Software Design Document

<Project Name>

Student Names

Augustine Kim (s5125270)

Kai Chun Yu (s5309764)

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

Improving road safety helps alleviate the corresponding social and economic costs brought about by road trauma. Victoria carries the 2nd biggest cost burden across Australian states, valued at $6 billion per year, which not just affects the crash victims, but also their families, other road users, the Commonwealth, and the Victoria State Government. This includes direct costs such as hospitalisation, medication, rehabilitation, and property damage, as well as other potential costs due to loss in productivity of patients, those who allot time to provide informal care, and the surrounding community (Steinhauser & Lancsar, 2022). Raw data that shows details of each road crash between the 3rd quarter of 2013 to 1st quarter of 2019 within Victoria is already available. The goal of the proposed system is to make the said data be more meaningful and useful by providing a user interface where data can be aggregated or broken down into more detailed information, to show status and trends by user-selected parameters (e.g., period, area, type of accident etc.). This will allow the government to measure if the polices that were set to improve road safety are effective, based on actual performance versus identified goals or metrics (e.g., lowering the number of alcohol-related accidents by a set percentage versus previous year).

## System Overview

The system will be capable of performing simple data analytics tasks, with the following output that will be visualised on a dashboard:

* Information of all accidents based on a user-selected period.
* A chart showing the average number of accidents in each hour of the day based on a user-selected period.
* Retrieve all accidents caused by an accident type that contains a keyword entered by the user (e.g., collision, pedestrian), based on a user-selected period.
* A chart that shows the impact of alcohol in accidents with the following filters:
  + Period Covered: Number of accidents by year, by month across a 5-year period, by day across a 5-year period, day of the week, hour of the day.
  + Type of accident
* A chart showing accidents that caused harm to each type of road user i.e., pedestrian, cyclists, motorists, older people etc., based on user-selected period and area.

## Potential Benefits

The primary benefit of the system is its potential to lower the social and economic costs brought about by road crashes, by serving as a tool to measure the performance of polices and departments in improving road safety, which in turn will serve as basis for policy improvements.

# 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?**

**Users’ needs**

* A user guide that describes how to use the system.
* A selection of a period (eg. Start date and end date) for viewing specific data in the period.
* A selection of accident categories for viewing specific information in selected accident types.
* A selection of alcohol-related cases to identify the influences of alcohol in accidents.
* All information on accidents that match the selected period and category.
* A diagram (line chart) that can show the average number of accidents in each hour of the day based on the selected period, category, and relationship of alcohol.
* Some diagrams (bar chart, pie chart) that show the number of accidents based on selected categories and, the relationship of alcohol.

Once the user accesses the user interface dashboard of the proposed system, the user will be able to view a user guide on how to use the system. Then, the user can use the filter to select a specific period and category to retrieve the accident data and diagram based on their needs. In addition, a filter for selecting alcohol-related cases will be available for users to identify the influences of alcohol. After selection, the user can click the search button to retrieve accident data and analyse diagrams based on the selected requirement.

## 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.

* The program shall show a Victoria State logo for users to recognise the authenticity of the website on the homepage.
* The program shall contain a system use guide to describe how to use the function.
* The program shall have two filters for the user to select a start day and an end date.
* The program shall have a filter that accepts typing for the user to select the accident category.
* The program shall have a filter for users to select alcohol-related accidents.
* The program shall contain a search button for the user to click and execute the search action.
* The program shall remind the user start date and end date must be filled in order to search.
* After performing a search, the program shall display all accident information that matches the selected filters.
* After performing a search, the program shall contain a line chart to show the average number of accidents in each hour of the day based on the selected period.
* After performing a search, the program shall contain a bar chart and pie chart to show the accident number based on the selection of category and alcohol-related filter.
* Once a selection has been made on the accident category filter, the program shall only display the information that is related to the selected category.
* Once a selection has been made on the alcohol-related filter, the program shall only display the information that based on the selection.

## Use Cases & Use Case Diagrams

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

|  |  |
| --- | --- |
| **Use Case ID** | 1 |
| **Use Case Name** | View all information of accident. |
| **Actors** | User |
| **Description** | The user will be able to view all accident information from a selected period. |
| **Flow of**  **Events** | 1. The user will access to the system user interface. 2. A system user guide will tell the user to select a period for search. 3. The user will select the start date and end date. 4. The system will display a table with all relevant data between the period on the screen. |
| Alternate Flow | None |

|  |  |
| --- | --- |
| **Use Case ID** | 2 |
| **Use Case Name** | View average number of accidents in each hour chart. |
| **Actors** | User |
| **Description** | The user will be able to view a line chart that show the average number of accidents in each hour of the day. |
| **Flow of**  **Events** | 1. The user will access to the system user interface. 2. A system use guide will tell the user to select a period for search. 3. The user will select the start date and end date. 4. The system will display a line chart on the screen. |
| Alternate Flow | None |

|  |  |
| --- | --- |
| **Use Case ID** | 3 |
| **Use Case Name** | Search by accident category. |
| **Actors** | User |
| **Description** | The user will be able to search by accident category and view the accident information based on the selected period and accident category. |
| **Flow of**  **Events** | 1. The user will access to the system user interface. 2. A system use guide will tell the user to select a period and accident category for search. 3. The user will select the start date and end date. 4. The user will type/select the accident category. 5. The system will display a table that contains relevant record on the screen. |
| Alternate Flow | None |

|  |  |
| --- | --- |
| **Use Case ID** | 4 |
| **Use Case Name** | View alcohol analyses. |
| **Actors** | User |
| **Description** | The user will be able to use an alcohol related filter to sort the output diagrams and table. |
| **Flow of**  **Events** | 1. The user will access to the system user interface. 2. A system use guide will tell the user to select a period, accident category, and alcohol-related filter for search. 3. The user will select the start date and end date. 4. The user will type/select the accident category. 5. The user will select Yes/No on alcohol-related filter. 6. The system will display the table and diagrams on the screen. |
| Alternate Flow | None |

|  |  |
| --- | --- |
| **Use Case ID** | 5 |
| **Use Case Name** | View road user chart. |
| **Actors** | User |
| **Description** | The user will be able to view a chart that show the number of harm case to each type of road user. |
| **Flow of**  **Events** | 1. The user will access to the system user interface. 2. A system use guide will tell the user to select a period, accident category for search. 3. The user will select the start date and end date. 4. The user will select the road users on the accident category filter. 5. The system will display a bar chart on the screen. |
| Alternate Flow | None |

**User Case Diagram**

A diagram of a problem

Description automatically generated with medium confidence

# 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

**Functions**

* Load Data function (Search)
  + Description: This function will enable the system to load data from the original database which is the excel file.
  + Input parameters: start date (date), end date (date), accident category (string), alcohol-relation (string). These parameters are used for identifying which data are going to load from the database to system.
  + Side effects: None
  + Return value: Array that store all data sorted by the input parameters.
* Table display function
  + Description: This display function will enable the system to show the result of the search in a table from for user to view the accident records.
  + Input parameters: the result of load data function (array). The parameter is used for inserting what data need to be displayed in the table.
  + Side effects: None
  + Return value: This function should end up return a table that contain all data filtered by the search.
* Line chart display function
  + Description: The function will display the number of accidents by selected categories.
  + Input parameters: start date (date), end date (date). These parameters are used for inserting what period of data need to be displayed in the line chart.
  + Side effects: None
  + Return value: This function should end up return a line chart that showing the average number of the accidents (Y Axis) by each hour in the day (X Axis).
* Bar chart display function
  + Description: The function will display average number of accidents by selected category in a bar chart view by user.
  + Input parameters: start date (date), end date (date), accident category (string), alcohol-relation (string). These parameters are used for identifying which data are going to display in the bar chart.
  + Side effects: None
  + Return value: This function should end up return a bar chart that showing the number of the accidents (Y Axis) by different categories (X Axis).
* Pie chart display function
  + Description: The function will display a pie chart that present the number of accidents by selected categories in a percentage form.
  + Input parameters: start date (date), end date (date), accident category (string), alcohol-relation (string). These parameters are used for identifying which data are going to display in the pie chart.
  + Side effects: None
  + Return value: This function should end up return a pie chart that showing the number of the accidents by different categories in percentage.

### 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.

For the interface design, wireframes were initially developed and later, Pixso was used for creating the final UI mock-up. User research underscored the importance of a user-friendly, easy-to-navigate interface, which guided our design decisions.

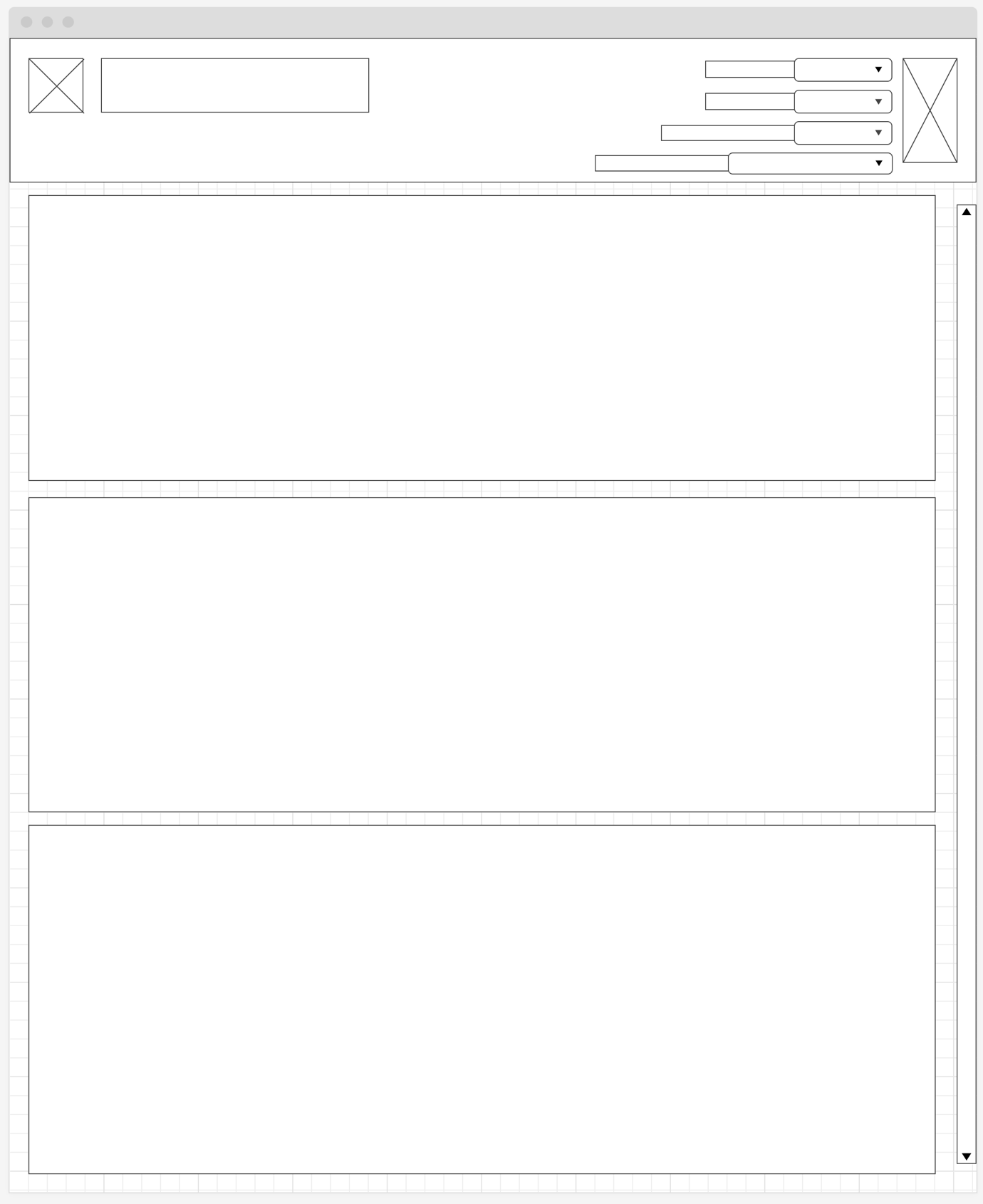
## Structural Design

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 interface is organized into several key sections:

* **Header**: Contains the system name and logo, with date selectors aligned to the right.
* **Search Bar & Dropdowns**: Below the header, users can set specific conditions and select desired search results via dropdown menus.
* **First Graph**: Positioned further down to provide an initial data overview.
* **Second Graph**: Offers a more detailed data analysis.
* **Table**: Situated at the bottom for granular data representation.

A vertical scroll is implemented to accommodate smaller screens or situations where a vertical layout is more practical.



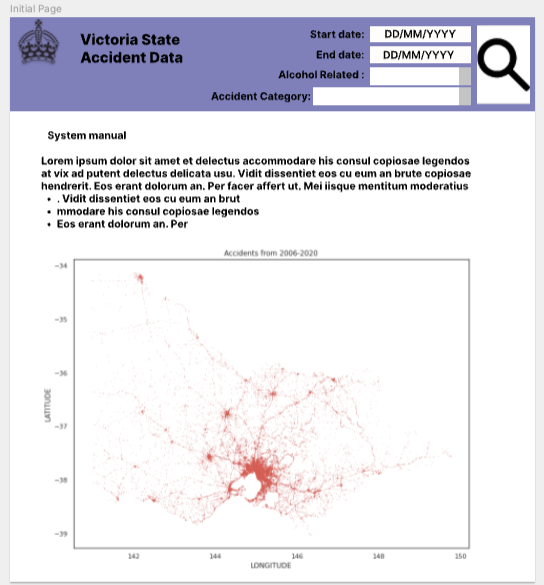
**Justification**

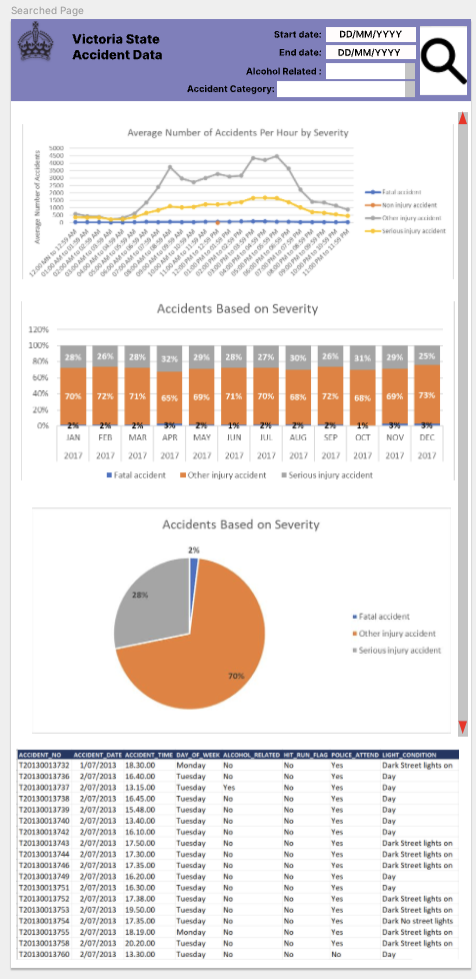
The header was designed for instant recognition, housing the system name and logo. Dropdowns are below the header to facilitate condition-based searching. Graphs and tables follow in a logical flow to present data from a general overview to specific details. The vertical scroll option was added to ensure user-friendliness regardless of screen size.

## 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**: The layout is dynamic, able to extend vertically or adapt via a scrollbar, depending on the user’s screen size.
* **Visual Elements**: Minimalistic shadows are used to elevate elements.
* **Icons & Graphics**: Simple, easy-to-understand icons are employed.
* **Style**: The design maintains a clean and modern aesthetic.
* **Color**: The header utilizes the signature color of the relevant government body to evoke familiarity and trust. The rest of the design sticks to neutral shades.
* **Fonts**: Sans-serif fonts are chosen for readability and modernity.





**Justification**

The dynamic layout was chosen for adaptability, ensuring an optimal user experience irrespective of device dimensions. Minimalistic shadows add depth without causing distractions. The government's signature color in the header was incorporated to build a sense of trust and familiarity with the users.

# REFERENCES

Steinhauser, R., & Lancsar, E. (2022, September). Social Cost of Road Crashes: *Report for the Bureau of Infrastructure and Transport Research Economics* (Final Report). The Australian National University. <https://www.bitre.gov.au/sites/default/files/documents/social-cost-of-road-crashes.pdf>