Software Design Document

<Project Name>

Nikkelas Raines, Taiki Matehe, Kiarna Broomhead

Table of Contents

[1.0 System Vision 3](#_Toc46748622)

[1.1 Problem Background 3](#_Toc46748623)

[1.2 System Overview 3](#_Toc46748624)

[1.3 Potential Benefits 3](#_Toc46748625)

[2.0 Requirements 4](#_Toc46748626)

[2.1 User Requirements 4](#_Toc46748627)

[2.2 Software Requirements 4](#_Toc46748628)

[2.3 Use Cases 4](#_Toc46748629)

[3.0 System Components and Software Design 5](#_Toc46748630)

[3.1 System Components 5](#_Toc46748631)

[3.2 Software Design 5](#_Toc46748632)

[4.0 User Interface Design 6](#_Toc46748633)

# System Vision

## Problem Background

What is the dataset –

The dataset that will be analysed is the NSW traffic penalty dataset which include reports from 2011 until 2017. The dataset holds a wide variety of information, including but not limited to, the financial year of the offence, the month of the offence, offence description, offence code, legislation the offence falls under and location.

Input/Output –

Users will be able to input selected time periods, penalty codes, keywords and phrases, if the penalty case was detected by cameras to receive and output of information within the given constraints. This output can be altered to produce information in the form of graphs, with additional information available such as the average value ($AUD) of the penalties.

What problem does this solve –

Who will use it –

## System Overview

What will the system do –

Features, functions –

The system will include browsing functions which will let the user view all information. This browsing function will include features such as the ability to filter information -this can be by date, case codes, words and phrases, or if the penalty case was detected by cameras-, as well as the ability to generate graphs for the information and calculate the average value ($AUD) of the penalties.

## Potential Benefits

How will this create benefits –

# Requirements

## User Requirements

In this section you detail how a user is supposed to interact with or use your program. What do they ***need*** to be able to do? This should all be from the end users perspective. Can be a combination of narrative text and listing of needs.

**Assignment note: You have not been given a client/user, so you can make one up. Who do you think would be using your software?**

## Software Requirements

In this section you detail what the requirements for the software are. What functionality will it provide? This is usually a formal listing, with requirements often using the word ‘Shall’. IE:

R1.1 The program shall accept multiple file names as arguments from the command line.

R1.2 Each file name can be a simple file name or include the full path of the file with one or more levels.

etc …

Can be primarily functional requirements, though you may include other types if you think of them.

* 1. The program shall accept multiple dates as arguments from \_\_\_\_\_\_.
  2. Each date must include at least the desired month and year, within the given time period.
  3. Information within the user-selected period shall have the option of being shown in a graph.

2.1 The program shall accept penalty case codes as arguments from \_\_\_\_\_\_.

3.1 The program shall accept keywords and phrases as arguments from \_\_\_\_\_\_.

3.2 Information shall then be able to be filtered further with the use of dates.

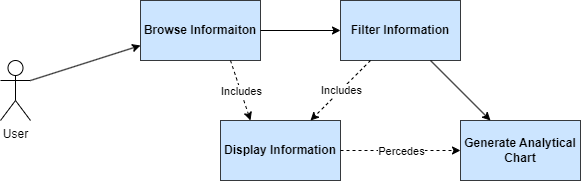
Not sure how the cameras or radar filer would work. That still needs to be added into this.

## Use Cases & Use Case Diagrams

In this section you provide some use cases showing how people may use your software.

Look im gonna be real. Not too sure what functions we are going to have.

Example below.



# Software Design and System Components

## Software Design

A block diagram/flowchart of how your software might work

## System Components

### Functions

Preliminary list of all functions in the software. For each function in the list the following information is provided:

* a brief description of what it does (1 or 2 sentences);
* a list of the input parameters, and their data types, and what they are used for;
* a list of any side effects caused by the function (ie change global or member variables, changes data passed by reference from calling function etc)
* a description of the function’s return value

1. Date filter

* This function filters the data with the use of selected dates.
* Only data within the given time period will be provided.
* User will need to input two different dates, the first will be the start date and the second will be the end date.
* Dates can be input by either a dropdown calendar or by using the search bar with the format “DD/MM/YYYY – DD/MM/YYYY”.

1. Keywords search bar

* This function filters the information with the use of keywords.
* Only data that contains the given keywords will be provided.

1. Penalty code filter

* This function filters the information with the use of keywords (Specifically Penalty case code/s).
* Only data that contains the given case code/s will be provided.
* User will need to input numbers as the case codes.
* When searching for multiple cases, a comma ‘,’ will act as the separating character. E.g. 74460,19584,74705

1. Receiving user selected information

* This function will be responsible for providing the user with the data they requested.
* This function will change and respond to the different prompts the user gives.

1. Graphs

* This function will take the data that the user requests and provide it in a graph form.

1. Penalty case average cost

* This function provides the user with the option to see the average cost of the penalties within their specified period.
* As this information is not directly taken from the data available, it will be provided to the user in the form of an integer.
* OPTIONAL – The information provided to the user can be presented in a graph to show the difference in each case cost. This graph will also provide the case average as a value.

### Data Structures / Data Sources

List of all data structures in the software (eg linked lists, trees, arrays etc) or eternal data sources. For each data structure in the list the following information is provided:

* Type of structure (tree, list etc),
* Description of where and how it is used
* List of data members, and what each one is for do
* List of functions that use it

### Detailed Design

Pseudocode for all non-standard / non-trivial algorithms that operate on data structures

# User Interface Design

This is your initial interface design. Describe the tools you used for this design stage and any key findings that informed your design. This introduction is descriptive and should explain what you have completed for the actual design work you will present in the sub-sections below.

## Structural Design

Structural design refers to the navigational and information structure of your product – the structure that supports the interface layout. How will you structure your product? How will you group your information? How will you navigate through your product? Why? This can take the form of a diagram showing structure and hierarchy, supported by a discussion and justification of your choices. Why have you made these design choices? Describe and outline the structure of your interface and of your information.

## Visual Design

Detail your visual design: Layout, visual elements, icons, graphics, style, colour, fonts general screen designs. This can be sketches, wireframes, mockups etc, supported by a discussion, explanation, and justification of your choices.