Software Design Document

Analysis tools of Victoria State Accident

S5288771 Chi Pang Cheng

S5323472 Kwangseok Choi

Table of Contents

[1.0 System Vision 3](#_Toc144412855)

[1.1 Problem Background 3](#_Toc144412856)

[1.2 System Overview 3](#_Toc144412857)

[1.3 Potential Benefits 3](#_Toc144412858)

[2.0 Requirements 4](#_Toc144412859)

[2.1 User Requirements 4](#_Toc144412860)

[2.2 Software Requirements 4](#_Toc144412861)

[2.3 Use Cases & Use Case Diagrams 4](#_Toc144412862)

[3.0 Software Design and System Components 7](#_Toc144412863)

[3.1 Software Design 7](#_Toc144412864)

[3.2 System Components 8](#_Toc144412865)

[3.2.1 Functions 8](#_Toc144412866)

[3.2.2 Data Structures / Data Sources 8](#_Toc144412867)

[3.2.3 Detailed Design 9](#_Toc144412868)

[4.0 User Interface Design 11](#_Toc144412869)

[4.1 Structural Design 11](#_Toc144412870)

[4.2 Visual Design 12](#_Toc144412871)

[Layout 12](#_Toc144412872)

[Visual elements 12](#_Toc144412873)

[Icons 12](#_Toc144412874)

[Colour 12](#_Toc144412875)

[Fonts 12](#_Toc144412876)

# System Vision

## Problem Background

By analysing data, organizations can identify the decision of their next business move, any update to their policy, and understand the trends related to their business. However, data usually comes with a huge set of data which may include data from decades, and unrelated data for business. To analyse this data often requires a lot of human resources, consuming huge amounts of time, and slowing the business process.

This project is aimed at developing software for data analysis to help the target audience fasten their data analysis process. Moreover, it helps the target audience be capable of keeping track of the latest trends related to their business. The dataset used on this project is about road crash accidents in Victoria State from 2013 to 2019 with over seventy thousand records. Each record contains sixty-three columns related to the details of the accident.

## System Overview

The system should be operated through a graphical user interface. It takes user inputs to search in the CSV file which inputs maybe a period of time/date, accident type, or alcohol related accident, etc. It should provide selections for users to decide what information to show on the system. Furthermore, the system should be able to produce a chart as per user request for them to easily analyse the data or clear to present the trends. For example, the user entered a period of time and only selected to display the related accident type. Then, the system will search for the accident type at the given time. The output should only contain two columns, which are date/time, and the accident type. If the user would like to generate a chart, there should be an option for the user to click on.

## Potential Benefits

To identify the benefits for our users, this project needs to first decide on who will be our target audience. The first thought of our target audiences are law enforcement agencies and insurance companies. As the dataset was released by law enforcement agencies, we believe that they should already have their own system for analysing the data. Therefore, the target audience of this project is insurance companies. Insurance companies will use the system to analyse the data and review or modify their policy based on the processed data. The potential benefits of using the system are listed below.

* Time is reduced on analysing raw data. Users only need to give a few inputs and click on some buttons, then the system can transfer the raw data to processed data as the user expected.
* Cost is reduced on processing raw data. Once insurance companies receive the raw data, they may need a certain number of human resources to manually separate the data based on different requirements. After that, they need another number of people to analyse the data at the same time. By using the system, the person who generates the required data and the person who analyses the data can combine into one position due to the efficiency of the system, which helps companies to save money from hiring people.
* Human error is reduced when processing raw data. When insurance companies manually process the raw data, staff may accidentally mis-entered data or used incorrect algorithm, which led to incorrect business decisions. After the system is implemented, users only need to input keywords or make some selections. Then the system will process the request and use the most suitable algorithm to output the processed data.
* Reduced business loss of the insurance policy. By analysing the processed data outputted from the system, insurance companies can easily keep track of the latest trends, such as accident rates in the suburbs. After that, insurance companies can make the decision quickly whether they will increase or decrease the insurance fee for the residents who live nearby the suburb.

# Requirements

## User Requirements

Firstly, users need to specify the file name of the dataset that they want to import. After that, the system will load the dataset and get the field header name and display the selections.

Secondly, users need to select the header name that they want to use for retrieving the related record and input a value to the related header name.

Thirdly, users need to select the header name of the data that users would like to display. If the user did not select any header name, all the information that related to the search will be displayed.

Finally, users can decide whether they want to generate a chart for the related search or not.

As an example, a user from an insurance company would like to review the accident frequencies in different suburbs to update their policy. The user selected a period of date for the search and selected field names ACCIDENT\_DATE and LGA\_NAME for the display. Then, the system will use the user input time to find any matched in the dataset and display each matched record with the data inside the field names ACCIDENT\_DATE and LGA\_NAME.

## Software Requirements

The purpose of the software is designed for analysing data which should be easy to use and should not be too complex. The general functional requirements are listed below.

* The program shall accept CSV as file name from the command line.
* File name can be the full path of the file with one or more levels.
* The Program shall allow users to select one or more field for the search.
* The Program shall allow users to input key information into the selected field.
* The Program shall allow users to select one or more field for the display.
* The Program shall allow users to generate a chart of their search results.
* The Program shall allow users to choose the X,Y axis for their chart of the search results.

## Use Cases & Use Case Diagrams

Figure 1 below shows the general ideas of the user identity, the functions they will use on the software. The following use cases will give more details on those functions.

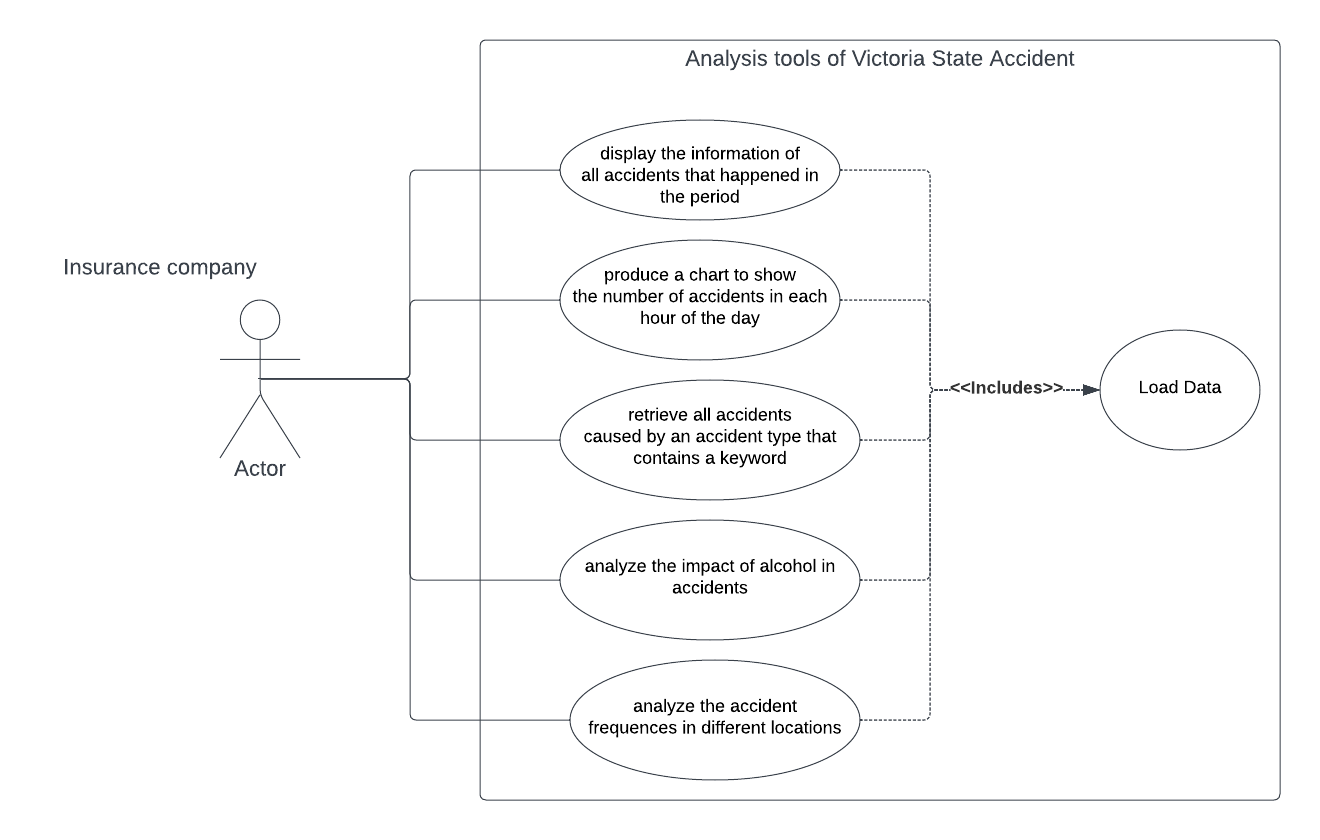


Figure 1: Analysis tools of Victoria State Accident

|  |  |
| --- | --- |
| Use case ID | 1 |
| Use case name | display the information of all accidents that happened in the period |
| Actor | Insurance company |
| Description | Information of all accidents of the actor selected period will be displayed |
| Flow of Events | 1. Insurance company will load the data to the system.  2. Insurance company will select a period of date/time.  3. Insurance company will select the field/s of information to display.  4. The system will display the requested information on the screen. |

|  |  |
| --- | --- |
| Use case ID | 2 |
| Use case name | produce a chart to show the number of accidents in each hour of the day |
| Actor | Insurance company |
| Description | Insurance company will be able to get a chart that shows the number of accidents in each hour of the day |
| Flow of Events | 1. Insurance company will load the data to the system.  2. Insurance company will select a period of date/time.  3. The system will display all accidents of the selected period  4. Insurance company will press the button of generating chart.  5. Insurance company will be prompted to select X,Y axis.  6. The system will generate a chart will the number of accidents in each hour of the day. |

|  |  |
| --- | --- |
| Use case ID | 3 |
| Use case name | retrieve all accidents caused by an accident type that contains a keyword |
| Actor | Insurance company |
| Description | Insurance company will be able to retrieve all accidents caused by an accident type that contains a keyword. |
| Flow of Events | 1. Insurance company will load the data to the system.  2. Insurance company will select ACCIDENT\_TYPE as the filter and enter key word.  3. Insurance company will select the field/s of information to display.  4. The system will display the requested information on the screen. |

|  |  |
| --- | --- |
| Use case ID | 4 |
| Use case name | analyse the impact of alcohol in accidents |
| Actor | Insurance company |
| Description | Insurance company will be able to analyse the impact of alcohol in accidents. |
| Flow of Events | 1. Insurance company will load the data to the system.  2. Insurance company will select ALCOHOLTIME as the filter and enter key word “Yes”.  3. Insurance company will select the field/s of information to display.  4. The system will display the requested information on the screen.  5. Insurance company will be given an option to generate a chart. |

|  |  |
| --- | --- |
| Use case ID | 5 |
| Use case name | analyse the accident frequency in different locations |
| Actor | Insurance company |
| Description | Insurance company will be able to analyse the accident frequency in different locations. |
| Flow of Events | 1. Insurance company will load the data to the system.  2. Insurance company will select a period of date/time.  3. Insurance company will select LGA\_NAME for the field to display.  4. The system will display the requested information on the screen.  5. Insurance company will be given an option to generate a chart. |

# Software Design and System Components

## Software Design

Figure 2 below shows the process and decisions that the software will interact with users. First, the system will ask users to provide a file name to load the data. If the user provided an incorrect file name, the system would ask the user to provide the file name again. After successfully loading the data, the system will take the header names as the label names which should be in the first row of the data. Second, users will need to decide what label/s they would like to use for searching and what label/s needs to be included in the output. Depending on the user's decision, reports will be output to the screen. Lastly, users will always have the choice to output a chart with their selected X,Y axis from the outputted report.

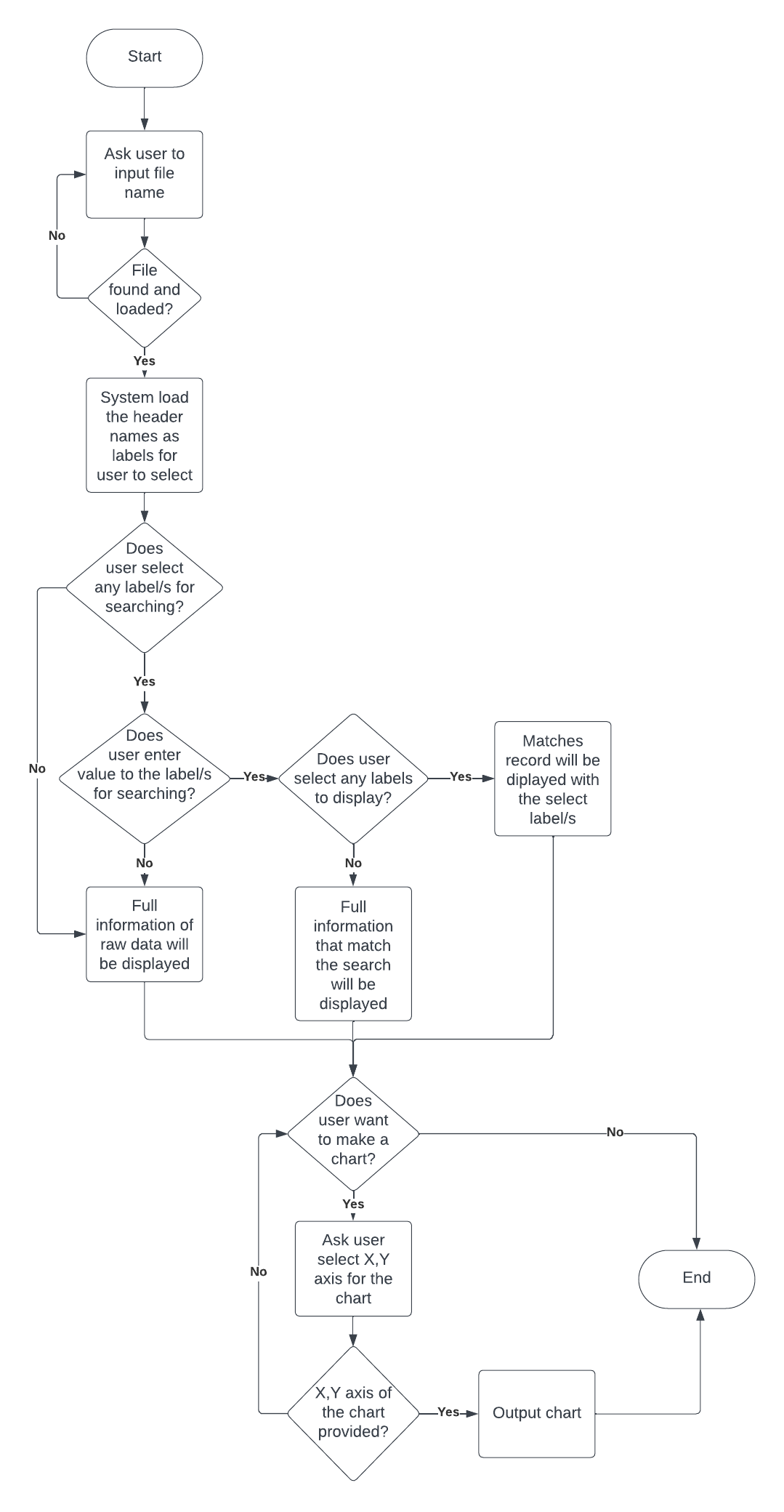


Figure 2: Flow chart of analysis tools of Victoria State Accidents

## System Components

### Functions

Function for loading the data:

This is the initial function of the system. Users will be prompted to ask for a file name for loading the data. After the user inputs the file name, this function should take the file name and load the data from the file. The parameters should use a string as a data type. As this is the initial function, data will be stored in a dictionary. The dictionary will be deleted after users close the system.

Function for searching and display:

Users will be provided with different labels for selection and a field to input the keyword of the selection. The system will take the inputted string to lookup into a dictionary and display the matched results on the screen. The matched results will be stored in a list which helps users easily generate the chart. Every time a user starts a new search, the previous results in the list will be rewritten to a new search result.

### Data Structures / Data Sources

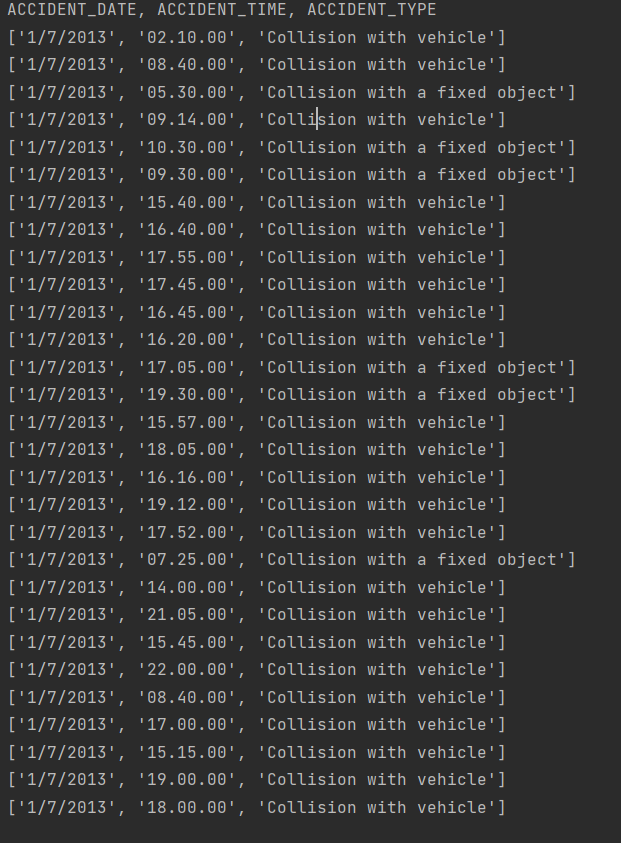
To maintain data integrity, all data from the file will be set as read only. Moreover, all attributes will be set as public. Data structures are listed below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Name: | Data Type: | Description: | Data members: | Shared functions: |
| label | List | First row of the data will be stored into the list. It used as labels for users to select and also used as a reference point to locate the data in another list or dictionary. | All columns name which should be stored as string. | +loadData  +searchData  +displayData |
| count | Int | This a flag for creating the “label” list. After stored value to the “label” list, value in count will become 1. | Itself. | +loadData |
| raw\_Data | dictionary | This is a dictionary that store all loaded data with “OBJECTID” as key. | OBJECTID as key.  Other 62 columns as value. | +loadData  +searchData |
| temp | List | This is a list that store temporary results of one record. After it append to another list, it will be overwritten by next record. | Results of the search which can be string or integer | +searchData |
| processed\_data | list | This is a list that store the all the results of the search. | This list may contain multiple lists and those lists may contain integer or string. | +searchData  +displayData |

### Detailed Design

Below are the Pseudocode for loading data from a file and providing results for a search.

import csv  
label = []  
raw\_data = {}  
count = 0  
processed\_data = []  
  
  
  
with open("./Crash Statistics Victoria.csv", 'r') as file:  
 csvreader = csv.reader(file)  
   
 for row in csvreader:  
 if count == 0:  
 label = row  
 count = 1  
 else:  
 raw\_data[row[0]]= row[1:-1]  
  
 if "Collision" in row[7] and "1/7/2013" == row[4]:  
 temp = []  
 temp.append(row[4])  
 temp.append(row[5])  
 temp.append(row[7])  
 processed\_data.append(temp)  
  
print(label[4], label[5], label[7], sep=", ")  
for i in processed\_data:  
 print(i)



# User Interface Design

In this stage, a low-fidelity user interface was designed. Wireframe is applied to the low-fidelity user interface which presents some key functions and layouts of the software. The wireframe shows the structure of the user interface and the functions used on different panels. By creating the wireframe, we have further developed the visual design which includes layout, visual elements, icons, colour, and font style.

## Structural Design

Figure 3 below shows the general structural design of the software. At this stage, the graphical user interface contains a tool bar and three panels. Buttons in the tool bar include importing files, searching, or generating charts.

After the user has imported a file, the left panel will show the file name and all the field names of the file. The right panel will show all the records from the file. The bottom panel will show the statistical results on each field, which may be presented as percentages or numbers.

If the user selects some field and gives input to the field, then press on the search button. The right panel will show all the filtered results related to the selected field. Statistical results on the bottom panel will also be affected.

After users have retrieved the results, they can use the generate chart button in the tool bar and select the X,Y axis for generating the chart.

A screenshot of a computer

Description automatically generated

Figure 3: Wireframe of analysis tools of Victoria State Accidents

## Visual Design

Refer to section 4.1 and figure 3 above, the visual design details will be described below.

### Layout

Tool bar – there are serval buttons which contain import file, search, and generate chart.

Left panel – here should shows the file name of the user imported, label names, option to select label, input field for the label, and a scroll bar for user to further explore the label options.

Right panel – here will shows the results related to the user input on the left panel. First row should be the field name and the rest will be the results.

Bottom panel – here will shows some statistical results related to the search. So, user does not need to count each record for the analysing the results.

### Visual elements

As this software is used for informational purposes, it should be able to clearly display and watch. Therefore, background colour and font colour should remain close to white and black.

For the generated chart, different colours will be used for the purposes of clearly identifying the lines. Sometimes one line may overlap another line. Using different colours may help the user easily identify the meaning of the lines.

### Icons

Icons should be meaningful which can represent their usage. For example, magnifying-glass represent search function, bar chart icon represents chart generation function, etc. Used icons should be consistent with the icons used on other applications.

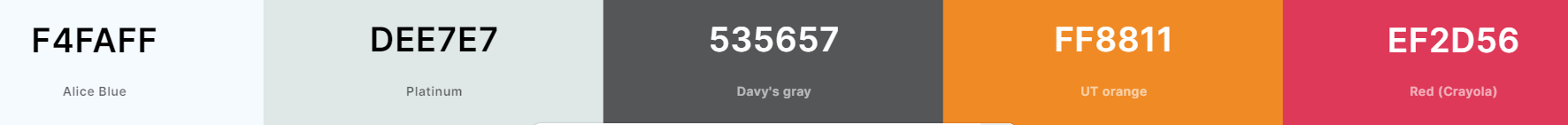
### Colour

Colour #F4FAFF will be used for the general background colour which is not too bright to watch.

Colour #535657 will be used as font colour. It is the contrast colour with the background. User can easily read the content applied with this colour.

Colour #DEE7E7 will be use on the first row of right panel. It is used for indicating that the row is the header of the results.

Colours #FF8811 and #EF2D56 can be used on the lines of the generated chart. As the colours are sharp, user can easily for the flow of the lines.



### Fonts

Fonts should be formal and easy to read. Therefore, sans serif will be applied to the software. This font style is most commonly used on different applications which users should be familiar with. Moreover, the spacing between characters is appropriate for reading, which will not overlap other characters in the same word.