Hugh Pearce Feeley

B00672792

Implementation of a Web Based Application that Alleviates Congestive Parking

COM547 – Project Plan

Table of Contents

[1.0 Introduction 3](#_Toc497077498)

[1.1 Purpose of Project 3](#_Toc497077499)

[1.2 Project Background 3](#_Toc497077500)

[1.3 Project Context 4](#_Toc497077501)

[2.0 Aims and Objectives 5](#_Toc497077502)

[2.1 Project Aim 5](#_Toc497077503)

[2.2 Project Stakeholders 5](#_Toc497077504)

[2.3 Project Scope 5](#_Toc497077505)

[2.4 Project Objectives 6](#_Toc497077506)

[2.5 Conclusion 6](#_Toc497077507)

[3.0 Project Deliverables 7](#_Toc497077508)

[3.1 Project Timeline 7](#_Toc497077509)

[Deadlines: 7](#_Toc497077510)

[3.2 Project Deliverables 7](#_Toc497077511)

[3.3 Resources Required 8](#_Toc497077512)

[4.0 Research 9](#_Toc497077513)

[4.1 Omniplex Cinema’s 9](#_Toc497077514)

[4.2 Park Mobile 9](#_Toc497077515)

[4.3 McCausland Car Park 10](#_Toc497077516)

[4.4 phpMyAdmin 10](#_Toc497077517)

[Advantages: 11](#_Toc497077518)

[Disadvantages: 11](#_Toc497077519)

[4.5 FireBase 11](#_Toc497077520)

[Advantages: 11](#_Toc497077521)

[Disadvantages: 12](#_Toc497077522)

[5.0 Software Development Lift Cycles 13](#_Toc497077523)

[5.1 Waterfall 13](#_Toc497077524)

[Advantages of Waterfall: 14](#_Toc497077525)

[Disadvantages of Waterfall: 14](#_Toc497077526)

[5.2 Agile 14](#_Toc497077527)

[Advantages of Agile: 14](#_Toc497077528)

[Disadvantages of Agile: 14](#_Toc497077529)

[5.3 Spiral 15](#_Toc497077530)

[Advantages of Spiral: 15](#_Toc497077531)

[Disadvantages of Spiral: 15](#_Toc497077532)

[5.4 Iterative 16](#_Toc497077533)

[Advantages of Iterative: 16](#_Toc497077534)

[Disadvantages of Iterative: 16](#_Toc497077535)

[5.6 Chosen Life Cycle 17](#_Toc497077536)

[5.7 Methods of Evaluation 17](#_Toc497077537)

[6.0 Methodology 18](#_Toc497077538)

[7.0 Risk Assessment 19](#_Toc497077539)

[8.0 Usability and Interaction 20](#_Toc497077540)

[9.0 Bibliography 21](#_Toc497077541)

1.0 Introduction

## 1.1 Purpose of Project

The purpose of this project is to create an online booking system for parking spaces in Belfast City Centre car parks whilst taking into consideration the project aims, objectives, timeline etc. in order to prevent time delays while trying to find a parking space.

## 1.2 Project Background

Belfast City Centre is a major city in Northern Ireland that attracts thousands of commuters a day. As a result of this there are many car parks located throughout the city. When in these car parks people can often spend a lot of time driving up and down until they find a space available. This is a lengthy task if the car park is busy, a problem, which could cause people to be late for work. Another problem with car parks is that in situations where they are unmanned an individual is required to have coins so they can pre pay for their parking before they can begin their stay.

There is currently no system in place that allows people to choose their specific parking space. There are systems that allows people to book their arrival and departure dates although sometimes the departure date may change suddenly.

The intended audience for this project are those that use car parks in Belfast city centre that have access to a mobile device or computer so that they are fit to reserve spaces. With the intention of widening the audience on a larger scale by expanding into other cities.

## 1.3 Project Context

The process of parking a vehicle in Belfast City Centre is an area that could do with a lot of work as there is room to make the task more efficient. This can be achieved by speeding up the process of parking your car and making payment easier as it is currently not the most efficient way. The plan for this project is to create a web application that allows users to pre book a specific space in a car park of their choosing. A QRCode will be given to each user on their booking which will allow them to scan into the car park. Alternatively, if the car park has a camera then there will be software in place that will read the registration of a person’s car. Confirming the person’s identity and granting them access to the space. The system will allow the user to view their parking history to keep track of the amount of money they spend on car parking. It will be maintained and managed by staff members of the car park that is currently in use. It will allow 24 hour access so that people can pre book their space at any time of the day so that it will allow for a quicker parking experience, reducing the risk of being late for work. Other features/advantages include the ability to pay digitally. Bank machines often only let users withdraw in multiples of £5, £10 or £20 meaning that someone will have to withdraw more cash than they need. Digital payment saves people money as they will have less lose change lying around.

2.0 Aims and Objectives

## 2.1 Project Aim

The aim of this project is to create a web based application that will include a system that allows users to book and pay for parking in Belfast City Centre. Improving on a system that has been in place for years. It will allow for quick and easy bookings, track a history of the cost of how much a user pays for parking as well as allowing admin users to generate reports easily, making the system leaner.

## 2.2 Project Stakeholders

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stakeholder** | **Contact Information** | **Impact** | **Influence** | **What is Important?** |
| Dr Daniel Guldenring | d.guldenring@ulster.ac.uk | Low | High | Communication with system developer |
| Hugh Pearce Feeley | [Feeley-h@ulster.ac.uk](mailto:Feeley-h@ulster.ac.uk) | High | High | Creating the system |
| Dr Kenneth Adamson | k.adamson@ulster.ac.uk | Low | Medium | Receiving a fully working final system |

## 2.3 Project Scope

When completed, the system will have a user friendly front end website supported by a JSON file database that people can use any time they like. Users will be able to login and register accounts on the system so that they can view parking history and a number of new parking sessions can be booked. This is a security measure that allows users information to remain private.

After selecting a car park and space users would be required to either show a QRCode or have their license plate read.

They will be charged on the length of time that they stay in the car park so a timer will begin whenever they enter the car park and end whenever they leave. Admin users of the system will be able to check the database for the amount of people using the system, most popular times and days of bookings. The database will hold information on each individual that has an account, recording things like name, car model, registration etc.

By using a NoSQL database relationships between data entities does not need to be defined; instead the database is designed on the queries that will be ran against it.

## 2.4 Project Objectives

The main project objectives are:

* Create user friendly front end website with consideration for HCI principles and usability.
* Identify and evaluate existing school car parks and payment options
* Create back end database to hold user information
* Streamline the car parking process through use of the website

## 2.5 Conclusion

By identifying the overall aims and objectives of the project it will help the system to achieve the overall aim and remain within scope. This will then boost the likelihood of staying within the time constraints and help to deliver the project successfully.

3.0 Project Deliverables

Project deliverables is a term for the services that will be provided on completion of a specific group of tasks. Deliverables are often specified characteristics or functions of a project. They are usually specified between stakeholders prior to the project commencing.

## 3.1 Project Timeline

The length of time it will take to plan, design, develop and distribute this project will take 2 full semesters to complete. The deadlines that must be kept are listed below.

### Deadlines:

* Project Plan - 29th October 2017, Semester 1
* ePortfolio - Week 11, Semester 1
* Project Monitoring & Control Report - Week 2, Semester 2
* Main Project Report and Oral - Week 11, Semester 2

## 3.2 Project Deliverables

In order for the project objectives to be delivered a number of different tasks will need to be completed. Appendix 1 is a Gantt chart that has been created to outline, plan, schedule and track the progress of processes and tasks. The project has been broken down into feasible tasks that have been scheduled respectively.

* Initiation
* Planning
* Requirements & Analysis
* Design
* Implementation
* Testing
* Distribution

## 3.3 Resources Required

The resources required to create and distribute this project are the following:

* Time
* Books, Magazines and Online Publications
* Self-taught resources
* Laptop/PC
* Coding framework
* Server to store data
* Camera (License Plate Reader)

All of these resources, albeit limited, will play a major role in the creation of this project.

4.0 Research

While carrying out research a number of systems were reviewed so as to give a better analysis of what would suit this project better.

## 4.1 Omniplex Cinema’s

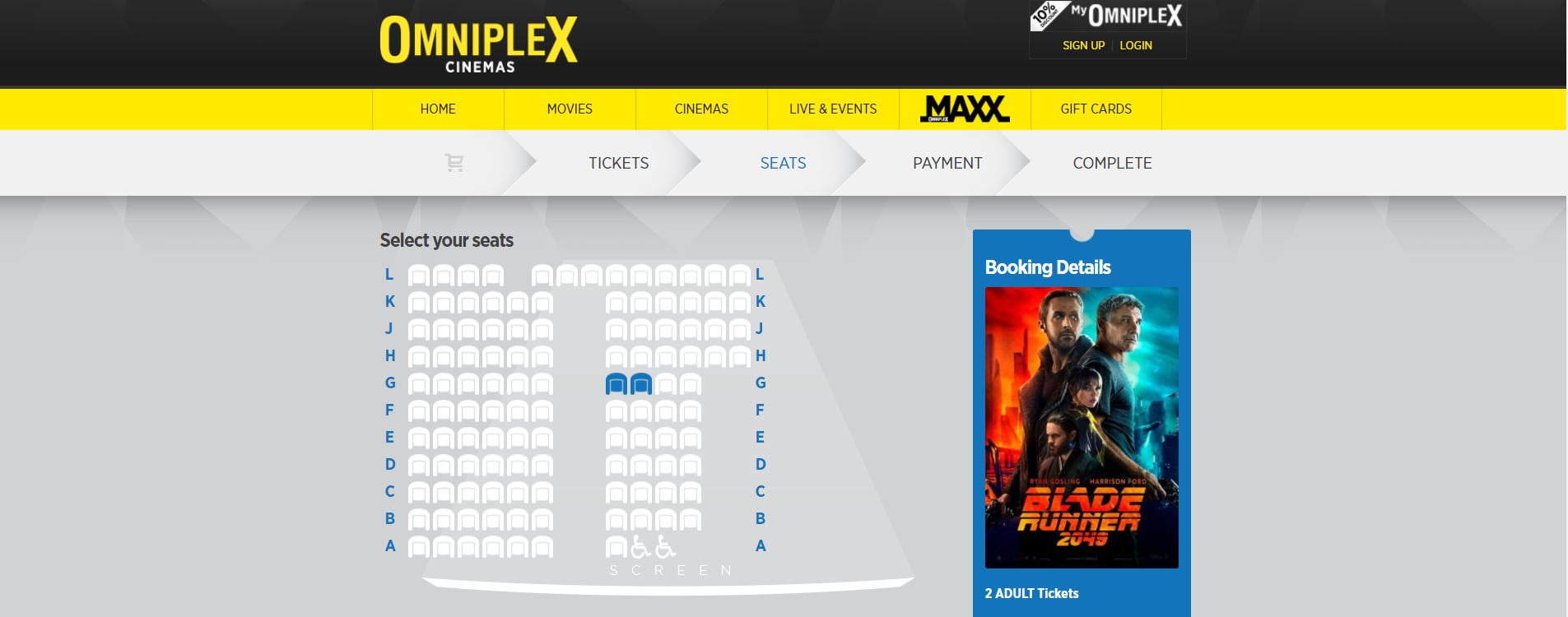


Figure 4.1 - (Omniplex, 2017)

Omniplex cinema was considered as it gives a different approach to how users book things. It allows them to select the specific seat that they wish to sit in. This would be a great feature when booking a car parking space as it would allow users to choose a space close to entrances/exits or one of their choosing.

## 4.2 Park Mobile

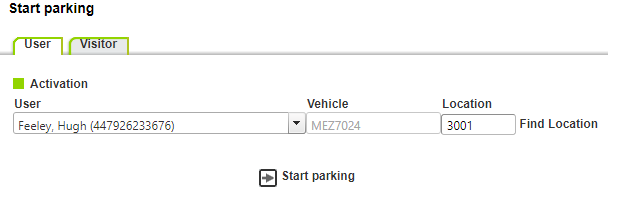


Figure 4.2.1 - (ParkMobile, n.d.)

Park mobile is another system that was researched. It allows users to pay for parking using their mobile devices while they are in the car park so that they do not have to use cash to pay for the parking space.



Figure 4.2.2 - (ParkMobile, n.d.)

With this project being a web based system it would be logical to incorporate a feature like this so that it gives users a real time booking system that can be used to pay for their space.

## 4.3 McCausland Car Park

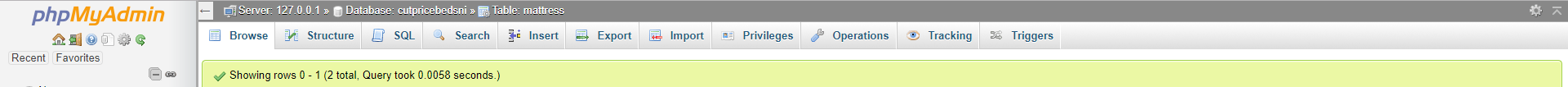


Figure 4.3 - (McCausland, 2017)

McCausland Car Park website was also researched as a part of the requirements phase. It was researched as it too is uses a booking system for parking. However it does not allow the user to book a specific parking space as previously suggested. It uses a simple system that allows the user to input their date of arrival and departure along with the time. This could be a useful tool to have included in the project.

## 4.4 phpMyAdmin

A system to store data for the project needs to be included for the project so one that was researched was phpMyAdmin.



### Advantages:

Figure 4.4 - (Xampp, 2017)

* High Speed
* Well Defined Standards Exist
* No Coding Required

### Disadvantages:

* Difficulty in Interfacing
* Partial Control

## 4.5 FireBase

Firebase is a Backend-as-a-Service that grew into a next-generation app-development platform on Google Cloud Platform.

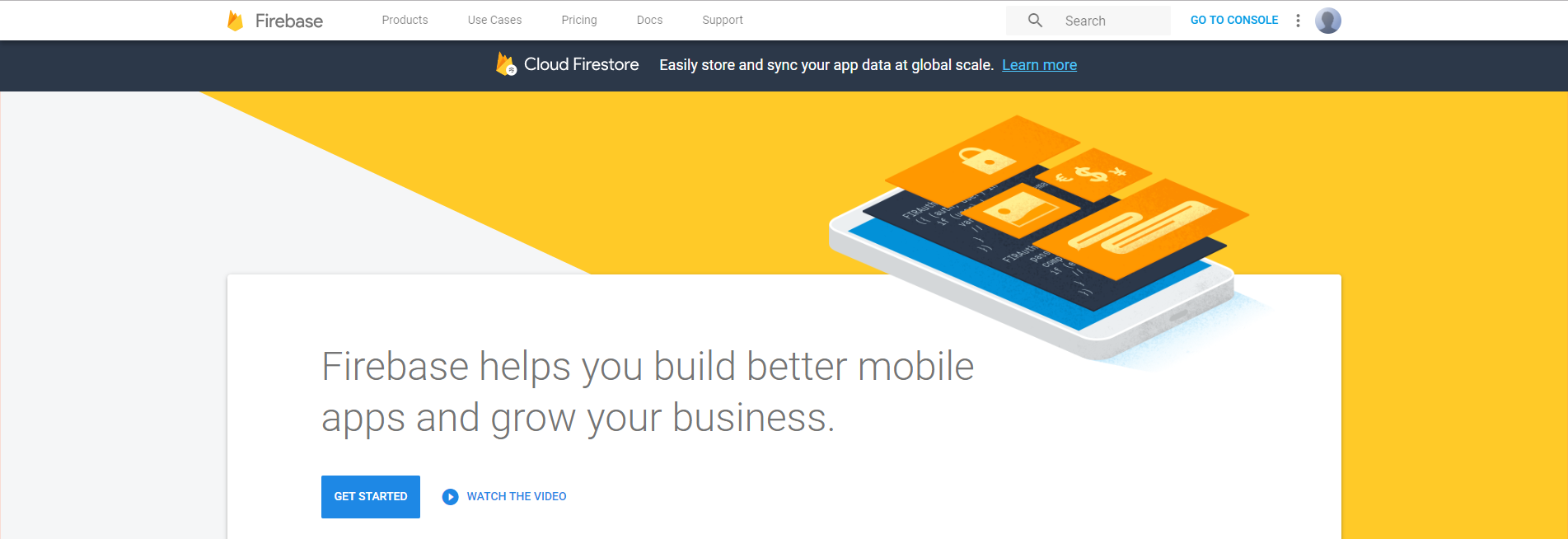


Figure 4.5 - (Firebase, 2017)

### Advantages:

* If your app is updated by a lot of users – It handles the Real-Time data updates between devices
* Stored in the cloud so readily available everywhere
* Cross Platform API
* Firebase hosts the data

### Disadvantages:

* Overkill if there is minimal users
* Difficult migration
* Reporting tools won't be anywhere near the ones of standard SQL
* Limited to 50 Connections and 100mb of Storage

5.0 Software Development Lift Cycles

The very first presentation of a software development life cycle model was put forward in 1956 by Herbert D. Benington. As systems were being developed and becoming more complex, coding along with debugging and testing were no longer efficient enough to continue. It didn’t make sense to have requirements defined from the start; instead the requirements had to be clarified first to give an outline of the project ahead. An example of a complex system during the 1950s was the Semi-Automatic Ground Environment (SAGE) project. “The work performed in SAGE asked for a systematic structure of the tasks involved, leading to the first explicit representation of a software development life cycle, as presented by Benington at a symposium on advanced programming methods” (Kneuper, 2017).

Benington defined the life cycle phases and their sequence, modelling the software development life cycle, whereas before his presentation they weren’t described but simply performed. Since Benington’s proposal, software development life cycle models have burgeoned.

## 5.1 Waterfall

The Waterfall model (shown in Figure 5.1) is a very linear and sequential life cycle model. It is considered a traditional method as it is one of the earliest SDLC approaches to be used in software development. In this model, a new phase cannot begin until the previous phase has been completed. Overlapping of phases is not allowed. There are 5 stages to the Waterfall model:

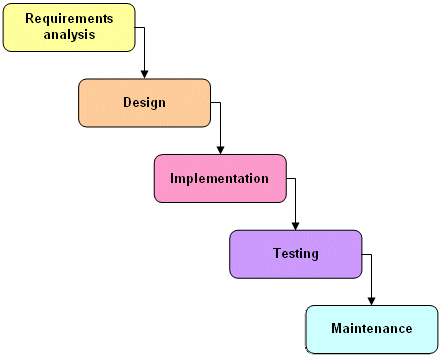


Figure 5.1 - Waterfall Model (TechnologyUK, 2017)

* Definition of Requirements
* System Design
* Implementation of code
* System Testing
* Maintenance

### Advantages of Waterfall:

* Easy to understand and manage.
* Step by step phases means no overlapping of tasks.
* Works well with smaller projects where requirements limited.

### Disadvantages of Waterfall:

* Poor model for long and ongoing projects.
* Software isn’t produced until late in the cycle.
* High risk and uncertainty.
* Difficult to go back and change something that wasn’t well planned out.

## 5.2 Agile

The Agile model focusses on process adaptability and customer satisfaction through frequent delivery of software. Agile breaks up tasks into smaller sections

### Advantages of Agile:

* Working software is delivered frequently.
* Face-to-face conversation is the best form of communication.
* Regular adaptation to changing circumstances.
* Late changes are welcomed.

### Disadvantages of Agile:

* Documentation gets side tracked.
* You need a long term vision for the product and actively work on communicating it
* Difficult to measure progress as it happens over multiple cycles.
* Customers can’t know how much the project will cost.

## 5.3 Spiral

The spiral model (Shown in Figure 5.3) is made up of four different phases. These phases are:

* Planning
* Risk Analysis
* Engineering
* Evaluation

The project passes through these four phases repeatedly in iterations, these iterations are known as Spirals. The baseline spiral, which starts at the beginning of the planning phase, requirements are gathered and the risks are assessed. Each spiral after this adds to the baseline.

### Advantages of Spiral:

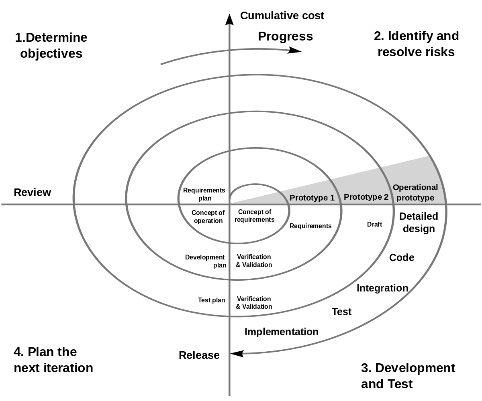


Figure 5.3 - (Wikipedia, n.d.)

* High amount of risk analysis
* Software is produced early in the software cycle.
* Strong approval and documentation control.
* Good for large mission-critical projects.

### Disadvantages of Spiral:

* Project’s success is highly dependent on the risk analysis phase
* Can be a costly model to use.
* Risk analysis requires highly specific expertise.

## 5.4 Iterative

Iterative process starts with implementation of a subset of requirements and iteratively enhances the evolving versions until the full system is complete. At each iteration new functional capabilities are added and design modifications are made. The idea behind this method is to enhance a system through iterative cycles and in smaller portions at a time (incremental).

### Advantages of Iterative:

* Progress can be measured.
* Less costly to change the scope/requirements.
* Testing and debugging during smaller iteration is easy.
* It supports changing requirements.

### Disadvantages of Iterative:

* More resources may be required.
* Although cost of change is lesser, but it is not very suitable for changing requirements.
* More management attention is required.
* System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.
* Defining increments may require definition of the complete system.

## 5.6 Chosen Life Cycle

After extensive research into development models it can be said that each model has its own benefits and hindrances. (Khari, et al., 2016) Say that “there are many SDLC models, which deals with the basic requirements to the final product. But, any software can be considered as a fail product, unless security is not properly used. Software is only secure only when it acts in a logical way in the existence of any malicious attacks”

Therefore, because of this, the chosen life cycle for this project is Waterfall method modified. This method is an extension of the Waterfall method. It uses the same phase process as the Waterfall model but is not subject to sticking to original requirements. This allows overlapping of phases when necessary.

Flexibility in requirements management including updating project requirements when needed is then allowed, and for prototyping, verification and validation to be carried out at each stage of the development process.

## 5.7 Methods of Evaluation

Evaluation will be carried out through frequent review sessions with the project supervisor and also a regular testing of software. Requirements will be reviewed and evaluated on a regular basis so that the project is at its best throughout the development.

6.0 Methodology

The project’s final version will be reliable, quick, easy to use and user friendly. It will be built in a stage of builds to make coding in the future, easier on an already working system.

A number of different technologies were taken into consideration, the following were:

* C#
* MVC
* ASP.NET
* PHP
* MySQL
* AngularJS
* CSS
* jQuery

In the end it was decided that the project will be coded in HTML using PHP while also including Javascript, CSS to create a designed user interface. FireBase, which helps you build better mobile apps, will be used to store the data. It is a NoSQL database meaning it reads data from JSON format only, making it easier to build on data without affecting other data’s information.

7.0 Risk Assessment

Risk assessment is an important step in being prepared for potential problems that is why the Health and Safety Executive (HSE) state that “A risk assessment is not about creating huge amounts of paperwork , but rather about identifying sensible measures to control the risks in your workplace.” (HSE, 2017).

There are limited risks when it comes to this project due to the nature of it, however if a risk that could possibly be dangerous is identified then a resolution to solve the problem should be developed.

The following represents some of the risks that have been identified and the resolution to take.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Risk** | **Consequence** | | **Likelihood** | | **Rating** |
| **Current** | **Target** | **Current** | **Target** | **Level of Risk** |
| 1 | Loss of IT (Data) | Major | Insignificant | Moderate | Unlikely | Extreme |
| 2 | Unavailable Resources | Major | Minor | Rare | Rare | Medium |
| 3 | Timeline identified Incorrectly | Major | Insignificant | Unlikely | Unlikely | High |
| 4 | Incorrect Requirements | Major | Minor | Moderate | Unlikely | Medium |
| 5 | Requirements Change | Moderate | Minor | Moderate | Moderate | Medium |
| 6 | Change in Circumstances | Moderate | Minor | Unlikely | Unlikely | Low |

8.0 Usability and Interaction

(Santoso, et al., 2016) States that “*The term of usability is one aspect of many that impact user experience. User experience may be described as the quality of interaction a participant has when interacting with a computer system. The interaction between a person and the computer system creates an experience called user experience.”* This gives a baseline to follow when creating the project. It is important to consider HCI and usability principles. Nielsen’s 10 Heuristics, Shneiderman’s 8 Golden Rules and Benyon and Turner’s 12 principles.

There are key principles, which will be taken into consideration during the design process are:

* Consistency
* Enable frequent shortcuts
* Feedback
* Design Dialogue
* Error Handling
* Easy Reversal
* Support
* Reduce Memory Load

9.0 Bibliography

Firebase, 2017. *Firebase.* [Online]   
Available at: https://firebase.google.com/  
[Accessed 24 10 2017].

HSE, 2017. *Risk Assessment.* [Online]   
Available at: http://www.hse.gov.uk/risk/controlling-risks.htm  
[Accessed 26 10 2017].

Khari, M., Vaishali & Kumar, P., 2016. *Embedding security in Software Development Life Cycle (SDLC).* s.l., s.n., pp. 2182 - 2186.

Kneuper, R., 2017. Sixty Years of Software Development Life Cycle Models. *IEEE Annals of the History of Computing,* 39(3), p. 43.

McCausland, 2017. *McCausland Airport Car Park.* [Online]   
Available at: https://www.mccausland.co.uk/  
[Accessed 23 10 2017].

Omniplex, 2017. *Omniplex Cinemas.* [Online]   
Available at: https://www.omniplex.ie/  
[Accessed 23 10 2017].

ParkMobile, n.d. *Park Mobile.* [Online]   
Available at: http://www.parkmobile.co.uk/  
[Accessed 23 10 2017].

Santoso, H. B., Fadhilah, S. & Nurrohmah, I., 2016. *The usability and user experience evaluation of web-based online self-monitoring tool: Case study human-computer interaction course.* Melaka, Malaysia, IEEE.

TechnologyUK, n.d. *The Watefall Model.* [Online]   
Available at: https://www.technologyuk.net/software-development/sad/waterfall-model.shtml  
[Accessed 19 10 2017].

Wikipedia, n.d. *Spiral model - Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Spiral\_model  
[Accessed 23 10 2017].

Xampp, 2017. *localhost:554.* [Online]   
Available at: http://localhost:554/phpmyadmin  
[Accessed 24 10 2017].