Project Plan

**NSW Traffic Penalty Analysis Software**

Elliott Horne - s5222361

Zoe Waters - s5131193

**Table of Contents**

[1.0 Introduction 3](#_gjdgxs)

[1.1 Problem Background 3](https://docs.google.com/document/d/1PgYs2NZUVuqTBIHwiiYkfg4ffAgiLldj/edit#heading=h.1t3h5sf)

[1.2 Scope 3](#_30j0zll)

[1.3 Document contents 3](#_1fob9te)

[2.0 Work Breakdown Structure 4](#_3znysh7)

[3.0 Activity Definition & Estimation 5](#_tyjcwt)

[4.0 Gantt Chart 6](#_3dy6vkm)

# **Introduction**

## **Background**

The New South Wales government has asked our team to take a significant amount of data that is formulated from the traffic penalties that the NSW police has given to the general public. This data ranges from 2011 to 2017 and contains all offenses recorded by the police in relation to the traffic offenses. This dataset contains over 100,000 entries and all the information associated with each offense, this is quite a significant amount of information to sift through so we’ve been tasked to create a program to sort through and display the information in certain circumstances.

## **Scope**

The scope of this project is to help streamline the accessibility and understanding the information presented in the data that is provided. This will be done through multiple coding practices that will help meet certain criteria in order to provide users with easy to use and understand information with the certain criteria in mind. This kind of program needs to be simple to follow and be accurate as although the users of the program should have basic computer operation skills, we want to minimize training required for this program.

## **Document contents**

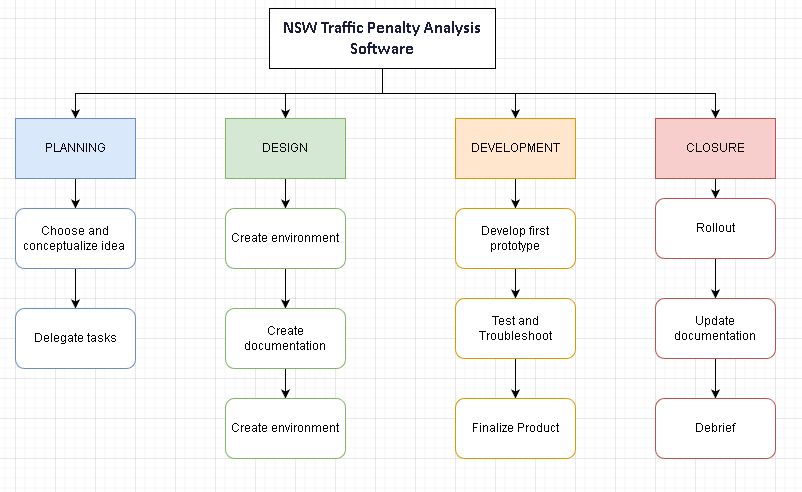
This document will contain

* A work breakdown structure
* The Activity description and estimation
* A Gantt Chart

All this information will help display our thought-process and the average timeframe we would recommend to complete this project.

# **Work Breakdown Structure**

The diagram below is a breakdown of the work that will be done across the entire project broken up into different sections. These will be discussed at greater length in the following Activity Definition and Estimation section.



*.*

# **Activity Definition & Estimation**

### **Planning**

1. Choose and conceptualize ideas (1 week)
2. Delegate tasks (1 week)

There are two main points in the planning stage of the project, being the conceptualizing and creating an idea in our group and then delegating tasks to each of the members to cover an evenly distributed workload. These are pretty self explanatory as they should only require a couple of hours but due to scheduling we have given ourselves a week to organize times to meet and discuss ideas and separate the workload accordingly.

### **Design**

1. Create environment (1 week)
2. Create documentation (2 weeks)
3. Create UI Template (1 week)

The design section of the WBS above is mainly taking the ideas and creating them into a more formal and structured approach. The first step in this section would be to create the environment for us all to access and update on our own personal machines, due to being online students or scheduling issues that make it hard to collectively work on the project. The second step would be creating the documentation for this project being this Project Plan and fleshing out the idea in the Software Design document providing a more extensive look at the project. Finally, the last step will include a UI template for us to have a visual aspect of what our design would look like for a user, giving us a rough idea what users would be working with.

### **Development**

1. Develop first prototype (3 weeks)
2. Test and Troubleshoot (2 weeks)
3. Finalize product (2 weeks)

The development cycle would be a large time sink for most of the work in order to properly go through all the processes and provide a working finished product. The first prototype should be up and running in approximately 3 weeks in order to properly build all the components of the project and apply them to the overall project. The testing step in the project should begin in the final week of the prototype development, in order to maintain a steady pace and give us ample time to troubleshoot any early problems that arise. The final step of the development stage will be the finalization of the project which should take approximately 2 weeks. This should be done in this time frame with all the troubleshooting and adjustments finalized.

### **Closure**

1. Rollout (1 week)
2. Update documentation (1 week)
3. Debrief (1 week)

The final section for the project would be the closure section which would be all done within a 1.5 week period. The rollout and documentation would be done in the same time period in order to get everything in order, beginning to revisit this document and add any more information about the progress of the project. The debrief can be carried out in a meeting of our choosing in order to see what our thoughts were as a unit with teamwork and the project work.

**3.1 Updated Documentation**

Several design changes were made to account for issues that occurred during development because of limitations on our programming knowledge, library limitations and data limitation.

The display design had to be changed due to wxpython and matplotlib and due to constraints of the library as well as the possibility of our lack of knowledge when using its libraries. As we first designed the program to run everything in the same window, changing frames in wxpython opens a new window as well as creating a graph in matplotlib opens a new window. We also could not get the grid or scroll working for data displays. The fact new windows had to be made, we changed the layout of the main page as each search must be open in a new window.

Due to the size of the database, we had to choose the top amounts in search searches such as the case code distributions, which has over 6000 unique case codes, this was the same for the mobile phone graph which shows the top 3 case codes. We also decided to use the number of entries that the case code had instead of the total amount of penalties for those case codes. This was attempted but the code ran for over 30mins before it was stopped prematurely as it was way too much data analyze in a timely manner.

# **Gantt Chart**

Chart

Description automatically generated with low confidence