

Technical Report

The Route Optimising SideKick



'The Route Planning Tool That Optimizes Your Journey'

Gavin Gaughran, x12107077, Gaving30@gmail.com

BSc (Hons) in Computing Evening
Networking and Mobile Technologies

11-05-2016

Declaration Cover Sheet for Project Submission

SECTION 1 *Student to complete*

Name:
Student ID:
Supervisor:

SECTION 2 **Confirmation of Authorship**

The acceptance of your work is subject to your signature on the following declaration:

I confirm that I have read the College statement on plagiarism (summarised overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature: _____ Date: _____

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

DECLARATION COVER SHEET FOR PROJECT SUBMISSION	1
EXECUTIVE SUMMARY	5
INTRODUCTION.....	5
BACKGROUND.....	6
RESEARCH.....	6
AIMS	7
TECHNOLOGIES	7
SYSTEM REQUIREMENTS.....	8
FUNCTIONAL REQUIREMENTS	9
REQUIREMENT #1 <GUEST ROUTE OPTIMIZE QUERY>	9
REQUIREMENT #2 < GUEST REGISTRATION >	11
REQUIREMENT #3 < USER LOG IN AND ROUTE OPTIMIZE QUERY >.....	14
NON-FUNCTIONAL REQUIREMENTS.....	17
PERFORMANCE/RESPONSE TIME REQUIREMENT	17
AVAILABILITY REQUIREMENT	17
SECURITY REQUIREMENT	17
RELIABILITY REQUIREMENT.....	17
PORTABILITY REQUIREMENT.....	17
REUSABILITY REQUIREMENT.....	17
RESOURCE UTILIZATION REQUIREMENT.....	17
DATA REQUIREMENTS.....	17
ENVIRONMENTAL REQUIREMENTS.....	18
USABILITY REQUIREMENTS	18
INTERFACE REQUIREMENTS.....	19
DESIGN AND ARCHITECTURE	19
OVERALL ARCHITECTURE	19
CODE ARCHITECTURE.....	20
SECURITY ARCHITECTURE.....	21
DATABASE SECURITY ARCHITECTURE.....	21
ALGORITHM ARCHITECTURE	21
SYSTEM ARCHITECTURE.....	23
IMPLEMENTATION	23
IDE	23
WEB APPLICATION	24
SQL SERVER.....	24
JAVASCRIPT	25
JQUERY.....	26
CSS	28

C#	28
TESTING	30
CLASS TESTS.....	31
TEST PLANS	33
CUSTOMER TESTING	35
EVALUATION	36
GRAPHICAL USER INTERFACE (GUI) LAYOUT	36
HOME PAGE (GUEST VIEW).....	36
HOME PAGE (LOGGED IN)	37
ROUTE OPTIMISED PAGE	37
LOG IN PAGE.....	38
USER REGISTRATION PAGE.....	38
ADMIN PAGE.....	39
CONCLUSIONS	39
FURTHER DEVELOPMENT OR RESEARCH	40
APPENDIX.....	40
PROJECT PROPOSAL	40
OBJECTIVE.....	40
BACKGROUND	41
SCOPE.....	42
TECHNICAL APPROACH/DETAILS.....	42
SPECIAL RESOURCES REQUIRED	43
PROJECT PLAN.....	43
EVALUATION	44
PROJECT ANALYSIS DESIGN	44
INTRODUCTION.....	44
PURPOSE OF THE PRODUCT DESIGN SPECIFICATION DOCUMENT	44
GENERAL OVERVIEW AND DESIGN GUIDELINES/APPROACH	45
ASSUMPTIONS	45
ARCHITECTURE DESIGN	45
OVERALL ARCHITECTURE.....	45
SYSTEM ARCHITECTURE	46
OVERALL ARCHITECTURE CLASS DIAGRAM	46
SYSTEM ARCHITECTURE CLASS DIAGRAM.....	47
SECURITY ARCHITECTURE	48
PERFORMANCE.....	48
GENETIC ALGORITHM	48
SYSTEM DESIGN.....	49

USE-CASES.....	49
SEQUENCE DIAGRAM	50
DATABASE DESIGN	50
APPLICATION PROGRAM INTERFACE	51
USER INTERFACE DESIGN	51
SECTION 508 COMPLIANCE	52
COMMUNICATION DIAGRAM.....	52
APPENDIX A: KEY TERMS.....	53
MONTHLY JOURNALS	54
INTRODUCTION – “3 DOWN AND 1 TO GO”	54
SEPTEMBER 2015	54
OCTOBER 2015.....	55
NOVEMBER 2015.....	57
DECEMBER 2015	58
JANUARY 2016.....	59
FEBRUARY 2016.....	59
MARCH 2016.....	60
APRIL 2016.....	60
MAY 2016	61

Executive Summary

This application, *The Route Optimising SideKick*, will assist businesses and people travelling for leisure when planning and optimizing their journeys. With time being so precious, businesses are continuously trying to save money wherever they can, this solution offers one of the easiest ways to do so. Using this application allows productivity to be enhanced and all without having to reduce employee's earnings or enforce cut backs of any sort.

Distribution companies, businesses delivering produce, repair companies and service companies all visit and travel to several destinations on a daily basis. By simply spending ten minutes planning the route, engineers and drivers can save so much time, every day. Other benefits include:

- ➔ Lower fuel consumption
- ➔ Enhanced productivity
- ➔ Reduced employee hours
- ➔ Lower wages for the company

The brilliance of this application is that people travelling for personal leisure can also use this for tracking routes when visiting several locations on a holiday or trip. This offers a massive gain, ensuring as little time travelling as possible and allowing them spend more time enjoying their actual destinations.

The initial release of this application is a free of charge application that can offer five free locations to guests. By simply registering to our site we will double that and allow you enter ten destinations at any given time. Once we have launched and are a recognized site we will eventually charge. The benefits of charging and offering this to businesses would be that they can save and store favourite routes or locations into our database. The plan is that we can again double or treble the allowed input, to twenty-thirty destinations in one go.

Introduction

Background

There are several reasons why I chose to create an application such as this, both from a personal and business perspective. Personally, throughout my college years I thoroughly enjoyed math, algorithms and of course, coding. AI had been my favourite subject, so I was trying to create an application that would let me work on improving my own algorithm/coding skills whilst also being a very useful tool to others.

Having previously worked in an IT company as a field repair engineer and delivering parts I have experience on how long the days can be. When I started, it was to my surprise, that my company, Wincor-Nixdorf would give their employees several locations to visit each day with no prior 'dispatch' assistance. By simply spending ten minutes planning the route, they could save engineers or drivers so much time each and every day. I later found out, other big companies such as Sky, still practice this and it is up to the engineer to decide what destination to visit first and then proceed in any order they wish. This application will let each driver plan their route exactly, using the most time-efficient and fuel-efficient route to take.

From a non-business related scenario, a friend of mine rented a car and wanted to travel around Ireland, he is from France. He planned to view several cities and knew the locations he wanted to go, entering all the locations into Google maps, it keeps the travel route in the exact order you enter them, so he spent a very long time trying to figure out what the best route would be to take. This application could be a very helpful for such an instance and result in less travel time and more time spent at each destination.

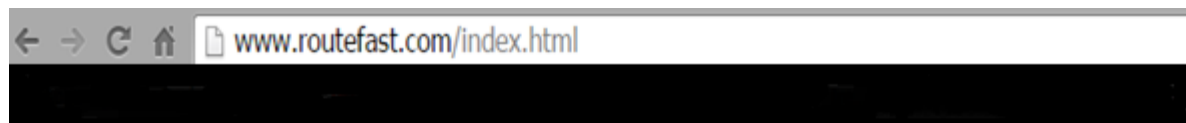
Research

My research included looking up the competition website and applications. The top Google listed apps coming up when searching was Road Warrior. My research into that revealed feedback such as *"Not enough stops I do over 30 stops a day and this app does not allow even 10"* So if this user simply signs up for free to *The Route Optimising SideKick* we will allow him 10 free locations to be entered for free. Once established we will advance to hopefully over the twenty mark. Another popular app is BestRoute and its feedback was *"I would love to see a "filing cabinet" with all of my customer job locations and simply be able to drag & drop from my filing cabinet into a route for that day"*. Once users are registered and paying for the service they will be offered a "Favourites" section, users should not need to enter the same locations, daily or weekly, they should be able to choose the location from a list of regularly used locations. The solution to help this is adding autocomplete textboxes so when they start typing in the address the autocomplete will finish it, saving time.

Top websites for “route planning” or “route optimizing” websites were AA and RouteXL. AA does not optimize locations and has a very inconsistent location finder, it could not find O’Connell Street, Dublin when I searched as the below image shows.



The RouteXL was a very sloppy website with sporadic searches and some very basic coding errors on its home page, clicking its icon on the home page lead to the below.



Not Found

The requested URL /index.html was not found on this server.

Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request.

Aims

In all aspects of a busy corporate or personal life, I believe that preparation is vital; I therefore believe having this application will benefit distribution companies wishing to optimize their routes or for people wishing to travel who want to plan ahead.

My aim is to create an application that can be used easily in a business and personal aspect. I want this to be something a business gets used to using daily and start relying on to assist them in their everyday tasks.

An application such as this has so much potential, ideally my aim would be to create this web application and once successfully completed and launched start looking into features and add-ons to further enhance it. Possibilities include adding a synchronized phone app to send locations to users on the road, or for personal users add in links to tourist’s spots or other locations of interest.

Technologies

The web application was developed in Visual Studio 2015 using the web framework ASP.NET and the programming language C#. The frontend consists of the ASP.NET framework, version 4.5.2 and Google APIs V3. The content of the site is constructed using HTML elements and styled using CSS, the use of JS and mainly JQuery is used to add a smooth and dynamic experience when using the Web pages.

The backend will be managed using SQL Server database and connected via C#. The server-based code connections allow interactions between our front and back end. A database will store the data that our site requires and we will use SQL commands to extract the relevant information.

A Google Maps API will be inserted into the web application UI, creating an easy-to-use frontend for the user. The algorithms themselves are done using C# and GAF (Genetic Algorithm Framework) which is a .Net/Mono assembly.

To improve the appearance of the application, CSS will be used to design. The reason behind using CSS is for consistency across all pages; I can design the page and use that consistent style across all pages within the application.

For design features I will use JS, using the JS library JQuery will increase the look of the web pages. It was used to add some slick features to the pages such as pop-out windows, and for creating dynamic pages, links and buttons. Adding new textboxes for more user entries to the algorithm can be easily done if required.

The final and main addition to the application will be the algorithm itself, which was coded up in C#. The user enters the destinations they want to enter simple textboxes and hitting a button injects the co-ordinates into the algorithm, returning.

To accommodate the websites look on hand held devices, the webpages will be Bootstrap responsive as to function correctly on these smaller mobile devices.

System Requirements

User Requirements

Looking for some professional opinions I chatted with two people about this, Rebecca Solomon (Wincor-Nixdorf, Dispatcher) and Jason McGuinness (Wincor-Nixdorf, Service Engineer) in relation to the requirements they would want in an application like this.

Below are the main features they would like to see in this type of application.

Dispatcher POV:

- Users can create an account and register to the site.

- When registered, the ability to login to their account.
- Use Google Maps API to choose destinations.
- Easy print option for chosen routes, post-optimisation.
- A backend database must be available to store data.
- Registered users should be able to store “favourite” locations.
- The GUI should be focused on being easy to use.

Engineer POV:

- Structured destinations
- Shorter day
- Less driving

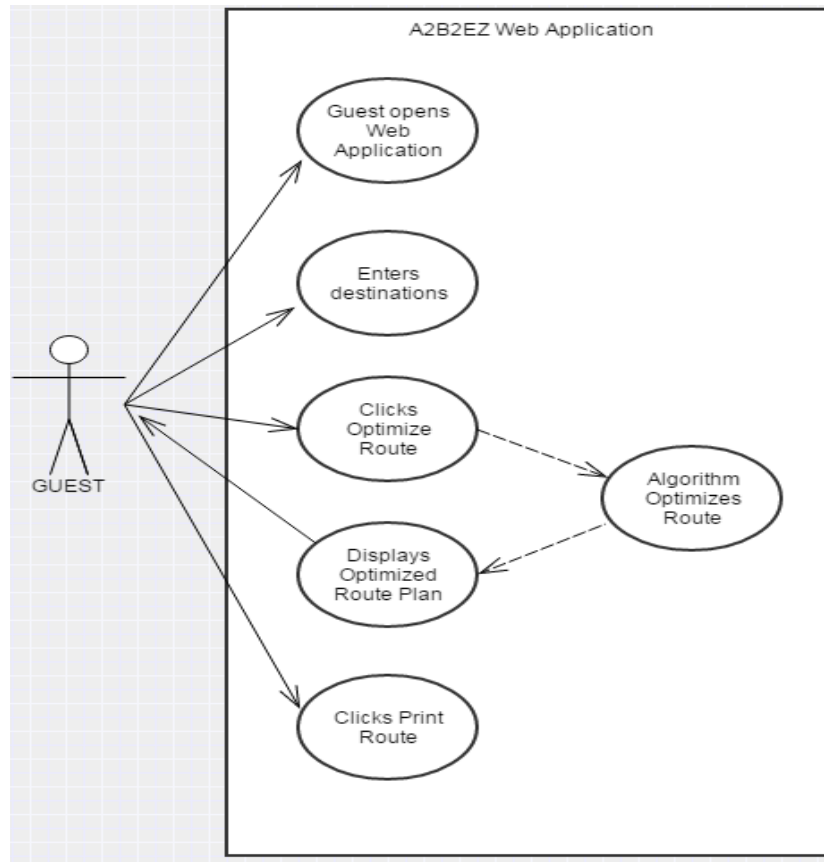
The dispatcher being the one using this application wants it easy to use and simplistic to enter data. The engineer wants this tool used by the dispatcher to benefit his day by helping him save some time where he can.

Functional Requirements

Requirement #1 <Guest Route Optimize Query>

Scope: The scope of this use case is to cover a guest to the application using the basic route optimization. The database needs to be involved in the process but it's not an external Actor as it's an internal feature of my application.

Description: This use case describes the process of a guest visiting the web application and using the route optimization tool. The user will open the web application. The system displays the 'home' page. The user will choose destinations from the map and run the optimization tool. The system uses the algorithm to generate the optimal route and return this to the user. The user can print the results for their own usage.



Flow Description

Precondition

The system is in initialisation mode and successfully online. The user logs in to the web application with the intention of using the optimization tool.

Activation

This use case starts when the user has an internet connection and opens the web application.

Main flow

1. The Actor opens the web application.
2. The Actor enters some destinations into the application.
3. The Actor clicks the optimize routes button.
4. The system computes the optimized route.

5. The system returns the optimized results.
6. The Actor clicks print in the application.

Alternate flow

<User Logs in to Account>

1. The Actor opens the web application.
 - A1. The Actor clicks the login button.
 - A2. The Actor signs in to the system
(The Actor is returned to Main Flow 2.)
2. The Actor enters some destinations into the application.
3. The Actor clicks the optimize routes button.
4. The system computes the optimized route.
5. The system returns the optimized results.
6. The Actor clicks print in the application.

Exceptional flow

1. The Actor opens the web application.
2. The Actor enters some destinations into the application.
3. The Actor clicks the optimize routes button.
 - E1. The system cannot compute routes.

Termination

The user exits the web application.

Post condition

The system continues to function as normal, offering route optimization.

Requirement #2 < Guest Registration > Description & Priority

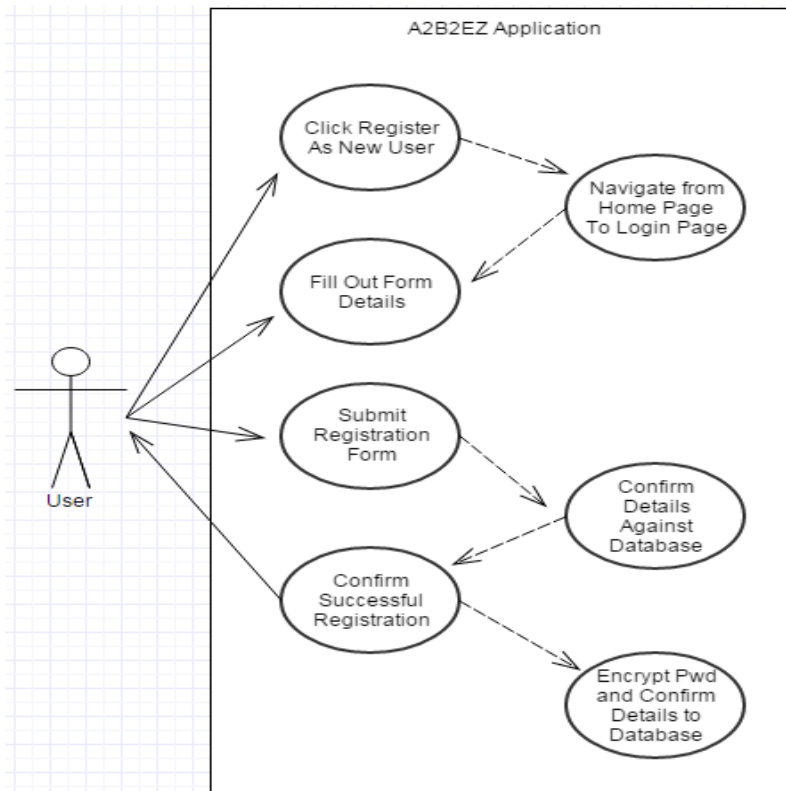
The application can be used as a guest but to get the full benefits of the system, user registration is essential. Guests can enter 5 locations in for route optimization but once registered it will more (to be confirmed exactly during final development).

Scope

The scope of this use case is to cover the user registration process. The database again needs to be involved in the process but it's not an external Actor as it's an internal feature of my application.

Description

This use case describes the user entering the web application and clicking on the “User Registration” link. The site will navigate to the User login registration form page. The user will then complete the form with their details and submit the form. The application upon submitting will perform the necessary checks against the database, i.e.: if the username meets criteria, if the password is strong enough and all necessary fields are filled in. Once successful the user will get a message confirming they are registered. The database will log and store all information for that new registered user.



Flow Description

Precondition

The system is in initialisation mode and connecting successfully to its database. The user logs in to the web application with the intention to create a new user account.

Activation

This use case starts when the user clicks the Register link within the application.

Main flow

1. The actor clicks the register link
2. The system navigates to the register page.
3. The Actor fills out the form.
4. The Actor submits the form.
5. The system checks against the database.
6. The system confirms success to the user.
7. The system stores the details correctly.

Alternate flow

<User Details Not Complete>

1. The system navigates to the register page.
2. The Actor fills out the form.
3. The Actor submits the form.
4. The system checks against the database.

A1. The system detects some details are missing.

A2. The system informs the user of the findings.

(The Actor is returned to Main Flow 2 so this is very similar to a Recovery flow for the fact it goes back some steps but due to the nature of the use case this is essential.)

2. The Actor fills out the form.
3. The Actor submits the form.

4. The system navigates to the register page.
5. The system checks against the database.
6. The system confirms success to the user.
7. The system stores the details correctly.

Exceptional flow

1. The actor clicks the register link
2. The system navigates to the register page.
3. The Actor fills out the form.
4. The Actor submits the form.
5. The system checks against the database.
 - E1. The system loses connection to the database.
 - E2. The system loses the Session

Termination

The system registers the user and saves details to database.

Post condition

The system continues to function as normal.

Requirement #3 < User Log In and Route Optimize Query >

Description & Priority

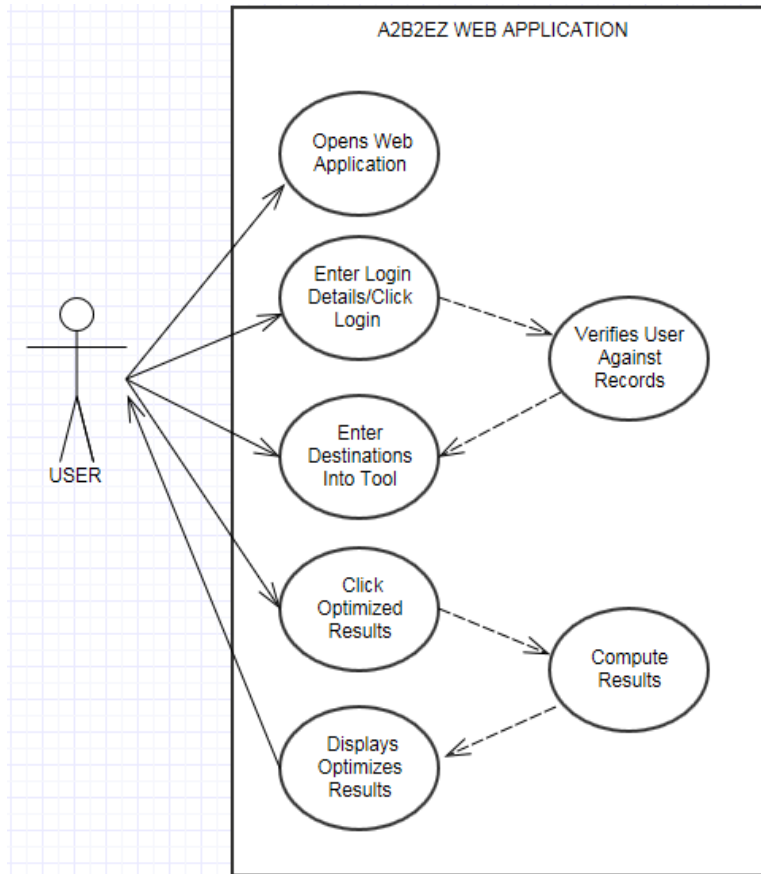
The application can be used as a guest but to get the full benefits of the system, user registration is essential. Guests can enter 5 locations for route optimization but once registered that will increase to 10.

Scope

The scope of this use case is to cover the user logging in to the application and using the tool. The service (algorithm) and the database need to be involved in the process but they are internal features, so they will not be additional Actors.

Description

This use case describes the user entering the web application and entering their login details, then clicking on the “Login” button. The database will verify the details against its data. The site will refresh the home page, welcoming the user. The user will then add some destination criteria for their query. The application will use its algorithm to compute the best route and return the results.



Flow Description

Precondition

The system is in initialisation mode and connecting successfully to its database. The user has created an account previously with the application.

Activation

This use case starts when the user opens the application in their browser.

Main flow

1. The Actor opens the web application.

2. The Actor enters his details and clicks Login.
3. The System verifies the user's credentials
4. The Actor enters some destinations into the tool.
5. The Actor clicks optimize.
6. The System computes the destinations.
7. The System displays the results.

Alternate flow

<User Details Not Complete>

1. The Actor opens the web application.

A1. The Actor had previously cached his details and is logged in automatically.

(The Actor goes immediately to Main Flow 4 as the details cached.)

4. The Actor enters some destinations into the tool.
5. The Actor clicks optimize.
6. The System computes the destinations.
7. The System displays the results.

Exceptional flow

1. The Actor opens the web application.
2. The Actor enters his details and clicks Login.

E1. The Actor loses internet connection.

Termination

The system returns the optimized results.

Post condition

The system returns to initialisation mode ready to optimize.

Non-Functional Requirements

Performance/Response time requirement

The communication between the system and the user should be averaging 1 second per destination.

Availability requirement

The database must always be available for users who have registered accounts so they can access their favourites or add more than 5 destinations into a new search.

Security requirement

The password details in the database must be encrypted to prevent unauthorised access.

Reliability requirement

The system should always return reliable and correct results from its code based algorithm.

Portability requirement

Using bootstrap for this application will allow the application to be used across different sized platforms and devices.

Reusability requirement

The favourites tab allows users to re-use their most used locations and this really

Resource utilization requirement

The code behind the application can be modified at any stage and tweaked to improve its utilization.

Data requirements

The map data is pulled from Google Maps API and Google libraries using JavaScript to perform client side commands. The data will be displayed to the user via the application.

All data entered into the application will be entered by the users. Addresses and locations on the Map need not be stored in the application unless the user has signed up for an account and wants their locations saved to the database.

The system will be designed so that the database will store all data entered for registered users, so it will be up to the individual to enter the information to be stored.

Environmental requirements

The environmental settings from a development point of view would require Visual Studio 2015 and browser access to test on the API section of the application.

From a user perspective the website portion of the project requires a suitable platform on which to run. The ideal environment would contain a web browser capable of processing HTML5, and CSS3. Google Chrome would be the recommended browser.

The application requires a device on which to run, such as laptop, mobile device or desktop.

Usability Requirements

The main usability requirements are tabled below.

Property	Measure	Reasoning
Ease of use	<1 hour learning	The ease of use should be main priority on this application for non-technical users.
Speed	Max of 1-2 seconds per destination entered.	Allowing time for algorithm to work out best route once entered by user.
Size	Amount of storage per user TBC	Database size tbc dependent on details stored and number of users
Portability	User unaffected on different device types.	The user should not be impacted dependent on the device they use for the application.

To acknowledge these requirements, I have implemented the following.

Ease of Use: Easy to use, Google Maps API and autocomplete textboxes are used to help the user experience.

Speed: Constant code modifications can be performed to enhance the speed of the algorithm.

Size: If the limit of 10 GB is reached on the local DB then SQL Lite can be implemented with 140TB limit.

Portability: Bootstrap is within the application and ensures it's responsive no matter what device size you are accessing the application on.

Interface Requirements

The GUI was created in ASP.NET as a C# web application and all graphical interactions were done using the HTML structures and designed using CSS with additional functionality being added through JS, namely the JQuery library.

The use of CSS was to define styles for my application, including the design, layout and variations in display for different devices and screen sizes.

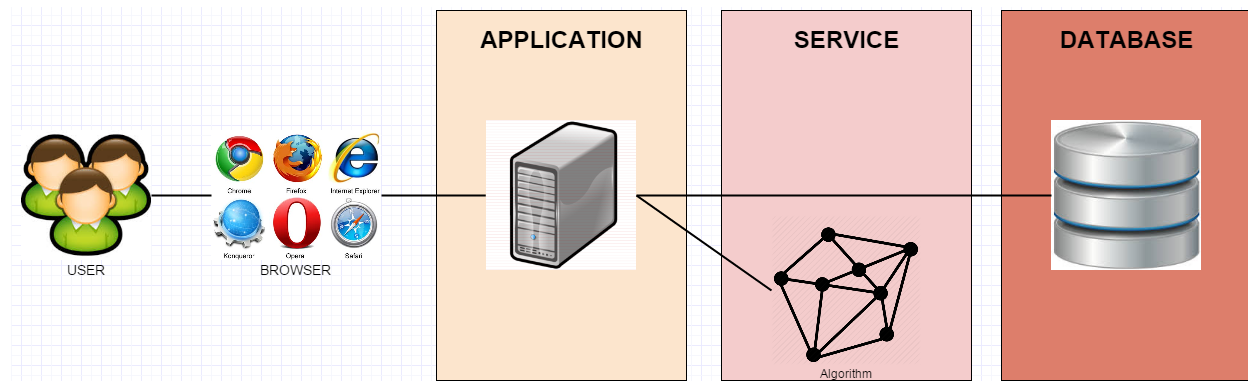
The web application has an interactive Google Maps API within the application to allow users to choose their destinations in the easiest way possible. The ability to choose Favourites from a dropdown will be available to ‘paying’ users, when implemented.

The database storing the data for user accounts is a simple SQL Server and users can just fill in a registration form with their details to avail of additional destination entries. They registration and logins are through stylish textboxes and allows users add details to their accounts at the click of a button. Forms were used to simplify this aspect of the application.

A contact form is available for users to submit queries or feedback they may have to the administrators and this will include a rating/feedback section for our own personal improvement.

Design and Architecture

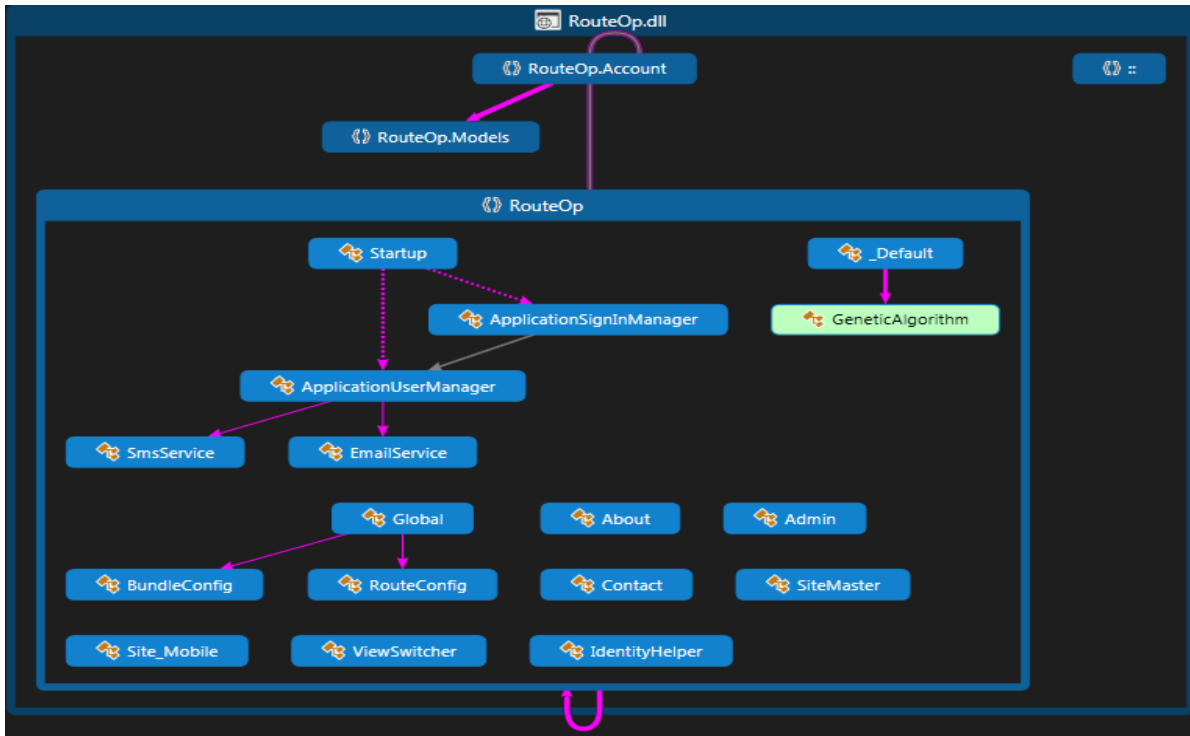
Overall Architecture



This is a high level description of the overall architecture. On the far left it starts with the user accessing their web browser. The application is opened using the chosen browser. Then application can then directly connect to the database or to its services. It's totally dependent on the users ask as to what it connects to, for example, the User login would connect to the dataset and would not need the Service section and if a guest ran a Route

plan they would not need to connect to the Database so just the Service section would be accessed.

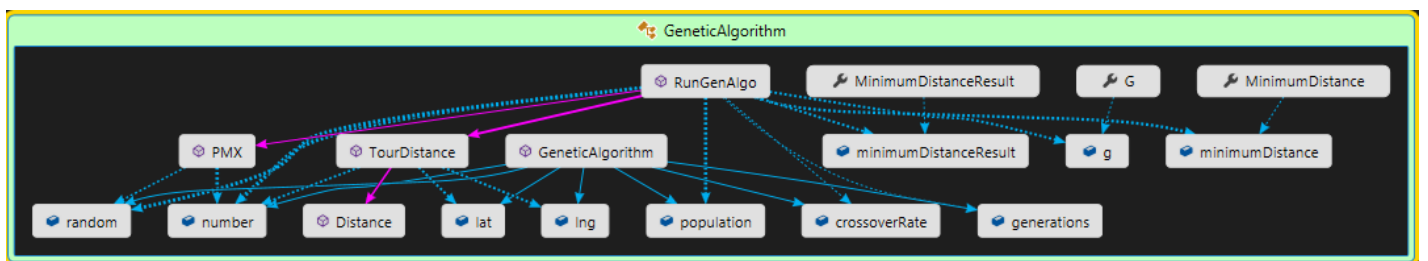
Code Architecture

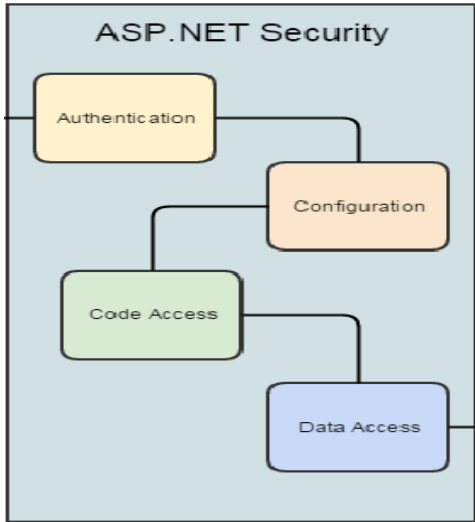


The above CodeMap displays the layout of all files with the Application (RouteOp.dll)

This is a high level diagram displaying the complete content of this application.

An example of how we can then enter into each file and get our class diagrams is below:





Security Architecture

ASP.NET applications can come with a host of features to secure their web applications but as developers, plenty of attention and time should be spent ensuring some basic features are enabled such as:

- Authentication and Authorization
- Application configuration
- Data validation, Code Access,
- User password policies and hashing.

Database Security Architecture

This scenario is the basic ASP.NET database security structure. As we store user details we apply the standard ASP.NET structure. This consists of two physical tiers so registered users can securely log in to the Web-based application using a Web browser. The ASP.NET-based Web application makes secure connections to a Microsoft SQL Server database to manage predominantly data retrieval tasks.

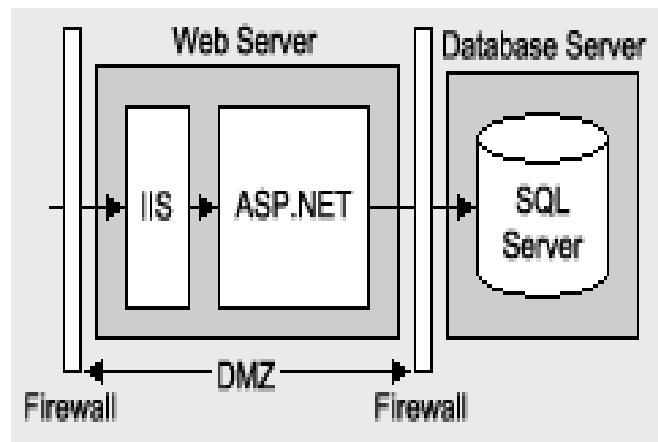


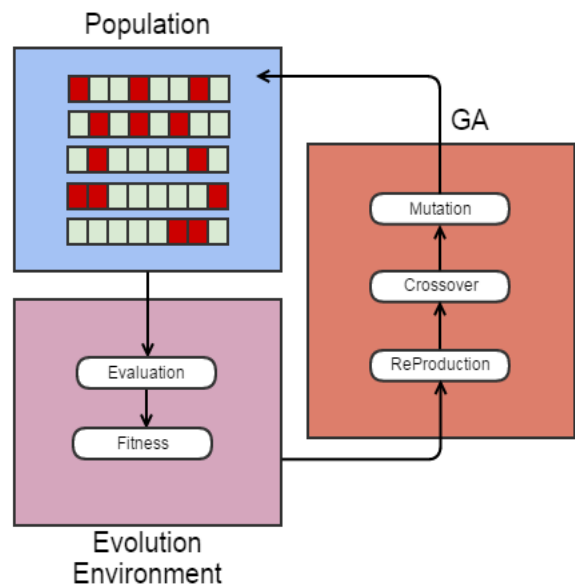
Image from: <https://msdn.microsoft.com/en-us/library/ff649341.aspx>

Algorithm Architecture

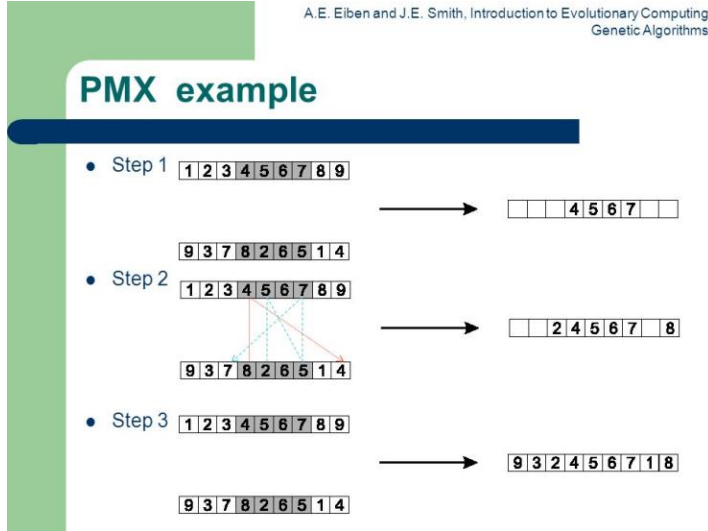
The population of the Genetic Algorithm is the locations on the map that are entered into the algorithm to be optimized.

The initialization occurs in the Evolution Environment where the fitness (criteria) is evaluated before the algorithm can compute. The latitude and longitude are gathered collected as a string before being entered into the code and separated into two arrays.

The GA itself consists of mutation and crossover and reproducing this until the best, or optimal



result has been reached.



Partially mapped crossover (PMX) is used. This genetic algorithm operator offers better performance than most other crossover techniques.

Parent 1 donates a section of genetic material and the corresponding section from the other parent is sprinkled about in the child. Once that is done, the remaining characters are copied direct from parent 2.

The Haversine formula was used as it calculates the distance between two points represented via latitude and longitude. Augments each relevant event with the resultant distance, stored in a field named 'distance' or another field as specified. Latitude and longitude for input must be represented in decimal degree format, though separate input fields may be specified for each.

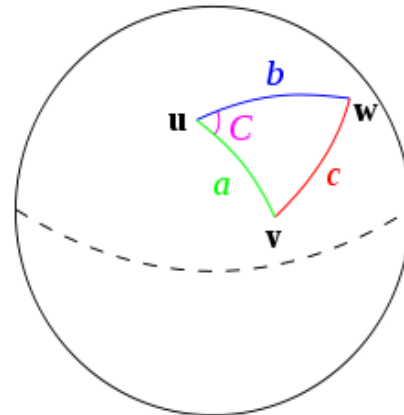
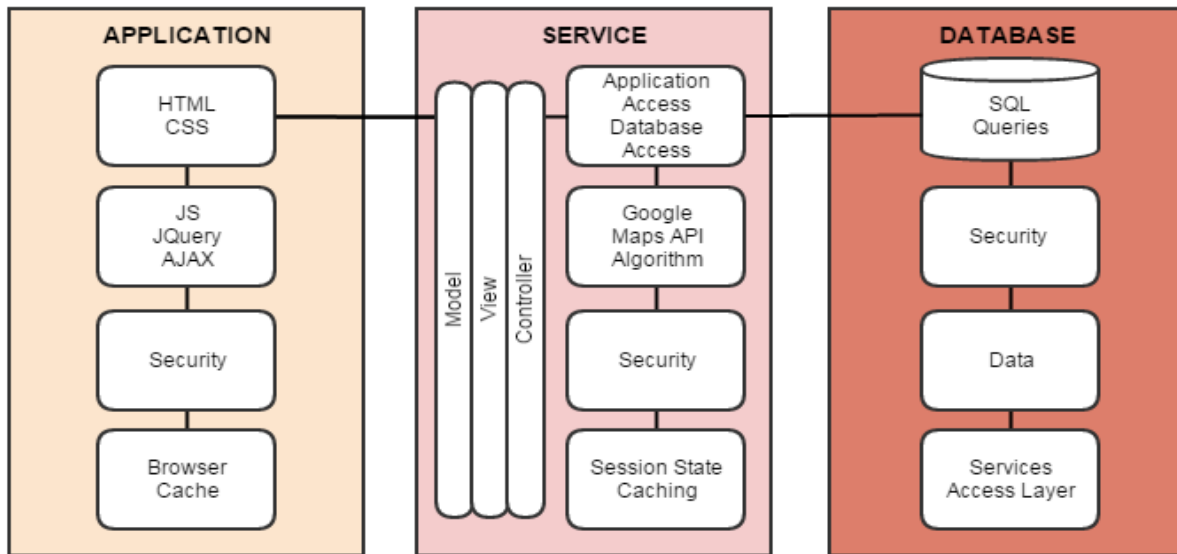


image from https://en.wikipedia.org/wiki/Haversine_formula

The user input will be strings containing Google Map exact addresses. This triggers hidden fields to store the latitude and longitude and they are transferred into the algorithm from our API and we can calculate the distances using that and return it in KM.

System Architecture



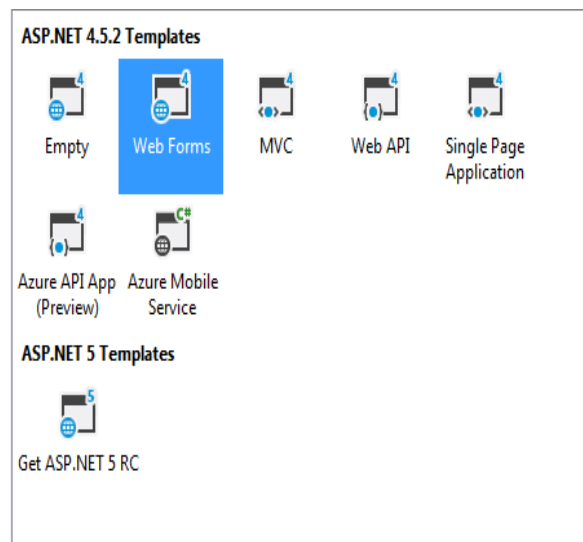
The architecture above is the System itself in some more details. The frontend system Application contains the HTML and CSS web forms, they receive extra functionality using JS, JQuery or Ajax. The forms will contain security within the browser and within the forms when it comes to passwords. The browser will contain a cache that links to the forms. The services behind the page include Access layers, algorithms and API's. Internal caching in the form of Sessions is built into the system. The backend database has obvious security features protecting the data within in it.

Implementation

There are several different coding concepts used within this application and within this section I cover some of the key features of this web application. This section will explain most of the concepts involved with the implementation of this application.

IDE

The application was developed in Microsoft's Visual Studio 2015. The .NET framework was 4.5.2 and this works well within this environment. I took full advantage of the IntelliSense, easy code navigation, fast builds, and quick deployment provided by Microsoft.



Web Application

With a vast array of application types to choose from within VS15, I went for an 'ASP.NET Web Application'. The 4.5.2 template was 'Web Forms' as this encourages dynamic websites and involves a drag and drop/event driven model. Data access was a big thing too as having users registering and logging in is important to the application.

SQL Server

Microsoft's SQL Server relational database management system was used as the backend to manage and store the required details. The data stored inside SQL Server will be housed in a "relational database" and the SQL Server is an entire "management system", not just a database.

A very brief summary of how we inserted information into the database using SQL is shown within the below code:

The user would complete the contact us form and click submit. The code then retrieves the input from numerous textboxes, shown in the one example below:

```
0 references
protected void SubmitFormButton_Click(object sender, EventArgs e)
{
    string strName = nameTextbox.Text.ToString();
}
```

The data is now collected and the new SqlConnection is created to the database. The SQL insert command is created that will inject the data to the database.

```
SqlConnection con = new SqlConnection();
con.ConnectionString = ConfigurationManager.ConnectionStrings["DefaultConnection"].ConnectionString;
SqlCommand cmd = new SqlCommand();
cmd.CommandText = "INSERT INTO QueryTable(Name,eMail,Subject,Comments,PostDate,Replied)" +
```

Once all settings are ready, the connection to the database is opened, the commands needed are run and the connection is closed.

```
con.Open();
int result = cmd.ExecuteNonQuery();
con.Close();
```

The user is informed of the success:

```
if (result > 0)
{
    successLabel.Text = "Your Query Has Been Sent Successfully!";
}
```

Retrieving data can be done using a GridView which is a .Net Framework Class Library that forms the connection to the database. It forms some of the Systems.Web.UI.WebControls namespace. This would create the database connection and pull the chosen data from using a command similar to this:

```
SelectCommand="SELECT * FROM [QueryTable]"
```

JavaScript

Mainly used for the integration of the Google Maps API within the application. Here are some examples from the application:

The below is the function for creating the Maps within my div 'map'. The latitude and longitude are set over Ireland as a default. The section from 'IF(navigator.geolocation)' gets the current position of the user and sets that as the default starting location of the map.

```
window.initMap = function () {
    directionsDisplay = new google.maps.DirectionsRenderer({
        suppressMarkers: true
    });

    //Create new map within the 'map' div with default lat and long and a default zoom of 7.
    map = new google.maps.Map(document.getElementById('map'), {
        center: { lat: 53.3498, lng: -6.2603 },
        zoom: 7
    });

    //Find current lat and long location and set this as default starting position
    if (navigator.geolocation) {
        browserSupportFlag = true;
        navigator.geolocation.getCurrentPosition(function (position) {
            initialLocation = new google.maps.LatLng(position.coords.latitude, position.coords.longitude);
            map.setCenter(initialLocation);
        }, function () {
            handleNoGeolocation(browserSupportFlag);
        });
    }

    // The else statement is for browsers not supporting Geolocation
    // Current browser support for the W3C Geolocation API as of 06-05-16
    // iPhone 3.0 +, Android 2.0 +, Firefox 3.5, Safari 5.0, Chrome 5.0, Internet Explorer 9.0+
    else {
        browserSupportFlag = false;
        handleNoGeolocation(browserSupportFlag);
    }

    directionsDisplay.setMap(map);
}
```

All the textbox related to the map section are autocomplete Google textboxes that makes user input extremely easy. This is implemented through listeners, such as:

```
autocomplete.addListener('place_changed', function () {
```

The listener constantly listens until a change within the box has been made and then offers the user a choice, example below:



Waypoints are used in conjunction with the DirectionRenderer to plot the markers on the map and display the connecting routes on the actual roads in Google Maps. Below I create an array called waypoints and start pushing locations into it.

```
var waypoints = [];  
  
stop = new google.maps.LatLng(lat1, long1)  
waypts.push({ location: stop, stopover: true });  
createMarker(stop);
```

JQuery

With a reputation for being a fast, small, and feature-rich JavaScript library, I chose JQuery for my responsive, dynamic features such as button clicks and ‘on-the-fly’ tasks. Some examples of its usage follow:

The result of the below code, left, is displayed, right. At the top is the alert, notifying the user there are still two boxes to add input to. The two boxes needing input are now highlights in red to make it easy for the user to identify the missing boxes.

```

$(document).ready(function () {
    //Monitor the button (id:verifyButton) for a click event
    $('#verifyButton').click(function (e) {
        //Array to store amount of boxes that need to be filled
        var array = [];
        var arrayOfDestinationNameValues = [];
        var isValid = true;
        var count = 0;
        //Look for any input textboxes inside the mapInputSection div
        $('#mapInputSection input[type="text"]').each(function () {
            count++;
            //If box is empty get the attribute id and store it to array
            if ($.trim($(this).val()) == '') {
                isValid = false;
                var id = $(this).attr('id');
                array.push(id);
                //If empty change CSS to make Red as not filled in yet
                $(this).css({
                    "border": "2px solid red", "background": "#FFCECE"
                });
            } else {
                //If details are in then store value of textbox into array
                var valueOfTextbox = $(this).val();
                arrayOfDestinationNameValues.push(valueOfTextbox);
            }
        });
    });
});

```

Fill In The 2 Empty Textboxes In Red And Then Verify To Continue.

OK

Enter Your Destinations Below:

Start

2

3

4

5

Below is the code to add more textboxes on the fly, for registered users. Read the comments within code for more information.

```

$(document).ready(function () {
    //Added to stop the "enter" button submitting any input buttons if pressed
    $('html').bind('keypress', function (e) {
        if (e.keyCode == 13) {
            return false;
        }
    });
    //var counter = 5;
    var max_fields = 10; //With original 4 and this adds 6 to allow 10 textboxes for now
    var wrapper = $(".input_fields_wrap"); //Fields wrapper

    var add_button = $(".add_field_button"); //Add button ID
    var x = 5; //initial text box count

    $(add_button).on("click", function (e) { //on add input button click
        e.preventDefault();
        if (x < max_fields) { //max input box allowed
            //alert("Counter" + counter + "x: " + x);
            x++; //text box increment
            //counter++;
            $(wrapper).append('<div><label class="numberLabelsForDestinations">' + x +
                '</label><input type="text" class="userInputTextbox" placeholder="Enter A Location" id="tb_' + x +
                '" name="tb_' + x + '" /><a href="#" class="remove_field">Remove</a></div>'); //add input box
            setUpHandler('tb_' + x);
        }
    });

    $(wrapper).on("click", ".remove_field", function (e) { //user click on remove text
        e.preventDefault(); $(this).parent('div').remove();
        x--;
    });
});

```

CSS

Bootstrap.css mainly controls the sizing and positioning of the page contents. I have made some small modifications to this file during development. These changed container sizes, background sizes and colours.

Most of the site.css file was created during the design implementation of the application and all divs, textboxes, labels, buttons and element settings are based here.

C#

The algorithm was done in C# using two code-behind class files. The first being behind the _Default page is Default.aspx.cs. We first gather the details for the algorithm from the user's inputs. They are saved a textbox as a large String. The code then breaks them up to latitude and longitude using the code below:

```
//Create Variable that takes in all our Lat and Long from destinations
string variable = hdnfldVariable.Value;

//Create int variable that is dynamically set from PageStyling.js.
//Counts textboxes submitted and sets below value (Converted from String to Int)
int arraySize = Convert.ToInt32(numFldVariable.Value);

//Create Array called words and use the comma as the split
string[] words = variable.Split(',');
int i = 0;
lng = new double[words.Length - 1];
lat = new double[words.Length - 1];
foreach (string word in words)
{
    string[] coords = word.Split(':');
    if (coords.Length == 2)
    {
        //Convert the String to a double
        lng[i] = Convert.ToDouble(coords[0]);
        lat[i] = Convert.ToDouble(coords[1]);
        i++;
    }
}
```

I introduced the Haversine formula as it seems to be a very popular method for measuring distance on a spherical object. As we use Google maps which in this case would be on a spherical surface, the minor allowances in the code give a slightly better accurate projection than those that would be on a standard (flat) form. The benefits to using the Haversine formula are:

1. Distances along lines of longitude will be reasonably accurate, a lot more so than other formulas such as Euclidean.
2. Distances along the Equator will be reasonably accurate.

- All other distances will be erroneous, in rough proportion to the differences in latitude and longitude.

```
double distanceNum=0;

//isNaN() function determines whether a value is an illegal number (Not-a-Number).
if (double.IsNaN(long1) || double.IsNaN(lat1) || double.IsNaN(long2) || double.IsNaN(lat2))
{
    // throw new ArgumentException(SR.GetString("Argument_LatitudeOrLongitudeIsNotANumber"));
}
else
{
    //Haversine Formula
    double latitude = long1 * 0.0174532925199433;
    double longitude = lat1 * 0.0174532925199433;
    double num = long2 * 0.0174532925199433;
    double longitude1 =lat2 * 0.0174532925199433;
    double num1 = longitude1 - longitude;
    double num2 = num - latitude;
    double num3 = Math.Pow(Math.Sin(num2 / 2), 2) + Math.Cos(latitude) * Math.Cos(num) * Math.Pow(Math.Sin(num1 / 2), 2);
    double num4 = 2 * Math.Atan2(Math.Sqrt(num3), Math.Sqrt(1 - num3));
    distanceNum = 6376500 * num4;
}
return distanceNum;
```

Partially Mapped Crossover was chosen as the crossover for the algorithm. Below is the declaration, the main contents of the PMX are defined below and my naming conventions make it obvious what each variable is used for.

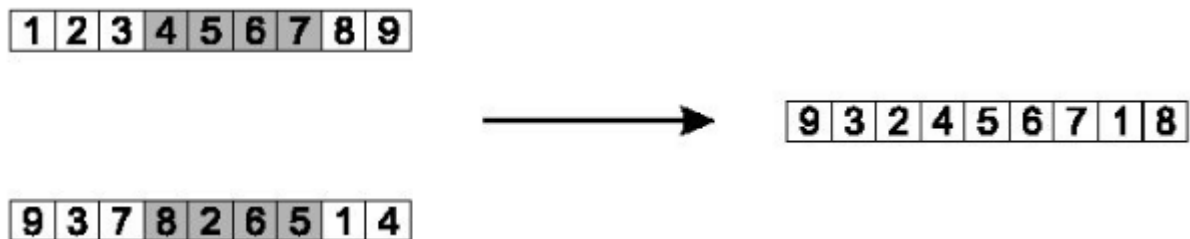
```
// Partially Mapped Crossover Section (PMX)
private void PartiallyMappedCrossover(int cutPoint1, int cutPoint2, int[] parentArray1, int[] parentArray2, int[] offSpring1, int[] offSpring2)
{
```

Two cut points are included; these are the positions in which we “slice” the parents to make the offspring’s.

9	3	7	8	2	6	5	1	4
---	---	---	---	---	---	---	---	---

 Slice occurs here, before the 8 and after the 5.

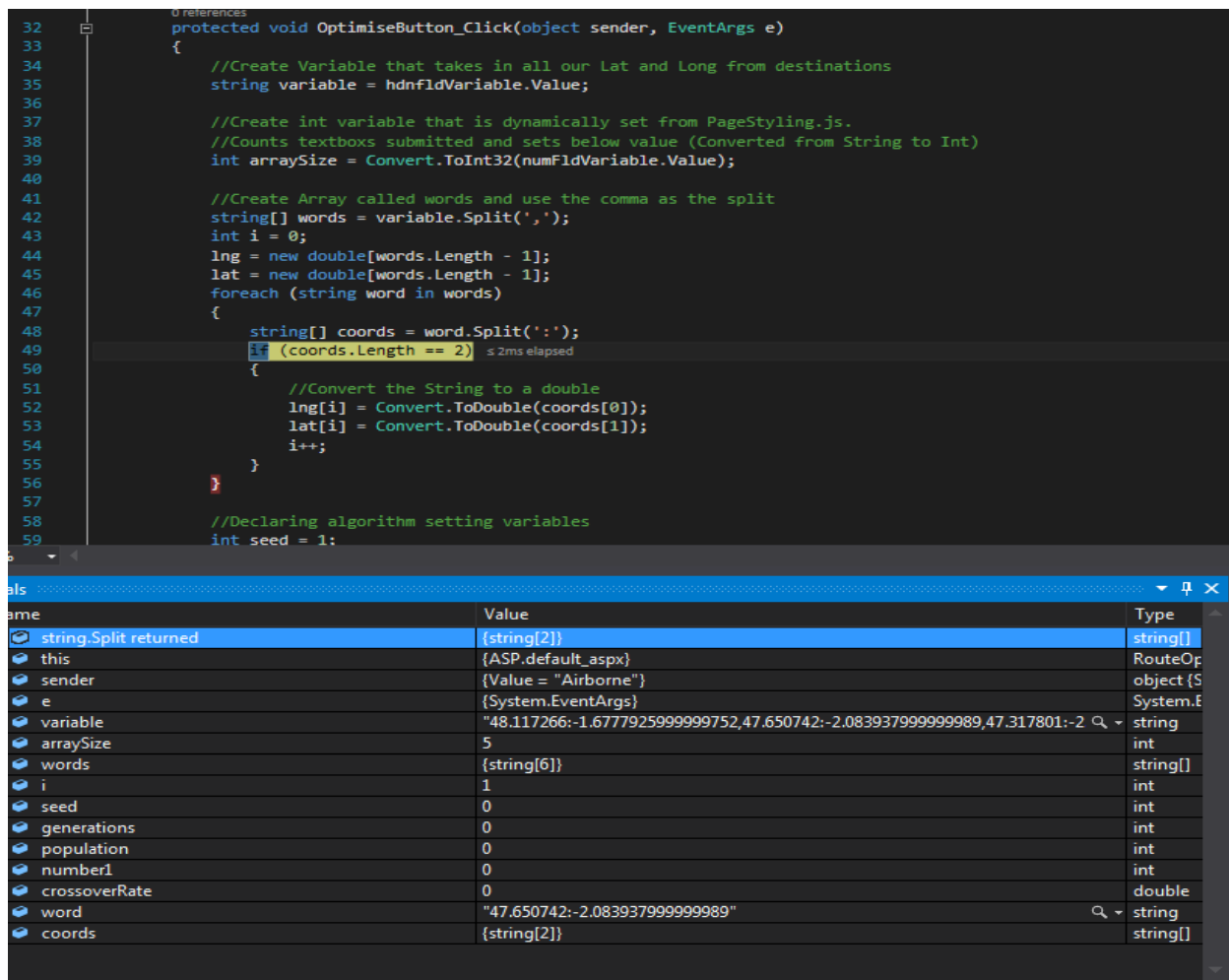
Two parent arrays are entered as we need to make the new offspring from these. Below is an example, straight swaps occur unless the gene already exists in the offspring then we use another gene to create a unique offspring.



Testing

All classes and code were debugged in Visual Studio 2015 using the Visual Studio debugger to observe the run-time behaviour of my program. Throughout the development life cycle all code was tested repetitively as entered and all syntax errors and even warnings were eliminated. Debugging my code allowed me to break into the execution of my program and to really to examine the code.

Breakpoints were used throughout and code was stepped into to examine the local properties and variables at runtime and witness how they evolve throughout the full cycle. The code below was an example of a Breakpoint and the variables were displayed below to ensure that evolve correctly. Extensive usage of the Console.WriteLine class were also used to display messages, errors and notes for myself

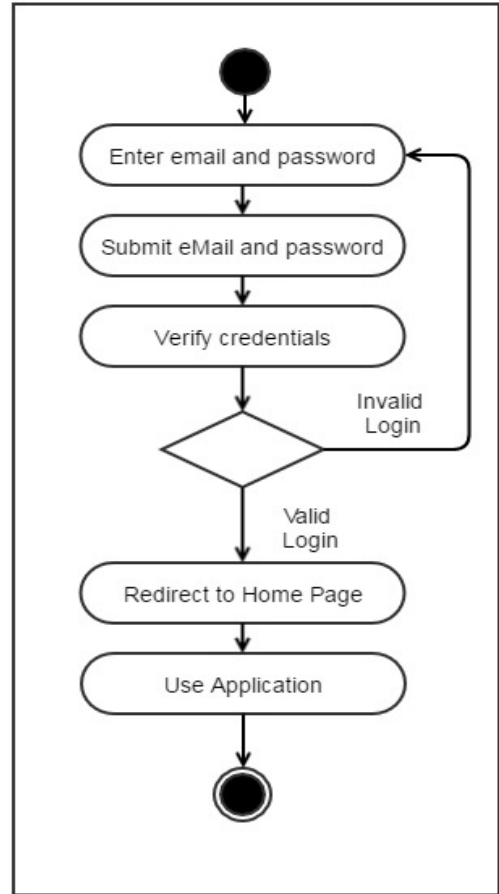
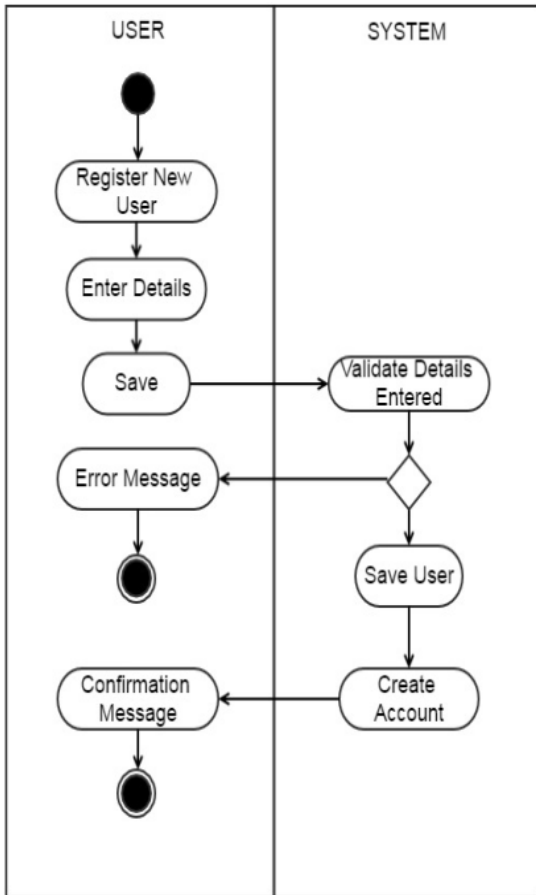


The screenshot shows a Visual Studio 2015 IDE. The top pane displays a C# code file with a breakpoint set on line 49. The code is as follows:

```
32 protected void OptimiseButton_Click(object sender, EventArgs e)
33 {
34     //Create Variable that takes in all our Lat and Long from destinations
35     string variable = hdnFldVariable.Value;
36
37     //Create int variable that is dynamically set from PageStyling.js.
38     //Counts textboxes submitted and sets below value (Converted from String to Int)
39     int arraySize = Convert.ToInt32(numFldVariable.Value);
40
41     //Create Array called words and use the comma as the split
42     string[] words = variable.Split(',');
43     int i = 0;
44     lng = new double[words.Length - 1];
45     lat = new double[words.Length - 1];
46     foreach (string word in words)
47     {
48         string[] coords = word.Split(':');
49         (coords.Length == 2)
50     {
51         //Convert the String to a double
52         lng[i] = Convert.ToDouble(coords[0]);
53         lat[i] = Convert.ToDouble(coords[1]);
54         i++;
55     }
56 }
57
58 //Declaring algorithm setting variables
59 int seed = 1;
```

The bottom pane shows the Variables window with the following data:

Name	Value	Type
string.Split returned	{string(2)}	string[]
this	{ASP.default_aspx}	RouteOf
sender	{Value = "Airborne"}	object {S
e	{System.EventArgs}	System.E
variable	"48.117266;-1.67779259999999752,47.650742;-2.0839379999999989,47.317801;-2	string
arraySize	5	int
words	{string(6)}	string[]
i	1	int
seed	0	int
generations	0	int
population	0	int
number1	0	int
crossoverRate	0	double
word	"47.650742;-2.0839379999999989"	string
coords	{string(2)}	string[]



Class Tests

To come up with the test cases for this application I created some flow diagrams to start covering certain aspects of the applications design.

These two diagrams were mocked up and lead to the inclusion of some of the test cases in the test plans below:

All JavaScript and JQuery within the application were tested independently of the project. This was done online using online tools such as JSFiddle and the Web Toolkit Online. Example below:

CONVERTER

- » Hexadecimal decimal
- » Hexadecimal color
- » Less to CSS
- » CSV to XML

ENCODER / DECODER

- » Base64
- » URL

FORMATTER

- » CSS
- » JSON
- » SQL
- » XML

GENERATOR

- » Lorem Ipsum
- » QR Code
- » Password

MINIFIER

- » CSS
- » JSON
- » SQL
- » XML

SECURITY

- » MD5 Digest

TESTER

- » JavaScript
- » JSON
- » Regular Expression
- » XML

JavaScript Tester

JavaScript code

```

1 window.initMap = function () { alert("lat1: " + lat1 + " long1: " + long1); }
2

```

Missing semicolon.

www.webtoolkitonline.com says:

Result=function () { alert("lat1: " + lat1 + " long1: " + long1); }

Prevent this page from creating additional dialogs.

OK

Execute Copy Clear

Elements Console Sources Network

top

```

▼ [document, context: document]
  ▼ 0: document
    ► I0_1462962477730: Window
      URL: "http://www.webtoolkitonline.com/jav
    ► ZeroClipboardMovie_1: embed#ZeroClipboard
    ► activeElement: input#test
      aLinkColor: ""
    ► all: HTMLAllCollection[257]
    ► anchors: HTMLCollection[0]
    ► applets: HTMLCollection[0]
      baseURI: "http://www.webtoolkitonline.com
      bgColor: ""
    ► body: body
      characterSet: "UTF-8"
      charset: "UTF-8"
      childElementCount: 1
    ► childNodes: NodeList[2]
    ► children: HTMLCollection[1]
      compatMode: "CSS1Compat"
      contentType: "text/html"
      cookie: "_ga=GA1.2.102526886.1462962478; .
      currentScript: null
    ► defaultView: Window
      designMode: "off"
      dir: ""
    ► doctype: <!DOCTYPE html>
    ► documentElement: html
      documentURI: "http://www.webtoolkitonline
      domain: "www.webtoolkitonline.com"
    ► embeds: HTMLCollection[1]
    ► f37acd072c56c: Window
    ► fb_xdm_frame_http: Window
    ► fb_xdm_frame_https: Window
      fgColor: ""
    ► firstChild: <!DOCTYPE html>
    ► firstElementChild: html
    ► fonts: FontFaceSet

```

Run Update Fork Tidy Collaborate Embed

```

1 <div id="map"></div>

```

```

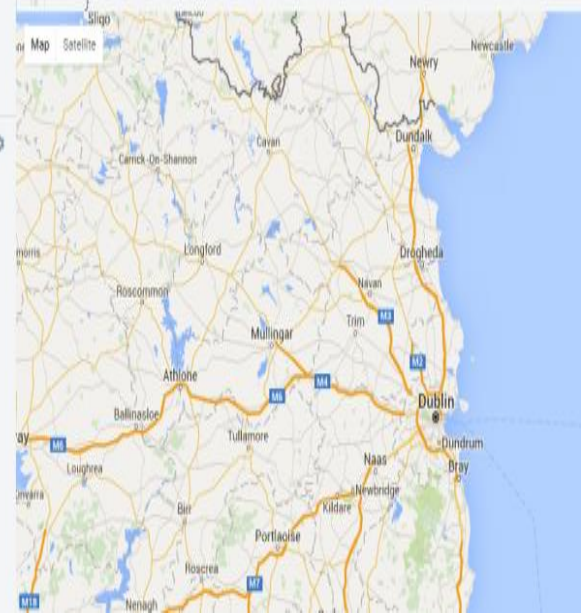
HTML
max-height:100%;
}
#map-canvas {
height: 90%;
margin: 0px;
padding: 0px;
float:left;
}
#sidebar{
background-color:grey;
width:50%;
height:100%;
float:left;
}

```

```

JAVASCRIPT
var map;
function initialize() {
var mapOptions = {
zoom: 8,
center: new google.maps.LatLng(53.397, -6.644)
};
map = new google.maps.Map(document.getElementById('map'),
mapOptions);
}
google.maps.event.addListener(window, 'load', initialize);

```



Test Plans

ID#	Page	Test Condition	Expected Result	Procedure/ Steps	Actual Result	Pass/ Fail	Comments
1	All Pages	Layout Consistency: Links	Each page has an identical nav-bar consisting of: • Site Name, • Home Link, • About Link, • Contact Link, • Register Link, • Login Link	1: Start Application 2: Visually inspect nav-bar for the link options.	Each page has an identical nav-bar consisting of: • Site Name, • Home Link, • About Link, • Contact Link, • Register Link, • Login Link	Pass	N/A
2	All Pages	Layout Consistency: Links	Clicking the links in nav-bar brings you to correct page • Site Name (Default.aspx) • Home Link (Default.aspx) • About Link (About.aspx) • Contact Link (Contact.aspx) • Register Link (Register.aspx) • Login Link (Login.aspx)	1: Start Application 2: Click on the nav-bar link options. 3: Verify navigation occurs to new page	Clicking the links in nav-bar brings you to correct page • Site Name (Default.aspx) • Home Link (Default.aspx) • About Link (About.aspx) • Contact Link (Contact.aspx) • Register Link (Register.aspx) • Login Link (Login.aspx)	Pass	N/A
3	All Pages	Layout Consistency: Background	Each page has an identical background image. • Default.aspx • About.aspx • Contact.aspx • Login.aspx • Register.aspx	1: Start Application 2: Click on the nav-bar link options. 3: Visually verify each page has the same background image	Each page has an identical background image. • Default.aspx • About.aspx • Contact.aspx • Login.aspx • Register.aspx	Pass	N/A
4	All Pages	Layout Consistency: Grammar	Correct spelling and punctuation throughout the application.	1: Start Application 2: Check all text visible on the site. 3: Check all punctuation visible on the site	Correct spelling and punctuation throughout the application.	Pass	N/A
5	Default	Home Page: Load	Home page loads without noticeable error/error messages	1: Start Application 2: Verify home page loaded successfully	Home page loads without noticeable error/error messages	Pass	N/A
5.1	Default	Home Page: Functionality	Google Map is displayed on home page	1: Start Application 2: Visually check Google Map is displayed in large div on the right	Google Map was displayed on home page	Pass	N/A
5.2	Default	Home Page: Functionality	Google Map defaults position is over Ireland	1: Start Application 2: Visually check Google Map is displayed in large div on the right 3: Check map position is over Ireland	Google Map defaults position is over Ireland	Pass	N/A
5.3	Default	Home : Functionality	When home page loads up, five textboxes will be in the div to left of map	1: Start application 2: Visually check div to the left contains five textboxes	Home page loads up and five textboxes were in the div to left of map	Pass	N/A
5.4	Default	Home Page: Functionality	The textboxes all have a placeholder with the text "Enter A Location"	1: Start application 2: Visually check div to the left contains five textboxes 3: Verify the text "Enter A Location" is within each textbox	All textboxes have a placeholder with the text "Enter A Location"	Pass	N/A
5.5	Default	Home Page: Functionality	"Verify Address" button is only button under textboxes on start up.	1: Start application 2: Visually check div to the left contains button with "Verify Addresses" text in it	"Verify Address" button is only button under textboxes on start up.	Pass	N/A
5.6	Default	Home Page: Functionality	Clicking "Verify Address" button displays user alert	1: Start application 2: Click "Verify Addresses" 3: Be prompted with alert box	Clicking "Verify Address" button displays user alert	Pass	N/A
5.7	Default	Home Page: Functionality	Alert Button Text is "Fill In The # Empty Textboxes In Red And Then Verify To Continue."	1: Start application 2: Click "Verify Addresses" 3: Be prompted with alert box 4: Verify text in alert is correct	Alert Button Text is "Fill In The # Empty Textboxes In Red And Then Verify To Continue."	Pass	N/A
5.8	Default	Home Page: Functionality	The # in the alert message, "Fill In The # Empty Textboxes In Red And Then Verify To Continue.", will be a number that matches up against the number of empty textboxes in the left div	1: Start application 2: Click "Verify Addresses" 3: Be prompted with alert box 4: Verify text in alert is "Fill In The # Empty Textboxes In Red And Then Verify To Continue." 5: Compare the amount of Red(empty) textboxes to the number in the alert.	The # in the alert message matches up against the number of empty textboxes in the left div	Pass	N/A
5.9	Default	Home Page: Functionality	Clicking into the autocomplete textbox and typing a letter will trigger the autocomplete listener to offer some options of addresses to choose.	1: Start application 2: Click into each textbox individually 3: Type a letter 4: Verify autocomplete works and offers several options beginning with that letter 5: Choose one of the options from the list 6: Verify that the textbox with that entry is now a dark shade of green	Clicking into the autocomplete textbox and typing a letter will trigger the autocomplete listener to offer some options of addresses to choose.	Pass	N/A
5.10	Default	Home Page: Functionality	Clicking one of the autocomplete options from the list will place a marker in that location on the Google Map.	1: Put a location in from TestID 5.9 2: Verify on Google Map that a marker is placed in the correct location you just chose.	Clicking one of the autocomplete options from the list placed a marker in that location on the Google Map.	Pass	N/A
5.11	Default	Home Page: Functionality	Filling in all textboxes successfully and hitting verify addresses will display two more buttons that users can see and use.	1: Fill all textboxes using the steps from TestID 5.9 2: Hit the "Verify Addresses" button 3: Verify two buttons appear under "Verify Addresses" 4: "Airbourne" and "Grounded" should be the text on the buttons.	Filling in all textboxes successfully and hitting verify addresses will display two more buttons that users can see and use.	Pass	N/A
5.12	Default	Home Page: Functionality	Pressing the newly created Airborne button will return a results div with the output and apply the markers onto the map connected via polylines.	1: Large "Results" div appears at bottom of page with optimised results. 2: Direct as the crow flies polylines are drawn onto the grid between markers	Pressing the newly created Airborne button will return a results div with the output and apply the markers onto the map connected via polylines.	Partial Fail	Div displayed but polylines didn't. Will be fixed by 16.05.16
5.13	Default	Home Page: Functionality	Pressing the newly created Airborne button will return a results div with the output and apply the markers onto the map by the correct roads.	1: Large "Results" div appears at bottom of page with optimised results. 2: Routes are displayed on the roads within the grid between the markers	Pressing the newly created Airborne button will return a results div with the output and apply the markers onto the map by the correct roads.	Partial Fail	Postback issue, page needs to be refreshed, then it works ok. To be fixed by 16.05.16
6	About	About Page: Load	About page loads without noticeable error or any error messages.	1: Start Application 2: Click "About" link 3: Verify page loads up with no errors, or error message	About page loads without noticeable error or any error messages.	Pass	N/A

6.1	About	About Page: Content	About page has correct spelling and grammar.	1: Start Application 2: Click "About" link 3: Read all visible text and verify no spelling, grammar mistakes	About page has correct spelling and grammar.	Pass	N/A
7	Contact	Contact Page: Load	Contact page loads without noticeable error/error messages	1: Start Application 2: Click "Contact" link 3: Verify page loads up with no errors, or error message	Contact page loads without noticeable error/error messages	Pass	N/A
7.1	Contact	Contact Page: Display	Check throughout the page for spelling and grammatical mistakes.	1: Start Application 2: Click "Contact" link 3: When page loads, visually inspect all text within page	Check throughout the page for spelling and grammatical mistakes.	Pass	N/A
7.2	Contact	Contact Page: Display	The contact page contain 4 labels, with the headings with the text Name, eMail, Subject Comments.	1: Start Application 2: Click "Contact" link 3: When page loads, visually inspect all label elements within page	The contact page contain 4 labels, with the headings with the text Name, eMail, Subject Comments.	Pass	N/A
7.3	Contact	Contact Page: Display	The contact page should contain 4 textboxes with placeholder text inside: Name: (Placeholder) Name: (Please Enter Your Name) eMail: (Please Enter Your e-Mail) Subject: (Please Enter A Subject) Comments: (Please Enter Your Comments)	1: Start Application 2: Click "Contact" link 3: When page loads, visually inspect all label and textbox elements within page	The contact page should contain 4 textboxes with placeholder text inside: Name: (Placeholder) Name: (Please Enter Your Name) eMail: (Please Enter Your e-Mail) Subject: (Please Enter A Subject) Comments: (Please Enter Your Comments)	Pass	N/A
7.4	Contact	Contact Page: Display	The contact page contains a button with the text "Submit Form"	1: Start Application 2: Click "Contact" link 3: When page loads, visually inspect the button element within page	The contact page contains a button with the text "Submit Form"	Pass	N/A
7.5	Contact	Contact Page: Display	Clicking the "Submit Button" with all textboxes empty displays four Red alert messages on the page: A Name Is Required. An eMail Address Is Required. A Subject Heading Is Required. Some Comments Are Required.	1: Start Application 2: Click "Contact" link 3: When page loads, click the "Submit Form" button	Clicking the "Submit Button" with all textboxes empty displays four Red alert messages on the page: A Name Is Required. An eMail Address Is Required. A Subject Heading Is Required. Some Comments Are Required.	Pass	N/A
7.6	Contact	Contact Page: Display	Adding input to each textbox, one at a time and re-submit the form after each entry. The messages will dissappear ad the data is entered.	1: Start Application 2: Click "Contact" link 3: Click "Submit Form" 4: Verify the four alert messages 5: One at a time enter your data into textbox, submitting after every entry, ensuring the message disappear	Adding input to each textbox, one at a time and re-submit the form after each entry. The messages will dissappear ad the data is entered.	Pass	N/A
8.0	Register	Register Page: Load	Register page loads without noticeable error/error messages	1: Start Application 2: Click "Register" link 3: Verify page loads up with no errors, or error message	Register page loads without noticeable error/error messages	Pass	N/A
8.1	Register	Register Page: Display	To register users are asked to supply an email address and a password. The password must be entered twice for verification. The user then has the option to select a "Register" button. Users should also be given the option to navigate to the Log In page if they already have an account.	1: Start Application 2: Click "Register" link 3: Fill in details and click "Register" button 4: Verify there is a link to login page	To register users are asked to supply an email address and a password. The password must be entered twice for verification. The user then has the option to select a "Register" button. Users should also be given the option to navigate to the Log In page if they already have an account.	Pass	N/A
8.2	Register	Register Page: Funtionality	When no email is entered and the register button is clicked an error appears asking for email	1: Start Application 2: Click "Register" link 3: Enter no details, click register. Verify error messages.	When no email is entered and the register button is clicked an error appears asking for email	Pass	N/A
8.3	Register	Register Page: Funtionality	When invalid email format is entered and register button is clicked an error appears explaining format	1: Enter a non-valid email ie "TestName" 2: Click "Register" 3: Verify email error message.	When invalid email format is entered and register button is clicked an error appears explaining format	Pass	N/A
8.4	Register	Register Page: Funtionality	When no password is entered or an insufficient password and register button is clicked an error appears asking for password	1: Enter a letter for a password ie "A" 2: Click "Register" 3: Verify the password error message. 4: Enter no password 5: Verify the password error message.	When no password is entered or an insufficient password and register button is clicked an error appears asking for password	Pass	N/A
8.5	Register	Register Page: Funtionality	When the two password fields do not match an error message will display to tell user that passwords must match.	1: Enter all details but put two different words in the passwords fields 2: Click "Register" 3: Verify error message.	When the two password fields do not match an error message will display to tell user that passwords must match.	Pass	N/A
8.6	Register	Register Page: Database	New user account created and appears in database	1: Navigate to the AspNetUsers Table in the LocalDB and verify it has been updated with new user	New user account created and appears in database	Pass	N/A
9.0	Login	Login Page: Load	Login page loads without noticeable error/error messages	1: Start Application 2: Click "Login" link 3: Verify page loads up with no errors, or error message	Login page loads without noticeable error/error messages	Pass	N/A
9.1	Login	Login Page: Functionality	Entering log in details, email and password, users the option for their details to be remembered	1: Navigate to "Login" page 2: Verify the options to log in 3: Tick remember details	Entering log in details, email and password, users the option for their details to be remembered	Pass	N/A
9.2	Login	Login Page: Functionality	Auto complete should work for email	1: After successfully logging in 2: Log out 3: Attempt to login again 4: Verify autocomplete remmebers recent email address	Auto complete should work for email	Pass	N/A
9.3	Login	Login Page: Functionality	Users should be given the option to go to register page if they do not have an account.	1: Attept to login with no account 2: Verify there is a link to register page	Users should be given the option to go to register page if they do not have an account.	Pass	N/A
9.4	Login	All Pages	When logged in there is a label in the navbar stating "Hello <username/email>"	1: Login to Application 2: Visually inspect navbar for username/email	When logged in there is a label in the navbar stating "Hello <username/email>"	Pass	N/A
10.0	Admin	Admin Page: Load	When logged in as admin additional "Admin" link is available for admin to use	1: Login with admin credentials 2: On navbar verify additional "Admin" is now available	When logged in as admin additional "Admin" link is available for admin to use	Pass	N/A
10.1	Admin	Admin Page: Load	Admin page loads without noticeable error/error messages	1: Start Application 2: Click "Admin" link 3: Verify page loads up with no errors, or error message	Admin page loads without noticeable error/error messages	Pass	N/A
10.2	Admin	Admin Page: Functionality	When logged in as Admin and on the Admin.aspx a GridView (table) with all users queries and comments should be available	1: Log in as Admin user 2: Click on "Admin" link 3: Visually confirm all users queries from the database are displayed on the page	When logged in as Admin and on the Admin.aspx a GridView (table) with all users queries and comments should be available	Pass	N/A
10.3	Admin	Admin Page: Functionality	The option to sort the table by column is available	1: Clicking on any heading on the GridView will sort the table by that column	The option to sort the table by column is available	Pass	N/A
10.4	Admin	Admin Page: Functionality	The option to edit and record in the table is available	1: Clicking on edit on any line in the GridView will enable that row to be edited	The option to edit and record in the table is available	Pass	N/A
10.5	Admin	Admin Page: Functionality	All user emails are links to default email provider and create a mail with the subject line "The Route Optimising Team"	1: Click on any email address in any row 2: Default email provider will open with subject line of "The Route Optimising Team"	All user emails are links to default email provider and create a mail with the subject line "The Route Optimising Team"	Pass	N/A

Customer testing

User acceptance testing was performed on the application during the build of it. The initial thought now is to add a feedback section into the application that users can comment on the good or the bad of the application before we charge for it. This will lead to continuous improvement on our side and hopefully remove and bugs before customers commit to paying for the application.

UAT 1 was performed by Jason McGuinness.

The Route Optimising SideKick Test Scripts							
Test script name:		Smoke Test For Web Application					
Test script developed by:		Gavin Gaughran			Date: 10.04.16		
Test script completed by:		Jason McGuinness			Date: 15.04.16		
Tester's role:		Field Service Engineer, Wincor-Nixdorf					
ID #	Page	Scenario	Expected Result	Process to complete	Actual result	Pass/Fail	Comments
1	Register	Locate the registration page and register as a new user	Locate "Register" page on website and successfully registers for a new account.	1. Navigate to "Register" page 2. Complete all required fields 3. Submit registration	Located register page on website and successfully registered new account.	Pass	
2	Default	Log out of the application	Log out of application	1. Navigate to Log Off button 2. Select it	Logged out easily	Pass	
3	Login	Locate login link/page and log into website using their account	Find the login page/link on application and successfully login Verify email/username is displayed	1. Navigate to log in page 2. Complete all required fields 3. Submit details	Found login link on application and successfully logged in. Verified email/username is displayed	Pass	
4	Default	Open homepage, view text and images and check if any available buttons are working	Locate homepage, can view text, can view and interact with Google Map. After clicking on the "Contact" link they are redirected to a new page that displays a form to contact the website	1. Navigate to homepage 2. View content on homepage 3. Select button for contact us form	Opened homepage, viewed text, can view and interact with Google Map. After clicking on the "Contact" link was redirected to a new page that displays a form to contact the website	Pass	
5	Contact	Complete a contact form and submit it to the database	Complete each required field on contact form and submit it successfully. Message will appear on submission that it was successful	1. Click on contact us form button on homepage 2. Complete each field in form 3. Click the "Submit Form"	Complete each required field on contact form and submit it successfully. Message will appear on submission that it was successful	Pass	
6	About.aspx	Read the "About Us" page	View the information about the site in the "About" page	1. Navigate to About us page 2. Look over content of page 3. Provide feedback on how it looks	View the information about the site in the "About" page	Fail	Was not created at date of test
7	All pages	Locate the Admin link	Should not be able to find the link if not logged in as Admin	1: Search all pages to try access Admin page	Could not locate link	Pass	
8	Default	Create entries in Google Maps textboxes and request some results from algorithm as a guest	Should only allow legit addresses and should return results and display map with new route plan	1: Open home page 2: Enter some addresses to textboxes 3: Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass	
9	Default	Create entries in Google Maps textboxes and request some results from algorithm when logged in.	Should only allow legit addresses and should return results and display map with new route plan	1: Open home page 2: Enter some addresses to textboxes 3: Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass	

UAT 2 was performed by Rebecca Solomon.

The Route Optimising SideKick Test Scripts							
Test script name:		Smoke Test For Web Application					
Test script developed by:		Gavin Gaughran			Date: 10.04.16		
Test script completed by:		Rebecca Solomon			Date: 19.04.16		
Tester's role:		Dispatch Tem Lead, Wincor-Nixdorf					
ID #	Page	Scenario	Expected Result	Process to complete	Actual result	Pass/Fail	Comments
1	Register	Locate the registration page and register as a new user	Locate "Register" page on website and successfully registers for a new account.	1. Navigate to "Register" page 2. Complete all required fields 3. Submit registration	Located register page on website and successfully registered new account.	Pass	
2	Default	Log out of the application	Log out of application	1. Navigate to Log Off button 2. Select it	Logged out easily	Pass	
3	Login	Locate login link/page and log into website using their account	Find the login page/link on application and successfully login Verify email/username is displayed	1. Navigate to log in page 2. Complete all required fields 3. Submit details	Found login link on application and successfully logged in. Verified email/username is displayed	Pass	
4	Default	Open homepage, view text and images and check if any available buttons are working	Locate homepage, can view text, can view and interact with Google Map. After clicking on the "Contact" link they are redirected to a new page that displays a form to contact the website	1. Navigate to homepage 2. View content on homepage 3. Select button for contact us form	Opened homepage, viewed text, can view and interact with Google Map. After clicking on the "Contact" link was redirected to a new page that displays a form to contact the website	Pass	
5	Contact	Complete a contact form and submit it to the database	Complete each required field on contact form and submit it successfully. Message will appear on submission that it was successful	1. Click on contact us form button on homepage 2. Complete each field in form 3. Click the "Submit Form"	Complete each required field on contact form and submit it successfully. Message will appear on submission that it was successful	Pass	
6	About.aspx	Read the "About Us" page	View the information about the site in the "About" page	1. Navigate to About us page 2. Look over content of page 3. Provide feedback on how it looks	View the information about the site in the "About" page	Fail	Was not created at time of test
7	All pages	Locate the Admin link	Should not be able to find the link if not logged in as Admin	1: Search all pages to try access Admin page	Could not locate link	Pass	
8	Default	Create entries in Google Maps textboxes and request some results from algorithm as a guest	Should only allow legit addresses and should return results and display map with new route plan	1: Open home page 2: Enter some addresses to textboxes 3: Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass	
9	Default	Create entries in Google Maps textboxes and request some results from algorithm when logged in.	Should only allow legit addresses and should return results and display map with new route plan	1: Open home page 2: Enter some addresses to textboxes 3: Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass	

UAT 3 was performed by Sharon O’Doherty.

ID #	Page	Scenario	Expected Result	Process to complete	Actual result	Pass/Fail	Comments																				
<table border="1"> <thead> <tr> <th colspan="4">The Route Optimising SideKick Test Scripts</th> </tr> </thead> <tbody> <tr> <td>Test script name:</td> <td colspan="3">Smoke Test For Web Application</td> </tr> <tr> <td>Test script developed by:</td> <td>Gavin Gaughran</td> <td>Date:</td> <td>10.04.16</td> </tr> <tr> <td>Test script completed by:</td> <td>Sharon O’Doherty</td> <td>Date:</td> <td>25.04.16</td> </tr> <tr> <td>Tester’s role:</td> <td colspan="3">Student NCIRL, Regular Vacation Taker</td> </tr> </tbody> </table>								The Route Optimising SideKick Test Scripts				Test script name:	Smoke Test For Web Application			Test script developed by:	Gavin Gaughran	Date:	10.04.16	Test script completed by:	Sharon O’Doherty	Date:	25.04.16	Tester’s role:	Student NCIRL, Regular Vacation Taker		
The Route Optimising SideKick Test Scripts																											
Test script name:	Smoke Test For Web Application																										
Test script developed by:	Gavin Gaughran	Date:	10.04.16																								
Test script completed by:	Sharon O’Doherty	Date:	25.04.16																								
Tester’s role:	Student NCIRL, Regular Vacation Taker																										
1	Register	Locate the registration page and register as a new user	Locate "Register" page on website and successfully registers for a new account.	1. Navigate to "Register" page 2. Complete all required fields 3. Submit registration	Located register page on website and successfully registered new account.	Pass																					
2	Default	Log out of the application	Log out of application	1. Navigate to Log Off button 2. Select it	Logged out easily	Pass																					
3	Login	Locate login link/page and log into website using their account	Find the login page/link on application and successfully login Verify email/username is displayed	1. Navigate to login page 2. Complete all required fields 3. Submit details	Found login link on application and successfully logged in. Verified email/username is displayed	Pass																					
4	Default	Open homepage, view text and images and check if any available buttons are working	Locate homepage, can view text, can view and interact with Google Map. After clicking on the "Contact" link they are redirected to a new page that displays a form to contact the website	1. Navigate to homepage 2. View content on homepage 3. Select button for contact us form	Opened homepage, viewed text, can view and interact with Google Map. After clicking on the "Contact" link was redirected to a new page that displays a form to contact the website	Pass																					
5	Contact	Complete a contact form and submit it to the database	Complete each required field on contact form and submit it successfully Message will appear on submission that it was successful	1. Click on contact us form button on homepage 2. Complete each field in form 3. Click the "Submit Form"	Complete each required field on contact form and submit it successfully. Message will appear on submission that it was successful	Pass																					
6	About.aspx	Read the "About Us" page	View the information about the site in the "About" page	1. Navigate to About us page 2. Look over content of page 3. Provide feedback on how it looks	View the information about the site in the "About" page	Pass																					
7	All pages	Locate the Admin link	Should not be able to find the link if not logged in as Admin	1. Search all pages to try access Admin page	Could not locate link	Pass																					
8	Default	Create entries in Google Maps textboxes and request some results from algorithm as a guest	Should only allow legit addresses and should return results and display map with new route plan	1. Open home page 2. Enter some addresses to textboxes 3. Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass																					
9	Default	Create entries in Google Maps textboxes and request some results from algorithm when logged in.	Should only allow legit addresses and should return results and display map with new route plan	1. Open home page 2. Enter some addresses to textboxes 3. Submit entries to code and wait for results	Only legit addresses were entered and return results were displayed on map with new route plan	Pass																					

Evaluation

Only minimal evaluations of the JS have been carried out to date. Some testing for the autocomplete text box was performed using repetitive search entries.

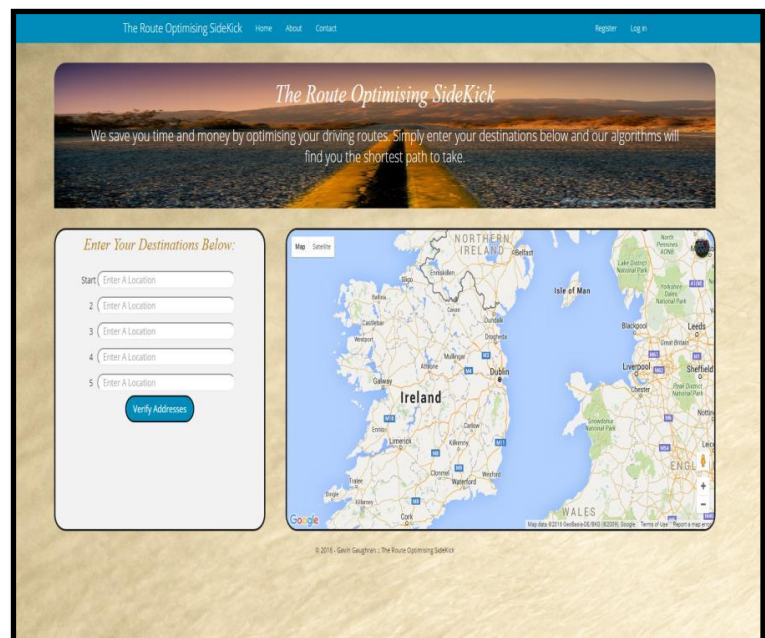
The feedback I received back pretty positive from Rebecca, Sharon and Jason was that they have agreed that if I can host this application they will perform more testing for me to improve. The testing was performed with the GUI not finished so they are welcoming testing the final product when it looks a lot better.

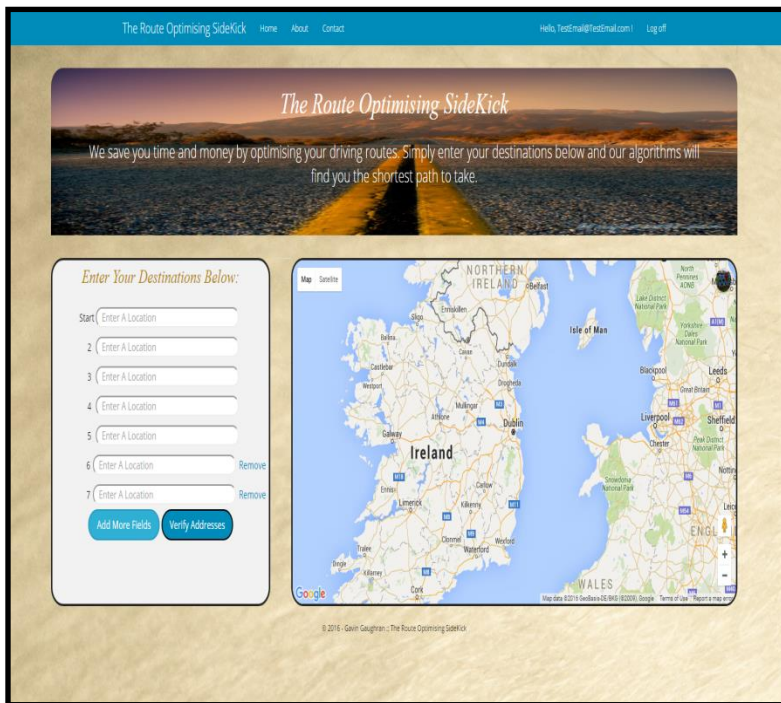
Graphical User Interface (GUI) Layout

Following on from Interface Requirements, below are some examples of the key features within the web application.

Home Page (Guest View)

This the home page as a Guest will see it. The user here has either not logged in or not registered. They user can see the Google Maps API and has the option to add 5 destinations into the application.





The user login panel is in the top right where registered users can login in to their account. If the user logs in to the application, they are brought to the below screen.

Home Page (Logged In)

The user has now logged in and receives the Welcome <email> line in the nav-bar of the application.

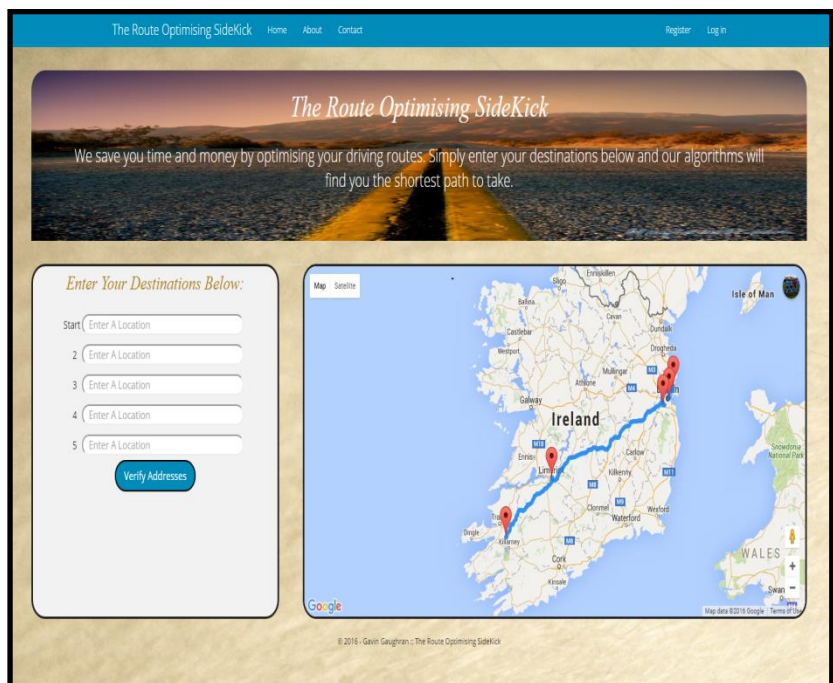
The user also can see the “Add More Fields” button, enabling the ability to add up to ten destinations into the route optimization tool.

Route Optimised Page

The user has now chosen some destination and hit the Optimise button. The application returns the results with the option to print, add more destinations and re-calculate.

The map will be displayed with the route and also a list of “Optimized Results” is available for viewing.

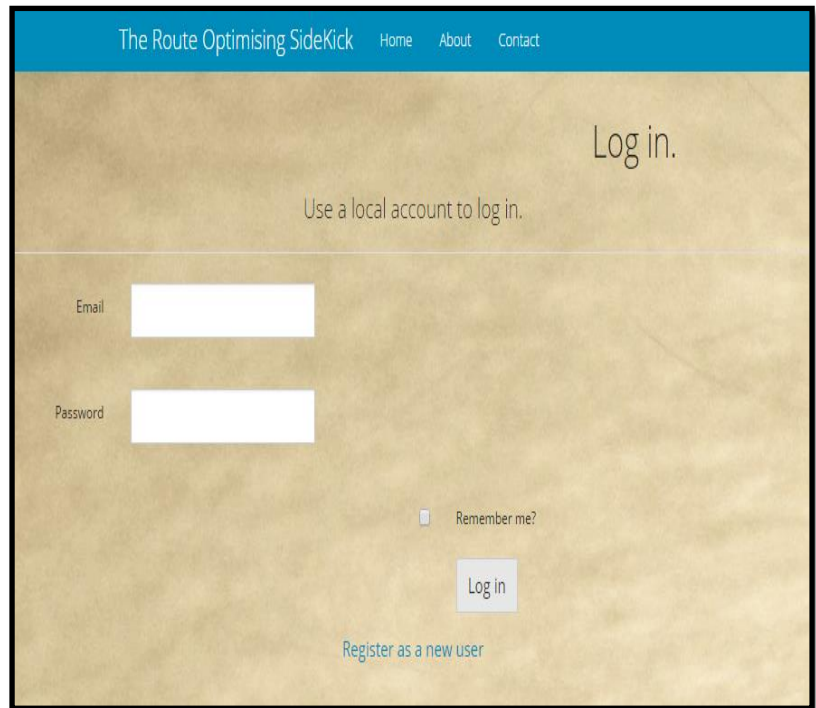
Print and Optimal result details to be completed by demo.



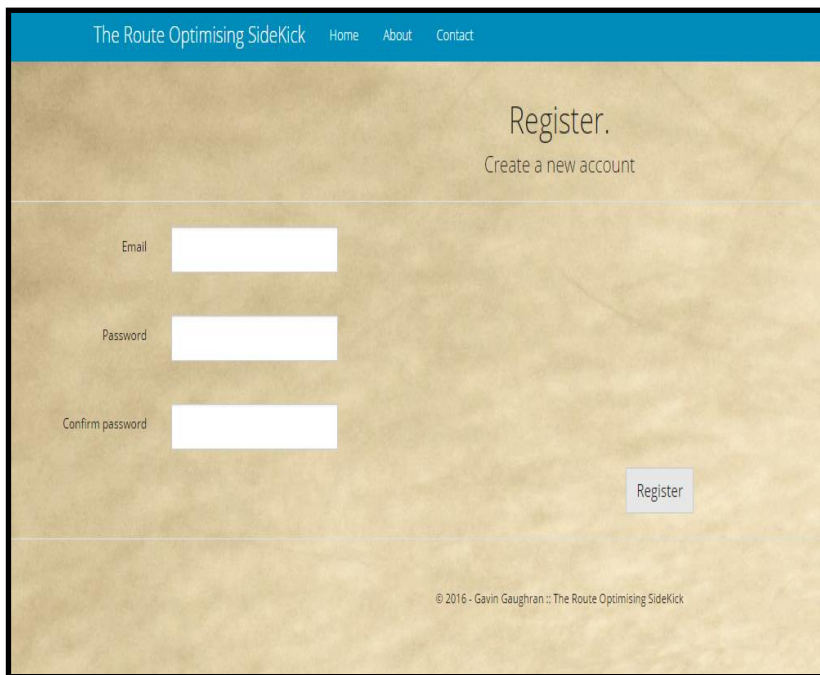
Log In Page

Located at the top right of every page there is a login button and once clicked it will bring the user to the page displayed to the right.

Logging in will allow additional destinations for the user.



The screenshot shows the 'Log In' page for 'The Route Optimising SideKick'. The page has a blue header with the site name and navigation links for 'Home', 'About', and 'Contact'. The main content area has a light brown background with the text 'Log in.' and 'Use a local account to log in.' Below this are two input fields for 'Email' and 'Password'. There is a 'Remember me?' checkbox and a 'Log in' button. At the bottom, there is a link for 'Register as a new user'.



The screenshot shows the 'Register' page for 'The Route Optimising SideKick'. The page has a blue header with the site name and navigation links for 'Home', 'About', and 'Contact'. The main content area has a light brown background with the text 'Register.' and 'Create a new account'. Below this are three input fields for 'Email', 'Password', and 'Confirm password'. There is a 'Register' button. At the bottom, there is a copyright notice: '© 2016 - Gavin Gaughran :: The Route Optimising SideKick'.

User Registration Page

Located at the top right of every page there is also a register button and once clicked it will bring the user to the page displayed on the left. This is the registration form page and users complete this to create an account. Once created the details will be stored in the database and users can login whenever they want to.

Admin Page

Below is the Admin page that only the admin(myself) of the site has access to. Within here I can view all the queries that have been sent to the site. The email links connect to the default email provider allowing a quick response and once responded to their is a checkbox section allowing me to modify the “Replied” to checkbox.



The screenshot shows the Admin Page of a website. The page has a blue header with the site name 'The Route Optimising SideKick' and navigation links for Home, About, Contact, and Admin. The user is logged in as 'Hello, Gaving30@gmail.com' and can click 'Log off'. The main content area is titled 'Admin Page' and contains a table of queries. The table has columns for Id, Name, eMail, Subject, Comments, Replied, and PostDate. The data in the table is as follows:

Id	Name	eMail	Subject	Comments	Replied	PostDate
Edite 1	Gav	mail@mail.mail	Subject	Comments	<input type="checkbox"/>	
Edite 2	test	t@t	y	comm	<input type="checkbox"/>	
Edite 3	test	t@t	y	comm	<input type="checkbox"/>	
Edite 4					<input type="checkbox"/>	
Edite 5	Gav	mail@mail.mail	Subject	C	<input type="checkbox"/>	
Edite 19	Guest	Guest@email.ie	Its now working	This just worked	<input type="checkbox"/>	
Edite 20	newTest	new@new.new	New Test	Date?	<input type="checkbox"/>	05/06/2016 00:00:00
Edite 21	Late	mail@mail.ie	Replied?	Replied Check NO	<input type="checkbox"/>	05/06/2016 00:00:00
Edite 22	o	o@o.ie	F	F	<input type="checkbox"/>	05/06/2016 00:00:00

At the bottom of the page, there is a copyright notice: © 2016 - Gaving Gaughran :: The Route Optimising SideKick

Conclusions

My sights were set on improving my overall understanding of programming and its concepts. For this personal goal I have achieved this, coding is what I want to be doing on a daily basis and doing this project felt like this, and I really loved it! The pressure was a lot with working full time but that was my choice so is not an excuse. Choosing to use new languages to me, I learned a lot about ASP.NET and C# and used JQuery a lot and for the first time also. They are some new languages I can now add to my CV. JQuery is a very quick, really smooth tool and I would recommend learning it as it improves sites by adding dynamic controls to web applications.

The algorithm itself, I learned that I will not be creating a brand new algorithm myself, just yet, but there were so many choices out there for types of crossover and types of distance formulas to use. Putting this together and seeing them fit, and eventually work, was amazing and knowing the way I have them together and with my application in the way it is structured is totally unique.

The realisation that the TSP is not intended to work on a map did hit me a long time after starting, as it can't take into account roads and other obstacles. All examples I looked at and learned from were using a basic web forms page, a basic graph that the points connect via a straight line. That's why I introduce polylines into the map. That was why I chose to use the Haversine formula as it takes its measurements based on a spherical shape and has allowances for curvature.

Further development or research

There are so many ideas I have now for this application. I always keep myself involved and I am interested in new IT technologies so I plan to learn Angular JS over the summer and can then introduce it to the application.

My next step when I am finished college is to continue improving this application as I think it will be, at least, a great addition to my portfolio. With also having interest in phone applications I am planning on creating a mobile application that could link in with this main web application to widen its overall scope.

The further development to the application will depend on my feedback, once I am 100% happy with the application I will encourage users to interact with it and provide me feedback. I think the only way to get real feedback is to host this and put my feedback form on the site, so people can submit real feedback.

Long term, the possibility of linking in with some advertises to put advertisements on the page is a possibility and maybe interacting with some popular restaurants/chains/tourist attractions for personal users to get some ideas of where to visit. I want to set up my database to store a list of the most visited addresses and store this and possibly create monthly lists of the most popular attractions.

Appendix

Project Proposal

Objective

The first objective of the project is the creation of a Web Application to display the results of algorithms for the travelling salesman problem (TSP). Algorithm comparisons will be performed using algorithms such as Brute Force, Greedy and Genetic Algorithms. The use of these algorithms will then be presented in an ASP.NET web application. The application will allow users to choose geographical locations on a Google map and the algorithms will work out the best route to take to reach all the chosen locations. This can be used for personal use when travelling to several locations or on a larger scale for distributive purposes.

Functionality within the application will include a user login. Once logged in, the ability to customize the home page, and chose from a simple drop down menu for regularly chosen locations. A Google maps API will allow users to choose locations easily and be used to display the algorithms results.

My second objective is to advance on the creation of a web application with the implementation of the web application to a mobile application. Similar web applications do already exist. My research into users experiences and opinions on the most popular existing applications, such as Road Warrior revealed *“Not enough stops I do over 30 stops a day and this app does not allow even 10”* I plan to allow more than 10 locations to be entered for users. Another popular app is BestRoute received the feedback *“I would love to see a “filing cabinet” with all of my customer job locations and simply be able to drag & drop from my filing cabinet into a route for that day”*, for this I want to create a “Favourites” section for this reason, users should not need to enter the same locations, daily or weekly, they should be able to choose the location from a list of regularly used locations.

An objective that will require further research will be how to integrate the application to mobile devices. All going good and all objectives been met, *The Route Optimising SideKick* will have a fully functioning web and mobile application available to users.

Background

There are several reasons why I chose to create an application such as this, both from a personal and business perspective. Firstly, throughout my college years I thoroughly enjoy math, algorithms and coding. AI was my favourite subject, so I was trying to create an application that would let me work on improving my algorithm/coding skills but would also be useful.

The second reason was a friend of mine rented a car and wanted to travel around Ireland as he is from France. He planned to view several cities and knew the locations he wanted to go, entering all the locations into Google maps it puts the travel route in the order as you enter the details so he spent a very long time trying to figure out what the best route would be to take so I thought this could be a very helpful application for such an instance.

Finally, I was working in an IT company as a field repair engineer and on some days we would work a Rota so that one of the engineers would just do deliveries of parts for cost saving purposes. For example a driver could be given 10 parts at the start of the day and told to drop them off at several locations and an application like this could have let each driver plan their route exactly, using the most time-efficient and fuel-efficient route to take.

In all aspects of a busy corporate or personal life, I believe that preparation is vital; I therefore believe having this application will benefit distribution companies wishing to optimize their routes. For people wishing to travel who want to plan ahead, or don't know

the area they are travelling in, this will reduce their travel time and allow them to increase the time they have in each location.

Scope

The fact that this application can be used for corporate or personal reasons, means the user might not necessarily be overly literate on PC's or mobile devices. To increase the '*user scope*' across these two areas, the idea for the application UI will be to make this as easy to use as possible for all users.

The 'tech scope' is to make this app available to desktops PC's, laptops, and then increase this to mobile devices such as tablets and phones, thus increasing the popularity and available usage of this app.

Technical Approach/Details

The web application will be developed in Visual studio in ASP.NET and C#. The frontend consists of the ASP.NET framework, version 5.0, Web API frameworks and will contain HTML elements throughout the Web pages. The backend will be managed using MVC 6, C# and HTML server controls. The server-based code connections allow interactions between our front and back end. A database will store the data that our site requires and we will use either ADO.NET classes or LINQ queries to extract the relevant information. Google maps API will be inserted into the web application UI, creating an easy-to-use frontend for the user. At the moment the algorithms themselves will be done using C# and GAF (Genetic Algorithm Framework) which is a .Net/Mono assembly. Alternatively Java and C++ could be implemented dependent on time constraints.

To improve the appearance of the application, CSS will be used to design. The reasons behind using CSS are for consistency across all pages; I can design the page and use that consistent style across all pages within the application.

For design features I will use JavaScript, which will allow users to customize their home page. Using the JS library JQuery will increase the look of the web pages and add some slick features to the pages such as pop-out windows. Ajax will also be used on the client-side to create some asynchronous connectivity to the server, possibly when the results for the algorithm are sent back to the user the whole page may not need to be refreshed and just the map could be returned with the optimal route.

The final addition to the application will be the algorithms themselves and I am currently looking into algorithms and the best implementation of them. I want the user to enter the details they want to enter into the algorithm, so I will use either a database or an XML document to store these dynamic inputs.

The research into integrating this application to mobiles will look into making the webpages responsive to function correctly on mobile applications, another possibility would be using bootstrap, or finally a separate application could be developed and could possibly sync up with the web application.

Special Resources Required

Development of the web application will require Visual Studio. Once developed, the user will need a desktop PC or laptop to access the web application. Once the web application is available and functioning correctly, the mobile application can then be implemented and smaller handheld devices such as tablets and phones can be used.

Project Plan

Main phases throughout project:

Phase	Description of Work	Start and End Dates
Phase 1.0	Documentation and Planning	01-09-15 – 25-05-15
Phase 2.0	Create basic Web Application	01-10-15 – 15-10-15
Phase 3.0	Create Database for Application	15-10-15 – 08-11-15
Phase 4.0	Implement Google API	10-11-15 – 01-12-15
Phase 5.0	Implement Algorithms	01-12-15 – 01-03-16
Phase 6.0	Design Web Application	01-03-16 – 08-03-16
Phase 7.0	Integrate to Mobile Application	10-03-16 – 18-05-16

Phase breakdown below:

Phase	ACTIVITY	Start	Finish	Predecessor
1.0	Documentation and Planning			
1.1	Project Proposal/ Plan	01-09-15	02-10-15	
1.2	Journal	01-09-15	25-05-16	
1.3	Requirements Spec	13-10-15	06-11-15	
1.4	Project Analysis & Design	06-11-15	04-12-15	
1.5	Final Project Hard Copies Docs	01-09-15	11-05-16	
2.0	Create basic Web Application			
2.1	Create basic web application in VS	01-10-15	15-10-15	
3.0	Create Database for Application			
3.1	Create and connect DB	15-10-15	22-10-15	
3.2	Research data to input to DB	22-10-15	01-11-15	

3.3	Input some data for test purposes	01-11-15	08-11-15	
4.0	Implement Google API			
4.1	Research Google API	10-11-15	15-11-15	
4.2	Implement Google API	15-11-15	01-12-15	
5.0	Implement Algorithms			
5.1	Study and write algorithm code	01-12-15	20-02-16	
5.2	Implement to Web Application	20-02-16	01-03-16	
6.0	Design Web Application			
6.1	Design web application	01-03-16	10-03-16	
6.2	Add any JS, AJAX extras	01-03-16	08-03-16	
7.0	Integrate to Mobile Application			
7.1	Develop website to mobile Either responsive or create app	10-03-16	18-05-16	

Evaluation

Once the web application is developed I will do up a small test plan covering all buttons and functionality within the web application, as I want to “test early and often”.

The database connection will be tested by entering data into the database using the web application. All possible database controls will be tested including inserting, deleting, reading and writing to and from the database.

All the above web applications tests will be performed by me. Once I am satisfied the web application and database are running steadily and pass my initial unit testing I will work on the Algorithm.

When the algorithm is implemented I have a colleague in work and a colleague in college that I have organized to test for a few hours each. I will ask them to try and “break” my application and let them provide me with feedback on a test completion report I will give them.

For the design I have a family member who was an art student and designer that I will ask to give some opinions on the layout and color scheme of the application.

Project Analysis Design

Introduction

Purpose of the Product Design Specification Document

The Product Design Specification documents and tracks the necessary information required to effectively define architecture and system design in order to give the development team guidance on architecture of the system to be developed for *The Route*

Optimising SideKick web application. This Product Design Specification document was created during the Planning Phase of my project.

General Overview and Design Guidelines/Approach

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

Assumptions

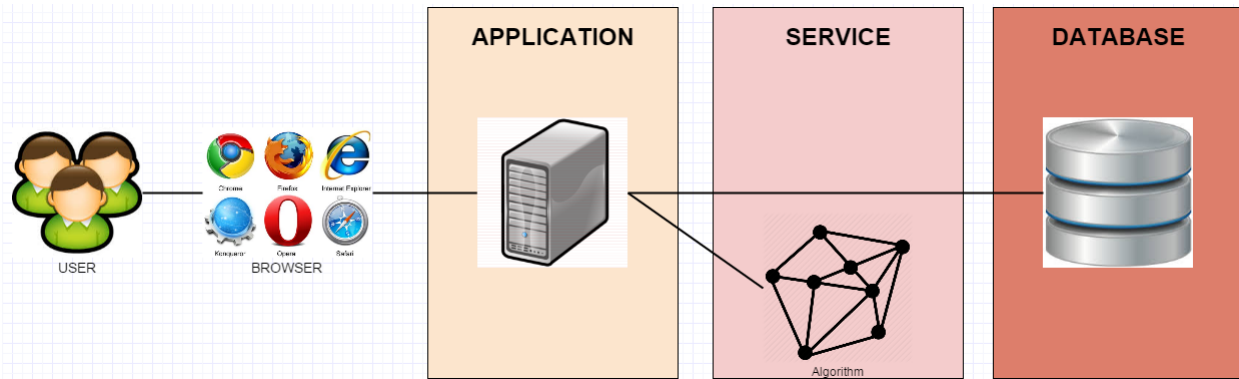
No.	Assumption	Status	Comment
1	The application allows users 5 free inputs as a 'guest'	Confirmed – Will remain	This will remain an assumption for the foreseeable future.
2	The application allows users to register and create an account.	Confirmed – Will remain	This will remain an assumption for the foreseeable future.
3	The application allows registered users more than 5 inputs.	Confirmed	This is confirmed but the exact number is to be confirmed during development.

Constraints

No.	Constraint	Status	Comment
1	The application only allows 5 free inputs for 'guests'	Confirmed – Will remain	This will remain an assumption for the foreseeable future.
2	Guest cannot save Favourites	Confirmed – Will remain	Registration required to save details/favourites.
3	There will be a max input of locations for registered guests.	To Be Confirmed	The final figure is yet to be confirmed.

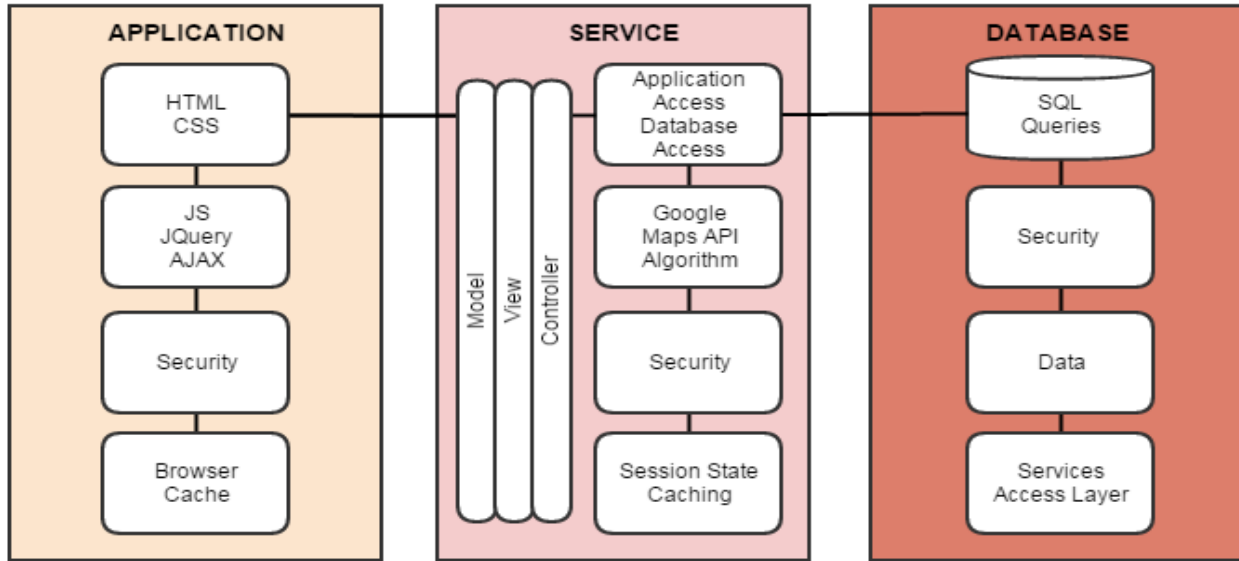
Architecture Design

Overall Architecture



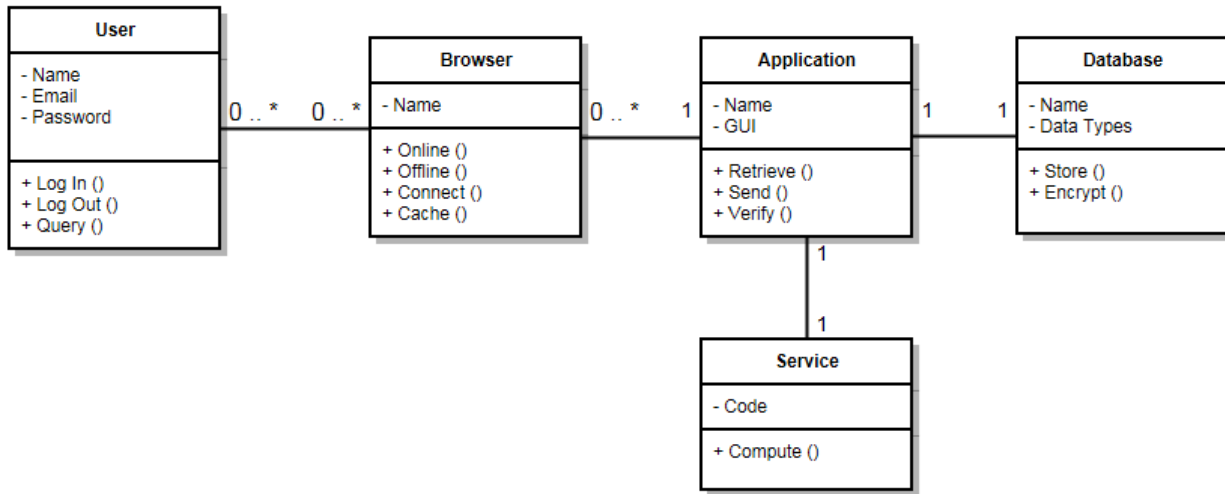
This is a high level description of the overall architecture. On the far left it starts with the user accessing their web browser. The application is opened using the chosen browser. Then application can then directly connect to the database or to its services. It's totally dependent on the users ask as to what it connects to, for example, the User login would connect to the dataset and would not need the Service section and if a guest ran a Route plan they would not need to connect to the Database so just the Service section would be accessed.

System Architecture

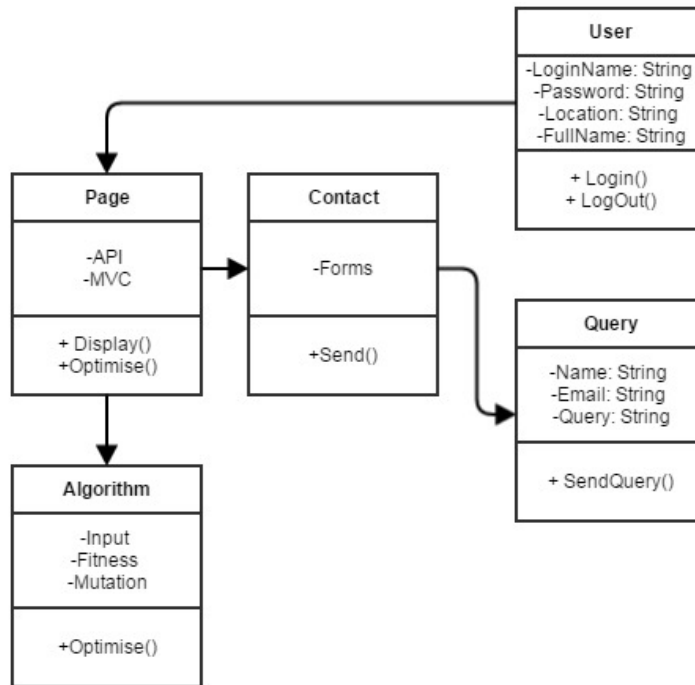


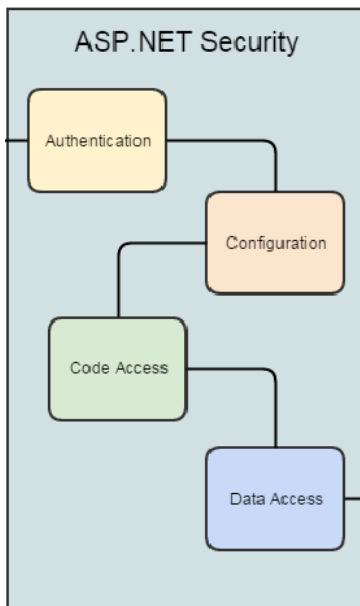
The architecture above is the System itself in some more details. The frontend system Application contains the HTML and CSS web forms, they receive extra functionality using JS, JQuery or Ajax. The forms will contain security within the browser and within the forms when it comes to passwords. The browser will contain a cache that links to the forms. The services behind the page include Access layers, algorithms and API's. Internal caching in the form of Sessions is built into the system. The backend database has obvious security features protecting the data within in it.

Overall Architecture Class Diagram



System Architecture Class Diagram





Security Architecture

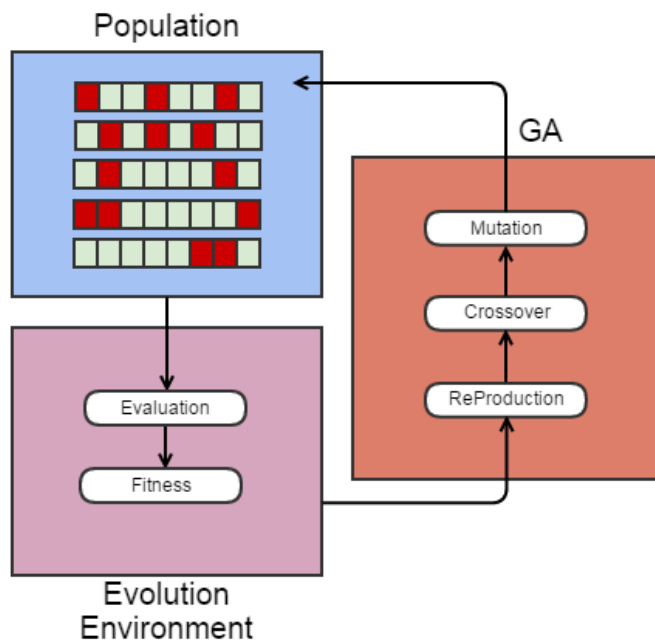
ASP.NET applications can come with a host of features to secure their web applications but as developers, plenty of attention and time should be spent ensuring some basic features are enabled such as:

- Authentication and Authorization
- Application configuration
- Data validation, Code Access ,
- User password policies and hashing.

Performance

The performance of the algorithm will be done at a later date. A theoretical analysis can be performed using the Big O notation and this would represent the complexity of the algorithm. This could not even be considered until full understanding of the algorithm is complete during development, which is due to begin mid-December.

Genetic Algorithm



The population of the Genetic Algorithm is the locations on the map that are entered into the algorithm to be optimized.

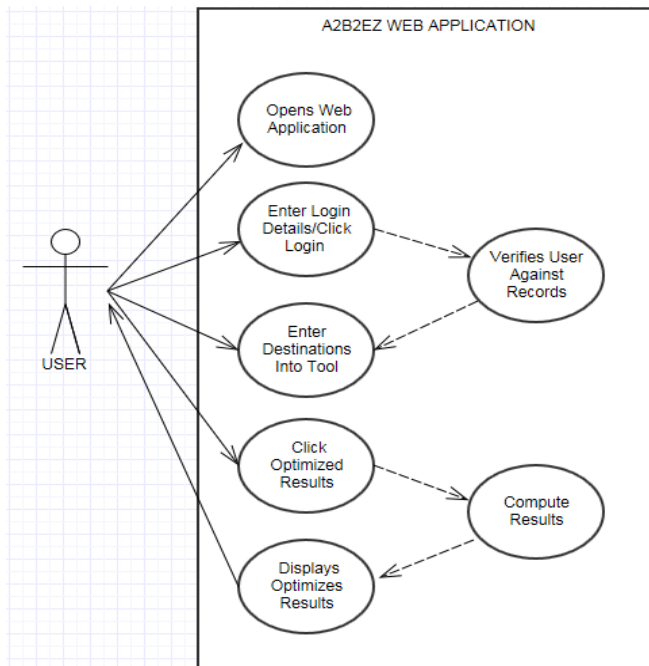
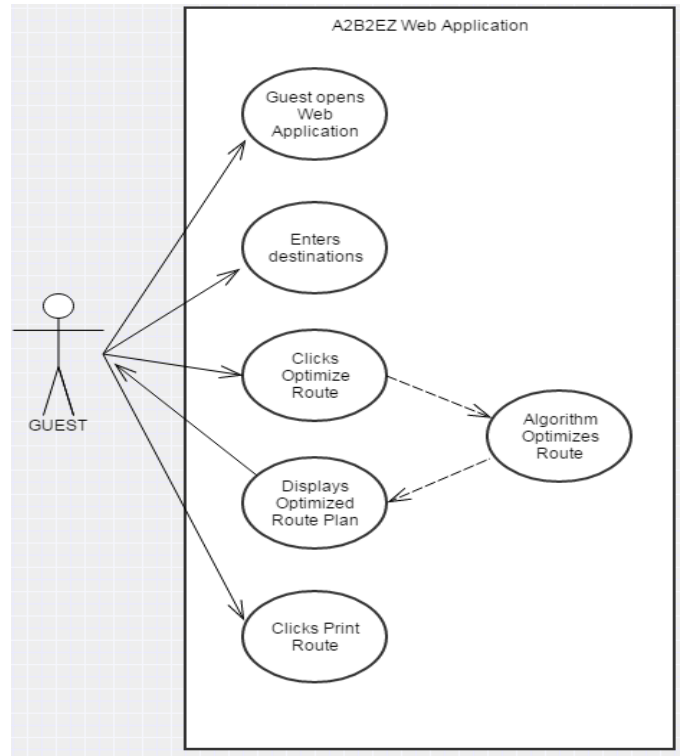
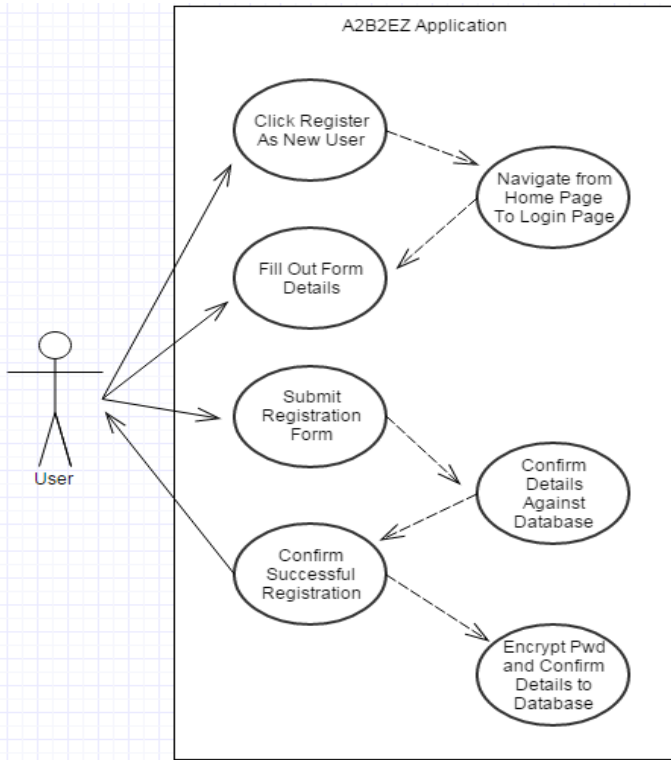
The initialization occurs in the Evolution Environment where the fitness (criteria) is evaluated before the algorithm can compute.

The GA itself consists of mutation and crossover and reproducing this until the best, or optimal result has been reached.

The input can be the latitude and longitude from our API and we can calculate the distances using that and return it in KM.

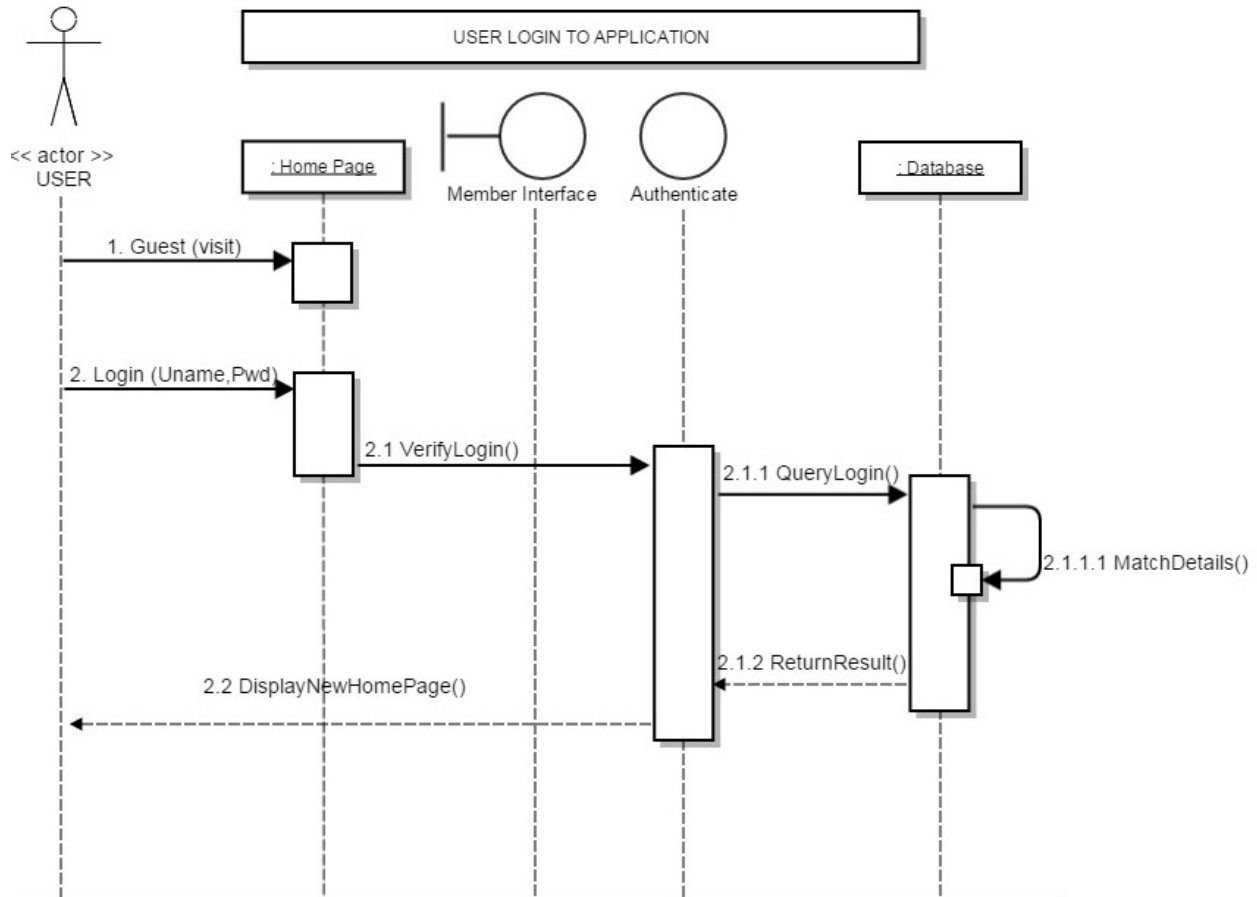
System Design

Use-Cases



Sequence Diagram

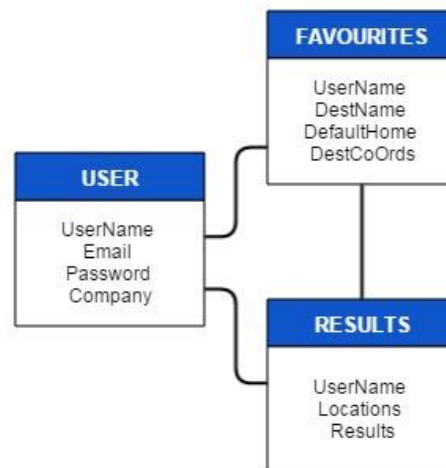
Below is the sequence of a user visiting the website as a guest and as a user that is logging in to their account



Database Design

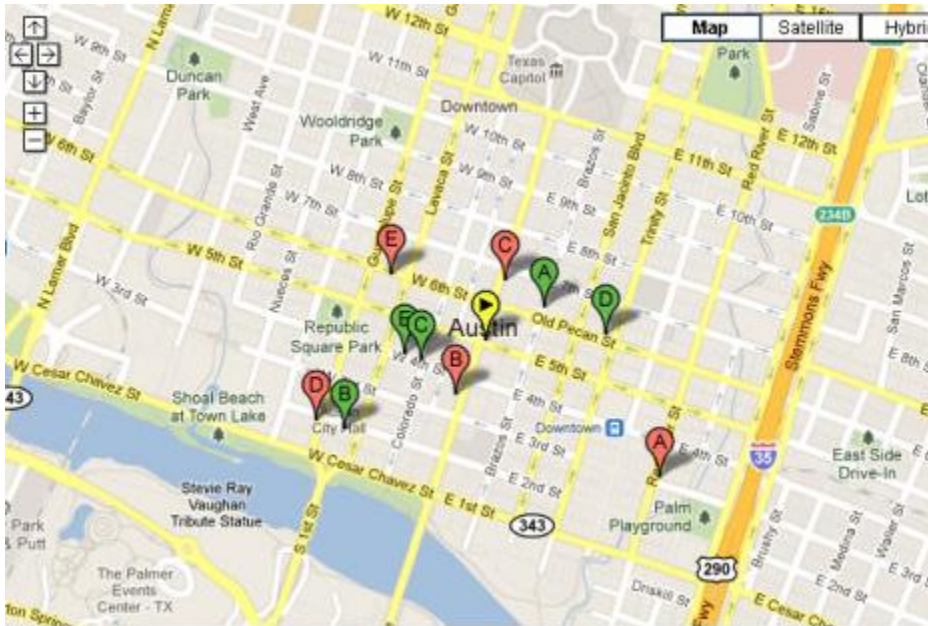
There will be three tables, one called USER to store registered users details.

FAVOURITES will store the favourite destinations for registered users so they can re-enter them easily. RESULTS will store results of queries for registered and guest users.



Application Program Interface

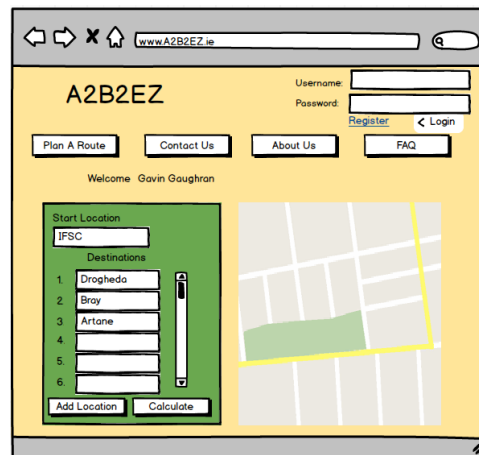
Google Maps API

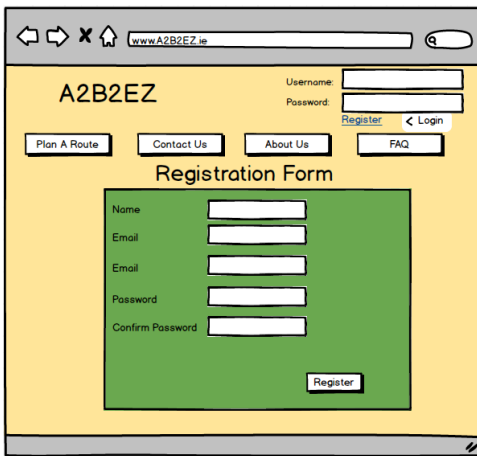


The Google Maps API will be used within the application. It's a standard Google map and I will need to implement that when clicking on a location within this map it will store that data as a variable within my application and be used for optimizing a route. The current strategy for plotting points on this is to then use the latitude and longitude distances to generate the algorithm.

User Interface Design

The website design will not be finalised until the first quarter of the New Year but the concept of how the below images look will remain. Colours and exact design might differ but the buttons and layout should not be too far from the final product.





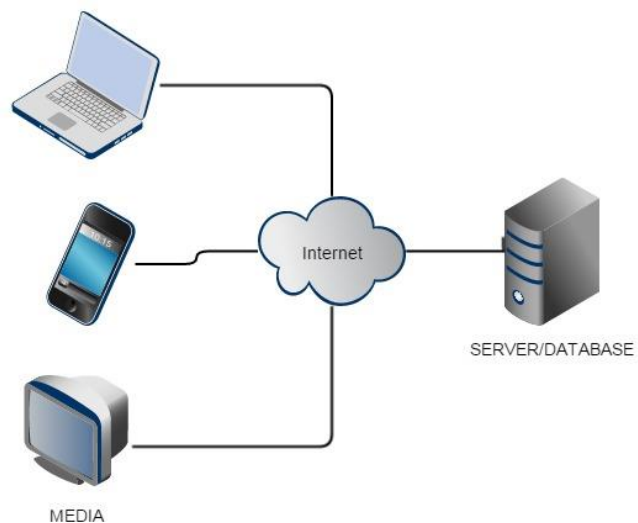
Section 508 Compliance



Initially the plan was not to have anything additional features such as visual impairment tools, but it could be possible to include an option such as Hi-Res images for users with visual impairment. Here is a mock-up of how using CSS we could change the layout of the web application for users with visual impairment. The high contrast colors are proven to aide people who are visually impaired.

Communication Diagram

The application will use standard browser and http/s connectivity.



Appendix A: Key Terms

The following table provides definitions for terms relevant to this document.

Term	Definition
Algorithm	A procedure or formula for solving a problem.
JS	JavaScript: A high-level, dynamic, untyped, and interpreted programming language.
JQuery	jQuery is a fast, small, and feature-rich JavaScript library
Ajax	Short for asynchronous JavaScript and XML. Is a set of web development techniques utilizing many web technologies used on the client-side to create asynchronous Web applications.
Media	Refers to IT hardware such as PC, laptops, phones etc.
GA	Genetic Algorithm: A method for solving both constrained and unconstrained optimization problems that is based on natural selection.
SQL	Structured Query Language: Is a special-purpose programming language designed for managing data held in a relational database management system.
API	Application program interface: Is a set of routines, protocols, and tools for building software applications.
IDE	Integrated Development Environment: This is a software application that provides the facilities to computer programmers for performing software development.

Monthly Journals

Introduction – “3 down and 1 to go”

From year one I told myself its only “4 years to go”, it will fly in, and every year I counted it up “1 down and 3 to go”, now look where I am. At times it did, and at times it certainly did not. Would I change it though!? Yes ☺ I would have studied harder and longer to be honest but it’s easier said than done.

My name is Gavin Gaughran and this is my Journal for my 4th year final project. I am now 29 years of age, I live with my girlfriend, Michelle, and I have one daughter- Ava aged 6. I am one year from my proudest personal achievement to date. Since beginning my degree I have changed jobs, bought a house, changed cars, and watched my baby grow up way beyond her years. Working full time, at least 40+ hours a week as a junior software developer, and also having a kid takes its toll and puts strain on you and your relationship but you need to realize and remember why you first chose to do this! “*I love writing code, I want a career in writing code*” and that has not changed, It’s where I want to be. My only regret now is that I didn’t start this Journal 3 year ago and cover my entire “college life”.

Background: The reason I am writing this Journal in such a personal way is mainly for my own personal reasons: I left school at 16 as I just could not go anymore, I just did not like it! So I left school to stack shelves in Tesco for several months. Half a year later I done what most kids at that time done and got a trade, so I was a 3rd year sheet metal worker before I realized I want to use my head for my job, not my hands. I moved into something I liked, I.T., and I started in a call centre for HP, I done call centre work for over a year and learned a lot about IT. My progression was then wanting to advance into the hardware sector and I went for an interview with Fujitsu Siemens, in the technical interview I got asked 20 questions, to 18 of them I replied “I’m sorry, *I don’t Know*” Worst day of my life! It was time to knuckle down; I went back the next day and requested to talk to the manager who interviewed me and offered to work for free for one month to prove my worth, I was in!!! 2 well-paying contracts later I left Siemens, (with the offer of a 3rd contract there for me) for Wincor-Nixdorf where I still to this day work, I have risen through the ranks from a Hardware engineer to now a software developer! I feel this is my mini-success story and it reminds me that I can do anything once I put my head into!

So let’s begin.....

September 2015

Tuesday 22-09-2015: Coming up with the idea is so pressuring, this defines what I have learned, what I want to and what direction I want to go in! I work closely with a very experienced coder who is soon leaving for Saudi Arabia and he told me “*Challenge yourself now with this project, it’s the beginning of your career now*” I love that attitude. With this in mind I’m going for something I love but also find very difficult, genetic algorithms. I’m going to challenge myself here I have 9 months, why not!?

Wednesday 23-09-2015: I downloaded all documents from Moodle and read over the past proposals and works. Also reading up on a few things I think I could use for my project: ASP.NET, C#, Databases, HTML, and SQL. Browsing has informed me that it could be maybe too difficult to implement C++ into an ASP.NET web application so I think I will go for C#. I am putting together a small email to send to Eamon and Eugene asking if they think my idea is ok to go with.

Thursday 24-09-2015: Today I emailed lecturer’s, they replied already and think I should start with my project proposal. Downloaded template and put in basic student details.

Friday 25-09-2015: Got my Project Proposal started and entered in some data. Read through some documents on what is expected from this project as I feel im really starting this now as I know what I want to achieve, this doesn’t necessarily mean I will be able but I know what I would like to do. The joys of parenthood just kicked in as Ava came into my office and got sick on my floor, seriously, enough done for tonight.

Saturday 26-09-2015: After a family day out in town, its 20:40 But I am going to do an hour as it all counts to keep moving toward completion. I told myself to just keep doing little bits, don’t need to finish it in one night but even a half hour a night helps massively. I added to my project proposal.

Sunday 27-09-2015: Again my project proposal was added to, well over 50% complete now with 5 days to go, so not feeling any pressure, yet.

Tuesday 29-09-2015: As I didn’t do anything yesterday I tried to get some done any second I could in work, and during my lunch and before college. Almost complete, took some time to do a project plan, very hard to predict the dates but I tried to think about all that lies ahead of me.

October 2015

Thursday 01-10-2015: Completed my Project Proposal, well I’m finished it but will reads over it tomorrow. Will ask a work colleague to look at it for me as she reads tech docs all the time and will read thoroughly! Didn’t put it in any fancy template or anything so I will ask what the guys in work think if I need to “pretty” it up.

Friday 02-10-2015: Put my journal and my project proposal into a new pretty template and sent it off to my friend in college for the once over before submission. Awaiting her reply. Few minor changes, spelling and grammar mistakes, time to upload ☺ First task done!!!

Sunday 04-10-2015: Took the weekend off my FYP as I had to complete and upload a slideshow on Security for my Business and Network Security module. Just uploaded, few hours early, feeling slightly ahead of things with the work I have been putting in.

Thursday 08-10-2015: Was sick from Saturday – Wednesday, worked from home, and didn't do much. Got no project work done and missed college on Tuesday night. Feeling better since yesterday so feel and feel a few days behind. The plan tonight is to get a web application started and have basic web application with a basic database behind it. With one eye on the Ireland Germany match and one eye on the laptop my application has finally started.

Wednesday 14-10-2015: Feel as though I am slacking a lot recently with nothing done during this week. Very busy in work and presentation preparation for Security subject has been the only thing I have done this week. Emailed Padraig De Burca who was assigned as my supervisor last night.

Friday 16-10-2015: Just downloaded some templates today for the next submissions still have some time before completing but I want to get started now and get back on track after a slow week.

Saturday 17-10-2015: Didn't hear from Padraig so going to re-send my mail just to be safe. Started my requirements spec and remember how helpful Padraig could be with these as they were always my struggling point. I like to jump in on my code and make mistakes and test it out, unfortunately the planning of it isn't exactly my forte.

Monday 19-10-2015: Starting my Requirement Specification tonight, kid just gone to bed so will get an hour in before my own bedtime. I will need to look up Use Case etc from year 2-3 as I have pretty much forgotten them at this stage. Hopefully it comes back to me quick enough.

Got started and added in purpose and scope. Not much, but at least it's started ☺

Thursday 22-10-2015: During work I added more to my requirement specification document. This is a bigger doc than I realized. Need to up my time on this over the next week or two!

Sunday 25-10-2015: Looking back over past submissions and over my past years notes to try completing the Use Case section. Talked with a colleague who works for my company, he drives everyday so I talked to him about the idea of my app and how it would help. Completed user requirement section.

Wednesday 28-10-2015: Downloaded information on Use Cases and diagrams from OOSE classes we had with Padraig in year 2. Added all the GUI section into Req Spec doc. Almost finished just use case and few bits left. Still have over a week, want to finish early though and make sure this one is done properly as 10% could be very important at the end if I want my first!!!

Friday 31-10-2015: All information apart from Use-Case diagrams added to req spec. #I am planning to use Gliffy for use-case diagrams.

November 2015

Sunday 01-11-2015: I have three requirements spec sections still to complete, the first one being the toughest as once I get this done I will have my idea of what to do and just change to suit the requirement. First one complete, including my use-case!!

Monday 02-11-2015: Second use-case complete and system architecture too. Stayed in work late to get as much done as possible as I'm 30 on the 5th and going away so need this finished for the 4th.

Wednesday 04-11-2015: Between yesterday and today I have finished my Requirement Specification document, and two days early ☺ checked over several times and got it proof read. Think it looks pretty good and I'm happy with the upload. Now to turn 30, its all down-hill from here ;-)

Thursday 05-11-2015: Happy with my finished requirement specification and uploaded a day early. Want to get started on Analysis document now as it's to be in, in four weeks.

Tuesday 10-11-2015: Took a few days away from project as had some business tutorials I wanted to complete and a presentation that I felt went really well. A lot of preparation seems to have paid off! Well we will see when the marks come out I suppose.

Wednesday 11-11-2015: Met with Padraig today, feelings nervous but good about project. He thinks it is going to be really difficult but if I can pull it off it will be good! Very happy I have him as my supervisor, he is so level headed and logical with his questions and discussions I love that. I like a challenge ☺. Dying to get this last doc done so I can start coding and learning more about algorithms.

Downloaded template and got my Analysis Design doc started.

Friday 13-11-2015: Started reading up on exactly what the design doc is, seems to me it's the same as a functional spec document. Going to start in the morning as only in from work and its after 8 so not thinking too clearly.

Saturday 14-11-2015: Started document and working on constraints and assumptions section. Slower process than I thought trying to document everything.

Monday 16-11-2015: Using Gliffy to complete diagrams, including Sequence, Use Case, Communication, Overall architecture, System architecture, Class Diagrams, Security Architecture, Performance Diagram, Database Design, API Design and UI.

Saturday 21-11-2015: Although I have not been entering into the journal I been spending any minute I get on all these diagrams for design document. Very tough work getting them done but gives a good idea of what way to design the application. Tonight I have finished all diagrams and the Design document is complete! Glad I'm over a week early as I have an essay for business due in two weeks so I will need to put some time into that now. Getting excited now as all preparation documents are done and now I can start coding my application and getting into the coding section of the project ☺

Thursday 31-11-2015: Spent a lot of time on my security essay over the last week so not much done apart from a review of my design document. Seems ok to me. Not overly happy but it's 12 pages long and has most types of diagrams I could think of in it.

December 2015

01/12/2015 – 09/12/2015: Spent most of the start of December completing a security essay. Went pretty well and will hope for my first to continue. Finding working and college tougher than ever at the moment. Have booked Christmas off though to get loads done so looking forward to that!

09/12/2015 – 31/12/2015: I have stopped writing daily notes as to be honest I forget a lot of the times and it also does take some time writing them in. So going to just add bits in in blocks more so than daily. Been reading an ebook: AI Techniques for Game Programming. Really good read on algorithms and talks about the TSP so I feel my understanding of the issue is improving. So my application has started and its just a basic template but it has me thinking exactly what I want to do and what way to do things. I feel starting to programme is the best way to start seeing what issues I will have and how best to resolve them.

Holidays for work over Christmas got accepted and then I was asked to come in so not too happy about that. That was when I knew I was off and could really get loads done but oh well nothing I can do. Moving slower than had hoped but getting there bit by bit. Even

if it is only a half an hour I am trying to do something for the project pretty much every single night and will need to continue that until the summer.

Overall December went only OK, thought I would have a good week or two off and didn't but still got application started and feeling slightly more encouraged about the algorithm itself.

January 2016

01/01/2016 – 08/01/2016: Only have one exam in Jan which is great so just focusing on that for now. Spend the week studying and I should do really well, so project taking a back seat for the week.

Ended up being pretty happy with exam, definitively passed it but not sure how good or bad it could be.

09/01/2016 – 31/01/2016: Applied and added my Google Maps API to my website, just a basic map for now. The plan is to have a map that users can click on but I have changed that now as I think having an auto-complete textbox is the better option. The clicking on the map option allows users to click locations that are not roads and leads to errors and mis-judgments from the map. So I am glad I got the click working, as I tested it well and realize it is not for my map and the autocomplete is a Google library I can call using JS and will pinpoint exact locations with great accuracy.

Struggling with some JS issues relating to my autocomplete boxes, silly issue, but cannot resolve! Want this done for mid-point presentation as really there isn't much to show as its all the code that does the "magic" and can't show that until I have it working.

February 2016

01/02/2016 - 10/02/2016: Coding the website now and have my Google Maps API working well. Some small JavaScript issues I want to try resolve but passing the map feature looks ok. Presentation uploads tomorrow so need to get all documentation up to date and finish of presentation slides.

Put together my tech report for this evening's submission. Have presentation half done so going to leave myself to only work on code for tomorrow and Saturday before mid-point presentation.

10/02/2016 - 20/02/2016: Have most Java working but it's slightly hacked that it's not the most efficient it should be. Going to work on it tomorrow and hopefully get it. If not will just leave it like that for now and move on as I don't want to lose any more time on it.

I am doing it every chance I get, really enjoying working on it at the moment but still very conscious this counts for everything really! No pressure.....

Padraig mentioned the “as the crow flies”, it really has me thinking and it shouldn’t! Why not give the user two options, for no good reason but travel on the road and also as the crow flies/aeroplane mode :-D I need to add this in.

Got mark for presentation... 73%, into my first but if anything I’m feeling more pressure now to keep it. The next two+ months will be tough.

20/02/2016 - 29/02/2016:

A lot on and been away for almost a week but still plodding along nicely. Feeling majorly stressed though for some reason, maybe due to CA’s and projects in other subjects. But I’ll get there!

March 2016

01/03/2016 - : Have decided to change the name of my project and the structure of my application. My name has gone from “a2B2ez” to “Route Optimising SideKick”. I do show this to a lot of people for feedback etc and everyone struggled with the name so it is getting changed. Everyone kept saying Z(Zed) and not Z(Zee) as I had hoped.

Also I have changed from using MVC to using Web Forms as from my research and a chat I had with someone I am only over complicating such a ”small” project. So I restarted and am back up to speed now after a good two days.

March was by far the most challenging month to date, Distributed Systems was extremely hard and the CA’s I could have done without! Miles behind I feel now but still doing late nights every night to catch up and I am still keeping the faith !

Code is coming together, it just better start working soon :-)

Not writing much in Journal as I really just want to spend as little time doing stuff like this and as much time as I can filling out my documentation and doing my actual project.

April 2016

The month of April went too quick, I created some test scripts and showed my application to people for the first time really. Very nerve-racking actually but the feedback was pretty positive, they appreciated that the web was prematurely displayed but the functionality was still there.

Exams and final projects for other projects took so much time from this project, was tough squeezing in a few hours late at night but wanted to keep going if even for an hour a night!

May 2016

I took the first two weeks of May off work and finalised all my documentation and all my code. Thank god I took so much time off and I wish I had more time to take off to be honest. Became a full time student for May and it was very much needed. Spending 12-15 hours a day on this and its really starting to come together.

I am happy with my documentation and I have all the bits I wanted in it. The code has to be submitted now on the 16th and I will be doing my code up until the last minute as I feel there is always something that can be improved!!

It has been the toughest few months ever, but seeing it coming together is nice and if I can get all the last few bits working the way I envisage them I will be really happy.

Gavin Gaughran

Gavin Gaughran

Date: 11-05-2016