Tim Scott

Tesco2@student.monash.edu

Abstract

A description of the requirements and outline of the Project to Calculate the area of Road across a Square Kilometer in Victoria  
  
Word Count: 1084

STREET aREA CALCULATOR

Project Proposal

1.0 Introduction

Calculating the area of roads has many different applications, from the military to Civil Engineers[SOURCE] to filming. Knowing where and how many roads there are is an important consideration for both moving and resurfacing roads. There are several different ways to tackle this problem as well as many resources to help you do so. Before there were databases of all the roads world-wide, such as those from Google or Open maps [SOURCE], or in regions where these resources still do not have a comprehensive catalogue of the roads, you had to use Satellite Data and image recognition. This is a process that required a lot of computation and was very complicated since not all roads are visible from above. Fortunately, we do have access to these types of databases, both containing images and vector data for the roads. It has made projects of this type a lot easier, although when you are trying to reach a greater degree of accuracy the complexity starts rising again as these databases rarely have all the data you need, such as the width of nature strips, the specific topographical information of the road, and the width of the shoulders[SOURCE].

For the project, we have been asked to calculate the total area to resurface in a given greater area. The nature of such an undertaking is that there will be a lot of unforeseen complications with the resurfacing of the road that can result in needing more asphalt than was absolutely necessary[SOURCE]. Such a degree of accuracy is unnecessary and as long as our inaccuracy is less than the natural error margin in the relaying of road it will be acceptable. Since it was difficult to find documentation on the average amount of asphalt over the bare minimum that was used I decided to go with a ballpark figure of 5%.

2.0 Project Requirements

2.1 Functional Requirements

* A website to Host the Project On. (Due to the nature of the project does not have to be a real web-address, locally hosting it will work
* Accuracy: must be able to Calculate the Area or Roads in the Given Sq. Km within 5%
* Ability to Differentiate between Different Kinds of Roads
* A GUI to allow Users to Input a Location
* Ability to Request Static Images from Google Maps

2.2 Non-Functional Requirements

* Completed within 13 weeks
* No Expenditure
* Easy to Use User Interface
* Real-Time: Calculation of Area must be Done efficiently (<10 seconds)
* Robust: The Average user should not be able to break the Program from the User interface

3.0 Project Plan

3.1 Overview

The Objective of this Project is to create a program with an elegant UI[SOURCE] that allows a User to Input a Location and then Calculates the Area of Roads with a square kilometre centred on that point, this project is to be Completed within 13 weeks without any external help or Collaboration from others and with a Budget of $0

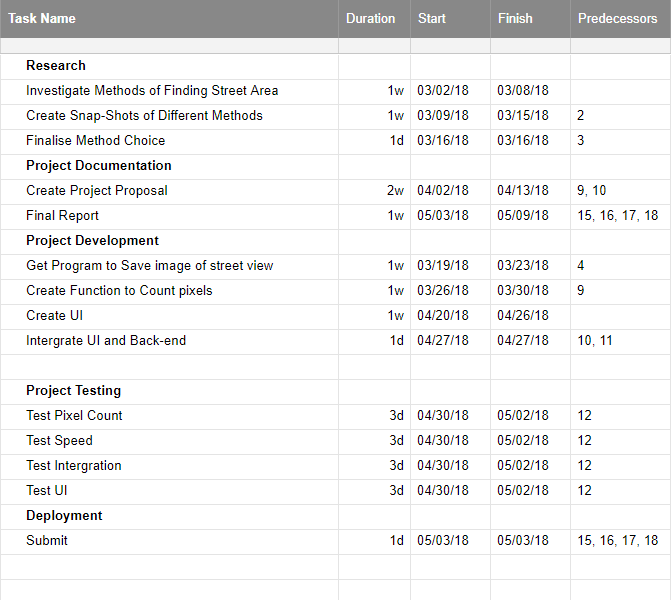
3.2 Risk Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Preventive Measures | Likelihood | Severity | Total |
| Time Mismanagement | Create a schedule | 7 | 4 | 28 |
| I get Sick | Give a few weeks leeway in schedule | 4 | 4 | 16 |
| Project Files Lost | Create Back-ups | 4 | 6 | 24 |
| Project Spec Misunderstood | Discuss with other Class Members | 3 | 7 | 21 |
| Project Changes | Discuss with Faculty | 1 | 7 | 7 |
| Aspect of Project is more Difficult than first Expected | Give a few weeks leeway in schedule | 4 | 6 | 24 |
| Hardware is Destroyed | Store back-apps Use Monash Computers | 3 | 8 | 24 |
| Google API Changes | Keep the google access sections Abstracted | 2 | 7 | 14 |

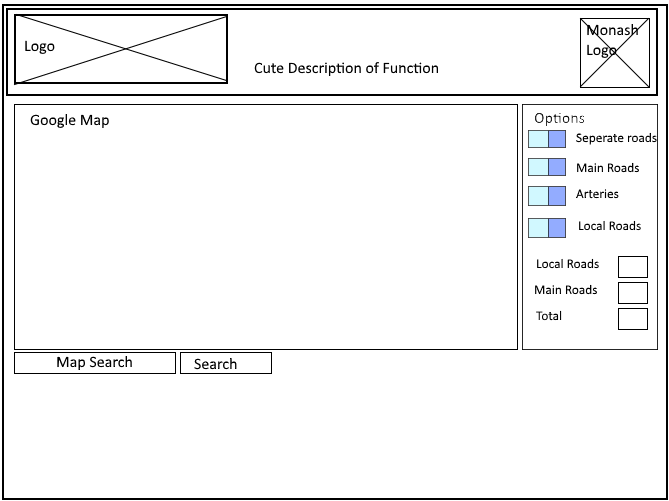
3.3 Resource Requirements

Hardware  
 A single Computer able to host the Website and a Internet Connection  
 Software

The Google Maps API, both Static and Dynamic, as well as Pycharm and Node.js for designing the program

3.4 Schedule  


4.0 External Design

4.1 User Interface  


4

3

2

1

1

4.2 Functionality

1. Interactive Google Map, Centre is marked with a Pin  
 2. Search Bar: Allow the User to Search for a Address  
 3. Options to Include Separate Kinds of roads  
 4. Totals Displayed

4.3 Performance

Space Requirements: < 6MB, Room for UI and space to save a Single 500 x 500 image

Time Performance: <10 seconds between request and displaying result

5.0 Internal Design

The Internal Design for this Project is actually Very simple with only a few Interacting Functions. The Basic Design will be that the User has a Interactive Google Maps Displayed on the webpage, and can control that through the Google Maps API, then when they click a button the Project will request the centre of the Currently Displayed Area and pass that off to a Python Program(GetMapImage).

The Python Program will take the Latitude and Longitude that was handed to it and Request a google static Image using the API of a Square Kilometre Area around that point and save that image with all extraneous information removed. A second Python Function (CaculateRoadArea) will then Go through that image and calculate the number of roads and then return that value to the webpage, where it will be displayed.

Web-Page

GetMapImage

CaculateRoadArea

Web-Page

6.0 Software Architecture

This is not a Very complicated Project from a architectural point of view since it only has 2 Methods and a Web-page Interface. And None of the Methods Inherit from any shared Parent Class. The Methods are, GetMapImage, which retrieves a image using the google API and the Requests Library in Python, and CaculateRoadArea, which using the Python Image Library to Analyse the Image, The webpage Also Just uses the Google API to Convert the Location the User wants into Latitude and Longitude.

7.0 Test Plan

7.1 Test Coverage

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement | Test Coverage | Test Methods | Sample Test Cases |
| Website Hosting | Test to see if a connection can be Established | Acceptance Testing | Connect to the Website |
| Accuracy | Using Sample Data test the Accuracy of the Result | Unit Testing | -37.753595 ,145.1198783 |
| User Input | Have the user enter Data and see if it can be retrieved | Unit Testing | 35 Atkinson St Templestowe entered in search Bar |
| Speed | Running Calculations and Recording the Time | Unit Testing | -37.753595 ,145.1198783 |
| User Interface | Have a User review the Interface for readability and try and break it | Acceptance Testing | - |
| Differentiate between roads | Use Sample Data and compare results to expected | Unit Testing | -37.753595 ,145.1198783 |
| Full Function | Run through complete operation of System | Systems Testing |  |