

CMO343 Final Year Project

# **Simply Wales Information System**

2011/2012

Author: Charlotte Swinnerton

Scheme: Information Systems Bsc

Supervisor: W K Ivins

Moderator: R J White

This project discusses the solution to the problem of limited resources within an SME. The project adopted a waterfall project development approach, and each aspect of this approach has been discussed in this document. A thorough description of the implementation and development stages are contained in this document since the development of a prototype was the intended deliverable. Although time constraints threatened the success of the project, it has succeeded in delivering two prototypes complete with not only requirements and interface designs, but a set of instructions on how to develop the prototypes further. There is an end goal in sight that can only be achieved through development after the project has ended.

## TABLE OF CONTENTS

|  |         |
|--|---------|
| Introduction                                 | Page 03 |
| Background                                   | Page 08 |
| Specification and Design                     | Page 10 |
| Implementation                               | Page 31 |
| Testing                                      | Page 43 |
| Evaluation of Project Deliverables using SSM | Page 44 |
| Evaluation of Project Approach               | Page 49 |
| Future Work                                  | Page 52 |
| Conclusion                                   | Page 55 |
| Reflection                                   | Page 56 |
| References                                   | Page 58 |

## INTRODUCTION

---

Simply Wales is a local organisation where I spent a significant amount of time assisting the company owner, Peter Morgan, with marketing activities organised through Go Wales.

Simply Wales is a Cardiff-based small-medium IT enterprise (SME) based within the heart of Cardiff. Peter Morgan has over 20 years experience in the IT and Telecoms industry which gives reason to the extensive list of products and services that Simply Wales offers.

The fundamental purpose of Simply Wales is to provide IT solutions to local small to medium sized enterprises (SMEs), but of course the extensibility of the products and services offered welcomes a wide variety of customers from home-owners to individuals requiring IT equipment, too large for Simply Wales to accommodate.

In hindsight, the problem we are attempting to solve is that Peter Morgan, the owner of Simply Wales, has a significantly large workload. Given that he is the only employee of Simply Wales at this time, progress and completion of business activities is slow. As a result, Simply Wales is limited to the number of customers it can service which of course produces a low turnover, little profit and restricts growth and expansion.

The proposed solution is an Information System that will automate specific business activities thus reducing Peter Morgan's workload, increasing efficiency and improving the customer service experience. I believe this will allow Simply Wales to expand and grow to a true medium-sized enterprise as the number of customers Simply Wales will be able to accommodate will no longer be limited.

### *Significant Project Changes*

This section of the report will discuss the change of direction in the project, detailing differences to the original direction set out in the Interim Report to highlight the impacts this change has had upon the project.

The original aim, as stated in the Interim Report, is to 'assess the benefits of an Information System and determine the capabilities and readiness of Simply Wales to successfully implement such an information system.' After further development of the project I would now consider the aim to be to 'demonstrate the need for an Information System in Simply Wales'. These aims are more alike than I first realised since 'assessing the benefits' and 'demonstrate the need' both give reason as to why Simply Wales should adopt the Information System.

Despite this slight change, the original aim to 'assess the benefits of an Information System and determine the capabilities and readiness of Simply Wales to successfully implement such

an information system' has been achieved to some extent. The literature review addresses the benefits of an information system, discussing potential problems that might occur before, during or after the implementation of an Information System. By highlighting the potential problems that could occur, Simply Wales can compile a risk assessment to determine how likely these problems are to occur, which would suggest their current level of capability.

Furthermore, the high-level requirements tell us what the system requires not just in terms of functionality, but in terms of the IT equipment or staffing levels needed. By looking at their assets, Simply Wales are able to determine if they are capable and ready to implement the Information System, they will be if they possess what is required.

### *Approach*

The key word in the revised aim is to 'demonstrate'. The original statement to 'assess the benefits' suggests the project involves a Soft Systems Methodology approach, the revised aim suggests an iterative project development approach.

In the Interim Report I had briefly discussed the potential to develop a prototype; this was dismissed when I assumed the project would not develop further than the design stages. A further reason for deciding against a prototype is that I assumed 'the creation of a prototype requires more skills relevant to the Computer Science degree scheme'.

It is clear I misjudged my skill set since I explicitly stated that I would make use of relevant tools and techniques taught throughout my degree scheme, I failed to notice that a proportion of those tools and techniques included aspects of the Computer Science degree scheme, and I am in fact very capable of implementing a prototype which I have achieved in this project.

The project development lifecycle, or system development lifecycle as titled in the Interim Report, was originally an alternative approach to Soft Systems Methodology. Soft Systems Methodology (SSM) did not remain the approach of choice after the change of direction as SSM is used as a means for finding the problem, whereas business process modelling is more suited to finding a solution.

The new change of direction focused on the solution to the problem as opposed to an understanding of the problem itself. As I already have a great understanding of the problem from the time I spent at Simply Wales, it seemed inappropriate to use the SSM approach.

Business process modelling, through the use of activity diagrams, focuses on the individual business processes which is a more efficient method to finding a direct solution. By modelling the activities, it is much easier to decide which of these activities are candidates for automation, which is what the Information System is intending to do.

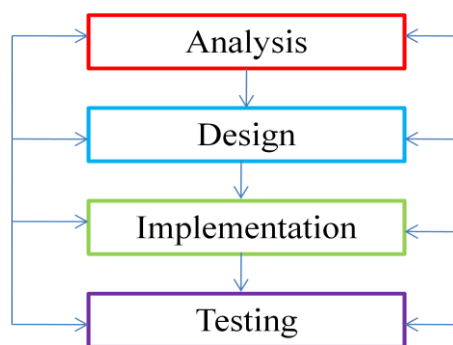
That said, Soft Systems Methodology(SSM) has been favourable in the evaluation of the system. The SSM is discussed in the evaluation section of the report and shows what I expect

the system to be able to do, and how the project has achieved these expectations. SSM has also helped identify areas for future work where the project has not addressed certain aspects of the system.

The second alternative discussed in the Interim Report is system dynamics. This would have had a similar output to Soft Systems Methodology in the sense that its focus is also upon the problem and not the solution. System dynamics is useful for determining what influences and factors exist to affect an organisation, but again this would not have any use in finding a solution to an already, well understood problem.

In addition, it would have been difficult to find factors that affect Simply Wales due to the size of the organisation and the small amount of resources they currently have. System dynamics would be much more useful to a larger organisation. There is a possibility for system dynamics to be used by the organisation if expansion and growth occurs, and more people are involved in the organisation.

The lifecycle the project adopted is illustrated in the diagram below.



I still intend to use heuristic evaluations as a means of testing although it will be considered future work. I have considered the use of test cases, even producing a potential test case for the prototypes in their current state. Throughout the implementation stage I have been informally testing the system to ensure it functions as intended.

Influence diagrams and SWOT analysis have been replaced with the use of business process modelling. SWOT Analysis is a useful tool for analysing an organisations ‘strategic position’ within its environment. While the identification of assets would help Simply Wales understand its current level of capability and readiness to implement an Information System, it is not within the scope of the project at this time.

Influence diagrams are a form of System Dynamics, as mentioned previously Influence Diagrams give more emphasis on understanding the problem, then actually finding a solution to the problem. Again, the problem is already understood without the aid of SSM, SWOT analysis or Influence Diagrams, and neither of these methods would add any insight into the solution.

## *Project Scope*

Due to time constraints the project has only drawn attention to the remote and business support activities. In summary, the remote and business supports are two separate packages that contain a number of pre-defined services that can be purchased by SMEs for a monthly fee. The full set of activities offered by Simply Wales, including the activities the remote and business support packages can be found in the Interim Report.

Although I have modelled all the activities in the remote and business support packages, only two of those activities have been implemented as a prototype. These two activities are customer registration and appointment booking, arguably two of the core activities within Simply Wales, and more than certainly the two that will have a significant impact on Peter Morgan's work load.

Previously I considered all of the activities in the business and remote support package, but due to time constraints and the change in direction it was appropriate to only develop two of the business activities that showed the most promise for success.

## *Assumptions*

For this project I have assumed Simply Wales has access to the Information Technology required to accommodate an Information System. I have also assumed Peter Morgan will have the finance available to implement the Information System, if there are any costs, should he choose to adopt my proposal put forward in this project.

## *Summary of Outcomes*

This section of the report summarises the key deliverables of the entire project, including those delivered within the Interim Report.

The project began with analysis and research, the research was compiled into a literature review. A large amount of my understanding of Simply Wales was due to observation and the time I spent working there. This was assembled into a number of documents that illustrated the current operation in terms of the full set of products and services Simply Wales offer and a brief comparison to the competition.

Following these initial introductory stages, I used business process modelling to model the activities. I have a set of fifteen activity diagrams with the corresponding use case diagrams. An in-depth description of each activity to justify its purpose is illustrated in a separate document that is discussed later in the report.

Through business process modelling I was able to derive a set of high-level requirements. This led to a conceptual database model design and thoughts about the design of the user

interface. This business process modelling also showed which activities were the best to be automated.

The implementation of the prototypes followed. This involved the installation and set-up of WAMP server which provided me with PHP and My SQL and a platform for developing dynamic web applications. Testing has been informally completed during the implementation stage, but formal test cases have been created based on the current state of the system as an example of how they can be used.

Soft Systems Methodology has been used to evaluate the system by highlighting what the project has achieved. Furthermore, the results of SSM have identified areas of future work where the project has not addressed an activity, and suggests further recommendations for the organisation.

The main outcome of this project will be a proposition to Peter Morgan. This proposition will be composed of a single prototype demonstrating the automation of 2 business activities performed within the remote and business support services.

### *Report Structure*

The background section of this report provides a generalised view of project, discussing existing solutions and alternative solutions that could have been used. This section concludes with statement of the aims and objectives.

The specification and design section provides an in-depth discussion of the design work that was carried out, more specifically the business process models and all other aspects of the design stage including requirements.

The implementation & testing section examines how the implementation was carried out to achieve the desired prototype, including challenges and snippets of code for justification and evidence. This section concludes with some thought to testing and potential test cases.

The evaluation section has been split into two parts; part one is the evaluation of the project deliverables using SSM, and the second part is evaluation of the project approach. Soft Systems Methodology has been used as an evaluative tool to evaluate the project deliverables by matching my expectations of the system to what the project has done to meet these expectations.

The future work section discusses the work that can be followed on from this project. This section counts for a large proportion of this dissertation as I have a lot of ideas about how this project can be taken further, and was unable to develop the prototypes as far as I had like due to time constraints.

The conclusion section offers a comparison between the objectives before the change of direction and after, discussing how each has or has not been achieved. Finally the reflection section ends the report with an account of my personal feelings toward the project and the skills learnt and developed.

## **BACKGROUND**

---

### *Existing Solutions*

IT support is fairly repetitive so there are of course overlaps between Simply Wales and their competitors in terms of the product and services offered, however I feel an Information System will give Simply Wales a great competitive stance through the reformed achievement of the business activities and the greater effect it will have upon the customers.

It is apparent from the variety of software available that there is a need for Information Systems within any sized organisation. In essence, an Information System is a combination of human activity and information technology to support operations and business activities. Therefore an Information System that may satisfy one organisation will not satisfy another since each organisation is unique.

It is highly unlikely that a solution to this problem already exists within Simply Wales since Peter Morgan's work load continues to expand. This expansion suggests no such IT support is currently used to enhance the business activities. I believe Peter Morgan is aware such enhancements exist but does not have the resources to consult experts in the field, nor does he have the knowledge to complete the ground work himself.

Since every organisation is unique and because this project is based upon the requirements and business activities unique to Simply Wales, then this project will be exclusive. Despite this exclusivity, there is software available to support the automation of the activities such as appointment booking solutions which are discussed further in the implementation section.

### *Aim*

The aim of this project is to 'demonstrate the need for an Information System in Simply Wales'

### *Research Question*

The intention of the Information System is to increase the efficiency of the delivery of the business activities involved in remote and business support. By delegating appropriate activities to the Information System, Peter Morgan will have more time to spend on completing valuable and high priority tasks.



In order to demonstrate the achievement of the stated aim, this project will analyse the need for an Information System within Simply Wales and the transition required for successful implementation, making use of UML use case and activity diagrams to model the proposed business processes. Two miniature prototypes will be developed using PHP/My SQL to demonstrate how the automation could be achieved.

### *Objectives*

- Create business process models to gain an understanding of the processes involved with each activity
- Create a prototype to demonstrate how the activities have been, and can be automated
- Understand the technologies that underpin the information system and how best to acquire and implement them.

## SPECIFICATION AND DESIGN

---

The designs are predominantly made up of business process models, more specifically activity diagrams with corresponding use case diagrams. Using the breakdown of the business and remote support activities listed in the Interim Report and illustrated in Figure 1 below, I produced an activity diagram for each activity finishing with a total of fifteen.

The products and services offered by Simply Wales within the business and remote support packages are shown below in Fig 1. Following this I have compared Fig1 with the activity diagrams to show the links between them; this is shown in Figure 2. As I have only offered an in-depth description of two activities that I am implementing, a descriptive table summarising the remaining activity diagrams is shown in Fig 5. All fifteen activity diagrams can be found in Appendix A.

*Fig 1 Business & Remote Support Activity Breakdown*

| Service Level                                  | Activities   |
|--|--|
| <b>Business Support</b><br>30p a day           | IT support for a server and up to 10 pc's or apple Macs.   |
|  | Unlimited telephone and e-mail support   |
|  | Unlimited remote repairs   |
|  | Online call logging & monthly fault reports  |
|  | Up to 4 free callouts per month & reduced extended callout fees  |
|  | Scheduled antivirus, malware & spyware checks  |
|  | Scheduled backup checks  |
|  | Real time pro-active sever monitoring  |
|  | Software updating and patching   |
|  | Security & penetration testing   |
|  | Competitive pricing on IT equipment, printer & toner cartridges & display kit.   |
|  | 8 hour response time   |
| <b>Remote Support</b><br>starts at just £19.99 | With your permission, we can connect to any PC, Mac or sever around the world within a few second. We can remote control your machine as if we were sitting right in front of it. Saving you time, money and hassle. |
|  | Perform powerful diagnostics on your computer  |
|  | Detect and remove viruses, malware and trojan's  |
|  | Update your operating system giving you access to the latest software and fixes  |
|  | Free hard drive space by deleting unnecessary files  |
|  | Speed up your PC start-up time   |

| Activity ID | Previous corresponding activity                      | Following corresponding activity | Activities in support package satisfied   |
|-------------|--|----------------------------------|---|
| B.A1        | A13  | Portal available for use         | IT support for up to 10 pc's or apple Macs. Unlimited remote repairs and competitive pricing on IT equipment.   |
| B.A2        | A11  | A12a<br>A12b<br>A14              | Up to 4 free callouts per month & reduced extended callout fees<br>8 hour response time<br>Monthly fault reports  |
| B.A3        | A11  | A12a<br>A12b<br>A14              | Unlimited Remote Repairs<br>Monthly fault reports<br>IT support for up to 10 PCs or Apple Macs.<br>8 hour response time   |
| B.A4        | A11  | A12a<br>A14                      | Scheduled antivirus, malware & spyware checks<br>Monthly fault reports<br>IT support for a server and up to 10 PCs or Apple Macs.<br>Software updating and patching<br>Security & Penetration Testing<br>Scheduled Back-ups |
| B.A5        | B.A1   | A12a<br>A12b<br>A14              | Real-time pro active server monitoring  |
| R.A6        | A11  | A12a<br>A14                      | Perform powerful diagnostics on the customer's computer   |
| R.A7        | A11  | A12a<br>A14                      | Detect and remove viruses, Trojans and malware  |
| R.A8        | B.A1<br>A11  | A14                              | Update your operating system giving you access to the latest software and fixes   |
| R.A9        | B.A1   | A14                              | Free the hard drive space by deleting unnecessary files   |
| R.A10       | B.A1   | A14                              | Speed up your PC start-up time  |
| A11         | B.A1   | Not Applicable                   | Not Applicable  |
| A12a        | All  | Not Applicable                   | Not Applicable  |
| A12b        | All  | Not Applicable                   | Not Applicable  |
| A13         | None   | B.A1                             | Not Applicable  |
| A14         | B.A2<br>B.A3<br>B.A4<br>B.A5<br>R.A6<br>R.A7<br>R.A8 | Not Applicable                   | Monthly progress reports  |

|  |               |  |  |
|--|---------------|--|--|
|  | R.A9<br>R.A10 |  |  |
|--|---------------|--|--|

*Figure 2 Corresponding Activities*

Four of the fifteen activities, A11, A12a, A12b and A13, do not directly satisfy the products and services in the remote and business packages, they do however support the activities that have a direct impact on these products and services. Although these activities did not pre-exist within the support packages they are necessary for the automation of the activities.

Fig 1 shows the extensive list of products and services offered by Simply Wales that compose the business support and remote support packages. It is from these I derived the set of activity diagrams. Fig 2 to show how the activities support and satisfy the remote and support packages, it demonstrates how the activities link together and how the automated activities support the remaining activities within the support packages.

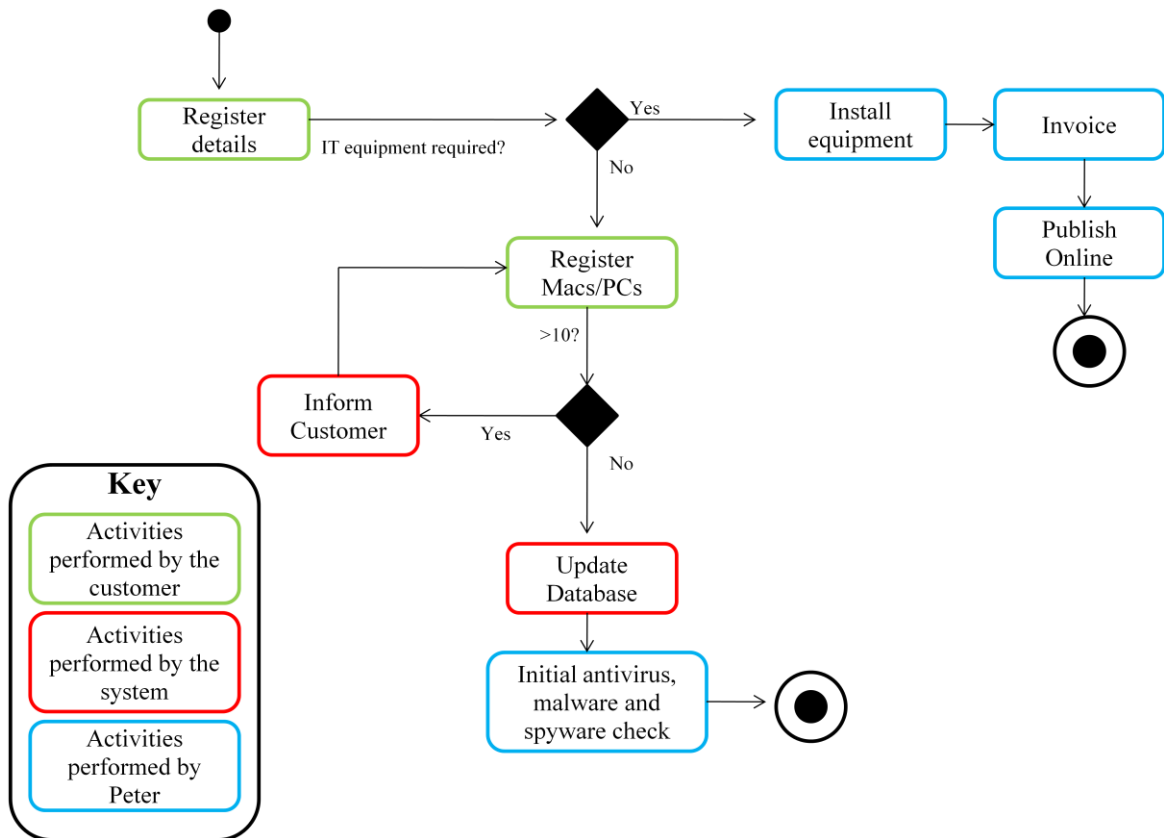
The activity diagrams have been evaluated to decide which of the activities would be best to automate in terms of how much it would reduce Peter Morgan's work load, and how much more efficient the activity in question would be.

From studying Fig 2 it is evident that the core activities B.A1 and A11 must be completed before a number of the activities can be achieved. Since these two activities play a major role in the achievement of these other activities, they have been selected as the two core activities to be implemented as a prototype.

I could have used a different layout when creating the activity diagrams by using partitions to show the different actors; instead I felt the use of colour easily shows the actors and is easy to interpret. The activity diagrams help to identify the core chunks of functionality that need to be developed during implementation.

For each activity I will discuss the comparison between the current processes and the proposed processes, as modelled in the activity diagrams, to show what improvements have been made by the proposed process.

## A1 Registering a New Customer



*Fig 3 Activity Diagram B.A1*

Fig 3 is the corresponding activity diagram to the activity B.A1 registering a customer. The following paragraphs will describe the process modelled above.

### *Current Process*

Prior to customer registration, the customer will have made a query about what Simply Wales can offer them; although customer queries is a separate activity, it is important to note how this activity can have a negative impact on the customer registration activity.

Currently potential customers make a query by contacting Peter via email or telephone. As a number of the existing customers are occupying Peter's time for trivial purposes then these queries are often missed and a potential customer is lost. Registration can be completed without a customer to register.

It comes as no surprise that customer registration does not currently exist. Should a customer choose to purchase support services from Simply Wales, after receiving a response to their query, Peter records the customer's details and discusses the agreements of the partnership between the customer and Simply Wales.

### *Proposed Process*

This activity will follow on from activity 13, customer queries. On registering, the customer will be prompted to fill in an online form detailing the contact name, organisation name, e-mail, contact number, level of service (business support or remote support) username and password, subscription to the e-newsletter and monthly payment details.

It is possible that the customer may wish to purchase IT equipment from Simply Wales since they refurbish and sell second hand equipment and are trained in the installation of this equipment.

One of the restrictions for the business support is that it will only support up to 10 PCs/Macs. This information will need to be added on registration where an ID number or IP address unique to each of the 10 PCs/Mac will be inserted. This may need to be added after registration once Peter has assigned an ID number to each PC/Mac or acquired the IP address of these machines.

A further activity delivered under business support is antivirus, malware and spyware check. This is a separate activity, however it is beneficial for an introductory check to be conducted on registration as it may have a negative impact on the other activities and become quite costly to the customer if the customer were to attract these threats.

Looking at Fig 2, activities B.A5, R.A8, R.A9, R.A10 and R.A11 are reliant on this activity before they can be performed. This is because in all these activities Peter needs information on the Macs or PCs that will benefit from the activities, this information is provided on registration when the customer details the Macs and PCs that will be included in the service. It is also important for Peter to know the identity of the PCs and Macs so that an accurate report can be produced on the outcomes of the activities.

### *Current Process Vs Proposed Process*

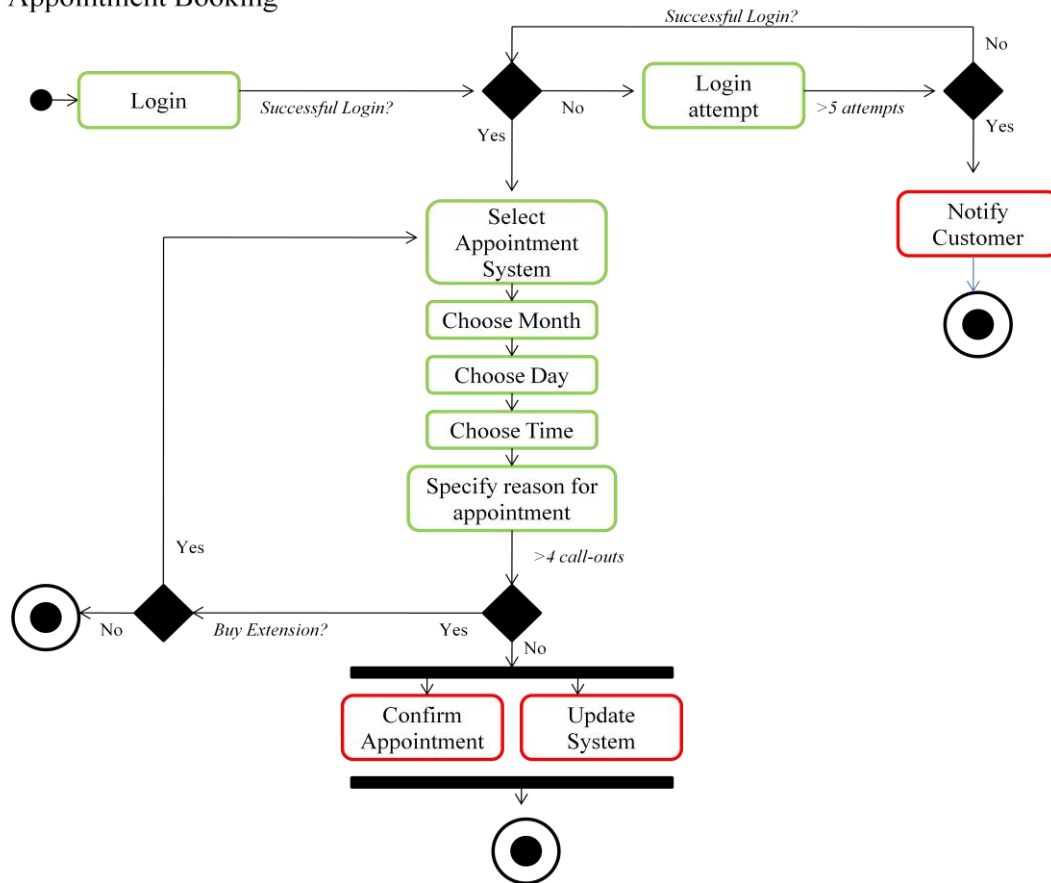
The 4-call outs a month restriction is not adhered to since certain customers require a lot of assistance from Peter for trivial repairs and problems, this may be because the customer is unaware of the conditions of the partnership with Simply Wales.

Currently, the information stored on customers is easily accessible and unprotected; therefore Simply Wales do not conform to the Data Protection Act which could have serious consequences if the information remains unprotected.

Rather than recording the information when communicating with the customer, and risk the loss of that information, the registration form allows the customer to input their own information in their own time, and the registration form controls what information is collected from the customer. It is then stored in a database which will have the necessary access controls to ensure the information is protected.

Not only will this activity store the information accordingly, it will ensure Simply Wales have the necessary information stored about the customers that may be needed in other activities.

#### Appointment Booking



S.

*Fig 4 A11 Appointment Booking*

#### Current Process

As it stands, appointments made by customers are recorded manually. A customer will contact Peter with a problem, Peter will make a note of the problem and will agree a date and time with the customer to resolve the problem. Usually, Peter will attend to the problem as soon as he has been contacted by the customer to keep within the promised 8-hour response time. If it is late in the day when Peter receives details of the problem then he will either attend to it on his journey home, or it will be his first priority the morning of the following day.

If Peter is completing a task with high-importance then he will leave the task un-finished to attend to the customer. Peter's task list is ever expanding due to the number of unfinished tasks that he begins but does not finish due to attending to customer problems.

There is no way of keeping record of the number of call-outs a customer has had in a month. Therefore, Simply Wales are missing out on collecting revenue from extended call-outs as Peter is not charging for these extensions.

### *Proposed Process*

Looking at Fig 2, this activity supports activities B.A2, B.A3, B.A4, R.A6, R.A7 and R.A8 as these are the activities that require the use of the appointment booking system.

Appointments are booked through a customer portal which the customers must login to to be able to access. B.A1 must be completed before this can take place since the login details are decided and confirmed on registration, without registration the customer will not have any login details and therefore will not be registered by the system and unable to login. After five failed login attempts the customer the customer will be notified and will be prompted to change their login details and security information.

When booking an appointment, the customer will first be presented with a calendar where they can scroll month by month choosing the day and time of the appointment they wish to book. On selecting the day and time, the customers information will automatically be added since they are logged in, but they will need to specify the purpose of the appointment. If the number of call-outs in a month exceeds four, the customer will be notified that they have exceeded their limitation and offered an extension where they can purchase the call-out at an additional cost.

### *Proposed Process Vs Current Process*

With an appointment system, Peter will be able to track the number of call-outs a customer has had as the system will restrict bookings if the limit has been reached. As they are prompted for an extension, additional charges will automatically be added to the invoice so that Peter can see the correct amount that a customer owes at the end of the month.

An appointment system will also allow Peter to finish tasks as he can plan his time accordingly. Customers will not be able to ring up and demand Peter's attention at that exact time; they will have to book a time slot from those available on the system.

Peter will be able to amend the time slots so that the time he has delegated to performing certain tasks will appear as unavailable on the system to ensure he is not disturbed.



### *Remaining Activity Diagrams*

The table below offers a description of the remaining activity diagrams; of course it does not include B.A1 and A11 since they have already been discussed. I needed to understand what each activity involved in order to determine if and how they can be automated, and what the requirements would be. From the activity diagrams I was able to derive a set of high-level user and system requirements.

| Activity ID | Activity Name   | Activity Description   |
|-------------|---|--|
| B.A2        | Call-out and response time log  | The customer will request a call-out using the online appointment booking system. The customer will specify the nature of the appointment; if they have exceeded their limit of 4 call-outs in one month then they will be prompted to purchase and extension. If they do not wish for an extension then they will be unable to book an appointment. If an extension is purchased, then the appointment will be confirmed and the appointment system will be updated.  |
| B.A3        | Remote Repairs  | The customer requests a repair using the appointment booking system. When the customer specifies that it is a remote repair they require, they will be prompted to give a short description of the problem to ensure the employee requested to perform the repair has all the equipment needed. The appointment will be confirmed and the appointment system will be updated.  |
| B.A4        | Scheduled antivirus, malware & spyware checks, Software updating and patching, Security & penetration testing and scheduled backup checks | <p>Activity A4 combines five activities. The purpose of grouping together these activities is to ensure it is minimally invasive and disruptive to the customer. This will be performed monthly and the customer will book this using the appointment system.</p> <p>The reason the customer books the appointment rather than Peter Morgan is to ensure it is performed on a date and time that will least disrupt their daily operations. The customer will be notified when this activity is due to be carried out.</p> |
| B.A5        | Real Time Pro Active Server Monitoring  | This is a relatively small activity since updates will be outputted regularly. If a fault is found, Peter will be notified who then notifies the customer recommending what course of action to take. The customer will continue by booking an appointment for the course of action to be  |

|       |                                   |   |
|-------|-----------------------------------|---|
|       |                                   | performed.  |
| R.A6  | Diagnostics                       | The diagnostic is performed monthly. The customer will be notified that this is due and can book an appointment for the diagnostics to be performed. Although this activity is performed remotely, it means the machine will be out of use to the customer, therefore it is favourable for them to book an appointment slot since Peter Morgan will not be available during this time when the activity is being performed. |
| R.A7  | Detect Viruses, Malware & Trojans | This will be performed monthly. The customer will book this using the appointment system when notified it is due to be performed. Again, this activity is performed remotely; it means the machine will be out of use to the customer, therefore it is favourable for them to book an appointment slot since Peter Morgan will not be available during this time when the activity is being performed.                      |
| R.A8  | Update Operating System           | This activity is due to be performed once a year. The customer will be notified when updates are due. They will then book an appointment. This activity is performed remotely.  |
| R.A9  | Hard Drive Space                  | This activity is due to be performed every 6 – 12 months. The customer will be notified when updates are due and will then book an appointment. This activity is performed remotely.  |
| R.A10 | PC Start-Up Time                  | This activity is due to be performed every 6 months. The customer will be notified when updates are due and will then book an appointment. This activity is performed remotely.   |
| A12a  | Invoicing – Direct Debits         | The two invoice and revenue collection has been separated into two activities as a direct debit payment is a set fee each month. This will be collected using BACS. The invoice will include all costs for that given month.  |
| A12b  | Invoicing – Additional Costs      | If a customer purchases an additional service then they will need to make a manual payment to Simply Wales as it will not have been collected by the direct debit. The invoice will include all costs for that given month.   |
| A13   | Customer Query                    | To date, the customer can make a query by either searching the information available on the website, or contact Peter by telephone or e-mail and await a response. This activity allows customers to make an online query using an enquiry form. There will be a link on the homepage so it is easily accessible and the query  |

|     |                 |   |
|-----|-----------------|---|
|     |                 | will be sent to Simply Wales' email address.  |
| A14 | Monthly Reports | A monthly reports activity is visible on the majority of the activities in both the business and remote support. The idea behind this activity is that the report will show the results of all the diagnostics and check and repairs etc in that one month. These reports will be available for viewing in the customer portal by the customer so that they compare the results and determine what improvements have been made. |

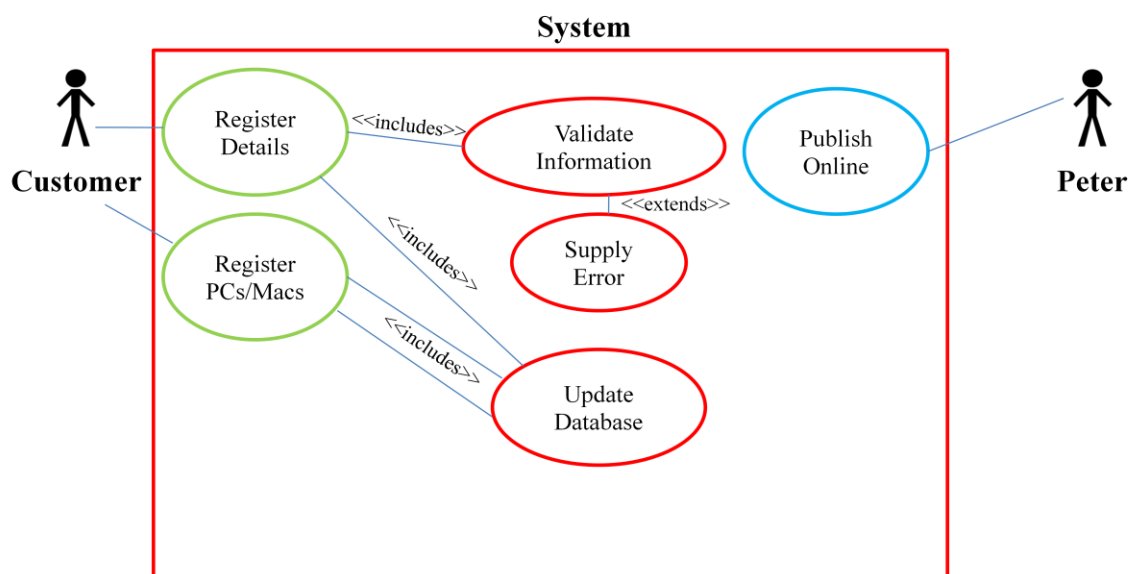
*Fig 5 Activity Diagram Descriptions*

### *UML Use Cases*

Although the use cases correspond to the activity diagrams, not all of the activities required a use case. The activities without a use case are B.A4, R.A9 and R.A10. The reason they do not have a use cases is because there is no use of the system in these activities.

These use case diagrams are useful for capturing the user's needs. They enable me to identify what tasks different users will perform with the system; these need to be reflected in the requirements. See Appendix B for the full set of UML Use Case Diagrams. As with the activity diagrams I will only discuss the two diagrams that relate to the two core activities in this section.

#### B.A1 Registering a New Customer

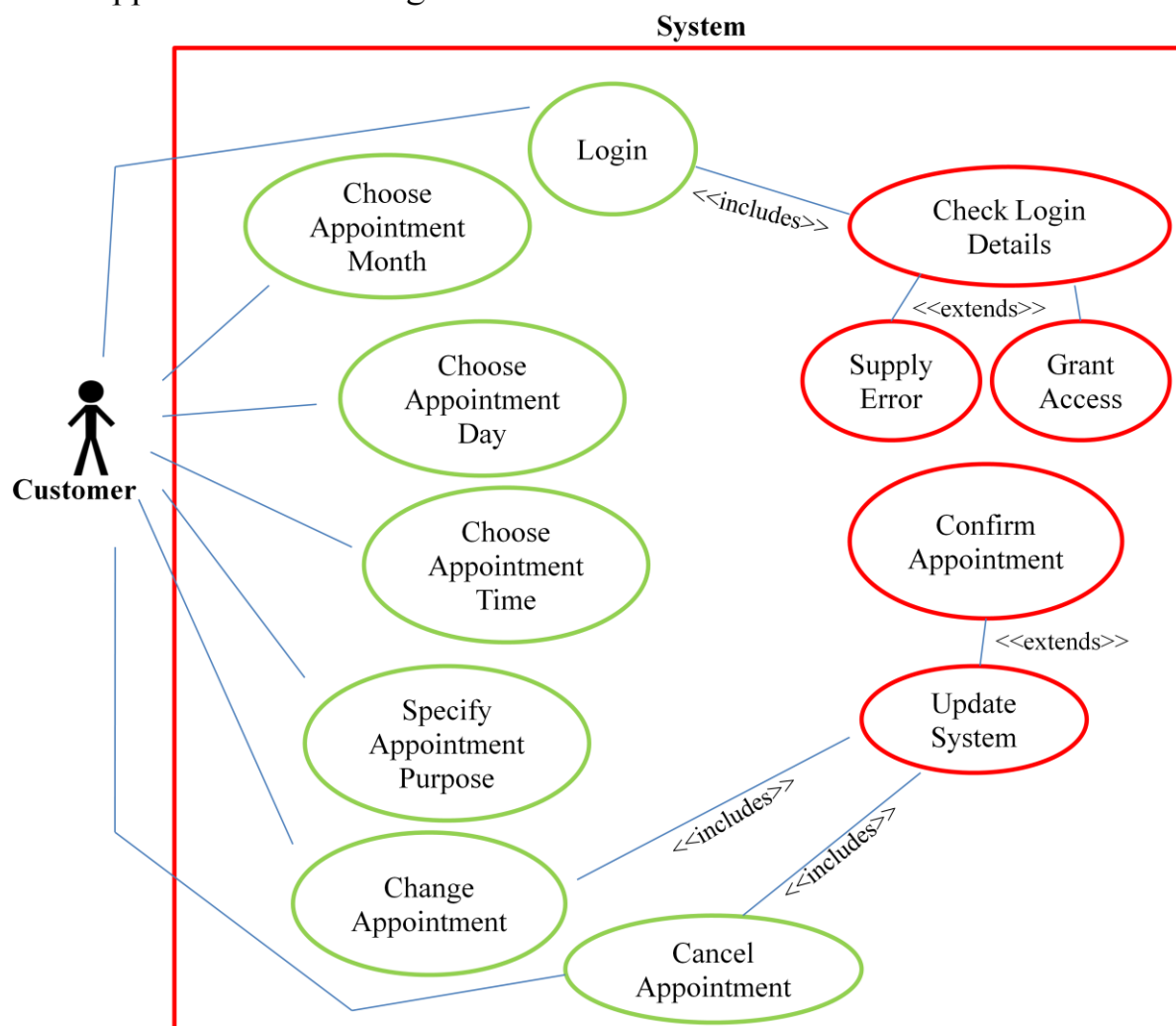


This use case diagram illustrates what tasks the customer, the system and Peter will perform during registration; it bears a lot of similarity to the activity diagram. The customer needs to be able to register their details and register the machines that will benefit from the services.

The system needs to validate the information inputted by the customer to ensure the information is correct, if it is not then it will supply an error message. If the information is correct then it will be added to the database. Peter needs to be able to publish an invoice online if they customer decides to purchase IT equipment.

You will notice that in the activity diagram Peter also performs the installation of IT equipment and invoicing, however this UML use case diagram only illustrates tasks relevant to use of the system and are therefore irrelevant to the diagram.

## A11 Appointment Booking



This use case shows what tasks the customer and the system will perform during the appointment booking activity. There is no need for Peter to be involved in this activity since it is the responsibility of the customer to make the appointment.

## *Requirements*

These requirements are only based upon the two prototypes with full functionality.

### *User Requirements*

#### ➤ Register details

New customers must be able to register their details which will allow access to the customer portal

#### ➤ Login

As a registered customer, they should be able to access the customer portal at any time using their username and password.

#### ➤ Access data

Via the customer portal customers can access their data such as invoices, historic and future appointments, their details, level of service, reports etc.

#### ➤ Upload Data

Authorised employees must be able to upload data, stored in the database, to the customer portal for the customers to view

#### ➤ Apply Restrictions

Authorised employees must be able to apply restrictions to prevent unauthorised access to the database. Restrictions should also be applied so that the uploaded data can only be seen by the parties involved. For example an organisation named Hudson can only be accessed once logged in using Hudson's username and password to access their account. It is the responsibility of the customer to apply controls as to who can access the customer portal.

#### ➤ Modify/Update data

Customers and employees must be able to modify the data in the customer portal and the database to ensure it is up to date and valid.

#### ➤ Book an appointment

Customers must be able to book an appointment and specify what the purpose of the appointment so Simply Wales can send out the necessary persons. As the customer will be logged into their account when booking an appointment it will automatically specify who the appointment is for and their location. They should be able to choose the month, date, day and time.

#### ➤ Cancel an Appointment

The customer must be able to cancel an appointment. If it is less than 24 hours they will not be able to cancel but will be provided with an instruction to contact Peter Morgan directly by

telephone instead. When logging into the customer portal customers can view past appointments and upcoming appointments.

➤ **Change an Appointment**

Customers must be able to change an appointment, as long as the modification takes place for more than 24 hours before the scheduled appointment. If it is less than 24 hours then the customer will be presented with an instruction to contact Peter directly by telephone.

*System Requirements*

➤ **Validate Registration**

When a customer registers their details, the system will check to ensure the information entered does not already exist. The information will then be validated to eliminate mistakes and added to the database. Confirmation will be sent to the customer via email detailing how to use the customer portal. An error message will be displayed if data is invalid and it will highlight which bit of data is invalid.

➤ **Automatic Database Updates**

Appointments, customer details, reports, invoices etc are all stored in a database. When data is modified via the customer portal the system will automatically modify that data in the database.

➤ **Notification**

Customers will be notified when they are due to book an appointment for one of the services.

➤ **Enforce restrictions**

After 5 login attempts the customer account will be frozen until an employee can confirm no unauthorised persons had attempted entry. An email will then be sent to allow the customer to change their login details.

The customer cannot cancel or amend an appointment if it is less than 24 hours until the scheduled appointment. Instead the system will output an instruction to contact Peter by telephone.

➤ **Prompt frequent password changes**

After 3 months a new password will be requested upon logon

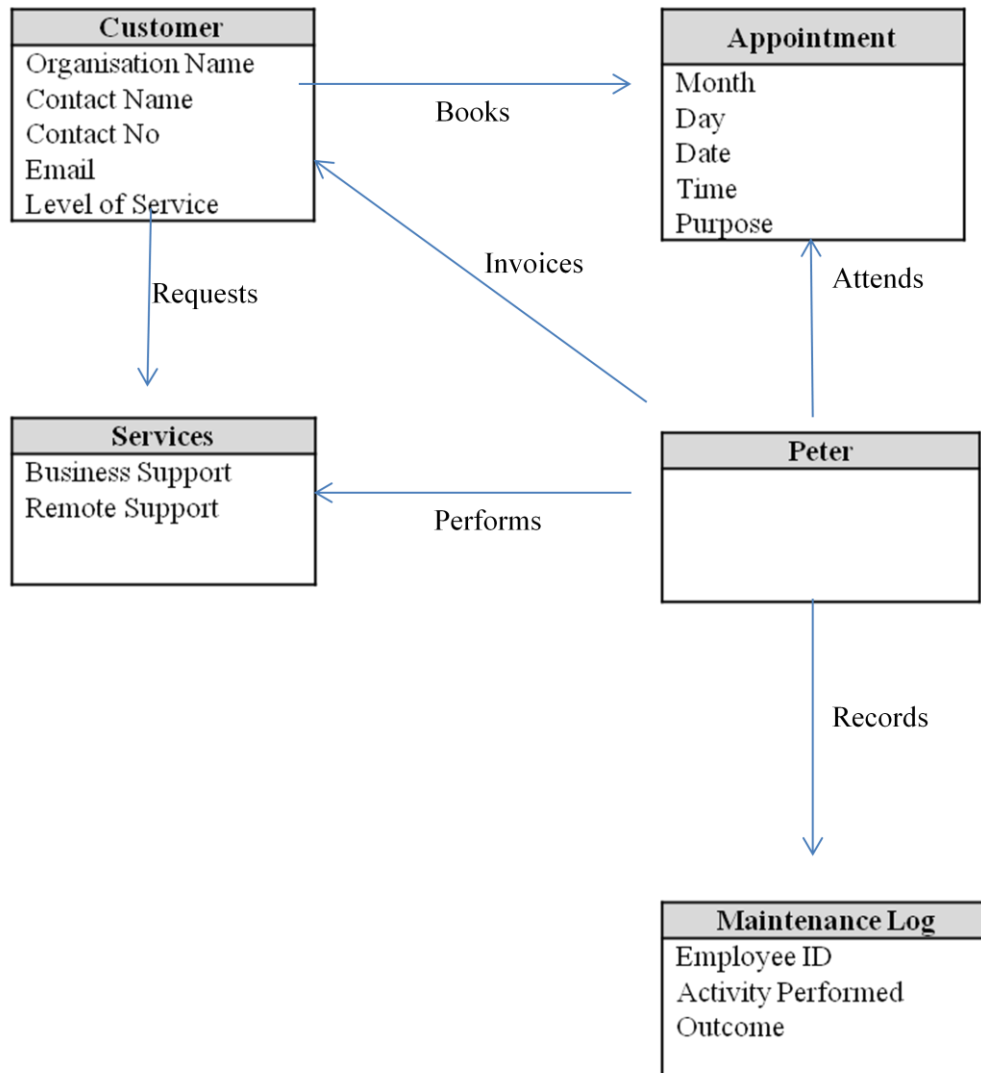
➤ **Confirm appointment**

When a customer has booked an appointment the system should update the database, update the booking system and email the customer with details of the appointments

As these requirements have been derived from the use case and activity diagrams, from comparing all three it should be clear how these requirements have been reached and what is expected of the system

### Database Conceptual Model

As the approach to this project is iterative, the database design was completed once I had begun implementation and had a better understanding of what was needed. The databases support the prototypes by storing updated information inputted to the system.



This conceptual data model identifies the entities that need to be stored, and what information we need to know about these entities. Since Peter has no attributes, it is clear he will not be an entity in the database but rather an actor instead. Likewise, the maintenance log is recorded by Peter and does have entities, but it has no direct relationship with the other entities.

## *User Interface Design*

Since this project is intending to improve the customer experience and overall customer satisfaction, it is important to think about user interface design as good interface design will improve both the customer experience and satisfaction.

The activity diagrams are not only useful for deriving a set of requirements, but also for determining what route the user is likely to take when completing the activity. This is beneficial to know so that I can apply Shneiderman's eight golden rules of interface design to the relevant web pages.

In this section I will discuss the designs for the web pages I have identified as the most likely pages the customer will visit when completing activity B.A1 customer registration and A12 appointment bookings, explaining how they do or do not satisfy Schneiderman's principles.

The eight principles are;

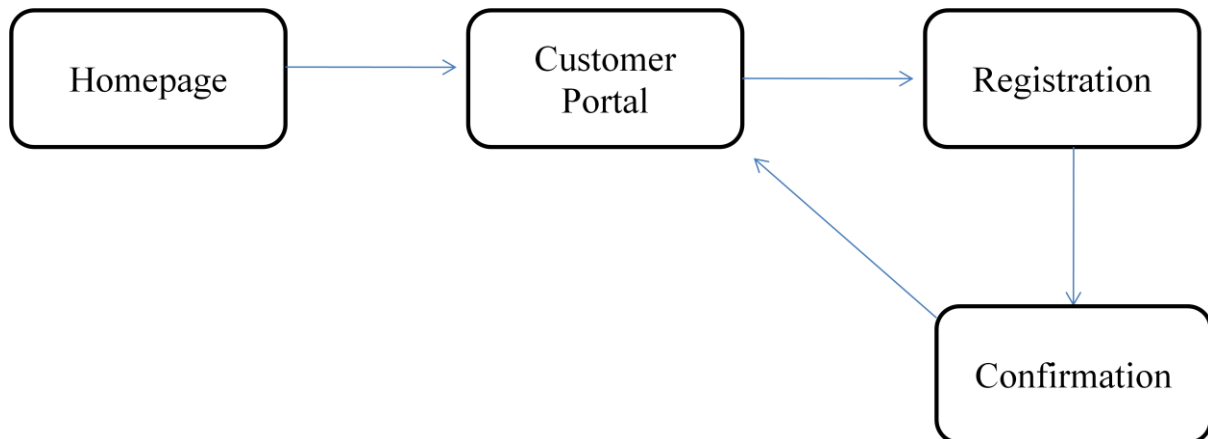
- 1 Strive for consistency.
- 2 Enable frequent users to use shortcuts.
- 3 Offer informative feedback.
- 4 Design dialog to yield closure.
- 5 Offer simple error handling.
- 6 Permit easy reversal of actions.
- 7 Support internal locus of control.
- 8 Reduce short-term memory load.

Of these eight principles only four are relevant to the current stages of the project; strive for consistency, informative feedback, error handling and short-term memory load. These are the only principles I will apply to the design.

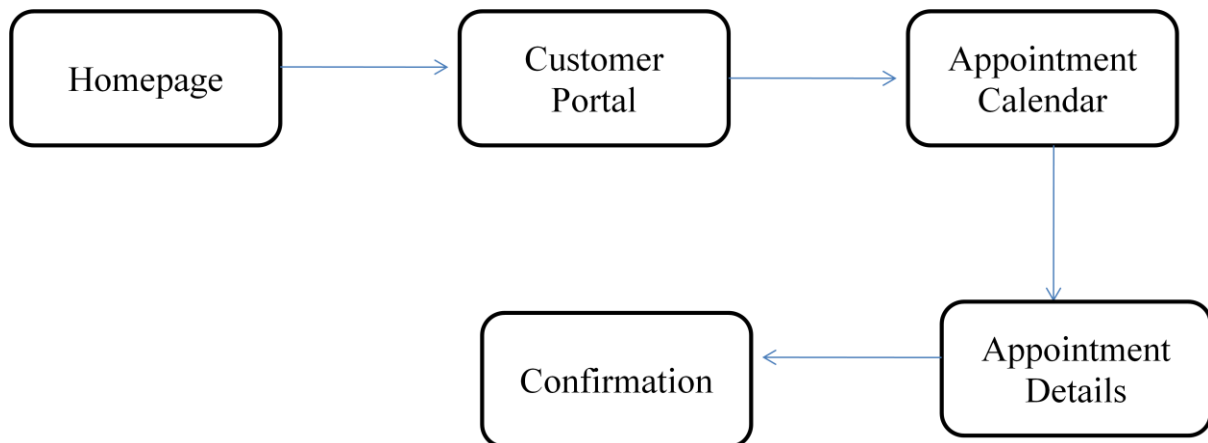
Using shortcuts, dialog, reversal of actions and locus of control are not relevant as the prototypes only include the core functionality; these principles will be relevant when future developments are made. Also, Peter Morgan is currently the only user of the system and so it will be choice discretion whether to include shortcuts.



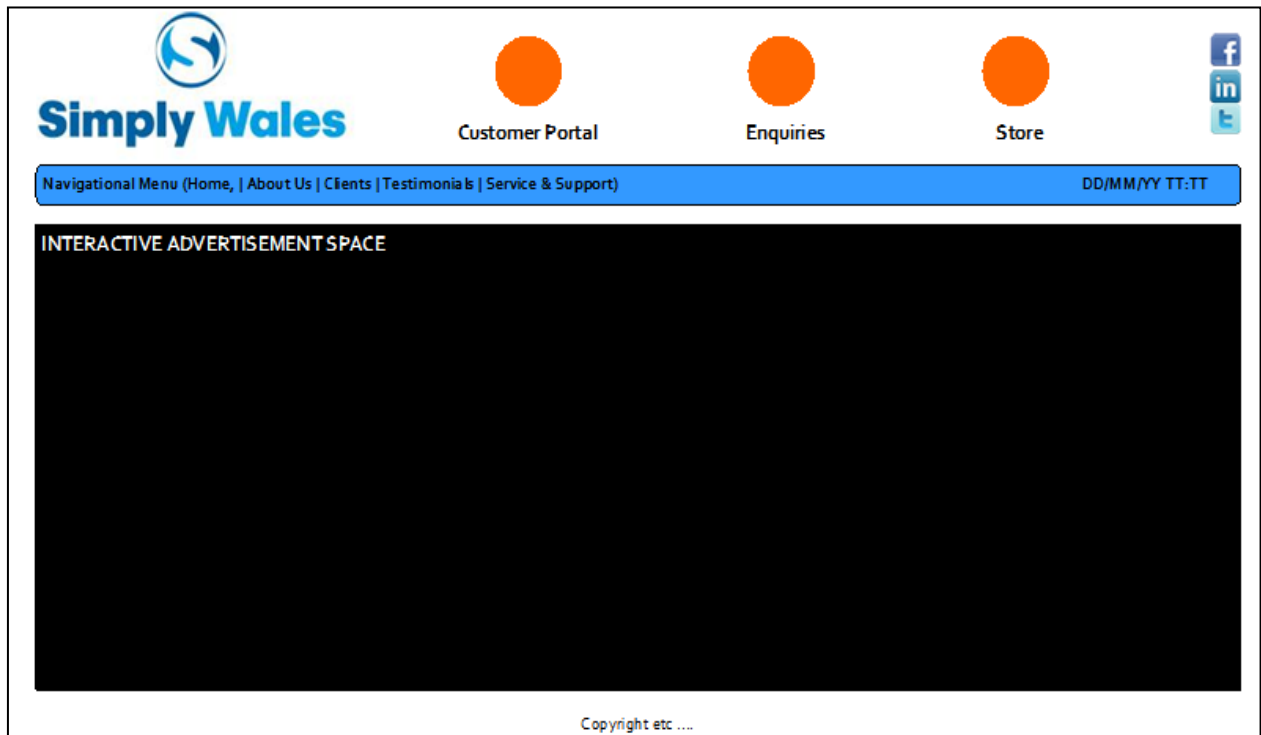
To register, the customer will take the following route through the website.



The route that the customer will take to complete the customer booking is illustrated in the diagram below.



The purpose of these diagrams is to identify what route the users will take to complete the two activities that have been implemented as prototypes. By identifying what route the user will take, it limits what web pages I need to design as it would be inappropriate to attempt to design every web page even if they have no relevance to the activity. The idea is to keep the designs relevant to the activities.



### *Homepage Design*

Consistency is a principle that I will strive for across the entire website, it will be achieved as by using the same layout across all web pages. The only differences between the web pages will be the use of the space underneath the navigational menu.

Informative feedback and error handling are not relevant to this webpage other than that users will be able to click on the advertisements that appear in the advertising space, which will direct them to the relevant corresponding webpage which could be considered as feedback. Short-term memory load is achieved by limiting the amount of information featured on the page.

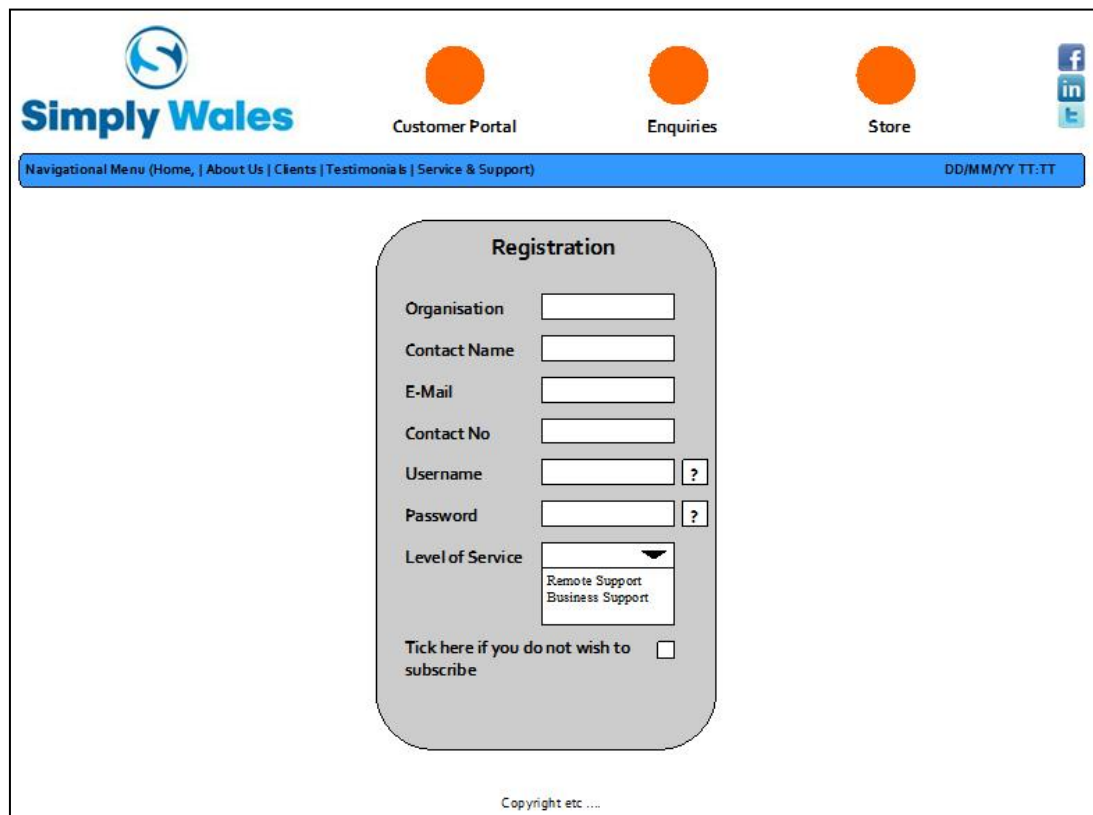
## Customer Portal Login Page

The screenshot shows the Customer Portal Login Page for 'Simply Wales'. The header features the 'Simply Wales' logo on the left, three orange circular icons labeled 'Customer Portal', 'Enquiries', and 'Store' in the center, and social media icons for Facebook, LinkedIn, and Twitter on the right. Below the header is a blue navigation bar containing the text 'Navigational Menu (Home, | About Us | Clients | Testimonials | Service & Support)' on the left and a date/time display 'DD/MM/YY TT:TT' on the right. The main content area is white and features a central grey rounded rectangle titled 'Member Login'. Inside this rectangle are two input fields labeled 'Username' and 'Password', and a link 'Forgotten Login Information? Click Here'. Below the grey rectangle is a blue button labeled 'New Customer? Register Here'. At the bottom of the page, there is a small text 'Copyright etc ....'.

Consistency is achieved as the head of the page has remained the same as the homepage; the same layout has been used. I feel short-term memory load has been achieved well as the login form is simple. There is a clear instruction for users who need to register to click the link underneath the form.

The user will receive feedback when they click on the links and are directed automatically to another page. The login form is missing a login button which will need to be added on later. This is a crucial part of the form as it will send the data entered into the form to be checked against the login data stored in the database to ensure they match and the details are correct.

## Customer Registration Form



The screenshot shows the 'Simply Wales' website header with a logo, three orange circular buttons for 'Customer Portal', 'Enquiries', and 'Store', and social media icons for Facebook, LinkedIn, and Twitter. A blue navigation bar contains a menu and a date/time display. The main content area features a 'Registration' form with the following fields: Organisation, Contact Name, E-Mail, Contact No, Username, Password, and Level of Service (a dropdown menu with 'Remote Support' and 'Business Support' options). A checkbox for 'Tick here if you do not wish to subscribe' is at the bottom of the form. The footer of the form area says 'Copyright etc ....'.

Simply Wales

Customer Portal Enquiries Store

Navigation Menu (Home, | About Us | Clients | Testimonials | Service & Support) DD/MM/YY TT:TT

**Registration**

Organisation

Contact Name

E-Mail

Contact No

Username  ?

Password  ?

Level of Service


Remote Support  
Business Support


Tick here if you do not wish to subscribe ☐


Copyright etc ....


Again this webpage is consistent with other web pages. The colours, shapes and fonts used are all consistent as is the look of the form. The registration form is simple and extra instructions have been included next to the boxes that require special information, for example the conditions the password has to meet such as a minimum of 5 characters and a mixture of numbers and letters. Again this form is missing a submit button that will check the information is valid and will then be added into the database. This will need to be added later.




## Appointment Booking Calendar



**Customer Portal**

**Enquiries**

**Store**



Navigation Menu (Home, | About Us | Clients | Testimonials | Service & Support)DD/MM/YY TT:TT

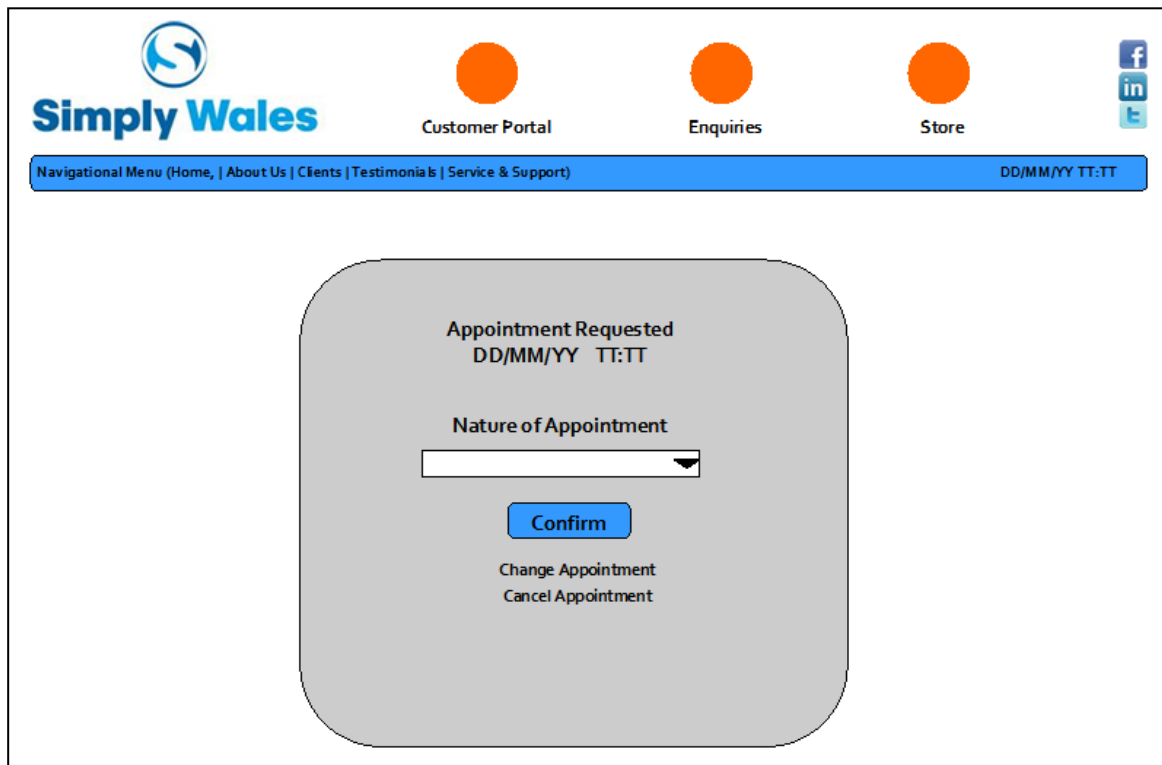
**Appointment Booking**  
**Month**

|    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  |
| 8  | 9  | 10 | 11 | 12 | 13 | 14 |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 29 | 30 | 31 |    |    |    |    |

Copyright etc ....

Consistency is apparent and the appointment booking system is in a format that users can recognise. All users have used a calendar before and it is common for them to have one displayed in their home, therefore it seems the obvious format for booking appointments. The left and right arrow will move between the months of the year.

## Appointment Confirmation



The image shows a web page for 'Simply Wales' with a header containing the logo, navigation links (Customer Portal, Enquiries, Store), and social media icons. A blue navigation bar at the top contains a menu and a date/time display. The main content area features a grey rounded rectangle with the text 'Appointment Requested DD/MM/YY TT:TT', a dropdown menu for 'Nature of Appointment', a blue 'Confirm' button, and links for 'Change Appointment' and 'Cancel Appointment'.

Simply Wales

Customer Portal Enquiries Store

Navigation Menu (Home, | About Us | Clients | Testimonials | Service & Support) DD/MM/YY TT:TT

Appointment Requested  
DD/MM/YY TT:TT

Nature of Appointment

Confirm

Change Appointment  
Cancel Appointment

Again the layout and design is consistent with the other web pages. The page is simple displaying the details of the appointment day, time and month the customer has selected. There is a confirm button on this page to confirm the booking, this matches the button that will be added to all the earlier forms.

## IMPLEMENTATION

---

This section of the report discusses how the implementation has been carried out, justifications for the choice that have been made and a discussion of the challenges faced.

I will begin by discussing my choice of software. The programming language I used to produce the prototypes, and that will also be used to implement the full system is PHP and My SQL managed through WAMP Server.

WAMP Server was suggested by a friend who is completing a similar project, although I am not familiar with it I found it very easy to use. Video tutorials were available to show how to correctly download it to your PC; I took full advantage of these to ensure the installation was correct so as to avoid any problems later on.

Although computer programming, particularly PHP, is not a personal strength there is a large amount of information available on the internet and printed literature to help with any problems I may face. Of course every attempt has been made to avoid plagiarism. Where I have used extracts of PHP code shared on the internet, I have been sure to reference every adaptation of this code.

An advantage of PHP is that it is interactive. I have suggested that I would like to add an interactive advertising space on the website and PHP will help to achieve this level of interactivity.

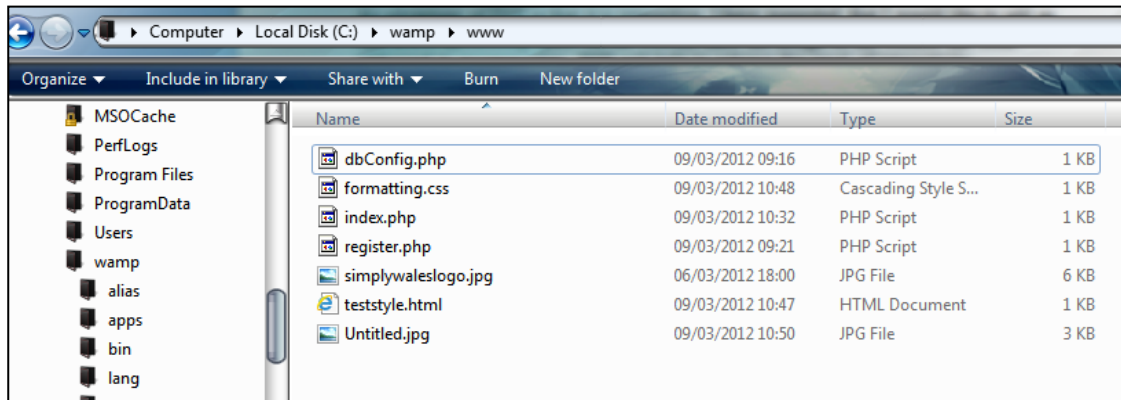
PHP is open source therefore there are no costs involved. Simply Wales already has a website which means there will be no additional costs. Since I have written the PHP code for the website, it will only need to be added to the existing website scripts. Earlier, I assumed Peter Morgan has the IT available to accommodate the Information System and the existence of their website proves that assumption.

Since I only have experience and familiarity with Java and PHP through the completion of certain modules in the degree scheme, the only other option available to me is Java. I am aware other programming languages exist such as C++, but because of time constraints it is more beneficial to use a language I am familiar with than attempt to learn a new language altogether that may threaten the success of the project.

I am familiar with MySQL from the module Database Management and again I found it relatively simply to use. phpMyAdmin was very self-explanatory for creating databases and tables within that database. I created the database simplywales which contains two tables, customerregistrationdetails for use with the registration form, and calendar\_events for use with the appointment system.

To begin I downloaded WAMP server from [www.wampserver.com/en/](http://www.wampserver.com/en/), this gave me use of PHP and My SQL. The screen shot below shows where all of the files are stored, they all

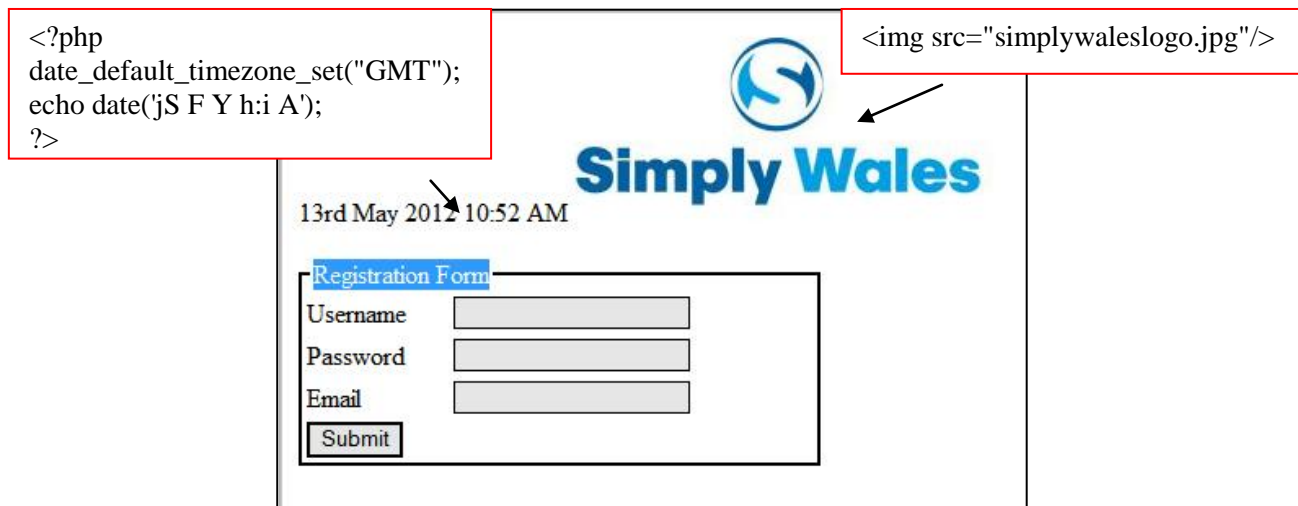
need to be stored in the www folder so they can be accessed through local host. I will first discuss the php code and what it does, and then provide screenshots to show how it looks.



### *Registration Form at Index.php*

The registration form is composed of three PHP pages. The first page is the index.php page that displays the registration form, the register.php page that inserts the data into the correct table in the database, and finally the dbconfig.php that connects to the database. I have used a CSS style sheet for formatting called formatting.css.

The registration form is a fairly simply entity to create, most of the code was familiar to me from using it in the first year module Internet Computing. The following screen shot shows index.php that contains the registration form.





You can see from the screenshot above the pieces of code that relate to the output on the screen, the most important snippet of code from index.php is the following;

```
<form method="get" action="register.php">
<fieldset>
<legend>Registration Form</legend>
<table width ="300" border = "0">
```

This snippet of code links to the register.php page which performs the function of adding the data into the database. The <legend> adds the registration form text to the border.

### *Register.php*

Rather than adding the database connection to each page, I found it easier to have it in a separate php document and link each page requiring database connection to it. The following snippet of code includes dbconfig.php in the register.php page; this means all the code within dbConfig.php is included within register.php without it physically having to be copied into register.php.

```
include"dbConfig.php";
```

The following snippet of code gets the information entered in the fields of the registration form. The md5 encrypts the password when it is entered into the database.

```
$username=$_GET['username'];
$password=md5($_GET['password']);
$email=$_GET['email'];
```

The following inserts the data that has been collected from the registration form and adds it into the customerregistrationdetails table in the simply wales database.

```
$insert="INSERT into customerregistrationdetails (username, password, email) VALUES
('$username', '$password', '$email)";
mysql_query($insert) or die(mysql_error());
```

If the data is successfully added to the database then the following is outputted to confirm that it has been successful.

```
echo "<h2>Thanks for registering!</h2>";
```

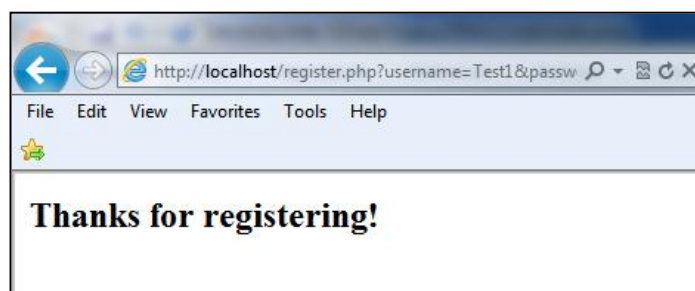
The screenshot below demonstrates how the information is displayed when the customer enters their information into the form; notice that the password is not visible. Take note of the information that is being inputted as this is what we will see in the database.

The image contains two side-by-side screenshots of a web application titled "Simply Wales". Both screenshots show a "Registration Form" with fields for Username, Password, and Email, and a "Submit" button. The left screenshot shows the form with pre-filled test data: Username "Test1", Password masked with five dots, and Email "Test1@Testing.com". A red box highlights the timestamp "13rd May 2012 10:59 AM" in the top left corner. The right screenshot shows the same form but with empty input fields. A red box highlights the timestamp "4th May 2012 09:50 AM" in the top left corner.

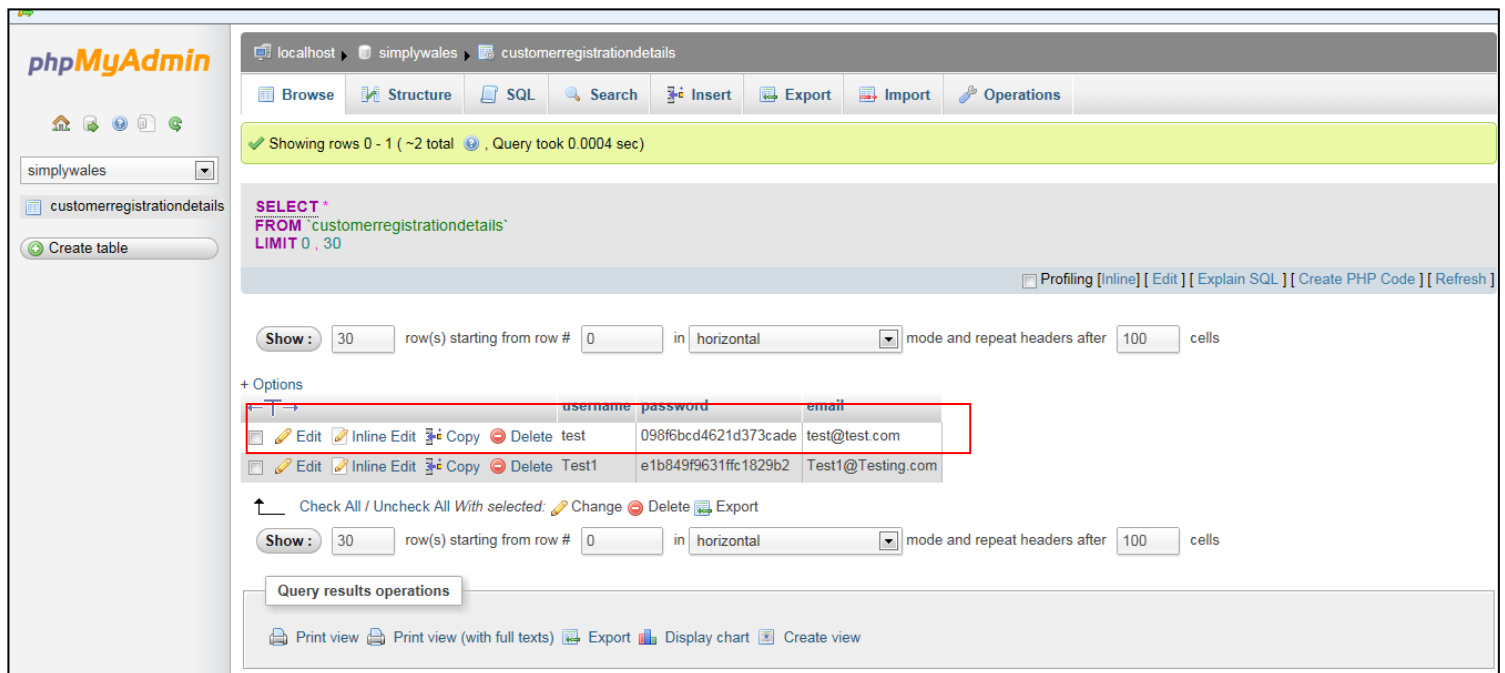
The customer registration prototype functions as expected with one minor error. The date should display as, for example, 3<sup>rd</sup> May 2012, however there is a 1 before the 3. I am unsure at this time as to why it is being displayed as previously the code stated `echo date('1 jS F Y h:i A');` which should have displayed the day before the date, however the number 1 has been removed so there is no reason why a number 1 should be displayed.

When I run local host on another laptop, where the number 1 has also been removed from the code, the data appears as expected. I have noticed that because the clocks were put forward after implementation the time is an hour behind. I intend to rectify this after submission.

The following shows the output to the screen if the registration is successful.



The following screen shot shows phpMyAdmin which is the interface that lets me add and remove databases, tables and entries. You can see that the information entered in the registration form has been added to the table customerregistrationdetails.



### *dbConfig.php*

As mentioned previously, this PHP page connects to the database, without this connection the registration form will not work as data inputted in the registration form will not be added to the database.

The following snippet of code lists the login details to My SQL that gains access. Under the privileges table in phpMyAdmin I added a new user, Charlie, and added a password for extra security.

```
$host = "localhost";
$dbuser = "Charlie";
$dbpass = "Cardiff2009";
$dbname = "simplywales";
```

The following code sets up the actual connection using the details listed above to gain access

```
$connect = mysql_connect($host, $dbuser, $dbpass);
mysql_select_db("$dbname", $connect);
```

### *Formatting.css*

As mentioned previously I used a cascading style sheet to format the web pages. The index.php is the only page that requires formatting since register.php and dbConfig.php do not display any outputs to the screen but rather perform the functions. CSS will ensure consistency is achieved across all webpages.

The pages that require formatting import the style sheet using the following piece of code:

```
<LINK rel="stylesheet" type="text/css" href="formatting.css">
```

### *Appointment Booking System*

I have never developed a booking system using PHP, so I used a recommended online tutorial to guide me through the development as it is quite a challenging task. The tutorial is hosted at [www.dreamincode.net/forums/topic/87835-event-tracking-calendar](http://www.dreamincode.net/forums/topic/87835-event-tracking-calendar). The appointment system is composed of two PHP pages; event.php and calendar.php

### *Calendar.php*

The first step is to check the user input. The purpose of this step is to display the calendar at the correct month when the user first views the calendar.

```
<?php
if ((!isset($_POST['month'])) || (!isset($_POST['year']))) {
    $nowArray = getdate();
    $month = $nowArray['mon'];
    $year = $nowArray['year'];
} else {
    $month = $_POST['month'];
    $year = $_POST['year'];
}
$start = mktime(12,0,0,$month,1,$year);
$firstDayArray = getdate($start);
?>
```

The `$_POST['month'];` and `$_POST['year'];` variables are stored for processing if the user has submitted any information. If no information has been submitted, then the current month and year will be selected as it is obvious it is their first visit to the calendar.

The next step is to build a HTML form. This HTML form provides an interface for users to select their desired month and year.

```
<html>
<head>
<title><?php echo "Calendar: ".$firstDayArray['month']. "" . $firstDayArray['year'];?>
</title>
</head>

<body>
<h1>Select a Month/Year</h1>
<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
<select name="month">
<?php
$months = Array("January", "February", "March", "April", "May", "June", "July", "August",
"September", "October", "November", "December");

for ($x=1; $x<=count($months); $x++){
    echo "<option value=\"\$x\"";
    if ($x == $month){
        echo " selected";
    }
    echo ">".$months[$x-1]. "</option>";
}
?>

</select>
<select name="year">

<?php

for ($x=1980; $x<=2010; $x++){
    echo "<option";
    if ($x == $year){
        echo " selected";
    }
    echo ">$x</option>";
}
?>

</select>
<input type="submit" name="submit" value="Go!">
</form>
</body>
</html>
```

This section of code creates a form that can send the month and year parameters to itself. The third step is to create the calendar. This is all still within the calendar.php document, see appendix C for the full document.

According to the tutorial, PHP has no built in calendar(); functions, so the next section of code aims to create a table and populate it with the corresponding days manually.

```
<?php
$days = Array("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat");
echo "<table border=\"1\" cellpadding=\"3\"><tr>\n";
foreach ($days as $day){
    echo "<td style=\"background-color: #CCCCCC; text-align: center; width: 14%
\"><strong>$day</strong></td>\n";
}
```

‘The for loop to determines how many days are in each week, and what days need to be printed along the way.’

```
for ($count=0; $count < (6*7); $count++){
    $dayArray = getdate($start);
    if (($count % 7) == 0){
        if ($dayArray['mon'] != $month){
            break;
        } else {
            echo "</tr><tr>\n";
        }
    }

    if ($count < $firstDayArray['wday'] || $dayArray['mon'] != $month){
        echo "<td> </td>\n";
    } else {
        echo "<td>".$dayArray['mday']." </td>\n";
        $start += ADAY;
    }
}
echo "</tr></table>";
?>
</body>
</html>
```

According to the tutorial the calendar should be 100% functional at this stage, however when viewing calendar.php in local host I was presented with a repetition of the same error message, 'Notice: Use of undefined constant: ADAY'.

The screenshot shows a web browser window with the address bar displaying 'localhost/calendartest.php'. The browser's tab bar shows several open tabs, including 'Strategies for grow...', 'Php Simple Login T...', 'A Php Registration f...', 'www.publications.p...', and 'Test products | Tolu...'. The main content area displays a calendar titled 'Select a Month/Year'. Below the title, there are two dropdown menus for selecting a month (currently set to 'May') and a year (currently set to '1980'), followed by a 'Go!' button. The calendar grid shows the days of the week (Sun, Mon, Tue, Wed, Thu, Fri, Sat) and the date '1' is visible under 'Tue'. Below the calendar grid, there are four identical error notices displayed in a row. Each notice is a red box with a white exclamation mark icon and the text: 'Notice: Use of undefined constant ADAY - assumed 'ADAY' in C:\wamp\www\calendartest.php on line 79'. Below each notice is a 'Call Stack' table with the following structure:

| # | Time   | Memory | Function | Location              |
|---|--------|--------|----------|-----------------------|
| 1 | 0.0023 | 384896 | {main}() | ..\calendartest.php:0 |

I added the following snippet of code to line 2; `define("ADAY", (60*60*24);`

Testing the code again using local host, I received the error message; 'Parse error: syntax error, unexpected ',''

I removed the ; I had added, retested it and was presented with another error message; 'Parse error: syntax error, unexpected T\_IF'

I decided to carry on with the tutorial in the hope the issue would get ironed out with the addition of more code.

The next part is to add events to the calendar; to support this I need to create a database that will store the events. The tutorial suggests creating a calendar\_events table in the database with the following fields; start date and time (event\_start), the name of the event (event\_title), and a short description of the event (event\_shortdesc).

I created this table using the same name as suggested in the tutorial, but the table was added to the pre-existing simplywales database. This is evident in the screen shot below.

The screenshot shows the phpMyAdmin interface for the 'simplywales' database. The 'calendar\_events' table is selected. The table structure is as follows:

| # | Column          | Type         | Collation         | Attributes | Null | Default | Extra          | Action           |
|---|-----------------|--------------|-------------------|------------|------|---------|----------------|------------------|
| 1 | id              | int(11)      |                   |            | No   | None    | AUTO_INCREMENT | Change Drop More |
| 2 | event_title     | varchar(25)  | latin1_swedish_ci |            | No   | None    |                | Change Drop More |
| 3 | event_shortdesc | varchar(225) | latin1_swedish_ci |            | No   | None    |                | Change Drop More |
| 4 | event_start     | datetime     |                   |            | No   | None    |                | Change Drop More |

The SQL query shown is: `SELECT * FROM `calendar_events` LIMIT 0, 30`. The result is: 'MySQL returned an empty result set (i.e. zero rows). ( Query took 0.0005 sec )'.

Following this, I added php code that will modify the calendar creation script to 'add Hyperlink URL's on the days to show/add events', connect the system to the MySQL database and print any events on the calendar so users can see which days and times are unavailable.

On line 2 of calendar.php script I added the MySQL connection information as instructed. The tutorial used the following connection information.

```
mysql = mysql_connect("localhost", "username", "password");
mysql_select_db("database_name", $mysql) or die(mysql_error());
```

The connection information I actually used is:

```
mysql = mysql_connect("localhost", "Charlie", "Cardiff2009");
mysql_select_db("simplywales", $mysql) or die(mysql_error());
```

There were further modifications from line 82 onwards. When testing the php script in local host I was presented with the following error.

**(!) Parse error: syntax error, unexpected '{' in C:\wamp\www\calendartest.php on line 86**

On removing the { on line 86 as I was presented with another error. This error is the same error I received earlier on.

**(!) Parse error: syntax error, unexpected T\_VARIABLE in C:\wamp\www\calendar.php on line 87**

As before, I decided to continue onto the final piece of php script and tackle the errors that I get presented with once all the code has been added and finished. The last step is the



event.php page. The code in this script takes the ‘basic principals’ and ‘uses them to add data to our database’.

### *Event.php*

The following snippet of code sends the query and shows all events for that day

```
$getEvent_sql = "SELECT event_title, event_shortdesc,
    date_format(event_start, '%l:%i %p') as fmt_date FROM
    calendar_events WHERE month(event_start) = ".$m."
    AND dayofmonth(event_start) = ".$d." AND
    year(event_start)= ".$y." ORDER BY event_start";

$getEvent_res = mysql_query($getEvent_sql, $mysql)
    or die(mysql_error($mysql));
if (mysql_num_rows($getEvent_res) > 0){
    $event_txt = "<ul>";
    while($ev = @mysql_fetch_array($getEvent_res)){
        $event_title = stripslashes($ev["event_title"]);
        $event_shortdesc = stripslashes($ev["event_shortdesc"]);
        $fmt_date = $ev["fmt_date"];
        $event_txt .= "<li><strong>".$fmt_date."</strong>:
            ".$event_title."<br/>".$event_shortdesc."</li>";
    }
    $event_txt .= "</ul>";
    mysql_free_result($getEvent_res);
} else {
    $event_txt = "";
}

mysql_close($mysql);

if ($event_txt != ""){
    echo "<p><strong>Today's Events:</strong></p>
    $event_txt
    <hr/>";
}
```

The following code creates the form that is used to submit new events:

```
echo "  
<form method=\"post\" action=\"\".$_SERVER['PHP_SELF'].\"\">  
<p><strong>Add Event:</strong><br/>  
Complete the form below then press the submit button when you are done.</p>  
<p><strong>Event Title:</strong><br/>  
  
<input type=\"text\" name=\"event_title\" size=\"25\" maxlength=\"25\"/></p>  
<p><strong>Event Description:</strong><br/>  
<input type=\"text\" name=\"event_shortdesc\" size=\"25\" maxlength=\"255\"/></p>  
<p><strong>Event Time (hh:mm):</strong><br/>  
<select name=\"event_time_hh\">;  
for ($x=1; $x<=24; $x++){  
    echo "<option value=\"$x\">$x</option>";  
}  
  
echo "</select> :  
<select name=\"event_time_mm\">  
<option value=\"00\">00</option>  
<option value=\"15\">15</option>  
<option value=\"30\">30</option>  
<option value=\"45\">45</option>  
</select>  
  
<input type=\"hidden\" name=\"m\" value=\"\".$m.\"\">  
<input type=\"hidden\" name=\"d\" value=\"\".$d.\"\">  
<input type=\"hidden\" name=\"y\" value=\"\".$y.\"\">  
<br/><br/>  
<input type=\"submit\" name=\"submit\" value=\"Add Event!\">  
</form>";
```

With all the code implemented as guided by the tutorial, I am still receiving the same error.

 **Parse error: syntax error, unexpected T\_VARIABLE in C:\wamp\www\calendar.php on line 87**

Given my lack of experience in coding such advanced functionality, it is a relief that this is the only error that has occurred as it means the database connection where I had to input the details of my database has been successful.

On researching the meaning of this error, it is possible that I have forgotten to include any of the following syntax ;'"'([]. To see if this was the solution I printed out the script and scanned it through matching up all the { } [ ] "" and '' to see if I had mistakenly missed one.

I tried removing various things and tested to see if there was any change, unfortunately the same error message remains. Comments from other readers left suggestions about how to improve the code and what changes they made when errors arose. I am keen to address this later on when more time is available.

## TESTING

---

It seems inappropriate to test a prototype which is known to be missing functionality, and is still in its initial design stages. It is still in the design stages as once it is presented to Peter Morgan it will be subject to change to fit with the dynamic business environment that may have already been subject to modifications since my observations.

Despite this, informal testing has been ongoing through implementation. During coding I would open up local host and test the web page to see if any error messages appear and if it functioned as intended. Although it is undesirable to formally test the prototypes, I have created some example test cases based on the current state of the system to demonstrate how they could be used.

The test case will describe the test number to keep a log of how many have been performed, the step number as there is more than one step involved in each activity, the expected outcome and the actual outcome. If the actual outcome is not as intended, then I can decide on a course of action to take and conduct more tests based on this action.

### *Prototype Test Cases*

#### *Customer Registration Test Case*

| Test No | Step No | Expected Outcome   | Actual Outcome   |
|---------|---------|--|--|
| 1       | 1       | Customer can see the correct date and time                     | Customer can see the date and time, but there is a number 1 before the date that should not be there |
| 2       | 2       | Customer can see Simply Wales logo                             | Simply Wales logo is visible   |
| 3       | 3       | Customer can enter a username                                  | Customer can enter a username  |
| 4       | 4       | Customer can enter a password that is hidden                   | Customer can enter a password that is hidden   |
| 5       | 5       | Customer can enter an email address                            | Customer can enter an email address  |
| 6       | 6       | Customer can click submit button                               | Customer can click submit button   |
| 7       | 7       | Details are added to the customerregistrationdetails database  | Details are added to the customerregistrationdetails database  |
| 8       | 8       | The password is stored as encrypted                            | The password is stored as encrypted  |
| 9       | 9       | A page is displayed with the message 'Thanks for registering!' | A page is displayed with the message 'Thanks for registering!'                                       |

## *Appointment Booking System Test Case*

It doesn't seem appropriate to use a test case for the appointment booking system since the error message that currently appears prevents me from seeing any aspect of the system. When I have found a solution to this errors I will run a test case using the same format as the test case used for the customer registration form.

## **EVALUATION OF THE PROJECT DELIVERABLES USING SSM**

---

Although SSM is not the method of choice taken to approach the project, it has been used as a means for identifying the key activities that need to happen in the project, it models my viewpoint of what I would like the system to achieve which provides a list of outcomes for the project. The SSM has helped to develop an evaluation of the project deliverables and identify areas for future work.

### *Root Definitions*

I began by developing root definitions six. I will firstly state each root definition and provide a description underneath.

#### **[T1]**

|  |
|--|
| A system (A) to support the efficient operation of Simply Wales with the intention to ease and simplify Peter Morgan's task load (T) by automating specific business functions (W) |
|--|

T1 is the main transformation or purpose of the system, we want to achieve 'efficient operation' and this will be achieved by 'automating specific business functions'.

#### **[S1]**

|   |
|---|
| A system (A) that allows Peter Morgan to easily (T) by store, input, edit and access data (W) |
|---|

S1 is a supporting activity; in this root definition it is describing a database. Peter Morgan should be able to perform multiple functions within this database easily.

#### **[S2]**

|  |
|--|
| A system (A) that delegates appointment bookings to the customer (T) by hosting a user-friendly web-front appointment system (W) |
|--|

S2 is a second supporting activity that discusses the use of an appointment booking system. This root definition specifies that the appointment booking system must be 'user-friendly' which would suggest testing needs to be carried out to ensure it is user-friendly.

### [L1]

A system to promote customer interaction (T) using a web-front customer portal to not only register new customers but enable access and user rights with existing customers to view and modify relevant data (W) extracting data from the relevant databases in compliance with the Data Protection Act (E)

L1 is a linking system; it links the system to the customers. This root definition discusses the use of a customer portal that enables customers to be able to view and modify data relevant to them. The Data Protection Act is an environmental constraint as by law customer should be able to access their data at anytime, and this data should be up to date and valid.

### [PMC1]

A system to control access and protect privacy to Simply Wales' assets [T] by performing annual risk assessments and applying the necessary controls [W]

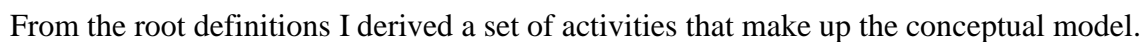
PMC1 is the first planning, monitoring and control activity, it attempts to control access to the system and planning what potential risks can arise so they can prevent them and/or deal with them accordingly if they are to occur.

### [PMC2]

A system to monitor the success of Simply Wales' automation (T) by producing monthly and yearly reports using data based on new customer registrations, number of appointments made, number of appointments attended, total number of customers and the type of service being delivered (W)

PMC 2 is the second planning, monitoring and control activity that exists to monitor the success of the system when it has been implemented in Simply Wales. It lists what will be monitored.

## |Conceptual Model



The conceptual model also neglects the T, in other words the purpose of the activity, this was not realised until I had the model validated with a second person. By this time the time constraints were too small for any changes to be made to the model. Despite these mistakes, the model contains enough activities and information to evaluate the system.

You will notice the table has been separated into two; the first table examines activities addressed by the system, and the second table discusses activities to be addressed by the organisation.

---

46

addressed in the future. By identifying what areas of the desired system the project has addressed, we can evaluate the overall achievement of the project.

| Activity Description                                     | Is it currently performed in Simply Wales?    | Has it been addressed in the project?  | Future Work Potential  |
|--|---|--|--|
| Identify the current business activities                 | No  | Figure 1 lists the products and services that compose the remote and business support packages | N/A  |
| Decide on which business activities to automate          | No  | Activity diagrams have identified which activities will work well with automation              | Automate all potential activities                                      |
| Decide how to automate activities                        | No  | The design stages has addressed how the activities will be automated                           | Automate all potential activities                                      |
| Automate activities                                      | No  | Implementation of Prototypes   | Added functionality  |
| Monitor the efficiency of the delivery of the activities | No  | No   | Gather progress reports and sales statistics. Conduct customer surveys |
| Design appointment booking system                        | No  | Activity diagrams and interface design   | Added functionality  |
| Gather appointment booking system requirements           | No  | High-level requirements have been derived from the activity diagrams                           | The requirements will change if more functionality is required         |
| Create appointment booking system                        | No  | Implementation of a prototype  | Added functionality  |
| Monitor customer satisfaction                            | Yes, Peter asks for testimonials occasionally | No, but has taken into consideration interface design principles                               | Customer feedback and user testing                                     |
| Take control action to ensure system is user friendly    | No  | No, but has taken into consideration interface design principles                               | Customer feedback and user testing                                     |
| Identify the relevant database options available         | No  | Made us of My SQL, and discussed the advantages and disadvantages of alternatives              | This is included in implementation                                     |
| Decide database provider to use                          | No  | Made us of My SQL, and discussed the advantages and disadvantages of alternatives              | This is included in implementation                                     |
| Create Database  | No  | My SQL on WAMP Server  | Expand database  |
| Input Data   | No  | Client data has not been added but test data has   | Input client data  |

|   |     |   |   |
|---|-----|---|---|
| Take control action to ensure the data entered is valid | No  | Some controls on the registration form        | Add validation checks on the data being added                 |
| Edit data   | Yes | Manually                                      | Customers can edit their own data                             |
| Take control action to ensure the data is up to date    | Yes | Checking the details every so often           | Customers can update their own data                           |
| Access data   | Yes | My SQL through phpMyAdmin                     | Authorised personnel and customers can access their own data, |
| Take control action against unauthorised access         | No  | No  | Add user access control                                       |
| Design customer portal                                  | No  | User interface designs                        | Derive requirements   |
| Create customer portal                                  | No  | There has not been time to fully address this | Using PHP, including it as an extension to the website        |

| Activity Description                               | Recommendations  |
|--|--|
| Decide what data will be accessible and modifiable | Customer requirements to determine what information they would need to be able to see and change |
| Set access and user rights for customers           | Customer requirements to determine what information they would need to be able to see and change |
| Take control action to prevent unauthorised access | Add controls to the database and set up users  |
| Set access and user rights for customers           | Database controls  |
| Perform annual risk assessments                    | Conduct risk assessments after creating a policy   |
| Implement controls                                 | During implementation  |
| Monitor performance                                | Using sales statistics and customer feedback surveys   |
| Collect data                                       | From new customers   |
| Monitor success                                    | Using sales statistics and customer feedback surveys, compare data between months and years      |
| Produce monthly progress report                    | Using sales statistics, problem logs and customer feedback surveys                               |



## EVALUATION OF PROJECT APPROACH

---

In this section I intend to evaluate the different tools and techniques to determine how beneficial they have been; I will include the tools and techniques that were used in the Interim Report.

### *Project Planning*

I began my project planning with a Gantt chart produced using Microsoft Excel, complete with a set of deliverables for each week. As some tasks took longer than intended, it wasn't long before modifications had to be made. With the lack of direction the project faced in these beginning stages, this attempt at project planning was not relevant since it did not keep up with the changes. I'd have spent more time modifying the plan than actually doing the work.

During summer I purchased an iPad 2, whilst this seems irrelevant to the project it did help with the project planning. Before this project was due to start I organised my priorities using a diary where I could plan my daily tasks. The iPad proved very efficient for project planning as it is easy to modify changes to my plans when using the built-in calendar. I feel it is easy to follow my progress on an interface that can adapt to changes easily than on an excel spreadsheet that is a nuisance to locate.

### *Research*

The first task in the project was research, with this research I planned to compile a literature review that discussed the benefits of Information Systems, and what challenges SME's face.

Much of the literature was found through internet research, I repeatedly used Google scholar searching for different key terms such as 'Information Systems', 'Challenges' and 'SMEs' to find a sufficient amount of results. I attempted to find information on UK-Based SMEs but this was not always possible, so I figured it was better to get some information on SME's even if it was based on worldwide organisations as it would still have some relevance to the topic.

A small amount of literature came from books hired in the University Library; I did not have the time available to read these books in full, so instead I studied the contents and index pages to identify areas of interest. In future, I will be more thorough with the research sourcing the material from different mediums to gain a wider picture.

### *Analysis and Design*

Some of the analysis of Simply Wales was completed in a separate module, Work Placement, where I examined the organisation and what I learnt from my time there. This helped me to

compile a few documents detailing their current operations. Using the information discussed in that module, and the information available on the website I compiled a full list of activities that Simply Wales offers, this can be found in the Interim Report.

This list was helpful when the scope of the project was determined, and it was decided that the focus of the project would only cover the business and remote support packages and the activities included in them. This is shown in Fig 2.

I used the results of the SSM that had been produced in the Work Placement module to develop my understanding of Simply Wales and what the project would include. As SSM had already been produced for Simply Wales, it seemed unnecessary to repeat the task. Instead I opted for business process modelling.

From studying the list of activities that were covered under the business and remote support packages I was able to produce activity diagrams based on my understanding of what the activities included.

The use of business process modelling was very beneficial, not only was it a task I felt at ease completing, but the activity and use case diagrams were particularly useful for addressing the user and system requirements, which lay the foundations for the functionality to be presented in the prototypes.

### *SSM*

Although SSM was not used as the main approach, it was carried out towards the end of the project serving a different purpose. The reason it was carried out at such a late stage is because it allowed me to refresh my ideas about what I wanted the system to achieve. From modelling the desired system I was able to compare it to the actual outcomes of the project to see how much the project has achieved.

### *Designs*

Before I started producing the interface designs, I identified what route the user would take through the website when completing both the customer registration and appointment booking activities. The purpose of this was to limit the designs to only these web pages, otherwise I would have got carried away designing aspects of the system that are outside the scope of the development.

Interface designs were created using Microsoft Publisher, purely on the basis that it is a program I am familiar with and have experience using. The interface designs are not intended to be high –level, but rather demonstrate the image I am trying to create, and how user-friendliness can be achieved through the correct positioning of certain links and icons.

Much like the intention of the SSM, I can compare these interface designs with the current design of the prototype to determine what extra functionality needs to be added and what additions to be made for it to achieve the desired state. This is an area that is discussed in the future work section. I used four of Schneider man's eight principles to interface design to determine the suitability of the designs.

### *Implementation*

I used WAMP Server so that I was able to develop the prototypes on my personal laptop. WAMP Server exists to allow users to create 'dynamic web applications' within an environment that contains My SQL and PHP, my two choices for implementation.

I have already discussed my reasons for choosing to use PHP and My SQL, so to recap I chose then because they are both open source; therefore no additional costs are involved which not only saves me money but Simply Wales too if they wish to adopt the system. In addition, I am familiar with both PHP and My SQL from previous modules within the degree scheme.

If I were to carry out this project again, I would still have opted to use PHP and My SQL primarily due to my experience using them, but also because they are widely used and quite high in popularity so there is a lot of literature available.

### *Testing*

The testing strategy I have adopted is informal, but I have considered using test cases to demonstrate how they can be used. I feel this was a good choice for testing since the entire project has been iterative, so the changes and tests have been ongoing.

## **FUTURE WORK**

---

The SSM activity analysis helped to identify areas of future development and change the ideas I already had into feasible areas of development. This section of the report will begin by discussing further developments with the prototypes and aspects of the system covered in the project, and then discuss the ways the project its self could be developed further.

This project only focuses on the business and remote support services. In the near future I would like to extend it across all products and services offered by Simply Wales to offer a fully-integrated system. The additional services offered by Simply Wales include server sales, Telecoms, web design, office moves, recycling and an online web store for refurbished IT equipment. It is clear Simply Wales has the expertise and business processed to expand into differing departments, but the Information System would guarantee this expansion.

There is also an opportunity to extend this system to include finance functionality. Integrating BACS services with the invoicing and outstanding balance system would drastically improve customer experience. With regards to the business and remote support, a monthly fee is paid by direct debit. When a customer purchases additional features the fee is added to the overall payment and an invoice accessible to the customer will detail the breakdown of payments due. The payment for the additional features will have to be paid manually since the direct debit collects a set value each month. This is reflection in activity diagrams 12a and 12b.

The starting point to further this project would be to add functionality to the prototypes, carrying it further to a fully implemented system. I am confident in my abilities that I would be able to achieve this.

The functionality I would add to the current prototypes is as follows. The registration form needs to be accessible from the internet via the customer portal and requires an extended online form, currently this registration form asks for only the username, password and e-mail address, with the added functionality the registration form will include the contact name, organisation name, e-mail address, contact number, username, password, a drop-down box to select the level of service required, and a tick box to receive the e-newsletter.

For the appointment booking system; once the user has selected the correct day they will be taken to a second screen displaying all the times available for that day. On selecting the time the user will be taken to a third screen where they confirm the day, date and time of the appointment and specify the reason for the appointment from a drop down box, and click confirm. If the appointment is confirmed then the system will connect the database and update the calendar so that slot is displayed as unavailable.

On finishing the prototypes, I will consider implementing them to a full system which automates all of the appropriate activities in the business and remote services. I would like to

present the prototype to Peter Morgan, before implementing a system, along with this project to demonstrate how it will achieve the desired benefits

If Peter Morgan decides that he would like to adopt the approach outline in the project, or is content with granting me access to Simply Wales to test the prototypes for myself then I will proceed in the following way.

Firstly, I would revamp the website since Peter Morgan does not have a strong grasp of user-friendliness. To begin this I would return to the design stages, using the relevant diagrams to understand the information flows between the web pages. I would then produce accurate designs of each webpage ensuring they meet requirements and fit with the overall branding of Simply Wales; note that I have already produced interface designs in the design stage but these are subject to change to meet Peter Morgan's tastes and aspirations for the organisation.

Peter is fond of advertising on the website, having already considered the design of the website I would like to add an interactive advertising space on the homepage of the website. I would also recommend Twitter, Facebook and Linked In icons since these are mediums also used by Peter Morgan. These are already included in the interface designs produced in the design stage of this project.

I have proposed implementing a customer portal as an extension to the website that will host a number of functions. It will firstly allow new customers to register their details which are added to the customer details database. Once a registered member, customers can then use their selected username and password to log into the customer portal for existing members.

On a successful login attempt, the customer will have access to their personal data as it appears in the database with the option of editing the data. They can subscribe or unsubscribe to the e-newsletter, book, cancel or amend an appointment and view all the invoices and outstanding balances.

Using the customer portal, customers can purchase extra PCs/Macs to receive the service which will update the owed monthly fee. This does not have any relevance to the system, other than when an invoice is compiled it will be added to the customer's portal page.

If the customer portal is a success, there would be an opportunity to introduce an employee portal to enable easier access inside and outside of the office.

For new customers, after registration I would like to prompt them to take a virtual tour of how to use the customer portal to its full potential, this tour may also extend outward to other sections of the website which may be of use such as the online web store.

I would invite local businesses with little to expert knowledge on using IT and run user tests and heuristic evaluations. Based on the feedback received from these tests I will amend the system and the website leasing with Peter throughout all the stages. I would also complete an

appropriate number of test cases to test the functionality of the system and to sure it all works as intended.

As the number of customers grows, I hope Peter Morgan will recruit more employees. This would allow Peter to further delegate the activities allowing him to explore areas of growth and improvement. With more employees, particularly those with IT expertise, the online appointment booking system will feature more time slots as more people will be available at any one time.

It would be desirable for Peter Morgan and potential employees to have contact outside of the office, particularly while attending appointments. When Simply Wales' budgets have increased, I would suggest investing in current mobile technology such as Apple iPad and iPhones so that the employees can interact with the system and access it at anytime. Particularly if there are two appointments back to back, when travelling from one location to another the system will tell the employee where the location is and act as a satellite navigation system to guide them to the appointment on time and with ease. This would also allow them to update the information while it is still fresh. To note, Peter Morgan has a partnership with Apple which provides him with a small amount of discount.

Employees will have an opportunity for career development by rotating around the different activities which would ultimately improve their skill-set. Working alongside Simply Wales widely recognised partners, the employees would also benefit from further recognition.

The overall goal is to have a fully functional system which reflects the interface designs and requirements identified in this project, as well as additions to these as future work goes underway. My intention is to monitor the success rate so that I have evidence of a personal achievement that will highlight my strengths when applying for future employment positions.

## CONCLUSIONS

---

In this section I will discuss which objectives were met in the Interim Report, what the revised objectives were after the change of direction and how these have been achieved.

In the introduction section I discussed how the original objectives stated in the Interim Report have been achieved. To reiterate I only reached the first stages of analysis before the direction of the project changed. In this time the literature had been completed with the intent to gain insight into the wider problem facing SMEs. I also produced a number of documents describing the current operations of Simply Wales.

As the direction of the project changed to develop a prototype and find a direction solution as opposed to modelling the problem, later aspects of the waterfall model were not addressed fully due to the time constraints.

The refined objectives after this change are;

- *Use business process modelling to gain an understanding of the processes involved with each activity*

Business process modelling forms a large proportion of the project; much of the time was delegated to thorough analysis of the activities as it was important to understand the needs of the system. This thorough analysis allowed me to address all of the activities within the business and remote support packages and I have a total of a total of fifteen activities and twelve use case diagrams.

- *Create a prototype to demonstrate how the activities have been, and can be automated*

The time constraints limited how far the prototypes could be developed, but this objective has been achieved even though one of the prototypes still requires attention.

- *Understand the technologies that underpin the information system and how best to acquire and implement them.*

Opting to develop a prototype gave me the opportunity to develop other areas of skill such as computer programming. I have a good basis for the overall design of the system and clear ideas for future developments to the system.

I engaged in frequent communication with Peter Morgan to ensure I grasped a full understanding of the activities. A key area of the project is user evaluation; in order to carry out user evaluation I need a full system for them to evaluate, at this time the prototypes are not enough for them to evaluate and user evaluations will be carried out on future versions of the system.

To conclude, the change of direction was appropriate even if it did cause further time constraints. I did not want to over simplify the activities which is why a large proportion of the project was delegated to business process modelling.

If I were to carry out a similar project in the future, I would be certain of the direction I want the project to take before it begins. If I had more time, I would certainly develop the prototypes further.

## **REFLECTION**

---

This section of the report discusses my personal feelings toward the project, what I have learnt, what challenges I faced and my overall confidence with the project.

I will begin with a discussion of my feelings toward the overall project. Progress at the beginning of the project was slow as it lacked direction and my understanding of the purpose of the project. After discussion with the supervisor the direction was clarified and I was able to progress at a sufficient rate with the project. Once the objective to develop a prototype was added, a number of the previous objectives were irrelevant or had to be removed due to time constraints. Although this new objective improved the overall outcome of the project I wish I had more time to fully implement the prototypes to their full functionality.

I am very confident with the idea of extending the website to accommodate a customer portal. Currently this customer portal does not exist as it was only decided late into the project that a prototype should be developed. I have discussed this idea in the future work section, but it is easy to see how this customer portal would fit in with the other activities; a number of the activities would be accessible through the customer portal, and customers would need to login to access it, therefore they would need to have registered; it is evident that this customer portal would link a number of the activities together and I regret that I was unable to develop this within the scope of the project.

I am proud that I developed two prototypes; however these prototypes only demonstrate basic functionality and do not reflect the full system requirements again this was due to time constraints.

### *Skills Development*

I have realised that I have used a large amount of information taught in the degree scheme from the following modules in particular, Human Factors, Database Systems, Internet Computing, Organisational Behaviour, Strategy and Information Systems and System Modelling. It is clear to me how the module selection for my degree scheme can be used during employment.

During implementation I developed my computer programming skills. Although I sourced material from the internet and used video tutorials to help start me off, I was able to adapt this material to the project. As a result, I have a fully functioning prototypes, experience in PHP, My SQL and WAMPS Server.



Soft Systems Methodology is a method that has been addressed repeatedly in the degree scheme, I now have insight of how it can be used in a project and in a real-world organisation to not only address a problem, but evaluate it.

Business process modelling is a module I had great success in when it was covered in the first year of the degree scheme. Since then it has not been addressed quite so much so again it was a refreshment of my learning when producing the activity diagrams. I found the production of activity diagrams quite a simply and enjoyable task, but I disregarded their importance to the project. It would appear these business process models laid the foundations for the new direction of the project, even fabricating requirements and I had not realised their importance until writing this dissertation.

I also now understand iteration within a project development lifecycle since that has been the basis for the approach to this project. Project development lifecycles were taught in one of the second year modules, having now experienced using a project lifecycle I have a better understanding of how they work.

### *Personal Challenges*

I would have to argue the personal challenges I faced during this project are self-motivation and personal confidence.

Project planning is one of my strengths, I have very strong organisation skills but what I lacked towards the beginning of this project was the motivation to actually do the work. I think because I lacked understanding of the project I was reluctant to progress with the project as any work just seemed meaningless. When I had a clear direction of the project I feel I pulled it together and worked hard to get the work done, even if I was up against time constraints. I am confident with the work that has been produced.

Prior to this project I was unsure of my computer programming abilities, even choosing not to dive into any aspect of the computer science degree scheme until it was decided that it was crucial for the success of the project. Now I am glad that I chose to develop a prototype and welcome aspects of the computer science degree scheme, and in future I will not disregard my abilities so quickly.

On completing this project I have a clear vision of how this project can be taken further. I have a good understanding of the needs of Simply Wales and am very confident that the Information System proposed in this project will achieve the intended benefits.

To summarise, I am confident with the outcomes of the project but I do regret not having enough time to add the desired functionality to the prototypes. I do intend to carry out the developments discussed in the future work should Peter choose or not choose to adopt the proposition set out in this project as it will form a portfolio of my work.

## REFERENCES

---

References used within the Literature Review are included in the Interim Report. I have included the references where I have adapted PHP code.

Anonymous. 2009

*Building an Event Tracking Calendar*

[Online]

Available at: [www.dreamincode.net/forums/topic/87835-event-tracking-calendar/](http://www.dreamincode.net/forums/topic/87835-event-tracking-calendar/)

[Accessed 4<sup>th</sup> May 2012]

Anonymous. 2012

*Creating a Registration Form Using PHP*

[Online]

Available at: [www.html-form-guide.com/php-form/php-registration-form.html](http://www.html-form-guide.com/php-form/php-registration-form.html)

[Accessed 4<sup>th</sup> May 2012]

Anonymous. 2012

*Review*

[Online]

Available at: <http://wampserver.software.informer.com/>

[Accessed 3<sup>rd</sup> May 2012]

Anonymous. 2012

*Date*

[Online]

Available at: <http://php.net/manual/en/function.date.php>

[Accessed 4<sup>th</sup> May 2012]

Anonymous. No Date

*SWOT Analysis*

[Online]

Available at: [www.managementstudyguide.com/swot-analysis.htm](http://www.managementstudyguide.com/swot-analysis.htm)

[Accessed 4<sup>th</sup> May 2012]

Skaalid, B. 1999

*Human-Computer Interface Design*

[Online]

Available at: [www.usask.ca/education/coursework/skaalid/theory/interface.htm](http://www.usask.ca/education/coursework/skaalid/theory/interface.htm)

[Accessed 3<sup>rd</sup> May 2012]