

Further to this time with the sponsor will be limited in terms of meetings, in the initial weeks of planning it has been decided that we will meet once every four weeks. However when the development of the prototype has begun this may need to be looked into.

The final constraint that with their being no funds from the constabulary to develop the project, the technologies that will be used will need to be open-source, other than the technologies that are readily available at the constabulary.

Resources

Guidance: These define the access and assets that will provided by the client, or available via the university, for use during the project and typically include: the number of people who will be involved in the project, their skills, possibly naming individual, Specific equipment, services.

Sample documents will be supplied by the client after they have been edited to removed any sensitive information

Dave Sampson will be available to discuss any system requirements that will be needed within

the system

The team that would be using the system will all have adequate computer skills as they have been working with the system previously developed.

Reporting to Sponsor

Guidance: This section identifies (i) who the student must liaise with, and report to, in the sponsoring organisation, (ii) what reporting mechanisms are to be used, and (iii) how frequently contact is to be made.

Initially contact was made with Dave Sampson of the Durham Constabulary via Dr. Albert Bokma, it was arranged that meetings would be held every 4 weeks. However due to new working commitments of the sponsor this will change as he is based in London 4 days a week and appointments will have to be organised to fit into his new timetable

Sponsor Sign-off

Guidance: The sponsor is asked to sign and date the first part of the ToR. This confirms acceptance of its content as a contract to be worked to by the student. The sponsor should not sign-off on this until they are satisfied with the content. The sponsor is asked to focus on the practical aspects of the project and the deliverable that is to be produced for them.

Signature *(this indicates acceptance of the scope of the practical component of the project)* Date

Project Objectives

Guidance: These are the major activities a student needs to complete to bring a project to a successful conclusion. Typically a project has between five and eight objectives. The objectives should cover: practical aspects required to enable delivery of the product to the client, research aspects: including identification of the research literature that will be critically reviewed, The objectives should be stated in such a way that it is possible to see how they can be evaluated that makes it easier to prove the success of a project. Definition of SMART objectives is sensible.

- Research into Semantic Web Technologies.
 - Research into the different technologies
 - Research into the frameworks available
 - Compare and contrast the technologies and come to a decision on the best technology to use.
- Determine clients Requirements
 - Meet with client and discuss the problem
 - Following this meeting generate requirements, and the proposed solution.
 - Meet client as discuss proposed solution
- Produce a prototype system
 - Following the meeting with client and finalising solutions, it is essential to thoroughly plan out the proposed system to ensure all aspects that are needed will be covered.
 - Define a development life cycle to ensure then project will me a success
- Evaluate the success of the deliverables
 - Create test session with potential users to ensure the prototype meets all of the requirements stated.
- Critically evaluate the project as a whole with reference to terms of reference
 - Evaluate with use of the terms of reference weather the project was successful, what went well, what went not so well, and what would be changed... if anything
- Write dissertation
 - Write a flowing document that is a high standard

Statement of Research

Guidance: This is (i) a statement of the research are to be investigated and (ii) a set of supporting initial references from reputable journals and conferences.(Between six and ten references)

The Semantic Web has a clear potential for managing data in a much more context sensitive way than traditional methods. The previous work upon which this project builds developed an ontology driven interface and document management system to represent the concepts of a typical investigation and associate textual documents in PDF format to it, so as to allow to record the results of the investigation. Besides textual documents seized in the course of an investigation, in financial investigations there are also records about the financial involvement of persons and organisations that are part of the investigation. This data comes from third party sources that show their bank accounts, credit cards and loans and mortgages, which are provided in XML format. This requires a different approach to relate these records to the ontology used in the investigation and make this data navigable alongside other records. Some work has been done on making databases amenable to semantic navigation but less work on making semi-structured resources such as these navigable in the same way and the project will focus on the integration of non-ontological structured resources.

Research will be conducted in Semantic Web Technologies and how they are used :

Fürber, Christian and Hepp, Martin: Using SPARQL and SPIN for Data Quality Management on the Semantic Web, in: BIS 2010. Proceedings of the 13th International Conference on Business Information Systems, May 3-5, 2010, Berlin, Germany, Springer LNBI Vol 47, pp. 35-46. <http://www.heppnetz.de/files/fuerber-hepp-sparql-spin-dqm.pdf>

Ontology-Based Integration of Information - A survey of Existing Approaches. H.Wache, T. Vogele - <http://www.let.uu.nl/%7EPaola.Monachesi/personal/papers/wache.pdf>

Foundations of Semantic Web technologies - Hitzler, Pascal; Krötzsch, Markus; Rudolph, Sebastian, Dr

NeOn Project documentation – http://www.neon-project.org/nw/Welcome_to_the_NeOn_Project

OWL 2prefuse API - <http://owl2prefuse.sourceforge.net/documentation.php>

Jena Framework - <http://openjena.org/documentation.html>

Statement of Level of Challenge

Guidance: This must identify how the skills and knowledge acquired during the taught part of the course are to be extended during the project in terms of the research topic investigated and/or the practical work undertaken.

The project is technically challenging in that the reports have to be mapped into the existing data structure and that data structure having to be extended without inadvertently causing the visual navigation of them to stop working. This will require a careful study of the previous prototype and making suitable modifications apart from developing an import component that can extract relevant data from the XML reports. The work included dealing with XML and OWL as well as proprietary data structures of the existing tool.

The project will draw upon various different aspects that have been thought thus far in the MSc course. Skills that were acquired in the following units

CETM11 – Research Skills and Academic Literacy
CIFM01 – Database Systems Engineering
COMM81 – Advanced Object Oriented Development
COMM83 – Software Production Measurement and Control
CSEM01 – Software Engineering and Management

Reporting to Supervisor

Guidance: This section identifies (i) who the supervisor is, (ii) what supervisory mechanisms are to be used, and (iii) how frequently contact is to be made.

The supervisor for this project will be Dr. Albert Bokma who has established a working partnership between the University of Sunderland and the Durham Constabulary. It has been arranged that a weekly meeting with Dr. Bokma will be on a Wednesday at 10am where we will discuss what tasks have been completed since the last meeting, and to ensure that the project is on track.

Supervisor Sign-off

Guidance: The supervisor is expected to sign and date the ToR. This confirms acceptance of its content as a contract to be worked to by the student. The supervisor is asked to focus on the entire project to ensure it is adequate as a terms of reference for a project for the specific masters programme being studied. The supervisor should not sign-off on this until they are satisfied with the content.

Signature *(this indicates acceptance of the scope of the entire project)*

Date

Appendix 4 – Project Schedule

Task	Objective	Task Description	Hours	Start Date		End Date		Deliverables
				Planned	Actual	Planned	Actual	
Planning								
1.	1	Initial Client Meeting	3		11 th March 2011		11 th March 2011	Requirements
2.	1	Generate project Proposal	3	25 th March 2011	25 th March 2011	30 th March 2011	30 th March 2011	Project Proposal
3.	1,2	generate TOR(Terms of Reference)	5	30 th August 2011	31 st August 2011	31 st August 2011	31 st August 2011	Completed TOR form
4.	1,2	Edits to TOR	2	31 st August 2011	1 st September 2011	10 th September 2011	9 th September 2011	Final TOR form
5.	5	Generate Schedule and Gantt Chart	5	1 st September 2011	1 st September 2011	5 th September 2011	5 th September 2011	Completed Gantt Chart and Schedule
6.	5	Confirm research area	5	10 th September 2011	10 th September 2011	10 th September 2011	10 th September 2011	
Total Hours : 23								
7.		Plan Dissertation Layout	3	6 th September 2011	6 th September 2011	11 th September 2011	11 th September 2011	Chapter Headings
Research and Literature Review.								
8.	2	Review of current system	15	12 th September 2011	13 th September 2011	19 th September 2011	19 th September 2011	Notes on topic
9.	2	Research into Jena Framework	20	20 th September 2011	20 th September 2011	24 th September 2011	26 th September 2011	Notes on topic
10	2	Research into NeOn Framework	20	24 th September 2011	24 th September 2011	29 th September 2011	29 th September 2011	Notes on topic
11	2	Investigation into Protégé	15	30 th September 2011	30 th September 2011	7 th October 2011	7 th October 2011	Notes on topic
12	2	Research into SPARQL	15	4 th October 2011	4 th October 2011	12 th October 2011	12 th October 2011	Notes on topic
13	2	Research into OWL	25	6 th October 2011	6 th October 2011	12 th October 2011	12 th October 2011	Notes on topic

14	2	Investigate text extraction from structured XML	15	10 th October 2011	10 th October 2011	12 th October 2011	12 th October 2011	Notes on topic
15	2	Investigation into Ontology-Based Integration of Information	20	11 th October 2011	11 th October 2011	13 th October 2011	13 th October 2011	Notes on topic
16	2	Research into RDF	20	12 th October 2011	12 th October 2011	17 th October 2011	17 th October 2011	Notes on topic
17		Compile all notes and information gathered	10	14 th October 2011	14 th October 2011	14 th October 2011	14 th October 2011	Notes on topic
18	2,3,6	Write Draft literature review	20	14 th October 2011	15 th October 2011	18 th October 2011	20 th October 2011	Draft chapter 3
19		Discuss literature review with supervisor	2	19 th October 2011	19 th October 2011	19 th October 2011	19 th October 2011	
20	2,3,6	Act upon comments in task 19	5	24 th October 2011	20 th October 2011	28 nd October 2011	29 th October 2011	Notes on changes/edits
Total Hours : 202								
Development of Prototype.								
21	4,6	Create prototype design	10	3 th October 2011	3 th October 2011	6 th October 2011	6 th October 2011	
22	4,6	Generate UML diagrams for system	5	7 th October 2011	7 th October 2011	9 th October 2011	9 th October 2011	
23	4,6	Create system to automatically read financial reports into the system.	20	15 th October 2011	15 th October 2011	19 th October 2011	27 th October 2011	
24	4,6	Create system to pull information from XML Documents to a temp source	20	24 th October 2011	28 th October 2011	26 th October 2011	30 th October 2011	

25	4,6	Generate data base to store data pulled from XML Documents	35	26 th October 2011	26 th October 2011	3 rd November 2011	3 rd November 2011	
26	4,6	Link to tasks 24 and 25	35	1 st November 2011	4 th November 2011	8 th November 2011	9 th November 2011	
27	4,6	Combine aspects with existing system	35	6 th November 2011	10 th November 2011	13 th November 2011	13 th November 2011	
28	4,6	Edit existing system to incorporate multiple accounts/mortgages ect	35	14 th November 2011	14 th November 2011	16 th November 2011	16 th November 2011	
Total Hours : 195								
Testing of Prototype.								
29	4,6	Write test suite	10	13 th November 2011	13 th November 2011	16 th November 2011	16 th November 2011	
30	4,6	White box testing of system	15	16 th November 2011	16 th November 2011	21 st November 2011	21 st November 2011	Test cases
31	4,6	Black Box testing of system	15	19 th November 2011	19 th November 2011	21 st November 2011	21 st November 2011	Test cases
32	4,6	Create questionnaire for end users test.	10	16 th November 2011	16 th November 2011	21 st November 2011	21 st November 2011	Questionnaires
33	1,4,6,	End User Testing	15	18 th November 2011	18 th November 2011	18 th November 2011	18 th November 2011	User Feedback Sheets
34	4,6	Edits to system	20	23 rd November 2011	23 rd November 2011	29 th November 2011	29 th November 2011	
Total Hours : 85								
Dissertation Write Up								
35	4	Write Chapter 1 (Introduction)	15	27 th September 2011	27 th September 2011	5 th October 2011	5 th October 2011	Completed Introduction Chapter
36	4	Write Chapter 2 (Analysis of Current Problem)	15	10 th October 2011	10 th October 2011	12 th October 2011	12 th October 2011	Completed chapter 2

37	4	Write Chapter 4 (Proposed Solution)	20	18 th October 2011	23 rd October 2011	22 nd October 2011	29 th October 2011	Completed chapter 3
38	4	Write Chapter 5 (Prototype evaluation)	20	23 rd November 2011	23 rd November 2011	25 th November 2011	25 th November 2011	Completed chapter 4
39	4	Write Chapter 6 (Project Evaluation)	20	26 th November 2011	26 th November 2011	28 th November 2011	28 th November 2011	Completed chapter 5
40	4	Write Chapter 7 (conclusion)	20	29 th November 2011	29 th November 2011	30 th November 2011	30 th November 2011	Completed chapter 6
Total Hours : 110								
41		Print Dissertation	1	30 th November 2011	30 th November 2011	30 th November 2011	30 th November 2011	
42		Proof read dissertation	4	30 th November 2011	30 th November 2011	30 th November 2011	30 th November 2011	
43		Make edits	2	3 rd December 2011	3 rd December 2011	4 th December 2011	4 th December 2011	Completed Dissertation
44		Bind Dissertation	2	5 th December 2011	5 th December 2011	6 th December 2011	6 th December 2011	
Total Hours : 9								
Total Hours : 627								

Appendix 5 – Gantt Chart

				Previous Weeks	Week 1	Week 2	Week 3	Week 4
					29th Aug 2011	5th Sept 2011	12th Sept 2011	19th Sept 2011
Task Number	Objective	Task Discription	Hours					
1	1	Initial client meeting	3		3			
2	1	Genertae Project Proposal	3		3			
3	1,2	Generate Terms Of Reference(ToR)	5			5		
4	1,2	Edit ToR	2			1	1	
5	5	Generate Schedule and Gantt Chart	5			5		
6	5	Confirm research area	5				5	
7		Plan Dissertation layout	3				3	
8	2	Review of current system	15					15
9	2	Research into Jena Framework	20					20
10	2	Research into NeOn Framework	20					5
11	2	investigation into Protégé	15					
12	2	Research into SPARQL	15					
13	2	Research into OWL	25					
14	2	Investigate text extraction from structured XML	15					
15	2	Investigation into Ontology based Intergration of Information	20					
16	2	Investigation into Non-ontological intergration	20					
17		comple all notes and information gathered	10					
18	2,3,6,	write draft literature review	20					
19		discuss literature review with supervisor	2					
20	2,3,6,	act upon comments for supervisor in task 19	5					
21	4,6	create prototype design	10					
22	4,6	generate UML class diagrams for system	5					
23	4,6	create system to auto read finicial reports	20					
24	4,6	create system to pull info from reports	20					
25	4,6	generate database to store data	35					
26	4,6	link tasks 24 and 25	35					
27	4,6	combine aspects with existing system	35					
28	4,6	edit existing system to incorporate multiple sets of data	35					
29	4,6	write test suite	10					
30	4,6	white box tesing of system	15					
31	4,6	black box testing of system	15					
32	4,6	create questionnaire of end user test	10					
33	1,4,6	end user test	15					
34	4,6	edits to system	20					
35	4	Write Chapter 1 (Introduction)	15					
36	4	Write Chapter 2 analysis of current problem)	15					
37	4	Write Chapter 4 (Proposed Solution)	20					
38	4	Write Chapter 5 (prototype evalyation)	20					
39	4	Write Chapter 6 (Project Evaluation)	20					
40	4	Write Chapter 7 (Conclusion)	20					
41		Print Dissertation	1					
42		Proof read dissertation	4					
43		make edits	2					
44		bind dissertation	2					

Total Planned Hours

627

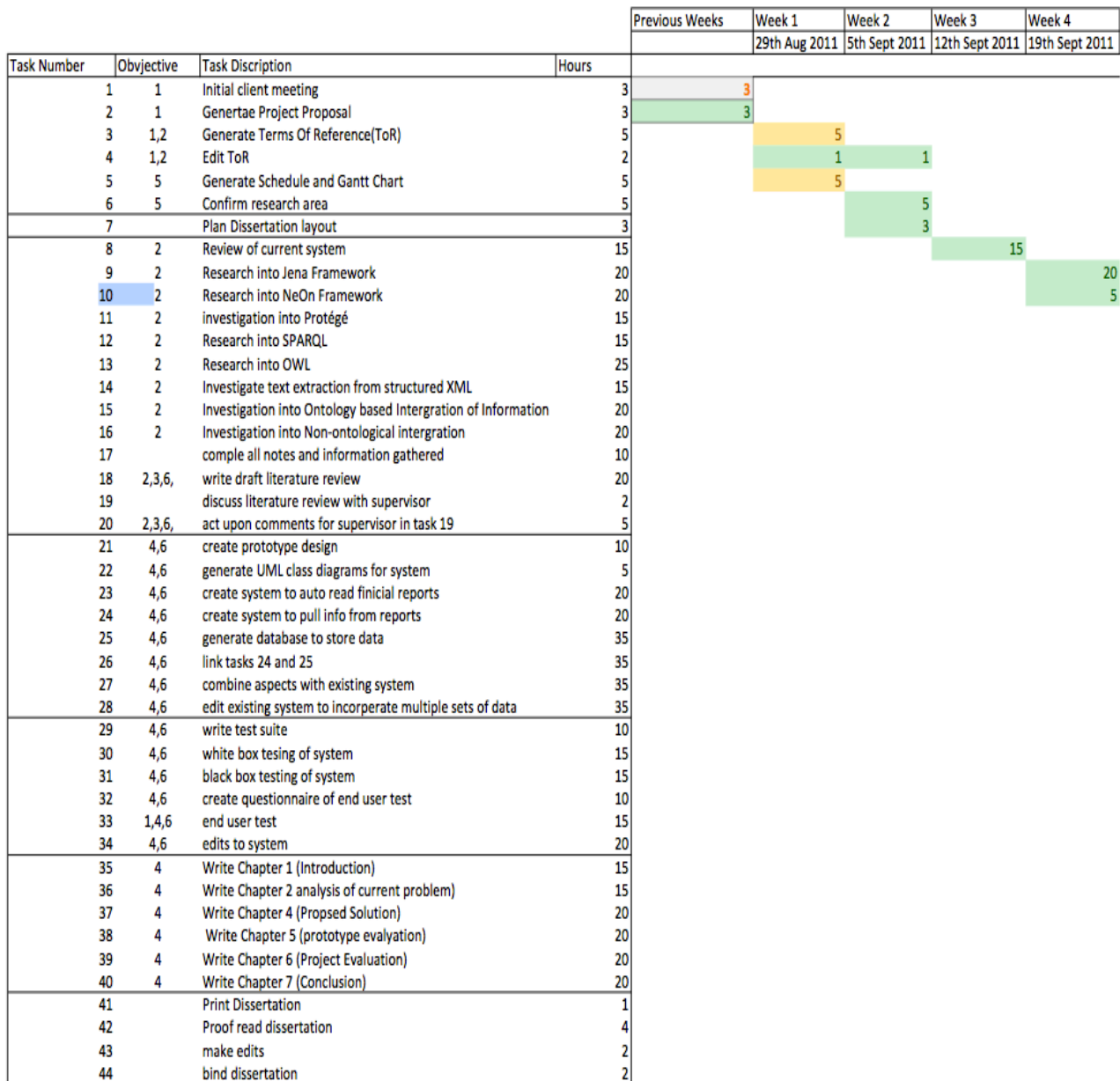
Task Completed on Time

Task Completed Ahead of time

Task completed Late



Appendix 6 – (up to date) Gantt Chart



Total Planned Hours

627

Task Completed on Time

Task Completed Ahead of time

Task completed Late



Appendix 7 – Risk Analysis Documents

Risk Item Name: Illness			Risk ID: 1
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the developer becomes ill during the project.			
Risk Statement Consequence(s): <u>then</u> the progress of the project may be affected.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the project	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> <p>This risk cannot be prevented.</p>			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> <p>Due to the working arrangements for the project minor illness will have limited effects on the development of the project, this is due to the amount of time the developer will be 'out of action' will be a short period of time which will mean at the most a task may run 2 days late.</p> <p>Being in constant contact with both my supervisor and my client, if illness should hinder any meetings that have been planned the developer can contact the client/supervisor to postpone.</p> <p>Should the developer suffer 'Major' illness, which would result in the developer being ill for more than a couple of days then the developer will have to liaise with the supervisor, and client to discuss the problem and if needs be change the deadline.</p>			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Change of Requirements			Risk ID: 2
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the client decided to change the requirements of the system.			
Risk Statement Consequence(s): <u>then</u> work completed up to that stage could be wasted, the prototype may have to be reworked or edited to accommodate the new requirements.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Following initial Project specification	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> <p>The requirements will be discussed with the client before any development begins, this will allow for the developer to ensure that the requirements meet the clients expectations for the system.</p> <p>The client will also be informed that large changes to the requirements will not be possible after they have been agreed. This is due to the short time frame that is in place to completed the project.</p>			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> <p>Should the client want to change any of the requirements, it is proposed that the client will meet with the developer and the supervisor to agree any requirements that are reasonable to incorporate into the system.</p>			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Task Timing Underestimate			Risk ID: 3
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the developers estimate for the competition of a task.			
Risk Statement Consequence(s): <u>then</u> a unfinished task, could result in the project failing			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the project	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> <p>When generating the schedule and Gantt Chart, estimates were given to time and effort that the developer believed would be needed to complete the project, these have to be checked by the project supervisor to ensure that they are achievable.</p> <p>Although the schedule and Gantt Chart have been created they are just a rough idea of timings. Regular meetings with both the client and the project supervisor will ensure that the project doesn't run to far behind schedule.</p>			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> <p>Actions will be taken by firstly myself if tasks begin to take longer than expected a meeting with the project supervisor to discuss what to do next.</p>			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Underestimate of Developers Skills			Risk ID: 4
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the developer underestimates their programming skills and the project becomes more complicated than 1 st thought			
Risk Statement Consequence(s): <u>then</u> this could result in the project not being completed to the satisfaction of the client.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the development	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> <p>The timings that have been applied via the Gantt Chart and schedule, however following the 1st review session the timing my need to be edited.</p>			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> <p>Progress will constantly reviewed with both the developer and the project supervisor. This will allow the developer to discuss any problems/concerns that may arise</p>			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Failure of Technology			Risk ID: 5
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the third party software used should fail			
Risk Statement Consequence(s): <u>then</u> the developer will have to edit the prototype.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the development	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> All work that is completed will be stored in several locations, in case of any program should fail. The main aspect that may fail is the open source software, should this happen the developer may have to edit the prototype to incorporate the changes.			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> With the developer storing several copies then should any aspect of software fail there will be very little work lost, this will mean less chance of a complete re-write. Should the worst happen then the developer will have to have a 'crisis' meeting with the project supervisor to discuss what the next step is.			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Technology not Available			Risk ID: 6
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the technology that has been specified as to be used within the prototype not being available.			
Risk Statement Consequence(s): <u>then</u> a completed task my have to re-developed.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the development	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> During the requirements phase the developer will check that all third party frameworks are current, and available to use.			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> If a specific framework is not available, the developer will have to ask the project supervisor which way I should deal with it best. Should this happen the developer will also have to report to the client and inform them of the changes.			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Client pulling out of project			Risk ID: 7
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the client decided that they no longer wanted to progress with the project			
Risk Statement Consequence(s): <u>then</u> . The project would become a failure.			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the project	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> Meeting will constantly happen between both the project supervisor as well as the client to keep them up to date with the progress of the project			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> If the client should pull out of the project, development will cease until a final decision has been made. And if the worst should happen then an emergency meeting will be held between the supervisor and the developer.			
Risk History:			
Date	Event	Author	
Current Date			

Risk Item Name: Client stops communication			Risk ID: 8
Author : Paul Fairley			
Risk Statement Condition: <u>if</u> the client stops responding to emails requesting their input			
Risk Statement Consequence(s): <u>then</u> the prototype that will be developed may not be to the clients satisfaction			
Probability	Low / Medium / High	Impact	Low / Medium / High
Earliest the risk could occur.	Beginning of the development	Latest the risk could occur.	Completion of the project
Mitigation Plan(s): <i>[to prevent/reduce the chance that the situation will occur]</i> During the requirements faze the developer will ensure they know exactly what is wanted from the prototype.			
Contingency Plan(s): <i>[to deal with situation when it occurs]</i> Should the client stop communications the developer will have a meeting with the project supervisor and try and work out a solution. Have a second contact within the organisation to ensure that should the sponsor cease contact then the developer can still continue with communication.			
Risk History:			
Date	Event	Author	
Current Date			

Risk Distribution Matrix¹²

		Impact		
		High	Medium	Low
Probability	High		T1	
	Medium	T5	T2	
	Low	T3/T4/T6 /T7/T8		

Risks Distribution on September 2011

		Impact		
		High	Medium	Low
Probability	High		T1	
	Medium	T5	T2	
	Low	T3/T4/T6 /T7/T8		

Risks Distribution on October 2011

		Impact		
		High	Medium	Low
Probability	High	T3	T1	
	Medium	T5	T2	
	Low	T4/T6/T7 /T8		

Risks Distribution on November 2011

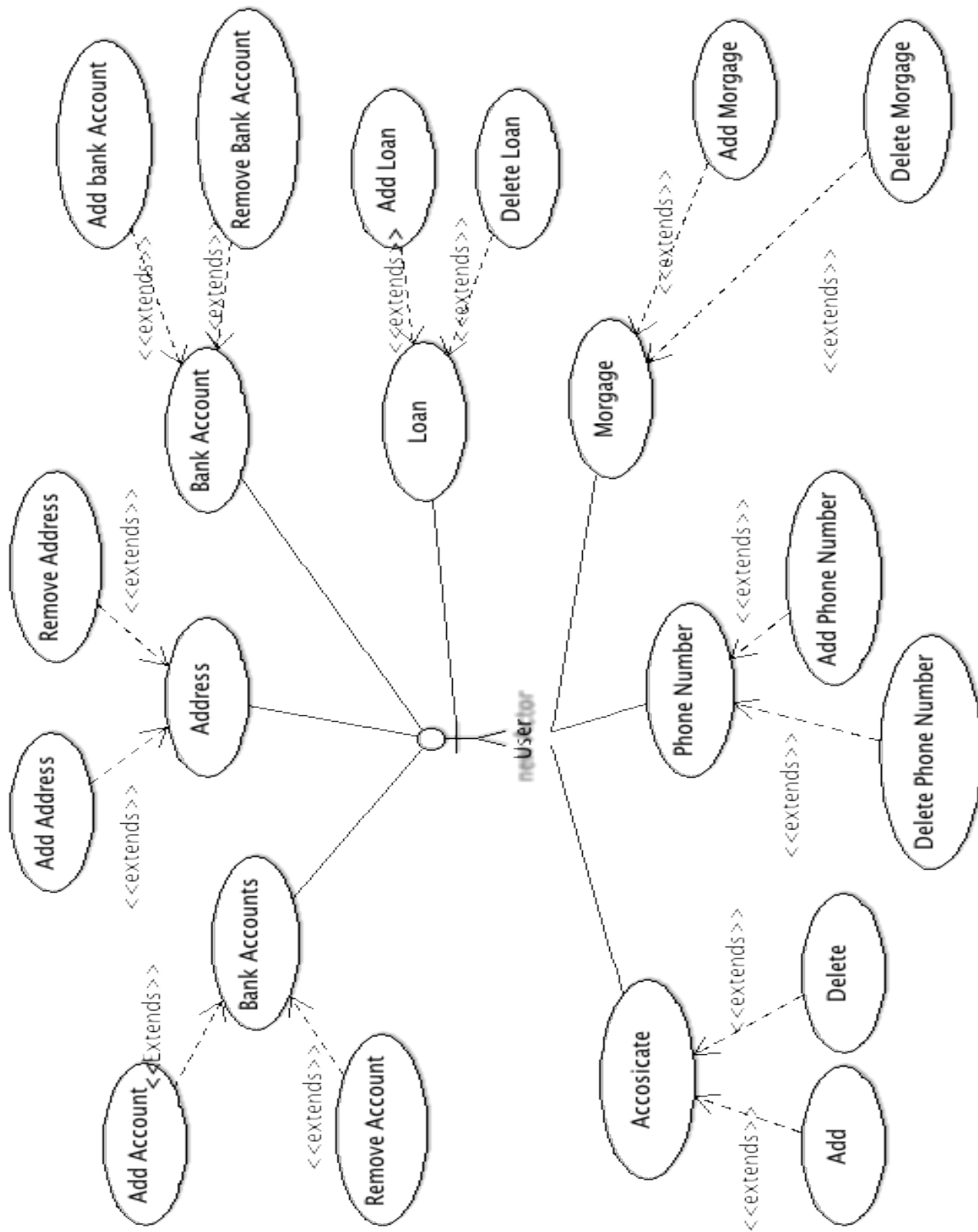
		Impact		
		High	Medium	Low
Probability	High	T3	T1	
	Medium	T5	T2	
	Low	T4/T6/T7/ T8		

Risks Distribution on December 2011

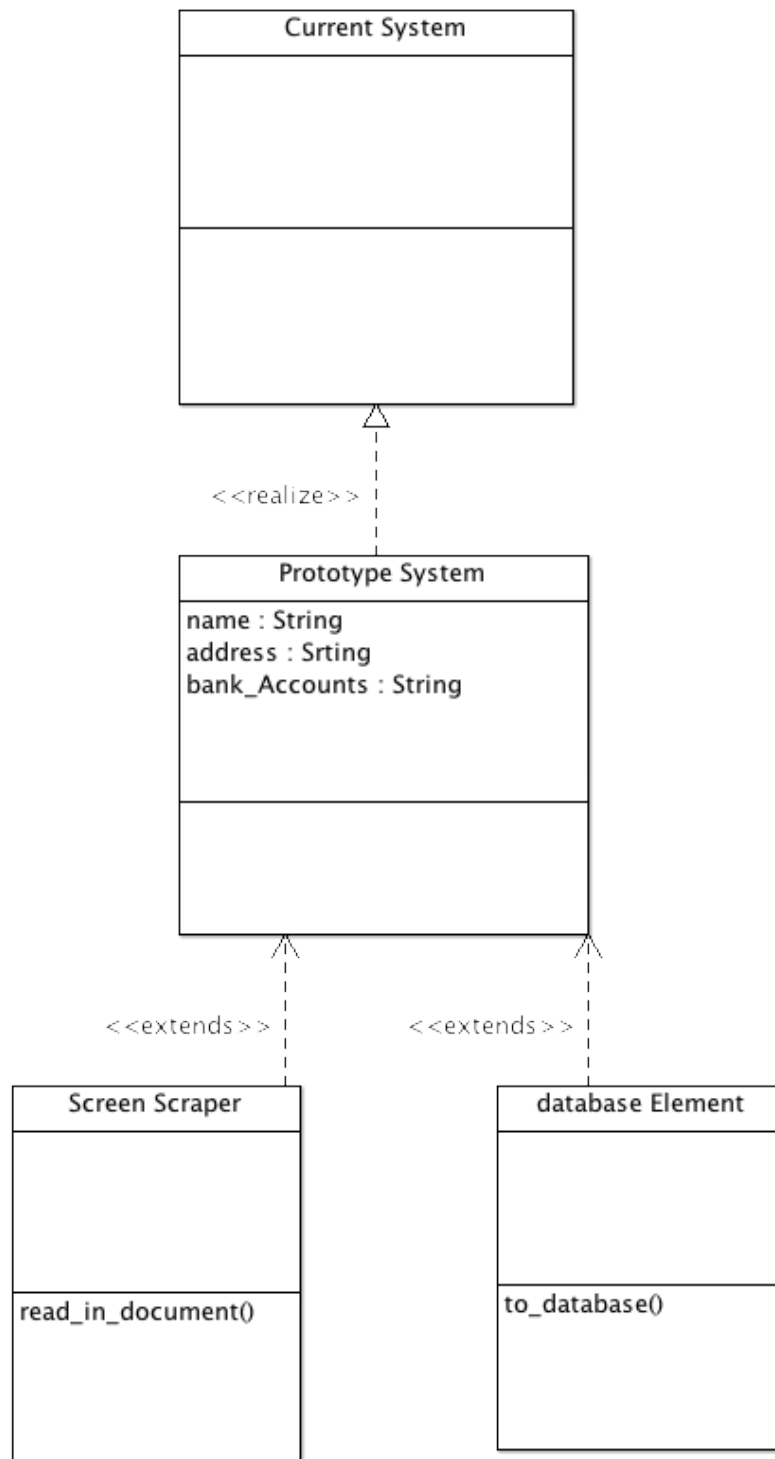
¹ Each time you review your risks you should map them against a risk matrix

² Use this in conjunction with your risk specifications.

Appendix 8 – System Use Cases



Appendix 9 – System Class Diagram



Appendix 10 – System Designs

accounts_held

Field	Type	Null	Default	Comments	MIME
investigation_number	varchar(8)	No			
default/active/closed	varchar(25)	No			
title	varchar(5)	No			
given_name	varchar(50)	No			
surname	varchar(50)	No			
house_number	varchar(5)	No			
address_1	varchar(50)	No			
address_2	varchar(50)	No			
town/city	varchar(50)	No			
postcode	varchar(6)	No			
date_of_birth	date	No			
company_type	varchar(250)	No			
company_name	varchar(250)	No			
account_type	varchar(100)	No			
account_number	varchar(15)	No			
status_1_2(36months)	int(3)	Yes	NULL		
status_3+	int(3)	Yes	NULL		
start_date	date	No			
default_date	date	No			
current_balance	double	No			
default_balance	double	No			
payment_terms	varchar(75)	No			
CAIS_last_update	date	No			
joint_account	varchar(1)	Yes	NULL		
current_status	int(2)	No			
worst_status	int(2)	No			

No index defined!

aliases_and_associates

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(8)	No		contact_details -> investigation_number		
name	varchar(250)	No				
is_alias_of	varchar(250)	No				
house_number	varchar(6)	No				
address_1	varchar(50)	No				
address_2	varchar(50)	No				
town/city	varchar(50)	No				
postcode	varchar(8)	No				
on	date	No				
source	varchar(100)	No				

No index defined!

contact_details

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(10)	No		contact_details -> investigation_number		
title	varchar(5)	No				
given_name	varchar(25)	No				
surname	varchar(50)	No				
house_number	varchar(5)	No				
address_1	varchar(100)	No				
address_2	varchar(100)	No				
town/city	varchar(50)	No				
postcode	varchar(8)	No				
number	int(5)	No				
last_amount	decimal(7,0)	No				
latest	date	No				

Indexes:

Keyname	Type	Unique	Packed	Field	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	investigation_number	0	A		

 [bel.sunderland.ac.uk](#) ▶  [bd79uq](#) ▶  [_credit_status](#)

credit_status

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(10)	No		contact_details -> investigation_number		
sreaches_0-3months	int(3)	No				
sreaches_4-6months	int(3)	No				
sreaches_7-12months	int(3)	No				
number_of_accounts_held	int(3)	No				
total_balance	decimal(10,0)	No				
worst_current	int(3)	No				
worst_historical	int(3)	No				

No index defined!

credit_appliction_serches

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(8)	No		contact_details -> investigation_number		
title	varchar(5)	No				
given_name	varchar(25)	No				
surname	varchar(50)	No				
house_number	varchar(5)	No				
address_1	varchar(50)	No				
address_2	varchar(50)	No				
town/city	varchar(50)	No				
postcode	varchar(8)	No				
aplication_type	varchar(100)	No				
company_name	varchar(50)	No				
company_type	varchar(50)	No				
aplication_date	date	No				
date_of_birth	date	No				
time_at_address	varchar(25)	No				

No index defined!

 [bel.sunderland.ac.uk](#) ▶  [bd79uq](#) ▶  [other_addresses](#)

other_addresses

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(8)	No		contact_details -> investigation_number		
number	varchar(2)	No				
title	varchar(5)	No				
given_name	varchar(25)	No				
surname	varchar(50)	No				
searched_address	varchar(1)	Yes	<i>NULL</i>			
house_number	varchar(5)	No				
address_1	varchar(50)	No				
address_2	varchar(50)	No				
town/city	varchar(50)	No				
postcode	varchar(8)	No				
link	varchar(5)	No				
date	date	No				

No index defined!

gone_away_information

Field	Type	Null	Default	Links to	Comments	MIME
investigation_number	varchar(8)	No		contact_details -> investigation_number		
title	varchar(5)	No				
given_name	varchar(25)	No				
surname	varchar(50)	No				
house_number	varchar(6)	No				
address_1	varchar(50)	No				
address_2	varchar(50)	No				
town/city	varchar(25)	No				
postcode	varchar(8)	No				
member_number	int(4)	No				
information_date	date	No				

No index defined!

Appendix 11 – Test Plan

Test Number	Description	Test Action	Expected Result	Actual result
1	Open HTML to check the content is correct	Open mr_peter_jones.html	Credit report opens in browser	As expected
2	Extract data from contact details	Turn HTML file to XML	new XML file is created	File created but only XML tags added (not all mark up included)
3	Extract data from contact details	Turn HTML file to XML	new XML file is created	File created but only XML tags added (not all mark up included)
4	Extract data from contact details as plan text	Use SimpleHTMLDom to extract data	New txt file is written to a temp directory	As expected
5	Extracted data is pulled to the xml file	Use java create a valid XML with Tags	Xml document to be created with tags	As expected
6	Move data from plan text to database	Use SQL commands to add data to database	New enter in contact_details table	SQL error
7	Move data from plan text to database	Use SQL commands to add data to database	New enter in contact_details table	SQL error
8	Move data from plan text to database	Use SQL commands to add data to database	New enter in contact_details table	Added MR Peter Jones in the same column
9	Use temporal database to split data	Use SQL commands to add data to database	New enter in contact_details table	Added MR Peter Jones in the same column

Appendix 12 – Document ‘A’ – HTML code

Below shows a extract from the code of the HTML document that the constabulary receive from the third party sites.

```
1 <DIV class=PrintOnly id=Banner>
2
3 <TABLE id=PrintBanner cellSpacing=0 cellPadding=0 width="100%"
4
5 summary="Print Banner" border=0>
6
7 <TBODY>
8
9 <TR>
10
11 <TD width="100%"><IMGOFF style="FLOAT: right" width="600"
12
13 src="../../../CEMS/IOL2/images/Experian_logo_prn.gif" alt="Experian Logo"
14
15 height="50"></TD></TR></TBODY></TABLE></DIV>
16
17 <DIV class=ReportSection id=SectionSummary name="rptSection"><A
18
19 name=summary></A>
20
21 <TABLE class=ReportSectionHeader style="WIDTH: 100%">
22
23 <TBODY>
24
25 <TR>
26
27 <TD style="WIDTH: 45%; TEXT-ALIGN: left">Summary</TD>
28
29 <TD style="WIDTH: 45%; TEXT-ALIGN: right">Search Date :&nbsp;
30
31 06/08/2010</TD></TR></TBODY></TABLE>
32
33 <DIV class=ReportSectionSubHeader style="WIDTH: 100%">Subject Details</DIV>
34
35 <TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
36
37 <TBODY>
38
39 <TR class=rptSectionStripe>
40
41 <TD class=rptdatalabel>Name:</TD>
42
43 <TD class=rptdata colSpan=6><SPAN style="TEXT-TRANSFORM: capitalize">MR
44
45 PETER JONES </SPAN></TD>
46
47 <TD class=rptdatalabel></TD>
48
49 <TD class=rptdatalabel></TD></TR>
50
51 <TR>
52
53 <TD class=rptdatalabel>Address:</TD>
54
55 <TD class=rptdata colSpan=8>2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8
56
57 8HH</TD></TR></TBODY></TABLE>
58
59 <TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
60
61 <TBODY>
62
63 <TR>
64
65 <TD>
66
67 <DIV style="FLOAT: left; WIDTH: 48%">
68
69 <TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>
```

Appendix 13 – XML Document

```
1 <?xml version="1.0" encoding="us-ascii"?>
2 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
3 <HTML>
4   <HEAD>
5     <META NAME="generator"
6       CONTENT="HTML Tidy for Linux/x86 (vers 11 February 2007), see www.w3.org" />
7     <TITLE>Data Subject Report - MR PETER JONES</TITLE>
8     <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=us-ascii" />
9     <META CONTENT="MSHTML 6.00.2900.6003" NAME="GENERATOR" />
10    <META CONTENT="C#" NAME="CODE_LANGUAGE" />
11    <META CONTENT="JavaScript" NAME="vs_defaultClientScript" />
12    <META CONTENT="http://schemas.microsoft.com/intellisense/ie5"
13      NAME="vs_targetSchema" />
14    <LINK ID="onetidThemeCSS"
15      HREF="Data%20Subject%20Report%20-%20MR%20PETER%20JONES_files/IOL21011-109.css"
16      TYPE="text/css" REL="stylesheet" />
17    <STYLE TYPE="text/css" XML:SPACE="preserve">
18  BODY {
19    FONT-WEIGHT: 100; FONT-SIZE: 0.8em; MARGIN: 0px; COLOR: #006; FONT-FAMILY:
20    arial,Verdana,sans-serif; BACKGROUND-COLOR: #fff
21  }
22  TABLE {
23    FONT-SIZE: 100%; COLOR: #006; FONT-FAMILY: arial, Verdana, sans-serif
24  }
25  TH {
26    FONT-FAMILY: arial, Verdana, sans-serif
27  }
28  DT {
29    FONT-WEIGHT: 700; MARGIN-LEFT: 12px
30  }
31  A {
32    TEXT-DECORATION: none
33  }
34  A:hover {
35    TEXT-DECORATION: none
36  }
37  INPUT {
38    FONT-SIZE: 100%; FONT-FAMILY: arial, Verdana, sans-serif
39  }
40  TEXTAREA {
41    FONT-FAMILY: arial,Verdana,sans-serif
42  }
43  #ifRunningProcess {
44  }
45  #lLeftBar {
46    BORDER-RIGHT: medium none; BORDER-TOP: medium none; BACKGROUND:
47    url(_layouts/CEMS/IOL2/Images/IOL_nav_logo.gif) #eee no-repeat 0% 100%; LEFT: 0px;
48    BORDER-LEFT: medium none; WIDTH: 212px; BORDER-BOTTOM: medium none; POSITION: absolute;
49    TOP: 0px; HEIGHT: 100%
50  }
51  #lProcessList {
52    BACKGROUND: url(_layouts/CEMS/IOL2/Images/proclist_bg.gif) no-repeat;
53    MARGIN-LEFT: 12px; WIDTH: 199px; POSITION: absolute; TOP: 32px
54  }
55  #lHeader {
56    LEFT: 0px; WIDTH: 700px; POSITION: absolute; TOP: 0px; HEIGHT: 370px;
57    BACKGROUND-COLOR: #eee
58  }
59  #lHeaderTitle {
60    BACKGROUND: url(_layouts/CEMS/IOL2/Images/IOL_Experian_background.jpg) #171774
61    no-repeat; LEFT: 0px; WIDTH: 100%; POSITION: absolute; TOP: 0px; HEIGHT: 369px
62  }
63  #divErrorMessages {
64    DISPLAY: none; MARGIN: 40px 0px 0px; WIDTH: 781px; HEIGHT: 0px
65  }
66  BODY.ProcessBody {
67    BORDER-LEFT: medium none; BORDER-TOP-STYLE: none; BORDER-RIGHT-STYLE: none;
68    BORDER-BOTTOM-STYLE: none
69  }
```

Macintosh HD:Users:Fazza:Desktop:p.iones.xml: 1/37

Appendix 14 – Document ‘A’ – Screenshots

Data Subject Report – MR PETER JONES

02/12/2011 11:10

Summary

Search Date : 06/08/2010

Subject Details			
Name:	MR PETER JONES		
Address:	2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH		
Public Information		Previous Credit Searches	
Number	1	0-3 months	0
Total amount	£1,200	4-6 months	0
Latest	05/06/2010	7-12 months	2
Messages (click for details)		Accounts Held	
CIFAS data present		Number	7
Reported Gone Away (GAIN)		Total balance	£8,694
Previous Address		Worst current	8
Notices of Correction		Worst historical	8
Notices of Correction			
You are legally obliged to read all NOCs before making any assessment or decision regarding the applicant(s)			
Y2			

Notices of Correction

Data Dispute - Y2	
/2 "THE INDIVIDUAL CONCERNED WISHES TO REQUEST THAT THIS DATA IS USED WITH CAUTION WHEN ASSESSING CREDITWORTHINESS."	

Voters Roll

2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH		
JONES	MR PETER	October 2000 to current register
STEPHENS	MISS MARIA	October 2000 to current register

Aliases and Associations

1 record

Aliases	
MR PETE M JONES	is an alias of MR PETER JONES
at 2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH	
on 11/02/2006	source EXPERIAN SYSTEMS DEPARTMENT,

Other Addresses

1 record

Previous and Forward Addresses			
	Name	Address	Link
2	MR PETER JONES	Searched Address - 2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH	
1	MR PETER JONES	54, CARDEN PLACE, ABERDEEN, ABERDEENSHIRE, AB101UN	To C
			11/2005

Telephone Numbers

0 records

NO DATA FOUND FOR THIS SECTION

Previous Fraud Cases (CIFAS)

1 record

CIFAS Records	
Name used:	MR PETER JONES
Address used:	2, ABBEY LANE, EDINBURGH, MIDLOTHIAN, EH8 8HH
Date of birth:	18/12/1972

file:///Users/Fazza/Documents/Data%20Subject%20Report%20-%20MR%20PETER%20JONES.htm

Page 1 of 4

Appendix 15 – Learning Logs

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 11 th September 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Terms of Reference</p>		
<p>2. <i>Time Spent on above work: 10</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>work on the terms of reference was essential to ensure that aspects of the project were heading in the right direction.</p> <p>Following interviews with my client I could define the first half of the terms</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>it was essential to have a meeting with my client to ensure both parties knew exactly which direction the project would be heading in.</p> <p>this also allowed me to get materials that will be needed for the project</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p> <p>I have gathered information that will be vital to the project</p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Amend Terms of Reference after supervisor has checked them.</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 18 th September 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Amending Terms of Reference, began research into 'Intergrating non-ontological resources into sematic based investigative systems' by looking into the existing technologies.</p>		
<p>2. <i>Time Spent on above work: 15</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>following my supervisors comments my terms of reference was edit to incorporate aspects that were missing.</p> <p>Research began into the Jena and NeOn framework</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>I felt it was important to get a grasp of how the two frameworks incorporate the aspects that I will be using within the system, and which would best suit what I am trying to achieve</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Further research into aspects of the frameworks and begin looking into clearly defining my research question.</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 25 th September 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>defining my research topic, editing my Terms of Reference as well as beginning work on making notes on my research to build my chapter 2</p>		
<p>2. <i>Time Spent on above work: 15</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>during my weekly meeting with my supervisor, we discussed what was already in place for the terms of reference, and how I could further what was already in place. During this discussion I asked what my supervisor thought of my research topic, as whether he felt that it could be applied to my proposed system. After he agreed, my week was to start reading into different papers and books to try and start to form my second chapter</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>with my research topic clearly defined I was able to clearly look into the defined subject and make a start on chapter 2.</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Following more research the last links can be added to the terms of reference and send it on to my sponsor.</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 2 nd October 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Completed terms of reference and emailed it onto my Sponsor (Dave Sampson) to sign. Completed work on Chapter 1 of dissertation</p> <p>More research into the chosen topic</p>		
<p>2. <i>Time Spent on above work: 25</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>The ToR was lacking the signature of the client so it was essential to sent that to them to get their approval for the proposed solution, and with the client being based in London for the majority of the week then it became apparent that email was the best way to get the ToR signed.</p> <p>Further to this completing the 1st chapter of the dissertation at this stage meant that the developer could concentrate on more important aspects of the work on the prototype and project for the remainder of the project.</p> <p>Finally further research was completed via both books and journals to give a better range of ideas from, the writers.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>all aspects of work were essential this week, and working in the manner that I did ensured the aspects were completed in good time.</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Problem : 1st review session on Monday, this could lead to edits in aspects of the project, and may cause a slight delay why aspects are resolved.</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 9 th October 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Beginning work Chapter 2, Beginning work on Chapter 3, had a client meeting, and further research into Information Management & the technologies I will be using.</p>		
<p>2. <i>Time Spent on above work: 35</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Continuing with the writing of the dissertation, to ensure that the documentation for the project would not run behind.</p> <p>Further research into the topics named allowed the chapter 3 to be more robust, as well as to give the developer to get a better grasp of what is needed to develop a successful project.</p> <p>Finally this week, the developer met with the client to ensure the requirements that have been worked on, were both achievable in the time that the developer has, as well as finalise details that they also wanted incorporating.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>The work on the chapters of the dissertation was essential to ensure that the developer doesn't have more, and more paperwork to work on as they are developing the prototype system.</p> <p>Meeting with the client was a great experience at this point in the project, as it allowed the developer to ensure that both the developer and the client are on the same 'wave length'</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p> <p>As described I think that meeting the client days before development is due to start has given the developer a better idea of what is expected from the prototype.</p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 16 th October 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Chapters 2 (finishing touches) Chapter 3</p>		
<p>2. <i>Time Spent on above work: 40</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Adding the finishing touches to chapter two, by finishing reference to back up arguments regarding UML</p> <p>Chapter three was concerning writing out the literature review.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Finish up chapter 3, and begin programming.</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 23 rd October 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Chapter 3 Began programming</p>		
<p>2. <i>Time Spent on above work: 35</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>while working on chapter i asked my supervisor to check I was on the right lines, he suggested slight changes to chapters one and two, as well as the structure of chapter 3</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>working on the edited that the supervisor advised me, would improved the flow of the dissertation.</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p> <p>however with the edits that were suggested, this knocked the timing of the schedule off ever so slightly, so programming began late.</p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later</i></p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 30 th October 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>8Worked on the programming of the prototype system beginning with the creation of the database, as well as working through chapter 4.</p>		
<p>2. <i>Time Spent on above work: 45</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Beginning work on the database for the system is essential for the running of the system, also working on chapter 4 as I worked it enabled me to do figures as I was going.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>after discovering the problem with the documents given by the constabulary was in a HTML document XML so rather than doing nothing the developer could work on aspects of the system such as the database.</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p> <p>beginning of next week I have a meeting with my supervisor to rectify the problem above.</p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 6 th November 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>meeting my supervisor regarding the problem, meeting with client, development of prototype as well as continuing chapter 4</p>		
<p>2. <i>Time Spent on above work: 45</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>after meeting with my supervisor he suggested that I meet with my client to see if the problem can be rectified. While waiting on interaction from the client I worked on chapter 4 further.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p> <p>following meeting with my client it was discovered that all documents are as HTML and not xml as first discussed. So following meeting my supervisor after meeting my client it was suggested that using screen scraping the data and generating XML from the screen scraped data.</p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p> <p>Research is needed into screen scraping</p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 13 th November 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>adding screen scraping to chapter three further development.</p>		
<p>2. <i>Time Spent on above work: 35</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Research was conducted into screen scraping both to be added to chapter three as well as to gain a better understanding of which tools to use in current system.</p> <p>This then led into further development of the prototype system</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 20 th November 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Further development Working on dissertation</p>		
<p>2. <i>Time Spent on above work: 40</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Developing the system so that data could be successfully extracted. Working on dissertation to ensure all tasks are completed for the hand in date</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 27 th November 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Further development Working on dissertation</p>		
<p>2. <i>Time Spent on above work: 40</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>Developing the system so that data could be successfully extracted. Working on dissertation to ensure all tasks are completed for the hand in date</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

--

Name: Paul Fairley	Programme & Level: MSc Software Engineering	Date: W/E 4 th December 2011
DESCRIPTION		
<p>1. <i>This week I worked on:</i></p> <p>Further development Working on dissertation</p>		
<p>2. <i>Time Spent on above work: 45</i></p>		
REFLECTION		
<p>3. <i>Explain how you did the work listed in section 1:</i></p> <p>moving the extracted data to the database finishing up on last touches on dissertation.</p>		
<p>4. <i>Explain why you worked in the manner described above:</i></p>		
<p>5. <i>Think about and write down what you have found out/learned from your actions this week:</i></p>		
CARRY FORWARD		
<p>6. <i>Highlight any questions, problems, tentative conclusions to follow up on next week or later.</i></p>		

Appendix 16 - Formal Meetings with Supervisor

RECORD OF FORMAL MEETINGS WITH SUPERVISOR

Student Name	PAUL PANDREY	Date: 15 SEPT 2011
Supervisor(s)	A. BOKNA	Time: 11AM

MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING

- Need to develop an integral of datasets + new biological resources
- Need to decide on focus of what to concentrate on in terms of work.

OTHER COMMENTS

- Need to concentrate on European report

AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING

- concentrate on integral of datasets with structural/datasets resources
 - ↳ do some research on list in Madrid
 - ↳ build up collection of literature
- hopefully meet with Don Gappa next week to discuss focus + get additional data

Date, Time & Location of next meeting:

Initialed Supervisor(s)

Initialed Student

RECORD OF FORMAL MEETINGS WITH SUPERVISOR

Student Name PAUL FAIRLEY Date: 21st SEP 201
Supervisor(s) A. BOWMA Time: 10AM

MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING

- Has worked on Andrew's Supph
 - ↳ trying to find how it stores data
- Need feedback for what the researchers in his project
 - ↳

OTHER COMMENTS

- Paul is able to abstract input to the E. reports
 - ↳ needs to incorporate the with Andrew's descriptions

AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING

- ▷ Albert to look at TOR about defining research objectives
- ▷ Paul to email and verify TOR to Albert
- > proposed next week with Don Supph 7 October TBC

Date, Time & Location of next meeting:
Initialed Supervisor(s)



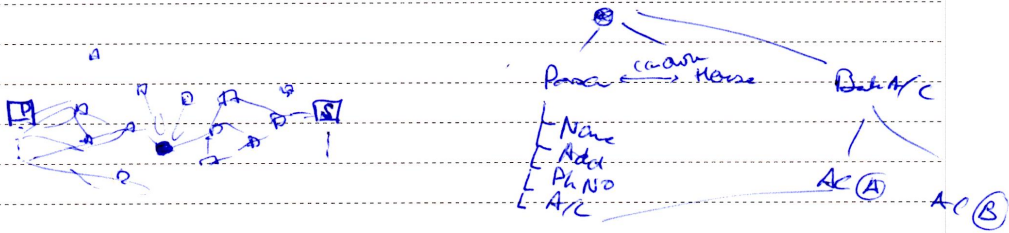
Initialed Student



RECORD OF FORMAL MEETINGS WITH SUPERVISOR

Student Name P. FAIRLEY Date: 17th Sept
 Supervisor(s) A. BOCKM Time:
 MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING

- TOP on the way and set to D. Castels
- Continuing Reading
- Continue work with technology



OTHER COMMENTS

- Have looked at technology
 - ↳ try to understand how Admin System works
 - ↳ look at his ontology
- Set a draft thesis

AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING

- Read Phil's ontology tutorial → for Bridge Workshop → Access by Hemridge
- Move to see Hemridge draft
 - ↳ Review order of public analysis, lit. Review & solution

Date, Time & Location of next meeting:
 Initialed Supervisor(s)

CB

Initialed Student

[Signature]

RECORD OF FORMAL MEETINGS WITH SUPERVISOR

Student Name	PAUL FAIRLEY	Date:	5/10/2011
Supervisor(s)	A. BOKHA	Time:	

MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING

- looked at what might be counts that would topic for 1st Exam
- looked at list of products for Medical
- looked at NEON toolkit plugins
- Nearly completed chapter 1
↳ N

OTHER COMMENTS

- Needs to work on problem sheet (chapters 1 + 2)
- Need to look at the book reviews
 - make sure to discuss w/ R/T + your problem
 - look at problem analysis
 - discuss & interpret problem + for possible conclusion

AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING

- get on with lit review + problem analysis
- keep expanding technology
- explain Andros system to use next week

↳ Meeting will start at Friday 10AM !

Date, Time & Location of next meeting:

Initialed Supervisor(s)

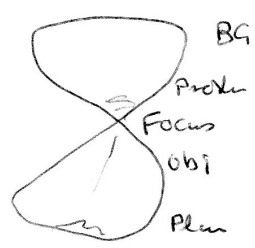

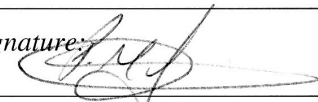
Initialed Student

CB

[Signature]

DEPARTMENT OF COMPUTING, ENGINEERING & TECHNOLOGY

Project Supervision Record

Student: PAUL FAIRLEY	Date: 20/10/2011
<p><i>Main points of discussion/Progress or issues since last time:</i></p> <p>Draft run of thesis with Chapt 1 - 3</p> <div style="text-align: right; margin-top: 20px;">  </div>	
<p><i>Recommended actions:</i></p> <ul style="list-style-type: none"> - Chapt 1 needs refining w/r/t focus + direction + background - Chapt. 2 needs refining with detailed problem analysis + <ul style="list-style-type: none"> • prob analysis • Lit Review • design + impl (test) • evaluate/expert. • writing 	
<p><i>Agreed deliverables for next time:</i></p> <ul style="list-style-type: none"> - Review draft thesis in line with discussion 	
<p><i>Other comments:</i></p>	
<p><i>Date & time of next meeting:</i></p>	
<p><i>Supervisor name & signature:</i></p> 	<p><i>Student signature:</i></p> 

DEPARTMENT OF COMPUTING, ENGINEERING & TECHNOLOGY
Project Supervision Record

Student: PAUL FAIRLEY

Date: 25/10/2011

Main points of discussion/Progress or issues since last time:

- Watched an interactive review → due in 4 weeks
↳ has considered the design of tests + evaluation early stage

Recommended actions:

Make sure you look at the breadth of literature on different topics
+ compare + contrast + justify your choices!

Agreed deliverables for next time:

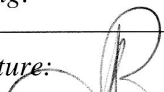
- Need to complete literature review draft.

Other comments:

-

Date & time of next meeting:

Supervisor name & signature:



Student signature:



SCHOOL OF COMPUTING AND TECHNOLOGY

Project Supervision Record

Student: PAUL FAIRLEY	Date: 9/11/2011
------------------------------	------------------------

Main Points of Discussion/Progress/Issues:

- Handed in 2nd Dev → Achat to give feedback shortly.
- Working on databases and Andrew's syth
 - ↳ working errors in current syth
 - ↳ problem with sample data as XML record was empty

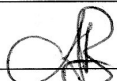
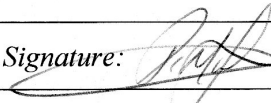
Recommended Actions:

- Request new test data for Cerastibulary.
- Delay the test by a week → 26 November
 - ↳ Currently working on reducing redundant records

Next Deliverables:

Other Comments:

Meets with Dave Supson on Monday 14/11/2011 at 10 AM confirmed.

Supervisor Signature: 	Student Signature: 
--	--

RECORD OF FORMAL MEETINGS WITH SUPERVISOR

Student Name PAUL FAIRLEY Date: 17/6/2014
Supervisor(s) A. BOKMA Time:
MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING

- saw Don Supra to get samples
 - ↳ problem with E-reports
 - ↳ his mail addresses arranged
 - ↳ scan sampling
 - ↳ finders for useful bits
- working on the dissertation
 - review of last Review
 - add new material & evidence plan.

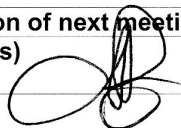
OTHER COMMENTS

- organised test session with Don next Tuesday
- making progress.

AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING

- Works on data report
 - ↳ have to reorg. Asper system
- problem of HTML to XML conversion

Date, Time & Location of next meeting:
Initialed Supervisor(s)



Initialed Student



RECORD OF FORMAL MEETINGS WITH SUPERVISOR (FACE-TO-FACE)		
Student Name	Paul Fairley	Date: 25 th November 2011
Supervisor(s)	Albert Bokma	Time: 11.25
MAJOR PROGRESS MADE & ANY PROBLEMS ENCOUNTERED SINCE LAST MEETING		
Spoke to albert on a skype call!!		
<i>Discussed the progression of the project, and the prototype system</i>		
<i>Spoke about test session on Tuesday (29th Nov), albert is teaching so it will just be Paul and Sponsor.</i>		
OTHER COMMENTS		
<i>We will have a meeting Monday/Tuesday (following test session to catch up on what has been completed)</i>		
AGREED ACTIONS & DELIVERABLES FOR NEXT MEETING		
Date, Time & Location of next meeting:		
Initialed Supervisor(s) Albert Bokma		Initialed Student Paul Fairley

DEPARTMENT OF COMPUTING, ENGINEERING & TECHNOLOGY

Project Supervision Record

Student: PAUL PAIRLEY	Date: 29/11/2011
<p><i>Main points of discussion/Progress or issues since last time:</i></p> <ul style="list-style-type: none"> - Have made progress on dissertation - Have made progress on the system 	
<p><i>Recommended actions:</i></p> <ul style="list-style-type: none"> - Used Access input fields to extract data - Left meeting → ensure conform to instructions → put docs in appendix → make sure chapters are connected 	
<p><i>Agreed deliverables for next time:</i></p> <ul style="list-style-type: none"> - Finished dissertation to submit by next Tuesday. 	
<p><i>Other comments:</i></p> <ul style="list-style-type: none"> - Problem with meeting with D. Symon by cancelled at last minute 	
<i>Date & time of next meeting:</i>	
<i>Supervisor name & signature:</i>	<i>Student signature:</i>