



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David



GIG
CYMRU
NHS
WALES

Bwrdd Iechyd Prifysgol
Aneurin Bevan
University Health Board



**DEVELOPMENT OF A&E GUIDELINES FOR NEWPORT MINOR INJURIES UNIT
USING MVC FRAMEWORK WITH A RESTFUL WEB SERVICE APPROACH**

PETER DAVIES

B.Sc. WEB DEVELOPMENT

Supervised by Dr Nicola Whitehead & Dr Kemi Ademoye

CONTENTS

Introduction.....	1
Background Research	2
Interaction Design & User Experience.....	2
Model View Controllers Framework using a RESTful Approach	6
Aims & Objectives	9
Research Methodology	10
Research Approach	10
Research Strategy	11
Document Review	11
Case Study	11
Development Methodology	12
Software Development Methodology.....	12
Analysis & Deliverables	14
Project Management.....	16
Risks Associated with Project	16
Ethical Issues.....	18
Project Resources	18
Authorities & Stakeholders	18
Physical Project Resources	19
Project Timeline	20
References	23
Bibliography.....	27

TABLE OF FIGURES

Figure 1 : The Five Dimensions of Interaction Design	2
Figure 2 : User Experience Honeycomb	4
Figure 3 : Model View Controllers	6
Figure 4 : RESTful Web Services	8
Figure 5 : System Development Lifecycle	12
Figure 6 : Risk Evaluation Matrix	16
Figure 7 : Gantt Chart	20
Figure 8 : Gantt Chart Timeline	21
Figure 9 : Project Resources	22

TABLE OF TABLES

Table 1 : The Five Dimensions of Interaction Design Description	3
Table 2 : The User Experience Honeycomb Description.....	4
Table 3 : Principles of RESTful Web Services	7
Table 4 : Objectivist & Subjectivist Philosophies	10
Table 5 : Risk Assessment Table.....	17

INTRODUCTION

A need has arisen within the Aneurin Bevan Health Board's Minor Injuries Unit in Newport to computerise their current paper documentation containing guidelines to aid staff to avoid potentially dangerous complications when patients attend with specific symptoms.

The purpose of the project is to develop a web application, accessible through an array of devices, which will display the guidelines currently laid out in the supplied MIU documentation whilst taking into consideration usability, security and the latest web technologies on information architecture.

This proposal will, in brief, research into current interaction design and user experience theories whilst also looking at existing online medical systems of this nature and whether these systems truly aid clinicians. This will be followed by research into the Model View Controller architectural pattern, REST architectural style and RESTful web services which will provide an interesting and innovative approach to a system in this field.

Aims and objectives of the project will then be identified, followed by identification of the research methodology, the strategies chosen and analysis of the deliverables. Project management points ensue with risks, resources and project timelines being discussed. A full reference list and bibliography conclude the proposal.

BACKGROUND RESEARCH

Interaction Design and User Experience

Interaction design is said to be one aspect of the broader field of user experience design (UX). It can be described as the design of interaction between a product and its users, it is the definition of the assembly and conduct of systems which we interact with. In a sense, it is the design of creating relationships between humans and computers, and the devices we use, it shapes digital products for people's use [1]–[3].

The beginnings of interaction design could be prehistoric, albeit not a recognized discipline at that point in history. Saffer argues that ancient civilisations, like the native Americans or European Celts, used smoke signals and stone markers respectively to communicate with one another [4]. A very early form of designing to interact with people? The phrase '*Interaction design*' stems from two industrial designers, Bill Moggridge and Bill Verplank, who invented the term to describe the work they were conducting on the first laptop computer [5]. However, the father of Interaction Design as we know it today is Doug Englebart, who, in 1968 presented what is now known as '*The Mother of All Demos*' [6], [7]. In this demonstration, Englebart showed the world the mouse, point and click interfaces, hyperlinks, copy & paste and networked collaboration [4][8], which paved the way for the future of interaction design.

There are 5 dimensions to interaction design, Gillian Crampton-Smith is accredited with introducing 4 of the concepts, with Kevin Silver of IDEXX adding a fifth [1]. These are words, visual representations, physical objects or space, time and behaviour [1][9].

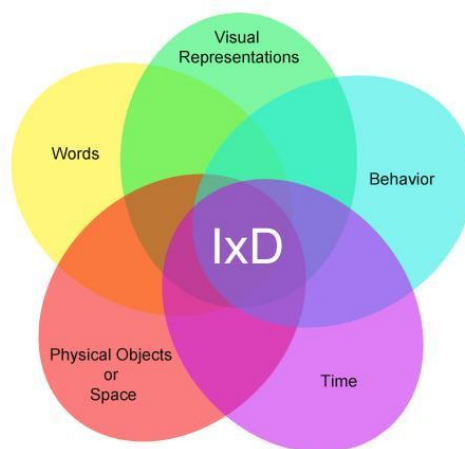


Figure 1. The five dimensions of interaction design by Samir Dash [10]

AREA	DESCRIPTION
Words	Words in interactions must be meaningful, concise & understandable
Representations	Graphical elements user interacts with – images, typography and icons.
Physical Objects	Physical object users interact with and physical space user does so.
Time	Media that changes with time. Time users spend interaction with product.
Behaviour	How do users perform interaction on website or operate the product?

Table 1. Descriptions of the 5 dimensions to Interaction Design [9], [11]

The other design area to research for the project is, *user experience*. UX is the reason that sites like Google and Facebook are easy to use, and so popular. UX is the overall experience a user will have when interacting with a product, it considers what they need, value, their abilities and disabilities. UX is not just concerned with interaction but how products feel, look and even the psychological impact the product has when delivering its outputs [12]–[15].

A good example of the differences between interaction design and user experience design is found with The Interaction Design Foundation. If a telepathic communication system was developed to simplify the process of communicating with a machine this would be considered an advance in interaction design, but what if the machine computed the interaction over several hours? This is where user experience plays its role, the UX process will find that the use of the product is limited although the interaction good [12]. This example illustrates the need to consider both interaction design and user experience principles when developing the web application for the MIU.

In his 2001 book *‘The Design of Everyday Things’* Don Norman states seven stages of action, questions that should be asked when designing a user interface for successful user experience, these are;

- i. What do I want to accomplish?
- ii. What are the alternative action sequences?
- iii. What action can I do now?
- iv. How do I do it?
- v. What happened?
- vi. What does it mean?
- vii. Is this okay? Have I accomplished my goal?

He argues that a user of a product should always be possible to determine the answers to these questions. He continues to state that the presentation of a system has to match how users will interpret the goal they are aiming to achieve, the information displayed much match human needs [16].

Krug further reinforces Norman’s views that answers to these questions should be obvious to the user, he states that when using a website/application the functionality should be obvious and self-

explanatory, that users should be able to understand the purpose and functions of a website without having to think about it, put simply – don’t make users think! [17]

Further research into UX revealed Peter Morville’s diagram from 2004 which helps illustrate the areas of good user experience design in web applications which will be considered when developing the project.



Figure 2. Morville’s User Experience Honeycomb [18]

AREA	DESCRIPTION
Useful	Define innovative solutions that are useful.
Useable	Ease of use is vital. Usability is necessary but not sufficient.
Findable	Strive to design navigable websites.
Accessible	Websites should be accessible to users with disabilities.
Desirable	Appreciation of image, identity, brand, and other elements of emotional design.
Creditable	Ensure design influences user trust.
Valuable	Websites must deliver value.

Table 2. Descriptions of Morville’s seven areas of UX [19]

Table one and two provides a brief description of the interaction design five dimensions and Morville’s user experience honeycomb. A more detailed discussion will be included in the following literature review.

Throughout the project there will be a need for emphasis on excellent interaction design and user experience. Given the stressful nature of the industry of which the web application will be used it is important that the system is simple, easy to use and avoids creating more stress. A whitepaper by Bacon agrees that when products are difficult to understand and use, they can cause users pain, whether in a physiological form as stress stemming from frustration of not understanding a system or

physical pain of an elderly person whose dexterity will not allow them to perform simple tasks with products [15].

Emotional experience is a large part of UX and interaction design, a user is less likely to revisit a website, or use a system which they attach with negative emotions, even if said website or system is aesthetically pleasing. An example given by Norman in 2004's *'Emotional Design, why we Love it or Hate it'* is that of mechanical drawing instruments, all very easy on the eye but very difficult and frustrating to use [20]. Further indication that a design of a health board documentation system must place high priority on its users who must be able to negotiate the system efficiently.

Again, Krug offers further insight to be considered when developing the project when he iterates that users of a website just glance pages, quickly scanning text and then clicking on the first link which vaguely relates to what they are looking for – rendering large parts of the page useless [17]. This point will be of special importance in an A&E ward where users will be under pressure, taking even less time to consider choices, with potentially grave consequences.

Do computerised medical documentation systems help clinicians? The need for the system is clear, it is easier to type keywords into a search engine than it is to fondle through the index of a book, especially when under pressure, however, Noyes et al. refers to a study by Dillion [21] looking at three aspects of computer vs. paper tasks, in all three areas of reading speed, accuracy and comprehension the results show that paper is favoured, especially reading speed which is 20-30% quicker [22]. However, when looking at studies surrounding clinical decision support systems (CDSS's) the positive evidence is overwhelming. Lobach and Hammond conclude from a 6-month trial testing the effectiveness of a CDSS for clinicians treating diabetes patients saw a 32% increase in clinician compliance [23], further studies by Roshanov et al. also concluded that computerised systems improved the process of medical care by 52-64% [24]. Furthermore, Moja et al. also discuss in their 2014 journal that the use of CDSS's "*increase clinician adherence to guidelines*" and they "*ultimately improve the overall efficiency and quality of health care delivery systems*" they continue to add that such systems could potentially save EU member states over €300 million [25]. Even given the evidence supporting computerised systems it is clear to see that there is a benefit for medical purposes, giving impetus to the proposed project.

When designing the system for the MIU it will be important to look at other CDSS currently in use. One system already in use is NHS Direct Wales' symptom checker [26]. This offers a simple front end which somewhat adheres to interaction design standards, however, it is felt that the user experience was poor, upon testing for six different ailments, after answering a set of questions for each the website simply directed the user to call NHS Direct, which defeats the point of the system. In comparison, the WebMd [27] is very intuitive and simple to use, offering a search engine for symptoms over the NHS's list. Results offer information on condition details and a tab containing details of treatment options. Interaction with the system is excellent with user experience ranking high.

From looking at existing systems it will be imperative to research and analyse both good and bad web based systems, and if possible, software based systems such as First Databank [28], Cerner [29], Elsevier [30] and even IBM's complex Watson [31].

Model View Controller Framework using RESTful Approach

Architectural patterns in programming, and now, in web applications are reusable solutions to problems which occur regularly. They are templates for solving problems which allow for high level solutions independent of the implementation criteria which can be easily maintained, tested and importantly, understood [32]–[34].

Numerous triad patterns are in use in today's web applications; Presentation Abstraction Control (PAC), Model View Presenter (MVP), Model-View View Model (MVVM) to name a few, however, they are all siblings and share similarities with the most common pattern in use, MVC or Model View Controller [35]–[37], which was first introduced by Trygve Reenskaug in the 1970's [38]. This proposal will focus on the MVP pattern, others will be reviewed and discussed in the final literature review.

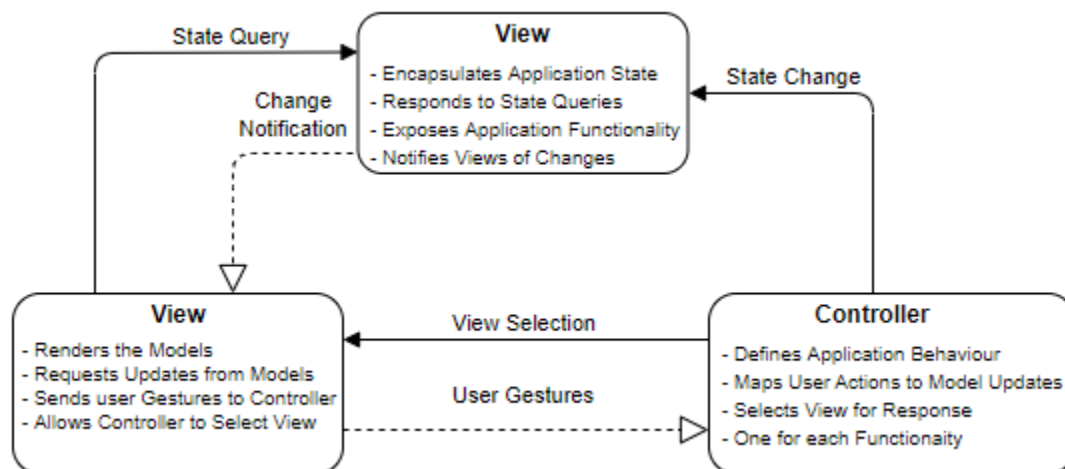


Figure 3. Model View Controller Diagram [39]

Figure two shows a diagram of an MVC where the;

Model – Manages the behaviour and data of the application, responding to instruction to change the state from the controller, and requests from the view for information about its state. It is the logic for the application – the code for the classes and functions [38], [40], [41].

View – Renders and displays the model content to the users, specifying how the data from the model should be presented. A view is html code which is read through a web browser [38], [40], [41]

Controller - Receives any inputs from users in the form of mouse or keyboard inputs, and calls to the model and view to update the data appropriately [37], [38], [40].

The benefits of using the MVC pattern are, as stated by MSDN; *separation*. The view and the controller both depend on the model, but the model does not rely on either the view or the controller, by separating the user interface logic and business logic, it allows the model to be developed and tested without the need for visual views [41]. The closeness of these objects ensures that changes to one component does not affect the rest of the system and removal and addition of features is possible without changes to existing code [42] as the data model is not tied to a singular presentation format which would limit the flexibility of the application [43].

In conjunction with an MVC framework, a RESTful API web service approach for development will also be researched and implemented into the web application. A brief background on REST and RESTful services is provided.

The Representational State Transfer (REST) is an architecture style which defines a set of constraints and properties based on, although not exclusive to, the HTTP protocol, which was first introduced by Roy Fielding in 2000 [44]. Fielding et al. state that *“REST is a coordinated set of architectural constraints that attempts to minimize latency and network communication while at the same time maximizing the independence and scalability of component implementations”* [45]. Ziemar adds in his 2002 paper that REST architecture is an abstraction of architectural elements within a distributed hyper media system [46], that is to say it focuses on a systems resources over HTTP by numerous clients written in different languages. REST has become the most popular architectural style, displacing SOAP and WSDL based interfaces [47].

A RESTful approach to a web application will adhere to the REST architectural style by obeying the constraint principles laid out by Fielding. These are briefly disclosed in the table below. More in-depth discussion will be included in following literature review.

PRINCIPLE	DESCRIPTION
Client - Server	Separate user-interface from data storage.
Stateless	Request from client to server must contain all necessary information.
Cacheable	Data must be labelled cacheable or non-cacheable.
Uniform Interface	Applying software principle of generality to component interface.
Layered System	Hierarchical layers constraining component behaviour.
Code on Demand	Extended functionality by downloading and executing scripts or applets.

Table 3. Guiding Principles of RESTful Web Services [48].

RESTful web services have a key characteristic in that they explicitly use HTTP methods, GET, POST, PUT, DELETE etc. to fetch data or execute queries from a web server, in turn establishing a one-to-one mapping between CRUD (Create, Read, Update, Delete) operations and HTTP methods [47][49].

A RESTful approach is simple and flexible, it is independent of platforms or languages, it allows for components to be developed in different languages (PHP, Python, Java, C#, Node.js etc.), on different servers, and it adapts to the type of syntax used. Japikse et al analogue a RESTful web service perfectly by stating “it is essentially a the MVC framework minus the “V” (view)” [50] that is understood in the sense that responses from the server are given in JSON or XML format which is universally readable by an array of devices, each with their own concept of a ‘view’. This allows for greater flexibility when developing and testing new environments [51]. Another key advantage includes the separation of the user interface from server data storage, this improves the interfaces portability and scalability of project [52]. These features will be of use when considering the scope of the project for the NHS.

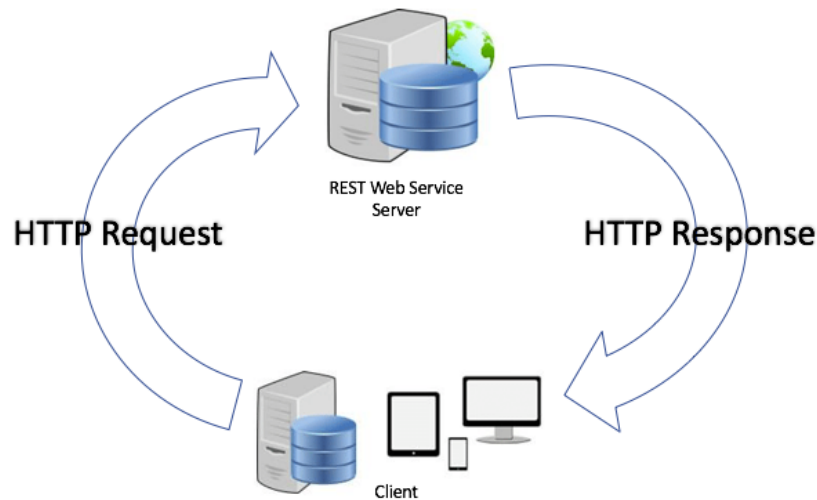


Figure 4. Restful Web Service [51]

When developing the web application there are many MVC frameworks available which allow for a RESTful service layer to be implemented; ASP.NET Core, Laravel, Node.js, Ruby on Rails, Symphony are just a few of many. Further research into frameworks available will be carried out before deciding on the best option to develop the MIU web application.

AIMS AND OBJECTIVES

The aim of the project is to design and develop a web application using an MVC framework with RESTful web service approach which will computerise current paper documentation of minor injury guidelines for Newport's MIU in a clear, concise format considering the latest studies in user interaction and experience for web front end design. If possible, the system will incorporate '*physical web*' technology utilising BLE (*Bluetooth Low Energy*) beacons around the unit.

To accomplish the propositioned aim, the following objectives will need to be realised;

- i. Compare current MVC frameworks which support RESTful web services and conclude a suitable choice to implement the web application.
- ii. Analyse and evaluate current medical information and diagnostic systems and current interaction design, user experience and presentation of data for medical purposes.
- iii. Write a system requirements specification for the proposed web application including a MoSCow analysis of the specified requirements.
- iv. To design, develop and construct the web application using the chosen MVC framework.
- v. To scrutinise and evaluate the completed system in the form of user feedback and reflect on findings to recommend future improvements.

RESEARCH METHODOLOGY

Research Philosophies

To conduct the conduct research correctly, the appropriate research philosophy must be selected. Firstly, table 4 below identifies the objectivist and subjectivist philosophies. Where objectivism seeks to research the truth of the world with observable and measurable facts, experienced by all social actors and subjectivists research considers different opinions, including that of their own and different social actors [53].

OBJECTIVIST	SUBJECTIVIST
Quantitative	Qualitative
Positivist	Phenomenological
Scientific	Humanistic
Experimentalist	Interpretivist
Traditionalist	Constructivist

Table 4. Objectivist and Subjectivist Philosophies [54]–[56]

In table 4 the two main research philosophies can be seen; positivist and phenomenological. Further reading of Hussey et al has pointed to the use of an objective, positivist philosophy, as the projects main aim is to develop a web (*software*) application. This has been decided taking into account ontological, epistemological, axiological, rhetorical and methodological assumptions where the positivist viewpoint best suits the project proposed, where, most notably, the epistemological assumption from a positivist view is to observe and measure to produce results which are measurable and credible, and the axiological assumption that research is carried out on facts not from interpretations [56].

The alternative phenomenological philosophy is where research is conducted and examined taking in to account individuals experiences and perceptions whilst considering the constructs of the social world into account [57]. While the main philosophy will be positivist, when user testing, at the end of the project, some phenomenological research in the form of a case study will be realised.

Research Approach

The approach taken to research can be done in two ways; *Inductive* or *Deductive*. In general, a positivist philosophy will utilise a deductive approach given that the area of research has a wide range of literature of which a hypothesis and theory can be tested, for example, the project may hypothesise that the web application developed will adhere to interaction design and user experience standards better than other systems in use, a research strategy can subsequently be designed to test the hypothesis [53]. A mainly deductive approach to this project will be taken.

The approach favoured by a phenomenological is inductive, where data will be collected and a theory will be developed on the back of the analysed data taking into account an understanding of the meanings of which humans attach to events, the researcher will also be a part of how the research is carried out and research will morph as the research is carried out [53]. Although not the main approach favoured by this project, the case study conducted upon completion of the project will be of an inductive approach.

Research Strategies

Focus is now turned to each of the research strategies which will be chosen for the project, Yin [2003] states that each of the strategies; experiment, survey, document analysis and case study can be used for explanatory, descriptive or exploratory research [58]. As the project will use both qualitative and quantitative methods of research in the form of a case study, document analysis and software development respectively the type of research is classed as explanatory as soft data will be collected upon completion, but statistics and hard data will be collected during research and testing.

Document Review

The first strategy employed, a document review, will be identifying the main theories in the fields of user interaction and user experience coupled with research into model view controllers and RESTful web services. Current research into medical systems which use an MVC framework will be looked at as well as research into front-end design to present data to users of a system. From this research a system will be designed, built and tested using practices and principles discovered. The project will then be evaluated to discover if work carried out supports or contradicts what has been researched.

Case Study

An inductive, phenomenological case study will show where a mixed methodology to research is evident in this project, where subjective evidence will be collected at the end of the project to fulfil aim v. of the aims and objectives laid out on page 9, where user feedback will be analysed and evaluated to recommend improvements to the system.

An experiment strategy was considered for this stage of the project as it would be possible to measure the change in independent variables [53], for example; experiment whether the implemented system actually made any change to medical diagnosis and missed symptoms by clinicians in the MIU, however a case study was chosen as research has already proven that a computerised system in a medical setting such as an MIU does increase clinician accuracy in diagnosis and this project is focused on delivering a system which is tailored around the users of which it is designed for.

The case study strategy is defined by Robson as “development of detailed, intensive knowledge about a single ‘case’, or of a small number of related ‘cases’” [59] where the MIU will be the ‘case’ studied, with different staff members being interviewed, surveyed and observed to obtain data from usability tests on how they interact with the system. The qualitative and quantitative data collected will be used to measure the staff’s satisfaction with the web application and implement improvements.

One method will be to use the System Usability Scale (SUS) which is described as a “quick and dirty, reliable tool for measuring usability” [60], it consists of 10 questions which will be presented to the user in the form of a Likert scale, scores of 0-40 are then collated and calculated to give a score out of 100 where 68 and above is considered above average. This technique was originally pioneered by John Brooke in 1986 to test systems of this nature [61].

Development Methodology

A software development strategy will be taken for the web application. Methodology surrounding software development is vast with many different techniques including the Waterfall, Spiral and Agile methods, however, it has been concluded that for meticulousness, the Structured Systems Analysis and Design Method (SSADM) will be taken, which is a modified Waterfall methodology.

SSADM was developed for the UK government in the 1980’s and is the standard method used today for government departments [62]. This methodology places an emphasis on the analysis and design stage of the project for user needs [63], [64] which makes it a perfect match to the web application proposed which requires an emphasis on usability.

In figure 5 a system development lifecycle is shown, of which the six main areas of the SSADM methodology encapsulate. These are;

- i. Feasibility
- ii. Investigation of Current Environment
- iii. Business Systems Options
- iv. Requirements Specification
- v. Technical Systems Options
- vi. Logical Design
- vii. Physical Design

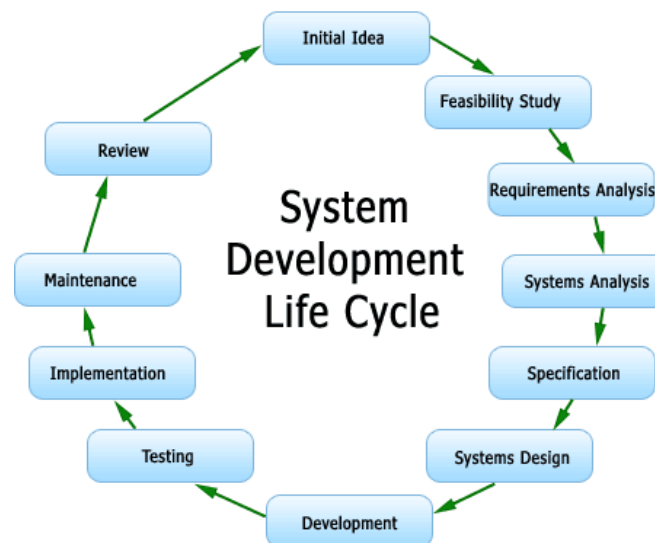


Figure 5. System Development Lifecycle [65]

Feasibility - a study is carried out to discover whether the project is technically feasible in terms of financial and social aspects. As client (NHS Wales) has requested the development of the system, this section of the methodology will be omitted.

Investigation of Current Environment – stage at where detailed research into the current system is carried out, however, as the current system is not computerised, detailed research will be carried out on existing systems in the medical industry.

Business System Options – from research carried out in the preceding stage, a range of solutions to the proposed web application will be analysed and one selected.

Requirements Specification – a full requirements specification, including a MoSCow analysis will be developed to outline in detail the systems capabilities.

Technical Systems Options – this section will outline the researched MVC frameworks with one framework being selected to continue implementation of system.

Logical Design – focuses on the user interaction requirements, updating the main methods of experience and interaction, specifying what the system will do.

Physical Design – logical system specification is used to implement physical design and complete the specification of the systems functions.

This methodology, although a difficult to implement, can produce well documented and accurate information systems – which is what is required by the project. Although the developers favoured methodology, the final methodology may include some aspects of the agile and time boxing methods, this will be decided between the developer and the MIU contact, Clare McCarthy during their meeting.

Analysis and Deliverables

The analysis of the finished web application will be carried out in three areas; analysis of the system and user requirements, software testing and finally user testing.

Data collected during the research and development stage in regard to system and user requirements, using the MoSCoW method will be revisited with comparisons drawn between the desired requirements and the requirement which have been implemented to discover if the project has achieved the aims laid out in this proposal.

Secondly, software testing will be carried out using a range of standardised methods including, black box testing, white box testing and unit testing to ensure that the web application is fit for purpose and up to the task of assisting medical professionals in the MIU.

Finally, with the aid of the case study into the system, user interaction and user experience will be measured to see if, firstly, the correct MVC framework has been implemented and the use of a RESTful service has been justified. Secondly, the front-end design will be scrutinised through the case study to ensure that the research into these fields of study has been translated into a useable system for the staff at the MIU.

PROJECT MANAGEMENT

Risks Associated with Project

With a project of this scale for an organisation of the NHS's nature, there are many risk factors involved which may interfere with the project's completion. In the following section of the proposal, the risks will be identified along with a level of probability for each risk, the risks will then be compiled into a risk evaluation matrix to visualise the risk to the project.

Identified Risks

R1. Underestimating the time required to build the web application.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R2. Incorrect balance of project aim and objectives with other modules of study throughout the final year of the B.Sc. course.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R3. Selecting an MVC framework which is unsuitable, or which is written in a programming language new to developer.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R4. Unrealistic targets set during planning process resulting in aims and objectives not being met.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R5. User testing may prove difficult due to nature of the organisation, staff often very busy, observations may mean observer in awkward positions.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R6. Input of data from MIU guidelines will be time costly.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R7. The requirements of the web application may change over the course of the project.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R8. Communication between developer and MIU staff contact will be limited.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R9. Illness could affect developer's time scale, and also affect communication between developer and MIU as A&E will be busier.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R10. Change in Government or NHS guidelines may not allow system developed out of house by a student to be implemented into a real world hospital.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

R11. Risk of 'Gold Plating' – adding unnecessary features which may please aesthetically but will set back the project timescales.

LOW	MODERATE	SIGNIFICANT	HIGH
-----	----------	-------------	------

Risk Evaluation Matrix

		SEVERITY			
		LOW	MODERATE	SIGNIFICANT	HIGH
PROBABILITY	LOW	R11		R7, R3	R10
	MODERATE		R5	R9	R4
	SIGNIFICANT	R8		R1, R2	
	HIGH	R6			

Figure 6. Risk Evaluation Matrix for Risks Identified

The risk evaluation matrix in figure 6 shows that the risks to the project mainly have a high probability of happening with a mainly significant or high severity attached, with 3 risks over the tolerance line - highlighting that delivering a project of this size and complexity with a small time budget will prove very challenging. In order to minimise the risks, each one will now be evaluated in an assessment table to provide methods which will aid in managing risk.

Risk Assessment Table

RISK	PROBABILITY	SEVERITY	MANAGEMENT METHOD
R1	Significant	Significant	Use project planning tools to schedule each process. Use a MoSCow analysis to identify areas where reduced functionality can be implemented if timescale is stretched.
R2	Significant	Significant	Realistic targets should be set and stuck to for the project with other modules in mind.
R3	Low	Significant	Conduct correct and thorough research to make the correct decision when deliberating over a framework.
R4	Moderate	High	Correct time management will ensure that all aims are achieved.
R5	Moderate	Moderate	Liaise with Clare McCarthy and discuss best times to perform testing, incorporate this into the project plan.
R6	High	Low	Initial system will omit the full 195 page guidelines – full data entry can be achieved if system is to be implemented into the MIU.
R7	Low	Significant	Requirements must be agreed before project commences between developer and Ms McCarthy.
R8	Significant	Low	Ensure that upon first arranged meeting enough information is gathered about the requirements of the system in preparation for minimal contact between parties.
R9	Moderate	Significant	Illness is unavoidable, without a larger team there will be no one to cover. Ensure that illness is factored in to project planning timescale.
R10	Low	High	The project can still be completed, however implementation into the MIU will not be possible.
R11	Low	Low	Ensure that any extra work is done if the project has met all the timescale targets. Only then can extra features be added.

Table 5. Management Methods for Identified Risks

Ethical Issues

No ethical or legal issues have been identified at this stage of the project, had the project involved patient records – which is in consideration for the future – then it would raise many ethical and legal issues. As the project progresses it is expected to reveal more issues of this nature, where the project will aim to adhere to the British Computer Society's (BCS) and Institute of Electrical and Electronic Engineers (IEEE) codes of conduct, these will be documented as and when discovered.

Project Resources

With any project it is important to identify the resources required for the task prior to commencing. In the following section of this proposal the projects stakeholders and authorities will be identified along with the physical resources which are required to meet the projects objective.

Authorities and Stakeholders

For this project, the following people have been identified as authorities and stakeholders.

Overall Project Authority & Stakeholder

Peter Davies – UWTSD student who has chosen to undertake the task of computerising the MIU's guidelines for his final year project. As project manager, control and overall responsibility for the project will be his.

Contact: pdavies.079868@student.uwtss.ac.uk

Secondary Project Authorities and Stakeholders

Claire McCarthy – NHS Wales' consultant nurse for emergency care at Newport's Royal Gwent hospital. The project has been commissioned by Ms McCarthy to aid her staff in diagnosis of patients.

Contact: claire.mccarthy@wales.nhs.uk

Dr Nicola Whitehead – UWTSD lecturer who commissioned this project to students of UWTSD. Dr Whitehead will be the authority on her areas of expertise surrounding this project; research and development, interaction design and user experience.

Contact: nicola.whitehead@uwtss.ac.uk

Dr Kemi Ademoye – UWTSD lecturer who has joined the project as joint supervisor to offer guidance in her specialist field of application development for the web. Dr Ademoye will provide support for all aspects of the backend development.

Contact: kemi.ademoye@uwtss.ac.uk

Physical Project Resources

The following physical resources have been acknowledged to meet project aims and objectives.

Hardware Resources

Personal Computer and Laptop – much of the projects work will commence on student's own time which will require the use of a personal desktop computer for work at home and personal laptop for work elsewhere.

University Computers & Labs – required for work which will be carried out at the university when supervision and/or support is required from project authorities and lecturers.

Software Resources

Microsoft Visual Studio – offering an excellent platform for development, MS VS will aid in the front and back end development, particularly if the chosen MVC framework is that of ASP.NET Core. If another framework is preferred then additional software will be required, for example, should Laravel be chosen, the dependency manager *Composer* will be needed.

Microsoft SQL Server – to build the initial database MS SQL Server will be required – if the *database first* approach to the MVC development is taken.

Chrome Web Browser – the most comprehensive browser including tools for development which will prove invaluable over the course of the project. Also required to conduct research into existing work of this nature.

Diagram Tools – to illustrate work throughout applications like Microsoft's Visio, Tableau or Draw.io's web application will be required for diagrams, for example; entity relationship diagrams.

Microsoft Project – project planning tool which will aid correct time management. This software will be used in the *project timeline* section of this proposal.

Other Resources

Project Team – Peter Davies, Clare McCarthy, Dr Nicola White Head and Dr Kemi Ademoye make up the team resource for the project.

USTWD Library – the new library at SA1 and the existing online library will assist in providing research materials in the form of books, journals and articles.

Minor Injury Unit Guidelines – the guidelines provided by Ms McCarthy is the resource of information to be computerised. This has been provided in Microsoft Word format.

System Usability Scale – the SUS will be required for collection of data upon completion and implementation of the project. A guide is available at Useability.gov [60].

Newport MIU Staff – required for observation and interviews regarding the system for results analysis and evaluation.

Project Timeline

To assist in the correct planning of the project a Gantt chart has been built using Microsoft Project. The project tasks have been broken down into five different sections; research, requirements & specification, development and implementation, testing and analysis and evaluation. In these five sections, each task is identified, in order, with timescale attached.

	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1	Research	110 days	Mon 04/06/18	Fri 02/11/18		
2	Literature Review	110 days	Mon 04/06/18	Fri 02/11/18		
3	Requirements & Specification	55 days	Mon 08/10/18	Fri 21/12/18		
4	Meeting with Clare McCarthy	1 day	TBC	TBC		Clare McCarthy
5	Identify Final Aims & Objectives	6 days	Mon 08/10/18	Mon 15/10/18		Peter Davies
6	Identify Final System Requirements	6 days	Tue 16/10/18	Tue 23/10/18	5	Peter Davies
7	Select Appropriate MVC Framework	1 day	Mon 05/11/18	Mon 05/11/18	1,5,6	Peter Davies
8	Identify Final Legal/Ethical/Risk/Sustainability Issues	5 days	Tue 06/11/18	Mon 12/11/18	1,5,6,7	Peter Davies
9	Write up Final Methodology	29 days	Tue 13/11/18	Fri 21/12/18		Peter Davies
10	Development & Implementation	43 days	Mon 07/01/19	Wed 06/03/19	1	
11	Evaluate Data to be Inputted to System	6 days	Mon 07/01/19	Mon 14/01/19		Peter Davies
12	Draw up ERD for System	6 days	Tue 15/01/19	Tue 22/01/19	11	Peter Davies
13	Implement Designed Database Structure into MVC Framework	11 days	Wed 23/01/19	Wed 06/02/19	12	Peter Davies
14	Testing	12 days	Thu 07/03/19	Fri 22/03/19	10	
15	Perform Software Tests (Black & White Box / Unit)	6 days	Thu 07/03/19	Thu 14/03/19		Peter Davies
16	User Testing - Observations	1 day	Fri 15/03/19	Fri 15/03/19	15	MIU Staff
17	User Testing - System Usability Survey	1 day	Fri 15/03/19	Fri 15/03/19	15	MIU Staff
18	Analysis & Evaluation	15 days	Mon 25/03/19	Fri 12/04/19	14	
19	Analysis of Results	6 days	Mon 25/03/19	Mon 01/04/19		Peter Davies
20	Evaluate Project	7 days	Tue 02/04/19	Wed 10/04/19		Peter Davies

Figure 7. Project Gantt chart.

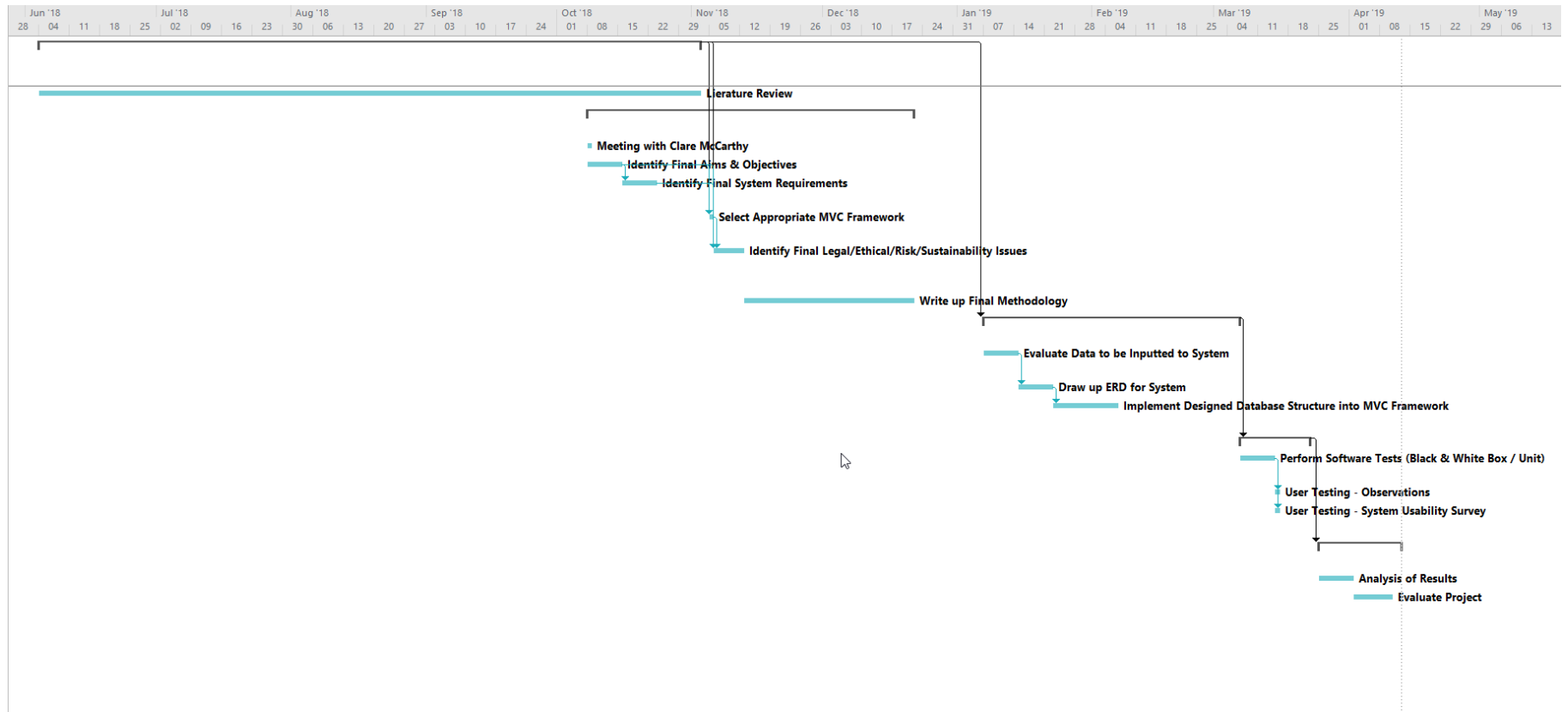


Figure 8. Timeline of Gantt chart

		Resource Name	Type	Material	Initials	Group	Max.	Std.	Ovt.	Cost/Use	Accrue	Base
1		Peter Davies	Work		P		100%	£0.00/hr	£0.00/hr	£0.00	Prorated	Standard
2		MIU Staff	Work		M		100%	£0.00/hr	£0.00/hr	£0.00	Prorated	Standard
3		Clare McCarthy	Work		C		100%	£0.00/hr	£0.00/hr	£0.00	Prorated	Standard
4		Home PC & Laptop	Material		H			£0.00		£0.00	Prorated	
5		Univeristy Computers	Material		U			£0.00		£0.00	Prorated	
6		SQL Server	Material		S			£0.00		£0.00	Prorated	
7		MS Visual Studio	Material		M			£0.00		£0.00	Prorated	
8		Chrome Web Browser	Material		C			£0.00		£0.00	Prorated	
9		Diagram Tools	Material		D			£0.00		£0.00	Prorated	
10		MS Project	Material		M			£0.00		£0.00	Prorated	
11		Library	Material		L			£0.00		£0.00	Prorated	
12		System Usability Scale	Material		S			£0.00		£0.00	Prorated	
13		MIU Guidelines	Material		M			£0.00		£0.00	Prorated	
14		Document Analysis	Cost		D						Prorated	

Figure 9. Resources Required for Project

It is important to stress that although every effort will be made to adhere to the proposed project plan, there will be instances where this will not be possible. As with every project, the unexpected can and will happen, which in turn will adjust the timescales given in the project plan. However, with determination, the project will see the finish line.

REFERENCES

- [1] T. Siang, "What is Interaction Design? | Interaction Design Foundation," *Interaction Design Foundation*, 2018. [Online]. Available: <https://www.interaction-design.org/literature/article/what-is-interaction-design>. [Accessed: 26-Apr-2018].
- [2] "About & History – Interaction Design Association – IxDA," *IxDA*. [Online]. Available: <http://ixda.org/ixda-global/about-history/>. [Accessed: 26-Apr-2018].
- [3] J. Lowgren, "Interaction Design - brief intro | The Encyclopedia of Human-Computer Interaction, 2nd Ed.," *Interaction Design Foundation*. Interaction Design Foundation.
- [4] D. Saffer, *Designing for Interaction*, 2nd ed. Berkely: New riders, 2010.
- [5] A. Cooper, R. Reimann, and D. Cronin, *About Face 3: The essentials of interaction design*, vol. 3, no. 3. Indianapolis: Wiley Publishing, Inc., 2007.
- [6] "The Mother of All Demos, presented by Douglas Engelbart (1968) - YouTube." [Online]. Available: <https://www.youtube.com/watch?v=yJDv-zdhzMY>. [Accessed: 26-Apr-2018].
- [7] "Doug's 1968 Demo - Doug Engelbart Institute." [Online]. Available: <http://www.dougenelbart.org/firsts/dougs-1968-demo.html>. [Accessed: 26-Apr-2018].
- [8] B. Moggridge, *Designing Interactions*. Cambridge, Massachusetts, 2007.
- [9] K. Silver, "What Puts the Design in Interaction Design :: UXmatters," *UX Matters*, 2007. [Online]. Available: <https://www.uxmatters.com/mt/archives/2007/07/what-puts-the-design-in-interaction-design.php>. [Accessed: 26-Apr-2018].
- [10] I. Instructor, "The Five Languages or Dimensions of Interaction Design | Interaction Design Foundation," *Interaction Design Foundation*, 2016. [Online]. Available: <https://www.interaction-design.org/literature/article/the-five-languages-or-dimensions-of-interaction-design>. [Accessed: 26-Apr-2018].
- [11] G. Smith Crampton, "What is Interaction Design," in *Designing Interactions*, Cambridge, Massachusetts, 2007, p. xvii.
- [12] "What is the difference between Interaction Design and UX Design? | Interaction Design Foundation," 2016. [Online]. Available: <https://www.interaction-design.org/literature/article/what-is-the-difference-between-interaction-design-and-ux-design>. [Accessed: 26-Apr-2018].
- [13] "What is User Experience (UX) Design? | Interaction Design Foundation." [Online]. Available: <https://www.interaction-design.org/literature/topics/ux-design>. [Accessed: 26-Apr-2018].
- [14] "User Experience Basics," Feb. 2014.
- [15] E. Bacon, *Interaction Design for Medical Systems*. Devise Consulting, 2013.
- [16] D. Norman, *The Design of everyday things*. New York: Basic Books, 2001.
- [17] S. Krug, *Don't Make Me Think!*, vol. 277, no. 1681. 2006.
- [18] P. Morville, "User Experience Honeycomb," *Intertwined*. [Online]. Available: <http://intertwined.org/user-experience-honeycomb/>. [Accessed: 26-Apr-2018].
- [19] P. Morville, "User Experience Design," *Semantic Studios*. [Online]. Available:

http://semanticstudios.com/user_experience_design/. [Accessed: 26-Apr-2018].

- [20] D. A. Norman, *Emotional Design - Why we love (or hate) everyday things*, no. 1. New York: Pegasus Books, 2004.
- [21] A. Dillion, "Reading from paper versus screens: a critical review of the empirical literature," *Ergonomics*, vol. 35, no. 10, pp. 1297–1326, Oct. 1992.
- [22] J. M. Noyes and K. J. Garland, "Computer-vs. paper-based tasks: Are they equivalent?"
- [23] D. F. Lobach and W. E. Hammond, "Computerized Decision Support Based on a Clinical Practice Guideline Improves Compliance with Care Standards," *Am. J. Med.*, vol. 102, no. 1, pp. 89–98, Jan. 1997.
- [24] P. S. Roshanov *et al.*, "Features of effective computerised clinical decision support systems: meta-regression of 162 randomised trials.," *BMJ*, vol. 346, p. f657, Feb. 2013.
- [25] L. Moja *et al.*, "Effectiveness of computerized decision support systems linked to electronic health records: A systematic review and meta-analysis," *Am. J. Public Health*, vol. 104, no. 12, pp. e12–e22, 2014.
- [26] "NHS Direct Wales | Symptom Checker." [Online]. Available: <https://www.nhsdirect.wales.nhs.uk/SelfAssessments/default.aspx>. [Accessed: 27-Apr-2018].
- [27] "Symptom Checker from WebMD. Check Your Medical Symptoms." [Online]. Available: <https://symptoms.webmd.com/#/info>. [Accessed: 27-Apr-2018].
- [28] "FDB MedKnowledge - Healthcare's Leading Drug Knowledge Solution | FDB (First Databank)." [Online]. Available: <http://www.fdbhealth.com/fdb-medknowledge-brochure-landing/?cid=US PS GG NA BN BN TXT 3 LPD 10232016 First Data Bank p 1t2&gclid=CMbHuLuvktMCFcKEswodPQoINg>. [Accessed: 27-Apr-2018].
- [29] "Provider Solutions | Cerner." [Online]. Available: <https://www.cerner.com/solutions/provider>. [Accessed: 27-Apr-2018].
- [30] "Clinical Solutions | Clinical Decision Support | Evidence-Based Content." [Online]. Available: <https://www.elsevier.com/clinical-solutions>. [Accessed: 27-Apr-2018].
- [31] "IBM Watson – Watson in healthcare – United Kingdom." [Online]. Available: https://www-05.ibm.com/innovation/uk/watson/watson_in_healthcare.shtml. [Accessed: 27-Apr-2018].
- [32] "Design Patterns." [Online]. Available: <https://msdn.microsoft.com/en-us/library/ff649977.aspx>. [Accessed: 28-Apr-2018].
- [33] B. L., C. P., and K. R., *Software Architecture in Practice: Second Edition*. Addison-Wesley, 2005.
- [34] C.-H. Chang, C.-W. Lu, C.-H. Lin, M.-F. Yang, and C.-F. Tsai, "An Experience of Applying Pattern-based Software Framework to Improve the Quality of Software Development: 4. The Design and Implementation of OS2F," *J. Softw. Eng. Stud. Vol. 2, No. 6*, pp. 185–194, 2008.
- [35] A. Ferrara, "Alternatives To MVC | ircmaxell's Blog." [Online]. Available: <https://blog.ircmaxell.com/2014/11/alternatives-to-mvc.html>. [Accessed: 27-Apr-2018].
- [36] H. B. Prajapati and V. K. Dabhi, "High Quality Web-Application Development on Java EE Platform," in *2009 IEEE International Advance Computing Conference*, 2009, pp. 1664–1669.
- [37] J. Deacon, "Model View Controller (MVC) Arcitecture," *Comput. Syst. Dev. Consult. Train.*,

2009.

- [38] R. Eckstein, "Java SE Application Design With MVC," *Oracle*, 2007. [Online]. Available: <http://www.oracle.com/technetwork/articles/javase/index-142890.html>. [Accessed: 29-Apr-2018].
- [39] C. Horstman, *Object Orientated Design & Patterns*, 2nd ed. San Jose: Wiley & Sons, 2006.
- [40] N. Latifi, "Architecture patterns in web applications and implementation of find local food," 2010.
- [41] "Model-View-Controller." [Online]. Available: <https://msdn.microsoft.com/en-us/library/ff649643.aspx>. [Accessed: 29-Apr-2018].
- [42] P. Worrall and T. Chaussalet, "Development of a web-based system using the model view controller paradigm to facilitate regional long-term care planning," *Proc. - IEEE Symp. Comput. Med. Syst.*, no. February 2015, 2011.
- [43] L. Shklar and R. Rosen, *Web Application Architecture: Principles, Protocols and Practices*. West Sussex: Wiley & Sons, 2003.
- [44] R. Fielding, "Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST)," Irvine, 2000.
- [45] R. T. Fielding and R. N. Taylor, "Principled Design of the Modern Web Architecture."
- [46] S. Ziemer, "An Architecture for Web Applications Essay in DIF 8914 Distributed Information Systems," 2002.
- [47] A. Rodriguez, "RESTful Web services: The basics," 2015.
- [48] "What is REST – Learn to create timeless RESTful APIs.," *RESTful API*. [Online]. Available: <https://restfulapi.net/>. [Accessed: 29-Apr-2018].
- [49] "RESTful Web Services Introduction," *Tutorials Point*. [Online]. Available: https://www.tutorialspoint.com/restful/restful_introduction.htm. [Accessed: 29-Apr-2018].
- [50] P. Japikse, K. Grossnicklaus, and B. Dewey, *Building Web Applications with Visual Studio 2017*, 1st ed. New York: Apress, 2017.
- [51] A. et al. Manitsaris, "Final Report on System Specification - FP7 i- Final Report on System Specification," no. September, 2015.
- [52] "REST API: What is it, and what are its advantages in project development?," *BBVA Open 4U*. [Online]. Available: <https://bbvaopen4u.com/en/actualidad/rest-api-what-it-and-what-are-its-advantages-project-development>. [Accessed: 29-Apr-2018].
- [53] M. Saunders, P. Lewis, and A. Thornhill, "Understanding research philosophies and approaches," *Res. methods Bus. students* 4, no. January 2009, pp. 106–136, 2009.
- [54] M. T. Holden and P. Lynch, "Choosing the Appropriate Methodology: Understanding Research Philosophy."
- [55] C.-C. Chen, "Quantitative Methodology: Appropriate use in Research for Blind Baseball Ergonomics and Safety Design."
- [56] J. Collis and R. Hussy, *Business Research - A Practical Guide for Undergraduate and Postgraduate Students*, 4th ed. London: Macmillan, 2014.

- [57] M. Saunders, P. Lewis, and A. Thornhill, *Research Methods for Business Students*. 2008.
- [58] R. K. Yin, *CASE STUDY RESEARCH Design and Methods Second Edition*, 2nd ed. London: Sage Publications, 2003.
- [59] C. Robson, *Real world research*. Blackwell Publishing, 2002.
- [60] A. S. for P. Affairs, "System Usability Scale (SUS)," Sep. 2013.
- [61] J. Brooke, "SUS -A quick and dirty usability scale Usability and context."
- [62] C. M. Ashworth, "Structured systems analysis and design method (SSADM)," *Inf. Softw. Technol.*, vol. 30, no. 3, pp. 153–163, Apr. 1988.
- [63] "What is SSADM?," *HN Computing*. [Online]. Available: https://www.sqa.org.uk/e-learning/SDM01CD/page_02.htm. [Accessed: 01-May-2018].
- [64] "Structured Systems Analysis and Design Methodology."
- [65] T. Trott, "System Lifecycles and SSADM - Azulia Designs," *Azulia Designs*, 2010. [Online]. Available: <http://azuliadesigns.com/system-lifecycles-ssadm/>. [Accessed: 01-May-2018].

BIBLIOGRAPHY

Morville, P., & Rosenfeld, L. (2007). *Designing Large-Scale Web Sites*. (S. St. Laurent & R. Monaghan, Eds.) (3rd ed.). Sebastopol: O'Reilly. Retrieved from

<http://yunus.hacettepe.edu.tr/~tonta/courses/fall2010/bby607/IAWWW.pdf>

Garret James, J. (2000). The Elements of User Experience. Retrieved from

<http://www.jjg.net/elements/pdf/elements.pdf>

Silver, K. (2007). What Puts the Design in Interaction Design :: UXmatters. Retrieved April 26, 2018, from <https://www.uxmatters.com/mt/archives/2007/07/what-puts-the-design-in-interaction-design.php>

Hargreaves, K. (n.d.). A Beginner's Guide to User Experience (UX) Fundamentals | General Assembly. Retrieved April 26, 2018, from <https://generalassemb.ly/design/visual-design/user-experience-fundamentals>

Neilson, J. (n.d.). A 100-Year View of User Experience (by Jakob Nielsen). Retrieved April 26, 2018, from <https://www.nngroup.com/articles/100-years-ux/>

interaction design. (n.d.). Retrieved from

<http://web.mit.edu/21w.789/www/spring2014/notes/designclass14.pdf>

Defining UX | deviseconsulting.com. (n.d.). Retrieved April 26, 2018, from

<https://deviseconsulting.com/defining-ux/>

Moja, L., Kwag, K. H., Lytras, T., Bertizzolo, L., Brandt, L., Pecoraro, V., ... Bonovas, S. (2014). Effectiveness of computerized decision support systems linked to electronic health records: A systematic review and meta-analysis. *American Journal of Public Health, 104*(12), e12–e22. <https://doi.org/10.2105/AJPH.2014.302164>

Kolandaisamy, R., & Noor Md, R. (2013). Web Based Online Medical Diagnosis System (Womeds), (June 2016).

IBM Watson – Watson in healthcare – United Kingdom. (n.d.). Retrieved April 27, 2018, from https://www-05.ibm.com/innovation/uk/watson/watson_in_healthcare.shtml

Turpin, P. G. (n.d.). Transitioning From Paper to Computerized Documentation Computerized Systems in Nursing. Retrieved from https://www.sgna.org/Portals/0/Education/PDF/Minimum-Data-Set/TransitionFromPapertoComputerizedDoc_Turpin.pdf

Crossing the Quality Chasm: A New Health System for the 21st Century. (2001), 364(9), 0–309. Retrieved from <http://www.nap.edu/catalog/10027.html>

What Is Using IBM Watson In Everyday Medicine Like? - The Medical Futurist. (n.d.). Retrieved April 27, 2018, from <http://medicalfuturist.com/what-is-using-ibm-watson-in-medicine-like>

Ross, C. (2017). IBM pitched Watson as a revolution in cancer care. It's nowhere close. Retrieved April 27, 2018, from <https://www.statnews.com/2017/09/05/watson-ibm-cancer/>

Orlowski, A. (2017). AI in Medicine? It's back to the future, Dr Watson • The Register. Retrieved April 27, 2018, from https://www.theregister.co.uk/2017/09/25/ai_in_medicine/

Gorski, D. (n.d.). IBM's Watson versus cancer: Hype meets reality – Science-Based Medicine. Retrieved April 27, 2018, from <https://sciencebasedmedicine.org/ibm-watson-versus-cancer-hype-meets-reality/>

NG, A. (2016). IBM's Watson gives proper diagnosis after doctors were stumped - NY Daily News. Retrieved April 27, 2018, from <http://www.nydailynews.com/news/world/ibm-watson-proper-diagnosis-doctors-stumped-article-1.2741857>

Friedman F, L. (2014). IBM's Watson May Soon Be The Best Doctor In The World - Business Insider. Retrieved April 27, 2018, from <http://www.businessinsider.com/ibms-watson-may-soon-be-the-best-doctor-in-the-world-2014-4?IR=T>

Farr, C. (2017). IBM's struggle with Watson in health care doesn't mean tech can't help. Retrieved April 27, 2018, from <https://www.cnn.com/2017/09/07/ibms-struggle-with-watson-in-health-care-doesnt-mean-tech-cant-help.html>

Nodder, C. (2013). *Evil by Design*. Indianapolis: Wiley Publishing, Inc.
<https://doi.org/10.1073/pnas.0703993104>

Vaqqas, M. (2014). RESTful Web Services: A Tutorial | Dr Dobb's. Retrieved April 27, 2018, from <http://www.drdoobs.com/web-development/restful-web-services-a-tutorial/240169069>

Mittal, A. (2016). RESTful Day #1: Enterprise Level Application Architecture with Web APIs using Entity Framework, Generic Repository Pattern and Unit of Work - CodeProject. Retrieved April 27, 2018, from https://www.codeproject.com/Articles/990492/RESTful-Day-sharp-Enterprise-Level-Application#_Toc418969116

Marty Stepp, by, Ernst, M., Reges, S., Notkin, D., Mercer, R., & Boswell, R. (n.d.). WEB APP ARCHITECTURES: MULTI-TIER (2-TIER, 3-TIER) MODEL-VIEWER-CONTROLLER (MVC) REST ARCHITECTURAL STYLE Slides created by Manos Papagelis Based on materials. Retrieved from <https://pdfs.semanticscholar.org/presentation/10b3/d6629330c9c1804847671a4815e125e63a67.pdf>

Web application architecture: Components, models and types. (n.d.). Retrieved April 27, 2018, from <https://www.scnsoft.com/blog/web-application-architecture>

Web Architecture. (n.d.). Retrieved April 27, 2018, from https://en.ryte.com/wiki/Web_Architecture

Common web application architectures | Microsoft Docs. (n.d.). Retrieved April 27, 2018, from <https://docs.microsoft.com/en-us/dotnet/standard/modern-web-apps-azure-architecture/common-web-application-architectures>

MVC Architecture. (n.d.). Retrieved April 27, 2018, from <http://www.tutorialsteacher.com/mvc/mvc-architecture>

ASP.NET MVC Overview. (n.d.). Retrieved April 27, 2018, from [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx)

Symfony Quick Tour: The Big Picture. (n.d.). Retrieved April 27, 2018, from https://symfony.com/doc/current/quick_tour/the_big_picture.html

Laravel - The PHP Framework For Web Artisans. (n.d.). Retrieved April 27, 2018, from <https://laravel.com/>

Dilloway, M. (n.d.). The Best MVC Frameworks For 2017 - Impression. Retrieved April 27, 2018, from <https://www.weareimpression.co.uk/best-mvc-frameworks-2017/>

General - Phalcon Framework. (n.d.). Retrieved April 27, 2018, from <https://docs.phalconphp.com/en/3.3>

Malhotra, M. (n.d.). 10 Top Web Development Frameworks In 2017 - DZone Web Dev. Retrieved April 27, 2018, from <https://dzone.com/articles/10-top-web-development-frameworks-in-2017-2>

Ferrara, A. (n.d.). A Beginner's Guide To MVC For The Web | ircmaxell's Blog. Retrieved April 27, 2018, from <https://blog.ircmaxell.com/2014/11/a-beginners-guide-to-mvc-for-web.html>

MVC architecture - App Center | MDN. (n.d.). Retrieved April 27, 2018, from https://developer.mozilla.org/en-US/Apps/Fundamentals/Modern_web_app_architecture/MVC_architecture

Taylor, R. N., Medvidović, N., & Dashofy, E. M. (Eric M. (2010). *Software architecture : foundations, theory, and practice*. Wiley. Retrieved from <https://search-proquest-com.ezproxy.uwtsd.ac.uk/docview/356979780/fulltext/FB1F79C30AEB4892PQ/1?accountid=130472>

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1993). Design Patterns: Abstraction and Reuse of Object-Oriented Design. In *ECOOP' 93 — Object-Oriented Programming* (pp. 406–431). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-47910-4_21

Evans, E. (2004). *Domain-driven design : tackling complexity in the heart of software*. Addison-Wesley.

Nilsson, J. (2006). *Applying domain-driven design and patterns : with examples in C# and .NET*. Addison-Wesley.

Taleb, M., Seffah, A., & Abran, A. (n.d.). Pattern-Oriented Architecture for Web Applications. Retrieved from <https://pdfs.semanticscholar.org/9243/a0bddd57c52943472e41adfc9241c3e14dca.pdf>

Professor, S. (n.d.). JOURNAL OF OBJECT TECHNOLOGY A Survey of Design Pattern Based Web Applications. *Journal of Object Technology*, 8(2), 61–70. Retrieved from http://www.jot.fm/issues/issue_2009_03/column5

Patrick, C. (2014). Literature Review Service Frameworks and Architectural Design Patterns in Web Development. Retrieved from <https://people.cs.uct.ac.za/~cpatrick/Honours/files/LiteratureReviewPTRCON001.pdf>

Skonnard, A. (2009). RESTful Services with ASP.NET MVC and XHTML. Retrieved April 29, 2018, from <https://msdn.microsoft.com/en-gb/magazine/dd943053.aspx>

Vaqqas, M. (2014). RESTful Web Services: A Tutorial | Dr Dobb's. Retrieved April 29, 2018, from <http://www.drdoobs.com/web-development/restful-web-services-a-tutorial/240169069>

REST Principles and Architectural Constraints – REST API Tutorial. (n.d.). Retrieved April 29, 2018, from <https://restfulapi.net/rest-architectural-constraints/>

Reenskaug, T., & Coplien, J. O. (2009). The DCI Architecture: A New Vision of Object-Oriented Programming. Retrieved April 29, 2018, from https://www.artima.com/articles/dci_vision.html

Snieder, R., & Lerner, K. L. (2009). *The art of being a scientist : a guide for graduate students and their mentors*. Cambridge University Press. Retrieved from <https://wenku.baidu.com/view/7a818118964bcf84b9d57bc9>

Affairs, A. S. for P. (2013). Usability Testing. Retrieved from <https://www.usability.gov/how-to-and-tools/methods/usability-testing.html>

User-Centered Design Process Map. (2013). Retrieved from <https://www.usability.gov/how-to-and-tools/resources/ucd-map.html>

User Research Basics. (2013). Retrieved from <https://www.usability.gov/what-and-why/user-research.html>

Sarma, R. (2012). RESEARCH: IN A METHODOLOGICAL FRAME. *Journal of Arts*, 3100(1). Retrieved from http://www.researchersworld.com/vol3/issue3/vol3_issue3_1/Paper_13.pdf

Rouse, M. (n.d.). What is SSADM (Structured Systems Analysis & Design Method) ? - Definition from WhatIs.com. Retrieved May 1, 2018, from <https://searchsoftwarequality.techtarget.com/definition/SSADM>

Wohlin, C., Host, M., & Henningsson, K. (n.d.). 3 Empirical Research Methods in Web and Software Engineering. Retrieved from <https://pdfs.semanticscholar.org/ad3a/417a546e714bec4a7ed176655e00ba715fe7.pdf>

Qualitative Research Methods: A Data Collector's Field Guide Qualitative Research Methods Overview Introduction to Qualitative Research. (n.d.). Retrieved from <https://course.ccs.neu.edu/is4800sp12/resources/qualmethods.pdf>

Wiles, R., Pain, H., & Crow, G. (2010). Innovation in qualitative research methods. Retrieved from http://eprints.ncrm.ac.uk/919/1/innovation_in_qualitative_research_methods.pdf

Research Onion - Explanation of the Concept. (2017). Retrieved May 1, 2018, from <https://www.ukessays.com/essays/psychology/explanation-of-the-concept-of-research-onion-psychology-essay.php#citethis>

Dillion, A. (1992). Reading from paper versus screens: a critical review of the empirical literature. *Ergonomics*, 35(10), 1297–1326. <https://doi.org/10.1080/00140139208967394>