Project Plan

-NSW Traffic Data Analysis Tool

## (Group 87)

## Anuj Khurana – s5281041

## Puneet - s5296567

## Raghav Kohli - s5326153

Table of Contents

[1.0 Introduction 3](#_Toc144513734)

[1.1 Background 3](#_Toc144513735)

[1.2 System Overview 3](#_Toc144513736)

[1.3 Potential Benefits 3](#_Toc144513737)

[1.4 Scope 3](#_Toc144513738)

[1.5 Document Contents 4](#_Toc144513739)

[2.0 Work Breakdown Structure 4](#_Toc144513740)

[3.0 Activity Definition & Estimation 5](#_Toc144513741)

[4.0 Gantt Chart 8](#_Toc144513742)

# 1.0 Introduction

Traffic penalty data analysis is crucial for understanding and improving road safety. This project aims to develop a Python-based tool with a graphical user interface (GUI) to facilitate the analysis and visualization of the New South Wales (NSW) Traffic Penalty Data dataset.

## 1.1 Background

This project embarks on an extensive examination of penalties issued in the New South Wales (NSW) region of Australia. The dataset, acquired from the NSW Revenue Office, covers the period from January 2012 to November 2017, offering an in-depth look into six years of fine issuance activities. Within this dataset, you will find detailed information about each penalty, encompassing details such as the nature of the violation, the legislative framework, the location, the penalty amount, and pertinent indicators.

The NSW Traffic Penalty Data analysis and visualization tool has been developed to empower users to analyse and visualize penalty data associated with traffic violations. This tool facilitates users in gaining insights into penalty cases, the distribution of offense codes, instances captured by radar or cameras, and trends concerning offenses related to mobile phone usage.

## 1.2 System Overview

The proposed system will provide a graphical user interface (GUI) for users to interact with the traffic penalty dataset. It will encompass functionalities like data retrieval, visualization, and analysis. The primary objective of this system is to convert raw data into practical insights that can benefit law enforcement, policymakers, and researchers.

## 1.3 Potential Benefits

The system offers several potential advantages, including:

* Streamlined data analysis and visualization processes.
* Enhanced decision-making capabilities in traffic management and law enforcement.
* Improved user experience through an intuitive graphical user interface (GUI).
* Access to insights regarding trends and patterns associated with traffic violations.
* The capacity to pinpoint and tackle specific concerns, such as mobile phone usage while driving.

## 1.4 Scope

The project encompasses the creation and execution of a user-friendly interface that empowers users to:

* Choose a particular timeframe for analysis.
* Generate reports concerning penalty cases.
* Visualize the distribution of cases based on offense codes.
* Retrieve cases associated with mobile phone usage.
* Conduct trend analysis for mobile phone usage cases across different time periods.
* Perform additional analyses as needed, tailored to the user's requirements.

## 1.5 Document Contents

This document provides an overview of the project plan for constructing the data analysis and visualization tool for the NSW Traffic Penalty Data. It encompasses the following components:

Project Plan; This thorough project plan makes sure that every work has a distinct goal and justification, enabling efficient project management and successful completion within the allotted six weeks. Note: Depending on the project's complexity and particular requirements, activity durations may change.

1. Work Breakdown Structure (WBS) - A hierarchical breakdown of project tasks and subtasks.

2. Activity Definition - Clear definitions and descriptions of each project activity.

3. Estimation - Estimations of time, resources, and effort required for each activity.

4. Gantt Chart - A visual representation of project scheduling, depicting task timelines and dependencies.

# 2.0 Work Breakdown Structure

The hierarchical organisation of project tasks and activities is shown in the Work organisation Structure (WBS). All activities pertaining to the project are presented in an organised manner. The following is the WBS for the project for the NSW Traffic Penalty Data Analysis Tool:

**Week 1-2: Initiation Phase**

* 1.1 Project Kickoff
* 1.2 Requirements Gathering
* 1.3 Define Project Scope and Objectives

**Week 3-4: Planning Phase**

* 2.1 Define Work Breakdown Structure (WBS)
* 2.2 Resource Allocation
* 2.3 Risk Assessment
* 2.4 Project Schedule (Gantt Chart)
* 2.5 Define Project Plan

**Week 5-6: Design Phase**

* 3.1 Define User Interface Specifications
* 3.2 Design the Graphical User Interface (GUI)
* 3.3 Define Software Design Document

**Week 7-8: Data Preparation Phase**

* 4.1 Analyse the Dataset and Data Processing Requirements
* 4.2 Database Setup
* 4.3 Data Import and Pre-processing

**Week 9-10: Development Phase**

* 4.4 Implement Core Functionality
* 4.5 Implement Data Analysis Algorithms using Python.
* 4.6 Implement Data Visualization using Libraries.
* 4.7 User Interface Design Development
* 4.8 Integrate User Interface with Analysis Modules

**Week 11: Testing Phase**

* 5.1 User Interface Testing
* 5.2 Perform Integration Testing

**Week 12: Deployment and Closing Phase**

* 6.1 Package Application
* 6.2 Prepare Tool for End-Users
* 6.3 Project Review and Evaluation
* 6.4 Project Closure

This hierarchical WBS provides a structured breakdown of the project tasks into phases and sub-tasks, making it easier to manage and track progress throughout the project. Each sub-task is associated with its respective phase for clarity and organization.

# 3.0 Activity Definition & Estimation

A thorough explanation of each task is listed in the Work Breakdown Structure (WBS) in this section, along with an expected time frame for each activity. Please be aware that as the project develops and additional information becomes available, these estimates will be revised.

**Week 1-2: Initiation Phase**

1.1 Project Kickoff (1 days)

* *Reasoning:* The project kickoff is essential to align team members, set expectations, and establish a clear understanding of the project's goals and objectives.

1.2 Requirements Gathering (2 days)

* *Reasoning:* Thorough requirements gathering ensures a complete understanding of user needs and guides project planning and development.

1.3 Define Project Scope (2 days)

* *Reasoning:* Clearly defining the project's scope sets boundaries and prevents scope creep, ensuring the project stays on track.

**Week 3-4: Planning Phase**

2.1 Define Work Breakdown Structure (WBS) (2 days)

* *Reasoning:* Creating a detailed WBS breaks the project into manageable tasks and aids in project management and tracking.

2.2 Resource Allocation (2 days)

* *Reasoning:* Allocating resources ensures that the necessary team members, tools, and equipment are available as needed.

2.3 Risk Assessment (2 days)

* *Reasoning:* Identifying potential risks and developing a risk mitigation plan is crucial to proactively manage and reduce project risks.

2.4 Project Schedule (Gantt Chart) (2 days)

* *Reasoning:* Developing a project schedule ensures that tasks are organized, sequenced, and assigned realistic timeframes, facilitating project tracking and completion.

2.5 Define Project Plan (1 days)

* *Reasoning:* Creating a comprehensive project plan documents the overall approach, goals, and deliverables of the project, guiding all subsequent phases.

**Week 5-6: Design Phase**

3.1 Define User Interface Specifications (2 days)

* *Reasoning:* Detailed UI specifications provide a clear vision for the GUI, aligning design efforts with user expectations.

3.2 Design the Graphical User Interface (GUI) (4 days)

* *Reasoning:* Designing the GUI based on specifications ensures an intuitive and user-friendly interface.

3.3 Define Software Design Document (2 days)

* *Reasoning:* Documenting the software design provides a blueprint for development, making it easier to implement and maintain the code.

**Week 7-8: Data Preparation Phase**

4.1 Analyse the Dataset and Data Processing Requirements (4 days)

* *Reasoning:* Understanding data requirements is critical for setting up the database and pre-processing data effectively.

4.2 Database Setup (2 days)

* *Reasoning:* Setting up the database ensures that data is stored securely and efficiently for analysis.

4.3 Data Import and Pre-processing (4 days)

* *Reasoning:* Proper data pre-processing ensures data accuracy and prepares it for analysis and visualization.

**Week 9-10: Development Phase**

4.4 Implement Core Functionality (8 days)

* *Reasoning:* Implementing core functionality ensures that the tool can perform its primary tasks effectively.

4.5 Implement Data Analysis Algorithms using Python (8 days)

* *Reasoning:* Developing data analysis algorithms is essential for deriving meaningful insights from the dataset.

4.6 Implement Data Visualization using Libraries (8 days)

* *Reasoning:* Data visualization enhances understanding and insight into the data.

4.7 User Interface Design Development (2 days)

* *Reasoning:* Developing the GUI based on design specifications ensures a cohesive and visually appealing interface.

4.8 Integrate User Interface with Analysis Modules (2 days)

* *Reasoning:* Integration ensures that the GUI effectively interacts with the analysis modules.

**Week 11: Testing Phase**

5.1 User Interface Testing (2 days)

* *Reasoning:* Testing the UI helps identify and resolve any usability or functionality issues.

5.2 Perform Integration Testing (2 days)

* *Reasoning:* Integration testing ensures that all components work together seamlessly.

**Week 12: Deployment and Closing Phase**

6.1 Package Application (1 days)

* *Reasoning:* Packaging the application prepares it for deployment to end-users.

6.2 Prepare Tool for End-Users (1 days)

* *Reasoning:* Preparing the tool includes documentation, training, and support to ensure a smooth transition for end-users.

6.3 Project Review and Evaluation (2 days)

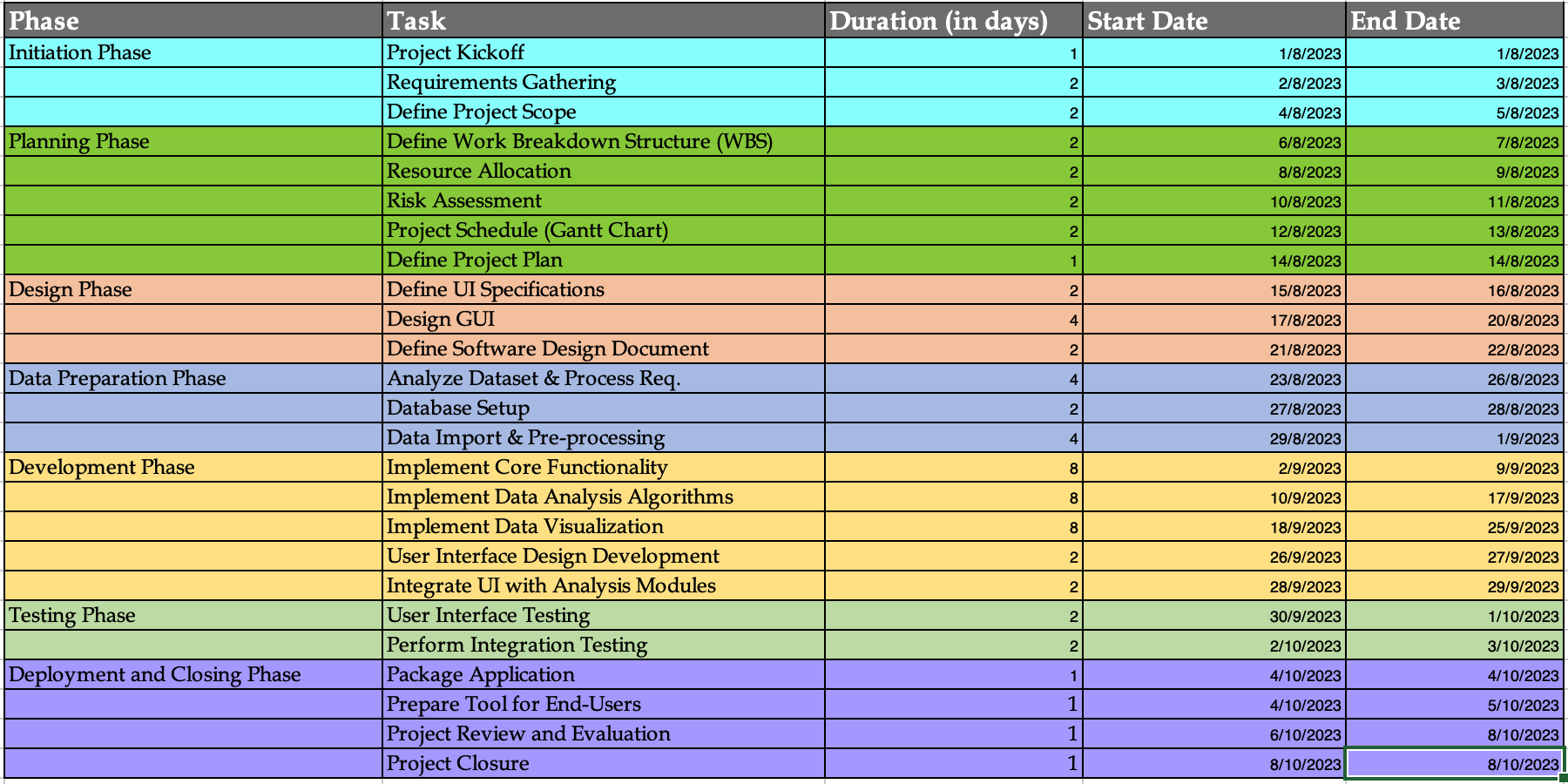
* *Reasoning:* Reviewing and evaluating the project outcomes helps identify lessons learned and areas for improvement.

6.4 Project Closure (1 days)

* *Reasoning:* Proper project closure includes archiving project files, conducting a final assessment of project goals, and ensuring a smooth transition to maintenance and support.

# 4.0 Gantt Chart

Based on the identified tasks and projected timeframes, the Gantt chart depicts the project schedule. Please be aware that this is only a high-level timetable and that it may change as the project progresses.



A graph with a number and a number

Description automatically generated with medium confidence