|  |  |  |
| --- | --- | --- |
| **New Zealand Diploma in Information Systems** | | |
| **Course No: 5607** | **IS Application Project** | **Level: 5**  **Credits: 30** |

|  |  |
| --- | --- |
| **Student Name: Sharish Bala** | **Student ID: 1458663** |
| **Assessment Type: Portfolio** | **Weighting: 100%** |
| **Due Date: 25/11/2021** | **Total Marks: 500 marks** |

|  |  |
| --- | --- |
| **Student declaration**  I confirm that:  • This is an original assessment and is entirely my own work.  • The work I am submitting for this assessment is free of plagiarism. I have read and  understood the [Academic Integrity Procedure](https://thenest.unitec.ac.nz/TheNestWP/wp-content/uploads/2019/05/AC-2.8-Academic-Integrity-Procedure.pdf) here. I have also read and understood  the [Student Disciplinary Statue](https://thenest.unitec.ac.nz/TheNestWP/wp-content/uploads/2020/07/Student-Disciplinary-Statute-FINAL-Feb-2020.pdf) here.   * Where I have used ideas, tables, diagrams etc. of other writers, I have acknowledged the source in every case. | |
| **Text  Description automatically generatedStudent Signature:** | **Date:25/11/2021** |

**New Zealand Diploma in Information Systems**

**HTCS5607 IS Application Project**

**TECHNICAL REPORT TEMPLATE**

**Project Name: NZ Wetlands**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Team Name(s)** | **Student ID** | **Email** | **Phone** |
| ***Project Manager***  Sharish Bala | 1458663 | sharishbala@hotmail.com |  |
| ***Deputy Project Manager***  Sharish Bala | 1458663 | sharishbala@hotmail.com |  |
| ***Team Members***  Sharish Bala | 1458663 | sharishbala@hotmail.com |  |

**Client Stakeholders**

|  |  |  |
| --- | --- | --- |
| **Client Stakeholders** | **Full name and title** | **Contact details** |
| **Project Sponsor(s)** | *Lei Song, Simon Dacey* | [lsong@unitec.ac.nz](mailto:lsong@unitec.ac.nz)  [sdacey@unitec.ac.nz](mailto:sdacey@unitec.ac.nz) |

**DATE OF SUBMISSION**

25/11/2021

Table of Contents

[1. Document Control 3](#_Toc88600163)

[1.1 Version History 3](#_Toc88600164)

[1.2 Contribution to Report sections 3](#_Toc88600165)

[1.3 Glossary 3](#_Toc88600166)

[2. Executive Summary 4](#_Toc88600167)

[3. Introduction 5](#_Toc88600168)

[4. Technology Review 6](#_Toc88600169)

[5. IT Methodology 8](#_Toc88600170)

[6. Project Management 9](#_Toc88600171)

[6.1 Project Management Narrative 9](#_Toc88600172)

[6.2 Project Plan with Milestones 9](#_Toc88600173)

[6.3 Project Governance Responsibilities 9](#_Toc88600174)

[6.4 Project Meetings 10](#_Toc88600175)

[6.5 Project Reports 10](#_Toc88600176)

[6.6 Project Risk and Issue Analysis 10](#_Toc88600177)

[7. Requirements Analysis 11](#_Toc88600178)

[7.1 Introduction 11](#_Toc88600179)

[7.2 Use Case Diagram 11](#_Toc88600180)

[7.3 Business Use Case Narratives (Descriptions) 12](#_Toc88600181)

[7.4 Activity Diagrams 12](#_Toc88600182)

[7.5 Overall Class Diagram 12](#_Toc88600183)

[8. Project Design 13](#_Toc88600184)

[8.1 Introduction 13](#_Toc88600185)

[8.2 Software List 13](#_Toc88600186)

[8.3 Version Control Software 13](#_Toc88600187)

[8.4 Design Use Case Narratives (Descriptions) 13](#_Toc88600188)

[8.5 Sequence Diagrams 13](#_Toc88600189)

[8.6 Deployment Diagram 14](#_Toc88600190)

[8.7 Database Design 14](#_Toc88600191)

[8.8 Annotated User Interface Designs 14](#_Toc88600192)

[8.9 Test Plan 14](#_Toc88600193)

[9. Project Training 15](#_Toc88600194)

[9.1 End User Background and Training Objectives 15](#_Toc88600195)

[9.2 Training Materials 15](#_Toc88600196)

[9.3 Training Deliverables 15](#_Toc88600197)

[10. Conclusion & Lessons Learned 16](#_Toc88600198)

[References 17](#_Toc88600199)

[Appendices 18](#_Toc88600200)

# 1. Document Control

## 1.1 Version History

This document has had the following revisions:

| **Version** | **Date** | **Author** | **Description of Change** |
| --- | --- | --- | --- |
| 0.1 | 23/11/2021 | Sharish Bala | Initial draft |

## 1.2 Contribution to Report sections

| **Project Team Member name** | **Student ID** | **Report Section** |
| --- | --- | --- |
| Sharish Bala | 1458663 | All Sections |
|  |  |  |
|  |  |  |
|  |  |  |

## 1.3 Glossary

To provide clarity, terms and acronyms used in this document are defined as follows:

| **Term / Abbreviation** | **Definition** |
| --- | --- |
| Supervisor | Technical Advisor |
|  |  |

# 2. Executive Summary

This report is for the development of management software being developed for NZ Wetland, this report contains the reasoning for the development and reasoning for what is to be used in the development of the product for NZ Wetlands in sections 3,4 and 5; the 6th section explains the project management, and planning of how the project is to be carried out; the 7th section is the analysis of the requirements given by NZ Wetlands to show what is need for the project to be developed; the 8th section takes the analysis and converts them into designs which will be developed and become the product; the 9th section is where the training material for the product is contained; the last section contains the conclusion if the project and what can be learned from having carried out this project.

# 3. Introduction

This is a technical report for the management software being developed for NZ Wetland. This report will cover the research, analysis, planning, design, creation and deployment of the application. The report will also serve as documentation of the development, the reasoning for decisions made for the application during it development life cycle. And the report will outline if the product is a success or failure and possible improvement on the development of the product. The actual development and testing of the application will not be included in this report.

# 4. Technology Review

The objective of this technological review is to examine and identify technical aspects of developing an application for operations of NZ Wetlands. The type of application needed for NZ Wetlands is management software which functions as a human, land and wildlife management system.

The core element for the project is the coding language which will act as the key framework for the application development. For the application the coding language which would need to be able to operate and function with the ability to interact with and use a database, the current coding languages that can do this are many but the most common are C#, Python and JavaScript. C# is a general-purpose programming language developed by Microsoft; it is an object-oriented language which is primarily used on the Windows .NET framework although it can also be used with an open-source platform, because C# is such an integrated coding language with .NET Widows it can cause issues functioning on more open-sourced platforms. Python is a general-purpose object-oriented programming language, which is an easy to learn and uses English keywords, and fewer syntactical constructs than other languages; python is also a highly versatile language which can be used in a more open operating environment, the negatives of python are that the coding language are that it us an inefficient coding language and it has issues with database access. JavaScript is a coding language based on Java, it was created to be a lighted version of Java for frontend developers; the JavaScript of today is a more versatile language which is compatible with other coding languages and because it was Java based and is compatible with other languages it is one of the most commonly used languages available; the negatives of JavaScript is that it may be difficult to develop larger applications and due to the code being visible to everyone can cause security issues. After considering the advantages and disadvantages to the requirements of the application the system does not need an overly complicated and overly efficient application, but it also does not need a very inefficient application; out of the 3 coding languages, Python due to its inefficient coding language it can be ruled out, and due to the complex development if JavaScript, in the timeframe it would be too difficult to develop leaving C# as the chosen coding language.

To develop the application there are many methods available to code, the most useful being an integrated development environment (IDE), the IDE is useful for coding, debugging and automation which saves time and money for development. There are many IDEs to choose from which can be used to develop the application but the most useful IDE for this development must be usable with the chosen coding language of C#, three common IDEs for C# are Visual Studio, MonoDevelop and VS code (Visual Studio code). Visual Studio is an IDE developed by Microsoft it is mainly used for C, C++, .NET and C#, but can support other languages like JavaScript, the downside of Visual Studio is that it is a very resource-intensive program. MonoDevelop is an IDE developed by Xamarin it supports cross-platform development and it can be used desktop and web applications on Windows, Linux, and Mac OS X, it can develop C# and F# programs but it can have some autocompletion and code formatting issues.

VS code (Visual Studio code) is another IDE developed by Microsoft, but it is not Visual Studio since Visual Studio is mainly aimed at .NET development. VS code is a lighter and more customizable version of Visual Studio which is less resource-intensive, VS code is very compatible with the development of C# applications; the downside of VS code is that due to being a lighter version of Visual Studio it is lacking in some of the more advanced debugging features. All the IDEs considered have the ability to work with C# but considering the application needing development and the resources required, Visual Studio is the most suitable IDE to be use for the application development using C#.

The application will also require a Database to sort the data for the application, this database will need to be JavaScript compatible. The most common databases that work with JavaScript are MySQL, Microsoft Access and SQLite. MYSQL is one of the most popular database management system available, it can be used for both small and large projects; but the downside to MYSQL is that it can be fairly slow to write data. Microsoft Access is still a common program for databases but little future proofing and due to having a file system which isn’t compatible with other system limits its uses. SQLite is a lighter version of MQSQL which is more suited for smaller applications, and due to being lighter than MYSQL is has a better performance on smaller scale projects. Considering the scale of the project and requirements of the application SQLite is the most suitable database for this project. And to access the actual SQLite database SQLite Studio is used.

For comparison to other management systems available, many such systems are made for more basic application an example being Reflex ERP, a management program specializing in land management. The application which is to be developed would meet the requirements and have a more customized system to function as is required for the application.

# 5. IT Methodology

For the Methodology used for the Systems Development Life Cycle (SDLC) of the application, agile and waterfall can both be used but for the chosen methodology waterfall is being used.

Agile does have a lot of good uses for the development of the application it requires a longer and more complex development structure would be needed for the development of the application but in the case for waterfall it is used due to the requirements having been out lined and a clear goal of the development of the application being given by the sponsors, allowing for a quicker and more efficient use of resources for the development of the application.

The waterfall SDLC is Project Initialization/Planning, getting the required information and planning how the development life cycle will happen; Analysis, taking the requirements and analysing them to figure out what us needed for the development of the application; Design, taking the analysed information and producing a structure which will be developed based on the analysed information; Development, the development of the application based on the designed structure and testing of the application; Deployment/Maintenance, deploying the application and creating training material based on the completed application to train the possible users of the application.

# 6. Project Management

## 6.1 Project Management Narrative

SDLC

1. Project Initialization/Planning: Technology review, tool selection, version control, risk management and Gantt chart all form the Project Initialization/Planning of this project. Version control GitHub contains a development history: <https://github.com/Hot-Choc/HTCS6507-IS-Project>
2. Analysis: Business use case narratives, Activity diagrams, use case diagram and an overall class diagram form the Analysis of this project.
3. Design: Design-Level Use Case Narrative, Sequence Diagrams, Database Design, Annotated Interface Designs, Test Plans and Deployment Plan form Design of this project.
4. Development: Creating the Database, Create Use Case Interfaces, Code & Test for Use Cases and Testing the program form the Development of this project.
5. Deployment/Maintenance: Deploy Program, Create Training Material and the Technical Report form the Deployment/Maintenance of this project.

## 6.2 Project Plan with Milestones

The overall plan for the development of the application required by NZ Wetlands is to gather information of the project and its requirements, initialize the project by creating a Gantt chart and getting the tools for development, analyse the information provided to create a plan for what needs to be developed, design the structure of the application and how it will function, develop and test the code for the application, and lastly deploy and finish the project.

## 6.3 Project Governance Responsibilities

The person responsible for the project management and quality assurance is the project manager, and the way these tasks were carried out was by reviewing the produced materials and comparing that what is required, and then making changes to bring it to the level required.

## 6.4 Project Meetings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Meeting | Date | Duration | Participants | Type |
| 1/ Week 2 | 29/09/2021 | 15min | Dacey, Simon  Bala, Sharish | Project Status review |
| 2/ Week 3 | 20/10/2021 | 15min | Lei Song  Bala, Sharish | Project Status review |
| 3/ Week 4 | 28/10/10 | 15min | Dacey, Simon  Bala, Sharish | Project Status review |
| 4/ Week 5 | 04/11/2021 | 15min | Lei Song  Bala, Sharish | Project Status review |
| 5/ Week 6 | 10/11/2021 | 15min | Dacey, Simon  Bala, Sharish | Project Status review |
| 6/ Week 7 | 17/11/2021 | 15min | Lei Song  Bala, Sharish | Project Status review |

## 6.5 Project Reports

The project status reports, report what has been done and what is to be done, and how the current work done meets the requirements set for completion and quality at the time which the project status reports is filled.

Project status report 1/ Week 3 Date:25/10/2021

Project status report 2/ Week 5 Date:05/11/2021

Project status report 3/ Week 7 Date:19/11/2021

## 6.6 Project Risk and Issue Analysis

The Risk and Issue analysis is to define possible risks to the project and outline possible measures to minimise and deal with those risk and issues.

# 7. Requirements Analysis

## 7.1 Introduction

The Requirements Analysis is to analysis the information given from NZ Wetlands and to figure out what is needed for the project and how they would like the project to function using diagrams and use case narratives

## 7.2 Use Case Diagram

A diagram to show who will use a given use case as described in the requirements given by NZ Wetlands.

Diagram

Description automatically generated

## 7.3 Business Use Case Narratives (Descriptions)

A step-by-step description of each use case based on how the use case will be used by the end user.



## 7.4 Activity Diagrams

A step-by-step diagram of each use case based on how the use case will be used by the end user.



## 7.5 Overall Class Diagram

A diagram of how the database will function based on the requirements.

Diagram

Description automatically generated

# 8. Project Design

## 8.1 Introduction

The Project design is to take the analysed information and convert them into a development structure that can then be used to develop the project and outline the final look and functions of the final product.

## 8.2 Software List

* Visual Studio: The Integrated Development Environment (IDE), highly suited for development of C# based programs.
* SQLite: The Database, light weight, easy to use and can be used with Visual Studio and C#.
* SQLite Studio: The graphics user interface (GUI) for managing SQLite databases, easy to use and good functionality with SQLite.

## 8.3 Version Control Software

* GitHub: A industry standard and effective use of GIT version management system. Simple to use and an effective way to manage version and maintain a backup of the project material.

## 8.4 Design Use Case Narratives (Descriptions)

A step-by-step description of each use case showing the interactions of the user and system and detailing each function as it would occur in a use case. The use cases were changed to remove the interface needing to prompt for the user to ‘add/update/delete/assign or remove another?’, the reason being that the user can do ‘another’ directly by adding/updating/deleting/assigning or removing in the shown interface.



## 8.5 Sequence Diagrams

A step-by-step diagram of each use case showing the interactions of the user and system and detailing each function as it would occur in a use case.



## 8.6 Deployment Diagram

The Deployment diagram show the method in which the product will be deploy for use.

Diagram

Description automatically generated

## 8.7 Database Design

The Database design is the structure of the database which will be used for the product and how that database with function.

 

## 8.8 Annotated User Interface Designs

The Annotated User Interface Designs are how the final products interface will look and how each part of the interfaces will function once coded.



## 8.9 Test Plan

The Test Plan is the testing criteria for the database and interface which will prove that they are functioning as intended with no major errors or missing functions.

 

# 9. Project Training

## 9.1 End User Background and Training Objectives

The end user for the provided use case is a site admin, there roll is to manages and adjust the data in the database, to do this they need to have access to the product and have adequate knowledge on how to use the product features and functions.

## 9.2 Training Materials

The training materials to help the end user develop the skills and knowledge to use the product with sufficient aptitude is done through a training video which show how the interfaces of the product function and what can be done using the ionterfaces.

## 9.3 Training Deliverables

## 

# 10. Conclusion & Lessons Learned

The outcome of the project can be considered a success, the reason being that a completed product has been delivered at the end of the project and the product functions as dictated in the use cases. The project also has a traceable development cycle to show how it was developed and how it met the requirements set by NZ wetlands.

During the development of the project C# was used to make the interfaces and code this allowed for a greater understanding of C# coding and how to use it effectively with SQLite. SQLite was used for the first time and by using it with SQLite Studio it provided a simple and easy to use database which could be used with C#. if I were to carry out this project again, I would continue to use SQLite, but I would try JavaScript or Python, the reason being that they are more light weight then C# but to use them I would need to learn more of how they function and how they would work with an SQLite database.

Another possible change that I would make is to the methodology use changing it from waterfall to agile, the reason being that waterfall does not allow for flexibility to make changes to the project, this results in a more rigid development structure and making decisions which can result in an inefficient product or a product with unnecessary elements in it.

# References

* Everything you need to know about C# (2019, May 15). Pluralsight [Article]; <https://www.pluralsight.com/blog/software-development/everything-you-need-to-know-about-c->
* Lysis (2021, January 11). Pros and Cons of Using C# as Your Backend Programming Language; Agilites [Article]; <https://www.agilites.com/pros-and-cons-of-using-c-as-your-backend-programming-language.html>
* What Is Python Used For? A Beginner’s Guide (2021, September 23). Coursera [Article]; <https://www.coursera.org/articles/what-is-python-used-for-a-beginners-guide-to-using-python>
* Python Tutorial (n.d.). tutorialspoint, retrieved 28 september,2021; <https://www.tutorialspoint.com/python/index.htm>
* Python Advantages and Disadvantages – Step in the right direction (n.d.). techvidvan, retrieved 28 September,2021; <https://techvidvan.com/tutorials/python-advantages-and-disadvantages/>
* Dionne, M. (2019, April 12). Reasons Why JavaScript is Omnipresent in Modern Development, snipcart; <https://snipcart.com/blog/why-javascript-benefits>
* abhishek0719kadiyan. (2021, August 9). Advantages and Disadvantages of JavaScript, Geeksforgeeks; <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-javascript/>
* Gupta, A. (2019, November 27). Advantages and Disadvantages of JavaScript, tutorialspoint; <https://www.tutorialspoint.com/advantages-and-disadvantages-of-javascript>
* Cooper, S. (2019, October 28). Microsoft Access: Is it still relevant?, Comparitech; <https://www.comparitech.com/net-admin/microsoft-access/>
* Kushnir, A. (2021, July 29). What Is the Best Database for Node.js?, bambooagile; <https://bambooagile.eu/insights/the-best-database-for-node-js/#sql-databases-for-nodejs-comparison>
* Reflex ERP (n.d.). capterra, retrieved 28 september,2021; <https://www.capterra.co.nz/software/149997/reflex-erp-software>
* MonoDevelop Review (n.d.). slant, retrieved 28 september,2021; <https://www.slant.co/options/12265/~monodevelop-review>
* HERRERA, C. (2021, June 11). The Ultimate List of C# Tools: IDEs, Profilers, Automation Tools, and More, stackify; <https://stackify.com/best-csharp-tools/>

# Appendices

Project Plan with Milestones – Gantt Chart



Project Meetings – Meeting minutes

     

Project Reports – Project Status Reports

  

Project Risk and Issue Analysis - Project Risks & Issues Register

