­­­



*Capable People, Capable Communities*

**Micro-Credentials in Software Development**

**Certificate in Software Development**

**MCSD51**

**Project Report**

**Zhengwen (Steven) Lei**

**Xian (Richard) Zhang**

**Maribeth Estimos**

**September 2022**

Contents

[**Introduction** 3](#_Toc113889345)

[**Initial Proposal** 3](#_Toc113889346)

[**Variations From The Initial Proposal** 3](#_Toc113889347)

[**Tools and Technologies Used (in alphabetical order)** 3](#_Toc113889348)

[**SDLC** 4](#_Toc113889349)

[**Project timeline** 4](#_Toc113889350)

[**Ethical and cultural impact** 4](#_Toc113889351)

[**Database** 4](#_Toc113889352)

[**Selected Screenshots of the completed App** 5](#_Toc113889353)

[**Reflection** 7](#_Toc113889354)

Executive Summary

In this project, we developed a web application to generate password. We used basic HTML, CSS, Embedded JavaScript, Node.JS and JavaScript to develop this app. This is a self-sponsored, individual project developed in an iterative manner. We created theUse Cases and User Stories based on our personal knowledge and experience related to water safety.

# Introduction

Why we need a password generator.

The number of people who get into trouble, and even lose their lives, in water in Aotearoa New Zealand is quite high as a proportion of the population. We have a very long coastline and treacherous rip currents at many beaches. Not many people, especially recent migrants, are not aware of the dangers of rip currents.

I have personally seen many rescues and even helped people get to safety. I am aware that there is a national website (<https://watersafety.org.nz/>) that helps promote water safety in New Zealand but not many people seem to be aware of its existence or use it to educate themselves.

# Initial Proposal

[Type the document title]

[Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.]

Kathiravelu Ganeshan

**Preface**

It is a great opportunity for us to have the Certificate in Introductory Software Development Level 5 (MCSD51) with Future Skills Academy.

In the accomplishment of this course we are submitting a system proposal on “Password Generator”. Subject to the limitation of time, every possible attempt has been made to study and discussed the project deeply.

The whole project is measured through questionnaire, data and other vital information that were further analyzed and interpreted and the result was obtained.

The whole project has been divided into 2 parts:

**Part1: System Proposal**

‣ Introduction

‣ User Stories and use cases

‣ Ethical and cultural impact

‣ Project Timeline

‣ Diagram

**Part 2: Product Implementation**

**Contents**

Preface …………………………………………………………………………………………….……2

Introduction ……………………………………………………………………………………….…... 4

User Stories and use cases…………………………………………………………………………..… 6

Ethical and cultural impact ………………………………………..………………………...………... 8

Project Timeline…………………………………………………………....……………….…..…….. 9

Diagram…………………………………………….…………………………….…………….…….. 11

References …………………………………………………………………………………………... 12

**Password Generator**

**Introduction**

The most prevalent way hackers can break through computers is by guessing your password. By using simple and common passwords creates an opportunity for hackers to get access to your electronic devices. Since no one would wish to have their personal information stolen, it is just ideal to use a strong passwords to protect your information.

Many people think good password management is all about creating difficult passwords from password generator Google. However, this is not the case. Creating a password is just the start of the management process. For instance, it is advisable to change passwords every three months. Due to the many passwords, you may need to record them on a notebook that you will carry everywhere you go.

With manual management, you might end up being mixed up by your passwords. At times, you may have to create own manual passwords that might be weak and predictable. In case the notebook is lost, all the credentials will get lost. To avoid these challenges, get a password generator that will help safeguard and manage your passwords use.

**What is a Password Generator?**

A password generator is a software tool that creates random or customized passwords for users. It helps users create a stronger password that provide greater security for a given type of access. Some password generators are simply random password generators. These programs produce complex/strong passwords with combinations of numbers, uppercase and lowercase letters, and special characters such as braces, asterisks, slashes, etc.

Other types of password generators are made to generate more recognizable passwords rather than a completely random set of characters. There are tools for generating pronounceable passwords, as well as custom tools that allow users to set detailed criteria. For instance, a user could set a request for a certain number of characters, a certain mix of letters and numbers, a certain number of special characters, or any other criteria for generating a new password.

**Objective**

Password generators help those who have to constantly come up with new passwords to ensure authorized access for programs and to manage a large number of passwords for identity and access management. Other kinds of tools include a password vault, where users manage large numbers of passwords in a secure location.

Passwords generator software offers a good display of all your credentials. As such, it saves you from memorizing hundreds of passwords except the generator's login details. The primary objective of using a password generator is to create a strong and unpredictable password for all of your accounts.

*(Ref:https://www.techopedia.com)*

**User Stories and Use Cases**

What is a User Story?

A simple way to define requirements from the view point of a user

As a (USER – WHO), I want to (The WHAT to Achieve), so that I (The WHY)

**User Stories in the perspective of the Password Generator User**

|  |  |
| --- | --- |
| **Story ID** | **User Story** |
| 1 | As a user, I want to be able to create a password that will require sets of numbers, letters and symbols so that I am assured that my password is secure. |
| 2 | As a user, I want to be able to get notify whenever I input the wrong parameter so that will give me the awareness. |
| 3 | As a user, I want to be able to create a password with dynamic length so that will make the password even stronger. |
| 4 | As a user, I want to be able to create a password that will be able to paste to another application so that will be less work and less worry for me. |
| 5 | As a user, I want to be able to save my settings so that I can use it on my next time. |

**Using Use Case to define requirements**

‣ UML (Unified Modelling Language) Use Case Diagram

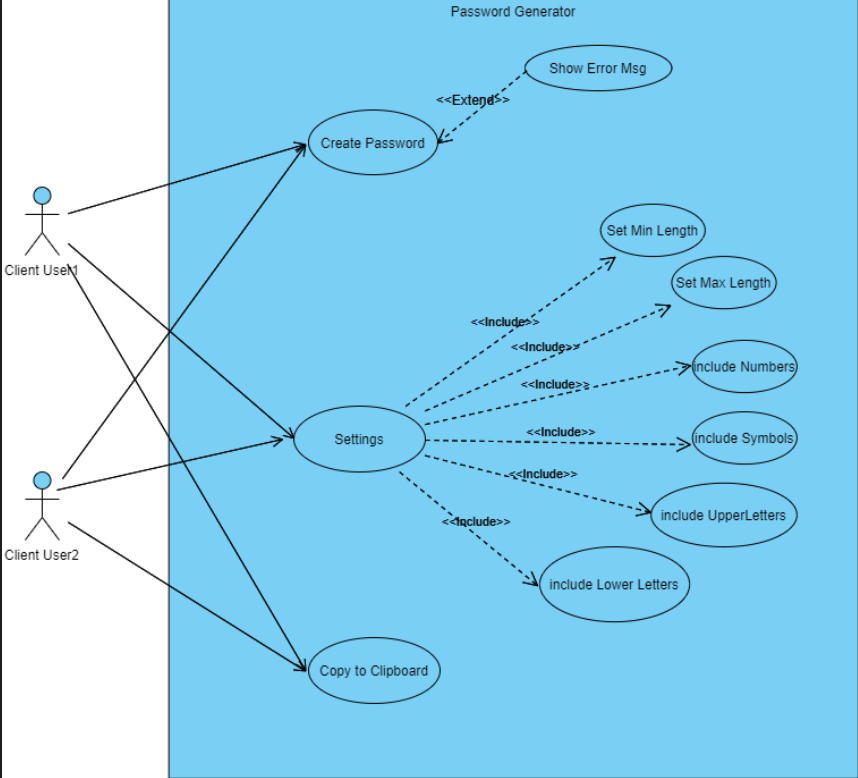
‣ Use Case Diagram is used to:

• Define and organizing functional requirements

• Define how user interact with the functions of the systems

• Model the basic flow of events in a use case

**USE CASE DIAGRAM OF PASSWORD GENERATOR**

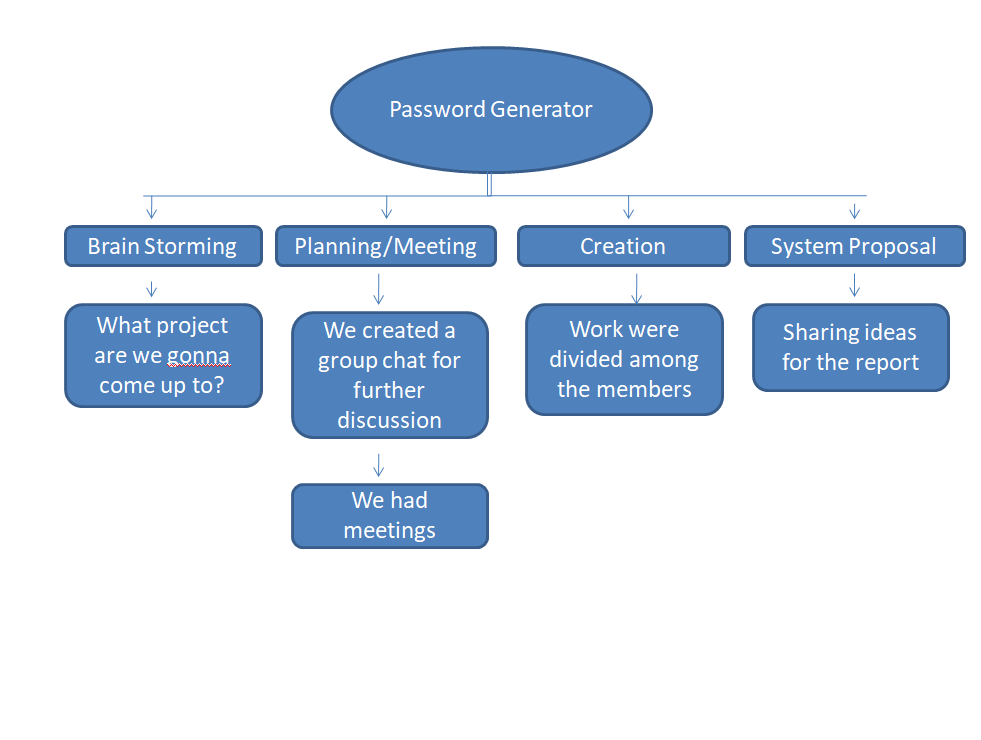
****

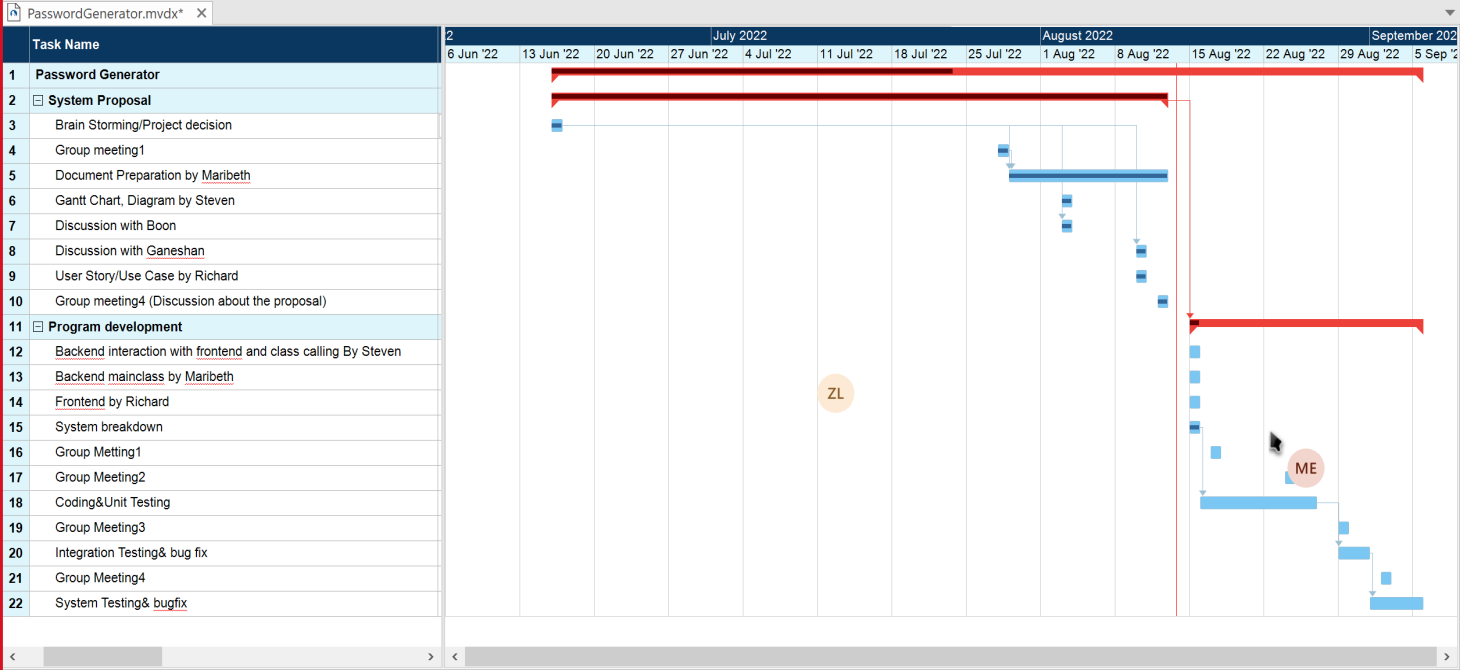
**Ethical and Cultural Impact**

To our knowledge, we are aware that there are lots of ways how these different groups of individual in different part of the world, may be able to get an access in one’s information and might have a negative effect to one’s information security.

Hence, as a software developer we know that there is a pressing need to improve the defenses. It is essential that we further enhance the security of our target users by understanding and analyzing how each user choose their passwords as this will give an important insights and will aid us on how to enhance the security of those users against password compromise attacks.

Having these to say, we all agreed to follow the best practices with regards to storing the information in an encrypted format and keep on working out how else we can improve the system.

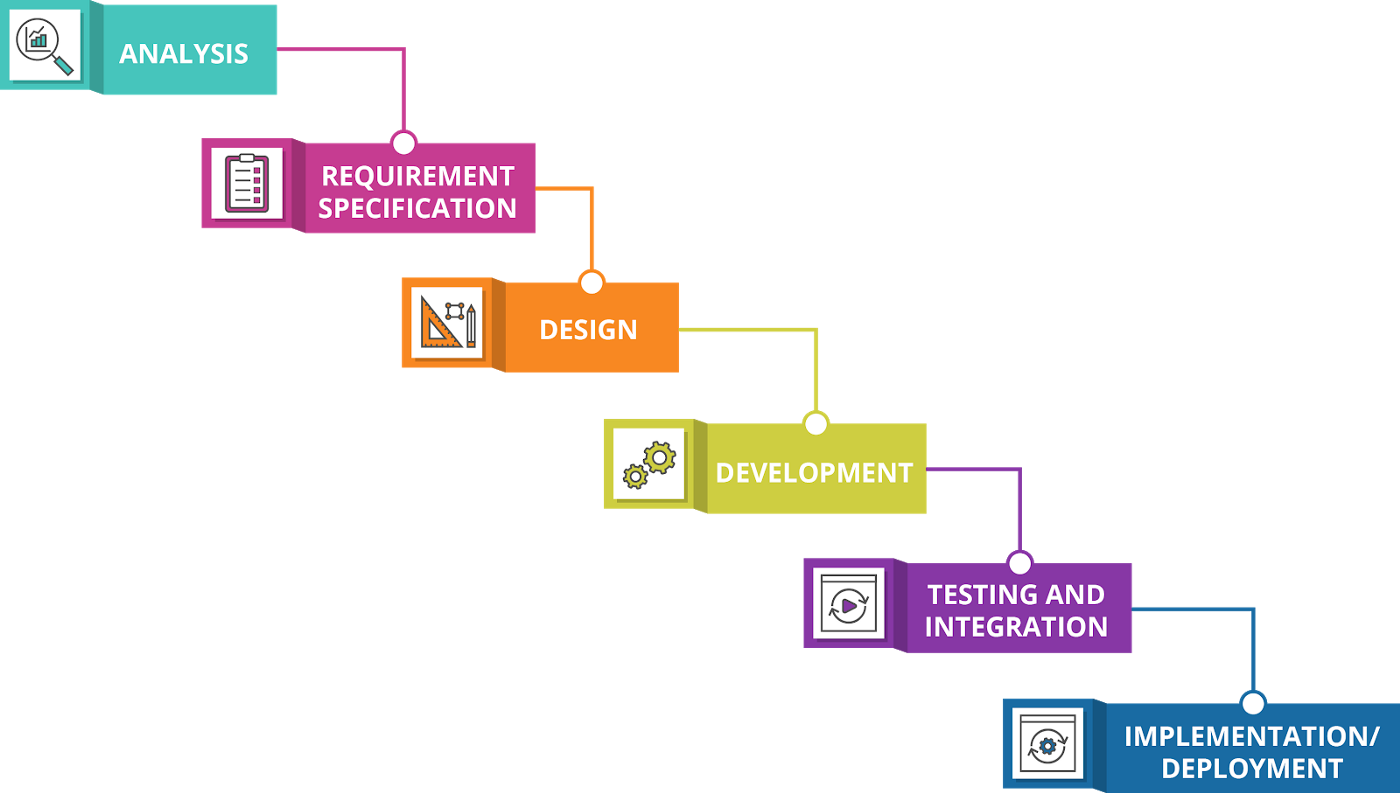
**Project Timeline**

**Gantt chart**

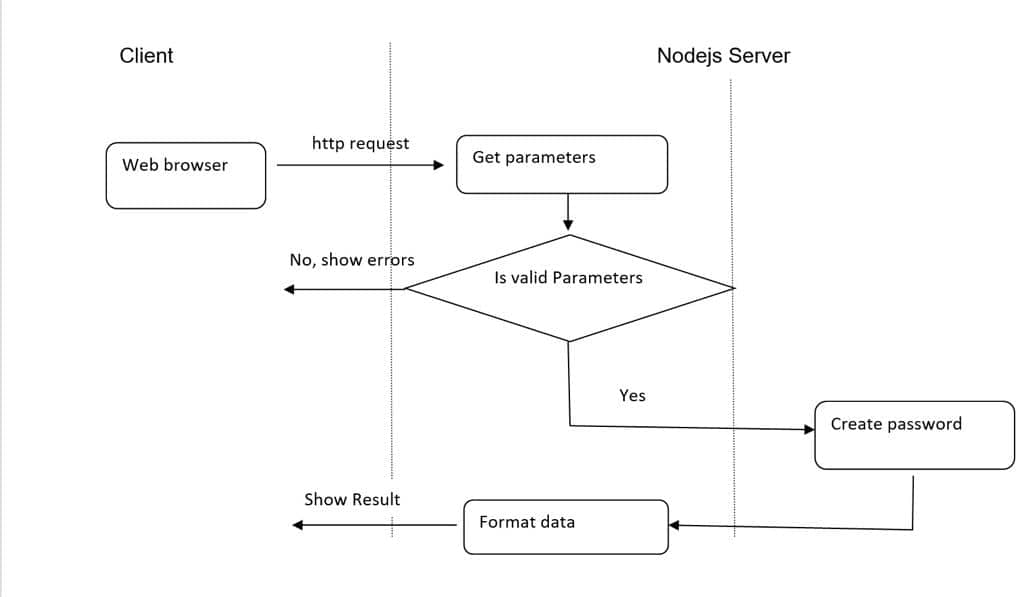
**About the project management methodology**

We will use the waterfall methodology to develop this software because the requirements of the project are explicit. This is a relatively simple system used to generate a password under some conditions. After we discussed the objective with the teacher, we know this software will only be used in certain circumstances when people want to create a random password. And this is a final project in this course, so it is unlikely to be changed in the future. Due to those reasons, we think the waterfall is an appropriate methodology for this project.*(Zhengwen Lei)*

**Table below depicts waterfall methodology**



**Diagram**



**References**

What is a Password Generator? [*https://www.techopedia.com/definition/31414/password-generator*](https://www.techopedia.com/definition/31414/password-generator)

Why should I use a Password Generator? *https://teampassword.com/blog/why-should-i-use-a-password-generator*

User Stories and Use Cases *SDLC – Define Phase: Use Cases 13th June 2022 MCSD51: Prepared by Boon*

Ethical and Cultural Impact[*https://arxiv.org/pdf/1712.08940.pdf%22%20%5Ct%20%22\_blank*](https://arxiv.org/pdf/1712.08940.pdf%22%20%5Ct%20%22_blank) *,* [*https://www.sain.ca/publication/thorpePDFS/Do\_Password\_Managers\_Nudge\_Secure\_Random\_Passwords\_SOUPS2022.pdf*](https://www.sain.ca/publication/thorpePDFS/Do_Password_Managers_Nudge_Secure_Random_Passwords_SOUPS2022.pdf) *,*[*https://www.researchgate.net/publication/309467190*](https://www.researchgate.net/publication/309467190)

Project Management Methodology, Water Fall Methodology Image from Google

# Variations From The Initial Proposal

Changes for proposal need be here.

I added the following extra features.

* Graph of deaths
* Links to videos on rip currents

I did not implement the Search functionality due to time constraints.

# Tools and Technologies Used (in alphabetical order)

* CSS
* Cookies
* EJS
* Express.js
* HTML
* JavaScript
* Node.js
* XAMPP - Apache
* Notepad++ / Edit plus

# SDLC

Water fall

As planned during the proposal stage, the incremental/ iterative methodology worked with more features and functionality added and tested at every stage.

# Project timeline

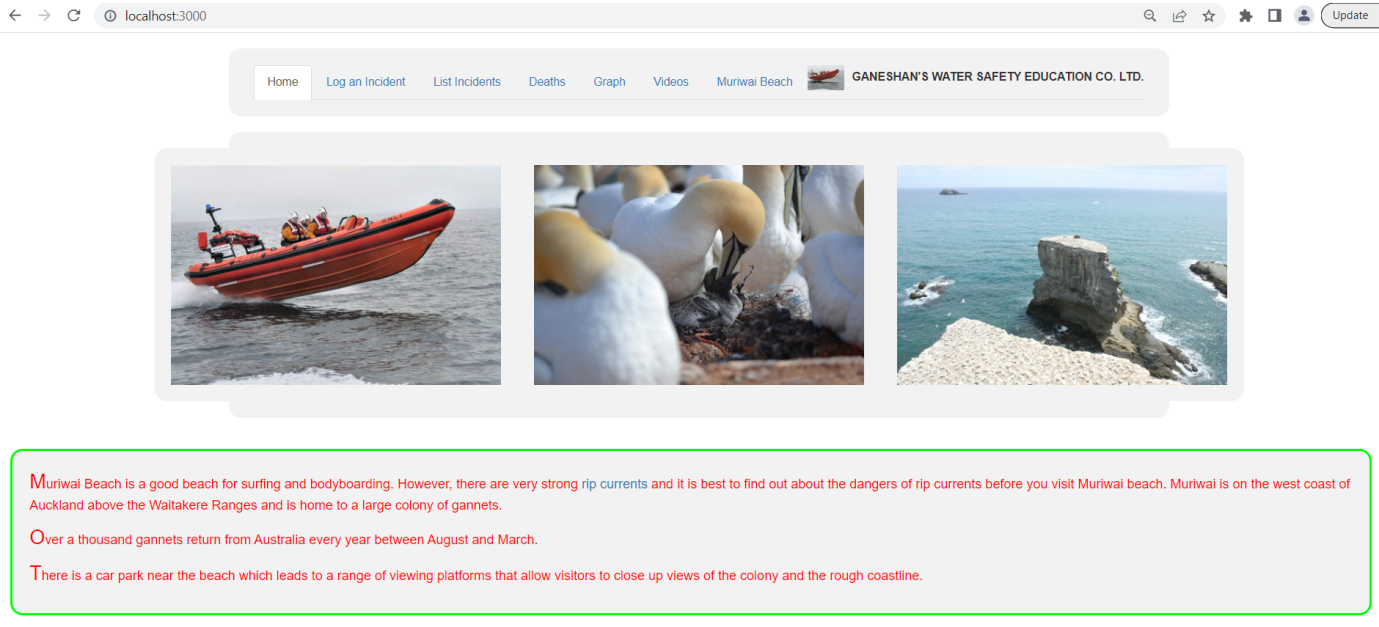
We have changed it…..

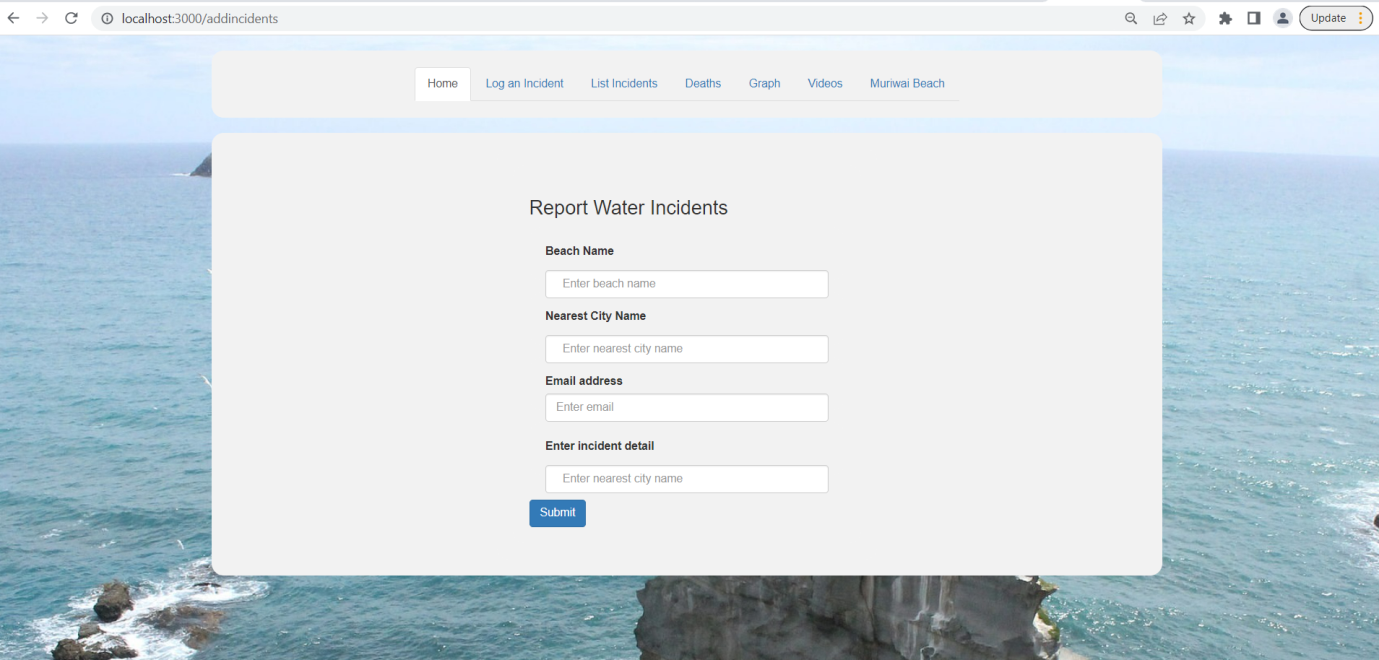
I pretty much followed the planned timeline as in the proposal.

# Ethical and cultural impact

As discussed in the Proposal, this project can impact the community by raising awareness of water safety.

# Selected Screenshots of the completed App





Graphical user interface, application, table

Description automatically generated

Chart

Description automatically generated

A picture containing text, indoor, screenshot, different

Description automatically generated

# Reflection

I started with very little knowledge in web app development. By doing this project I learned how to use basic HTML, CSS, Bootstrap, MYSQL, Node.JS and JavaScript. I am aware that there is much more to learn. I also learned about Use Cases and improved my project and time management skills. The learning environment was good and collaborative with people helping each other.

If I am to do this project or a similar project again, I will…………..

**References**

Future Skills Academy. August 2022. JavaScript.

What is a Password Generator? [*https://www.techopedia.com/definition/31414/password-generator*](https://www.techopedia.com/definition/31414/password-generator)

Why should I use a Password Generator? *https://teampassword.com/blog/why-should-i-use-a-password-generator*

User Stories and Use Cases *SDLC – Define Phase: Use Cases 13th June 2022 MCSD51: Prepared by Boon*

Ethical and Cultural Impact[*https://arxiv.org/pdf/1712.08940.pdf%22%20%5Ct%20%22\_blank*](https://arxiv.org/pdf/1712.08940.pdf%22%20%5Ct%20%22_blank) *,* [*https://www.sain.ca/publication/thorpePDFS/Do\_Password\_Managers\_Nudge\_Secure\_Random\_Passwords\_SOUPS2022.pdf*](https://www.sain.ca/publication/thorpePDFS/Do_Password_Managers_Nudge_Secure_Random_Passwords_SOUPS2022.pdf) *,*[*https://www.researchgate.net/publication/309467190*](https://www.researchgate.net/publication/309467190)

Project Management Methodology, Water Fall Methodology Image from Google

<https://www.w3schools.com/cs/trycs.php?filename=demo_helloworld>

# Appendix: Code

**server.js**

var express = require('express');

var path = require('path');

var cookieParser = require('cookie-parser');

var bodyParser = require('body-parser');

var flash = require('express-flash');

var session = require('express-session');

var db=require('./dbConfig');

var app = express();

// view engine setup

app.set('views', path.join(\_\_dirname, 'views'));

app.set('view engine', 'ejs');

app.use(express.json());

app.use(express.urlencoded({ extended: false }));

app.use(cookieParser());

app.use(express.static(path.join(\_\_dirname, 'public')));

/\* GET home page. \*/

app.get('/', function(req, res, next) {

res.render('home', { title: 'Home' });

});

app.get('/videos', function(req, res, next) {

res.render('videos', { title: 'Videos' });

});

app.get('/addIncidents', function(req, res, next) {

res.render('addIncidents', { title: 'Home' });

});

app.get('/getIncidents', function(req, res){

db.query("SELECT \* FROM incidents", function (err, result) {

if (err) throw err;

console.log(result);

res.render('getIncidents', { title: 'xyz', incidentData: result});

});

});

app.get('/deaths', function(req, res){

db.query("SELECT \* FROM deaths", function (err, result) {

if (err) throw err;

console.log(result);

res.render('deaths', { title: 'xyp', deathData: result});

});

});

app.get('/graph', function(req, res){

db.query("SELECT \* FROM deaths", function (err, result) {

if (err) throw err;

console.log(result);

res.render('graph', { title: 'graph', graphData: result});

});

});

app.post('/addIncidents', function(req, res, next) {

var beach = req.body.beach;

var city = req.body.city;

var email = req.body.email;

var detail = req.body.detail;

var sql = `INSERT INTO incidents (beach, city, email, detail, reported\_at) VALUES ("${beach}", "${city}", "${email}", "${detail}", NOW())`;

db.query(sql, function(err, result) {

if (err) throw err;

console.log('record inserted');

req.flash('success', 'Data added successfully!');

res.render('addIncidents');

});

});

// port must be set to 3000 because incoming http requests are routed from port 80 to port 8080

app.listen(3000);

console.log('Node app is running on port 3000');

module.exports = app;

**views/home.ejs**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Water incident reporting</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">

<link href= "/stylesheets/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

</head>

<body>

<div class="container">

<ul class="nav nav-tabs" role="tablist">

<li class="active"><a href=/>Home</a></li>

<li><a href=addincidents>Log an Incident</a></li>

<li><a href=getincidents>List Incidents</a></li>

<li><a href = "https://westauckland.co.nz/beaches/muriwai-beach"/>Muriwai Beach</a></li>

</ul>

</div>

<div class="container">

<div id ="leftbox">

<a href="https://en.wikipedia.org/wiki/Muriwai">

<img src="images/muriwai1.jpg" width="400px" alt ="Gannets at Muriwai, Auckland">

</a>

</div>

<div id="middlebox">

<a href="https://en.wikipedia.org/wiki/Gannet">

<img src="images/gannets.jpg" width="400px" alt ="Gannets at Muriwai, Auckland">

</a>

</div>

<div id="middlebox">

<a href="http://www.windsurf.co.nz/windsurf\_cam\_muriwai.asp">

<img src="images/muriwai2.jpg" width="400px" alt ="Gannets at Muriwai, Auckland">

</a>

</div>

</div>

<div id="divHome">

<p>

Muriwai Beach is a good beach for surfing and bodyboarding.

However, there are very strong

<a href="pages\ripcurrents.html">rip currents</a> and it is best to

find out about the dangers of rip currents before you visit

Muriwai beach. Muriwai is on the west coast of Auckland

above the Waitakere Ranges and is home to a large colony

of gannets.

</p>

<p>

Over a thousand gannets return from Australia every year

between August and March.

</p>

<p>

There is a car park near the beach which leads

to a range of viewing platforms that allow visitors

to close up views of the colony and the rough coastline.

</p>

</div>

</body>

</html>

**views/addIncidents.ejs**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Water incident reporting</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href= "/stylesheets/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

</head>

<body>

<div class="container">

<ul class="nav nav-tabs" role="tablist">

<li class="active"><a href=/>Home</a></li>

<li><a href=addincidents>Log an Incident</a></li>

<li><a href=getincidents>List Incidents</a></li>

<li><a href = "https://westauckland.co.nz/beaches/muriwai-beach"/>Muriwai Beach</a></li>

</ul>

</div>

<div class="container mt-4">

<div class="card">

<div class="card-body">

<h3>Report Water Incidents</h3>

<form method="POST">

<div class="form-group">

<label for="beach">Beach Name</label>

<input type="text" class="form-control col-lg-9" id="beach" aria-describedby="emailHelp" placeholder="Enter beach name" name="beach">

</div>

<div class="form-group">

<label for="city">Nearest City Name</label>

<input type="text" class="form-control col-lg-9" id="city" aria-describedby="emailHelp" placeholder="Enter nearest city name" name="city">

</div>

<div class="form-group">

<label for="email">Email address</label>

<input type="email" class="form-control col-lg-9" id="email" aria-describedby="emailHelp" name="email" placeholder="Enter email">

</div>

<div class="form-group">

<label for="detail">Enter incident detail</label>

<input type="text" class="form-control col-lg-9" id="detail" aria-describedby="emailHelp" placeholder="Enter nearest city name" name="detail">

</div>

<button type="submit" class="btn btn-primary">Submit</button>

</form>

</div>

</div>

</body>

</html>

**views/getIncidents.ejs**

<!DOCTYPE html>

<head>

<title>Incidents</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">

<link href= "/stylesheets/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

</head>

<body>

<div class="container">

<ul class="nav nav-tabs" role="tablist">

<li class="active"><a href=/>Home</a></li>

<li><a href=addincidents>Log an Incident</a></li>

<li><a href=getincidents>List Incidents</a></li>

<li><a href = "https://westauckland.co.nz/beaches/muriwai-beach"/>Muriwai Beach</a></li>

</ul>

</div>

<div id="divGet">

<table>

<tr>

<th>ID</th>

<th>Beach</th>

<th>City</th>

<th>Contact</th>

<th>Detail</th>

</tr>

<% if (incidentData.length !=0) {var i=1; incidentData.forEach( function(data) {

%>

<tr>

<td><%=data.id %></td>

<td><%=data.beach %></td>

<td><%=data.city %></td>

<td><%=data.email %></td>

<td><%=data.detail %></td>

</tr>

<% i++; }) %><%}%>

</table>

</div>

</body>

</html>

**public/stylesheets/style.css**

body {

background-image: url("paper.gif");

background-color: #e4f5cb;

}

h1 {

text-align: centre;

color: blue;

}

p {

color: red;

font-size: 16px;

}

p::first-letter {

font-size: 150%;

font-style: bold;

}

table {

border-collapse: collapse;

width: 38%;

}

th, td {

padding: 10px;

text-align: center;

border-bottom: 2px solid #ddd;

}

th {

background-color: yellow;

}

tr:hover {background-color: lime;}

div {

border-radius: 15px;

background-color: #f2f2f2;

padding: 20px;

}

#divHome

{

border: 3px solid #00ff00;

margin: 1cm 1cm 1cm 0.5cm;

}

#divGet

{

border: 4px solid #FF0000;

border-radius: 15px;

background-color: #f2f2f2;

padding: 20px;

display: flex;

justify-content: center;

}

#myChart

{

border: 4px solid #FF0000;

border-radius: 15px;

background-color: #f2f2f2;

padding: 20px;

display: flex;

justify-content: center;

}

input[type=text], select {

width: 100%;

padding: 12px 20px;

margin: 8px 0;

display: inline-block;

border: 1px solid #ccc;

border-radius: 4px;

box-sizing: border-box;

}

#leftbox {

float:left;

width:40%;

}

#middlebox{

float:left;

width:50%;

}

#rightbox{

float:right;

width:25%;

}

.container {

display: flex;

justify-content: center;

margin: 0.5cm 1cm 0.5cm 0.5cm;

}

.center {

display: flex;

margin: auto;

width: 60%;

border: 3px solid #73AD21;

padding: 10px;

}