2017-03-20

**Simple Route**

**MSc. Communications Software – Enterprise Web Development**

**Colm Carew - 20053766**

# Plagiarism Declaration

I certify that this assignment is all my own work and contains no Plagiarism. By submitting this assignment, I agree to the following terms:

Any text, diagrams or other material copied from other sources (including, but not limited to, books, journals and the internet) have been clearly acknowledged and referenced as such in the text by the use of ‘quotation marks’ (or indented italics for longer quotations) followed by the author’s name and date [eg (Byrne, 2008)] either in the text or in a footnote/endnote. These details are then confirmed by a fuller reference in the bibliography.

I have read the sections on referencing and plagiarism in the handbook or in the WIT Plagiarism policy and I understand that only assignments which are free of plagiarism will be awarded marks. I further understand that WIT has a plagiarism policy which can lead to the suspension or permanent expulsion of students in serious cases. (WIT, 2008).

Signed: Colm Carew

Date: 20/03/2017

# Introduction

Simple Route is a Web Application written using the React JavaScript Library. The core idea of the application is to; using a Map view, a Single Page Applications (SPA) which gives the user the easiest directions to their desired locations. In creating this solution the goal is also to exhibit some of the React concepts learned throughout the React lectures of the Enterprise Web Development module.

# Project Layout

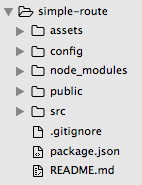
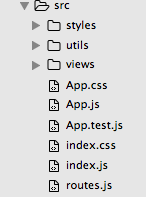


Figure 1 : Project Layout

As can be seen the folder structure is :

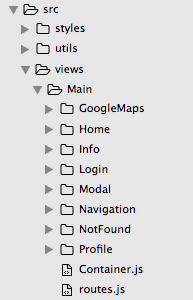
* assets (assets required for the webpage such as images)
* config – configuration files go here such as a config file for the Google Maps API Key
* node\_modules – this is the folder container all external components used in the project which are pulled using npm
* public – this is where the website favicon and index.html file are located
* src – this is where all the React Components that make up the website are located
* .gitignore – the file to tell git what files/folders should be ignored when committing
* package.json – the project configuration file which includes all relevant dependencies
* README.md – the read me of the project



**Figure 2 : src project layout**

Src is broken up into :

* Styles – where some of the css files reside
* Utils – where the utilities such as database writing or Google Maps services reside
* Views – where the actual react components are which comprised the website
* Route.js – used for the routing of the application
* Index – the entry point of the application
* App – where all of the components get rendered to before being rendered in index



**Figure 3 : Views Directory**

The views directory is inside the Main directory which is inside the src directory, it’s content are :

* GoogleMaps – all the components for GoogleMaps to be viewable + interactive
* Home – the main component of the application is located in this folder
* Info – the About page of the application is located here
* Login – the components related to the Login screen are here
* Modal – used to contain a custom modal which is used for display forms
* Navigation – folder containing the custom navigation bar
* NotFound – used for containing error pages, in this case just the NotFound page
* Profile – contains the components for the user profile
* Container – used as the main container of the home page
* Routes.js – used for the application routing – the other routes.js in the src directory uses this file

# NPM Modules Used

* auth0-lock – used for user authentication
* classnames – used for CSS so className is used in the JSX rather than class for defining css styling
* firebase – for this project firebase is being used as the database store – more on why later as auth0 could also have been used
* jwt-decode – used with auth0 authentication
* react – the react library
* react-bootstrap – a bootstrap library so bootstrap components can be used in React
* react-google-maps – a React module for Google Maps
* react-router – the react router used for rooting the application
* react-router-bootstrap – a module that allows the linking of the site within bootstrap components
* react-select – a module used for dropdowns
* superagent – a HTTP request library – used in obtaining location/address data from Google Maps API

# 3rd Party Libraries and Reasoning

Auth0 (<https://auth0.com/>) - Auth0 was used for sign in as at the time of project creation it was a popular choice for SSO and has an extensive list of providers that can be used for sign in. In this project it was used for Facebook, Twitter and Google sign in as well as a default registration if the user does not want to link an existing account.

Firebase (<https://firebase.google.com/>) – Firebase and Auth0 can have very similar uses. Auth0 primary goal is authentication and Firebase’s goal is overall application infrastructure. In this project it is being used for the database side of things storing the user’s preferences and directions they have searched. A major contributing factor in picking Firebase was that, in parallel with this project an Android App is being developed as part of another module that uses the similar idea of simple navigation. The Android App uses Firebase and thus would be nice if both applications could share the same user data so a user could use the website or the Android App.

Google Maps for React (https://github.com/tomchentw/react-google-maps)