Software Design Document

Accident Analysis Software

Nicolas Donaldson – S5256284

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

A dataset containing data on Victoria State Accidents presents opportunities to discover underlying causes and other statistics surrounding road accidents that occur in Victoria. Software is required to perform an analysis on the data and present the data to the user in a way that is accurate, interpretable, and relevant. An increase in data has required the Department of Transport for Victoria to seek software that can draw conclusions and present findings, so that the Department of Transport may overcome the challenges associated with data analysis and human error. As data becomes too large, it becomes impossible to analyse the data without errors, misunderstandings, and large amounts of human resources. This is the problem the software intends to solve.

## System Overview

The system intends to provide a user-friendly interface that correctly and promptly returns search queries. The system will include functions that:

* For a user-selected period, displays the information of all accidents that happened in the period.
* For a user-selected period, produces a chart to show the number of accidents in each hour of the day (on average).
* For a user-selected period, retrieves all accidents caused by an accident type that contains a keyword (user entered), e.g., collision, pedestrian.
* Allows the user to analyse the impact of alcohol in accidents – ie: trends over time, accident types involving alcohol, etc.
* One other ‘insight’ or analysis tool

## Potential Benefits

Implementing software in the context of data analysis, alleviates the pressure on humans to avoid producing errors in the analysis. Without software, the Department of Transport may spend countless human and financial resources on producing results that software can produce within seconds. By streamlining long tasks into functions that can be performed by software, the Department of Transport can benefit with increased employee productivity and streamlined processes. Software can discover patterns and statistics that are otherwise impossible for humans to observe from a dataset. With this knowledge, the Department of Transport can benefit greatly with the software’s ability to guide the data analyst to solutions not recognised from just the human’s perspective on the data.

# Requirements

## User Requirements

As the client is the Department of Transport for Victoria, it is expected that it is employees of the Department that are the end users of the program. The end user must first begin with inserting the dataset. Once entered, the user may toggle between variations of the user interface. It is only required that the user enters certain parameters surrounding the nature of the user’s query. Once decided, and entered, the results of the user’s query shall be viewable via the output.

In a sequential list of needs, the above would be translated to:

1. Open program
2. Insert Dataset (if first time using application)
3. Insert Query Parameters
4. View Results

Each function will be available from the main view tree in the user interface, with the required function being run dependant on the user input. A field will be available for users to input keywords, which will limit the data output to accident data only containing those keywords.

Options to limit the timeframe will be displayed, if a user decides to use this function the output will be limited to data from within this function. If no time is specified, then all relevant data from the dataset will be displayed.

All data queries will be returned in table format.

Once the data has been displayed, a button to display a chart for the average number of accidents in each hour of the user specified period will appear above the returned data. If this button is pressed a chart will be generated and will be displayed above the tabulated data.

There will also be a radio button which will allow the user to specify that they wish to analyse alcohol related incidents. If this button is checked, a chart will be displayed showing both alcohol and non-alcohol related incidents over the user-specified period, as well as the tabulated data from both alcohol and non-alcohol related incidents in different tables.

**Function 5 is still up in the air – to discuss @ next meeting.**

In addition to this, the end user should be aware that their query is being processed. To provide this context a loading prompt should be displayed to make the end user aware that their query has been accepted and should be displayed soon.

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

1. The program shall have a GUI implementation
   1. The program shall allow for user input through GUI
   2. The program shall accept .csv dataset files from user input *and convert to JSON*
   3. The program will limit returned data by user inputted keyword.
   4. The program will limit returned data by time range
   5. The program shall graphically display data through charts & tables
   6. The program shall output a brief summary of results
2. The program shall have maths & statistical modules
3. The program shall have a datetime module.
4. The program shall have a GIS module?

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.

## Use Cases & Use Case Diagrams

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

# 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

**Function 1:**

**Description**: For a user-selected period, display the information of all accidents that happened in the period.

**Input Parameters**: (Time1, Date1, Time2, Date2)

**Data Types**: (datetime (from datetime module)

**Side Effects**: NIL (pending)

**Return Value**: ALL results between Time1, Date1 AND Time2, Date2

**Function 2:**

**Description**: For a user-selected period, produce a chart to show the number of accidents in each hour of the day (on average). This function extends upon function 1.

**Input Parameters:** (Time1, Date1, Time2, Date2, show chart True/False [button])

**Data Types**: (datetime (from datetime module)

**Side Effects**: NIL (pending)

**Return Value**: IF TRUE: Return Chart between Time1, Date1 AND Time2, Date2 ELSE return information between Time1, Date1 AND Time2, Date2

**Function 3:**

**Description**: For a user-selected period, retrieve all accidents caused by an accident type that contains a keyword (user entered), e.g., collision, pedestrian. This function extends upon Function 1.

**Input Parameters:** (Time1, Date1, Time2, Date2, keyword[string])

**Data Types**: (datetime, string)

**Side Effects**: NIL (pending)

**Return Value**: ALL results between Time1, Date1 AND Time2, Date2 WHERE accident type = keyword

**Function 4:**

**Description**: Allow the user to analyse the impact of alcohol in accidents – ie: trends over time, accident types involving alcohol, etc.

**Input Parameters:** (Time1, Date1, Time2, Date2, keyword[string], predefined filter settings)

**Data Types**: (datetime, string)

**Side Effects**: NIL (pending)

**Return Value**: Graph of ALL results between Time1, Date1 AND Time2, Date2 WHERE accident type = keyword. Up to two sets of results (non-alcohol, alcohol) for user to compare.

**Function 5:**

**Description**: Visualisation tool using Google Maps overlay for user to understand accident data relating to geographical location information.

**Input Parameters:** (Time1, Date1, Time2, Date2, pulldown menu of accident type, alcohol related (true/false))

**Data Types**: datetime, boolean, string

**Side Effects**: NIL (pending)

**Return Value**: geolocation latitude, longitude, maps overlay?, accident details when clicked, google streetview of location (bonus function would be nice =3)

### Data Structures / Data Sources

CSV Dataset (Table) -> JSON file (key-value dictionary)

Search criteria

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.

The software consists of three windows including the main window. The structure of the main window is designed to indicate steps that users follow by placing sections from top to bottom. The estate of main window is limited, thus, charts will be displayed in a secondary window to create more display surface area.

### Main window

The main window of the software contains the following sections. In order to provide the best useability, there are grouped by action which are navigate, search, result (view), summary (view), and chart (view).

**Navigation tool bar**

This section is located at the top of the main window and contains buttons to perform the main functionalities. They are Dataset, Analyse, Alcohol, and Geographical.

**Data search box**

This section is located under the navigation tool bar and contains input fields for users to define search criteria such as dates, and accident type along with a search button to initiate data search.

**Data table view**

This section displays output data in a table view. It is located under the data search box and users can see search result.

**Chart selection bar**

This section is located at the bottom of the main window under the data table view. It shows available chart types which users can view by clicking. This section will not be available when there is no available option.

**Summary view**

This is located at the right side next to the data search box to show summary of result from a search. It can include number of record found, total number of target records, and total number of injury/fatality.

### Chart window

A secondary window will open when users select one of options in the chart selection bar. It displays chars with selected conditions. This window is independent from the main window and can be placed anywhere in users’ screen. Users can close this window without affecting the main operation of the software.

### Dataset window

A dialog window will open when users select Dataset menu from the navigation tool bar. It displays a file selection field for users to select a file to import as a dataset to use in the software. This window will close automatically after importing. In case of errors, an error message will be displayed.

## 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.

Layout

Visual elements (icons, graphics)

Colour

Key colours: Dark grey (#333434), Teal (#13a2a6)

Fonts

VIC, Arial, Helvetica, sans-serif

VIC font – free to download <https://www.vic.gov.au/brand-victoria-fonts>

<see <https://transport.vic.gov.au> to get info>





