Project Plan

Data Analysis & Visualisation Tool (DAVT)

s5273814 - Nathanael Gazzard

s5309988 - Stephen Urquhart

s5318167 - Jahanzaib

Table of Contents

[1.0 Introduction 3](#_Toc147696951)

[1.1 Background 3](#_Toc147696952)

[1.2 Scope 3](#_Toc147696953)

[1.3 Document contents 3](#_Toc147696954)

[2.0 Work Breakdown Structure 4](#_Toc147696955)

[3.0 Activity Definition & Estimation 5](#_Toc147696956)

[4.0 Gantt Chart 8](#_Toc147696957)

[4.1 Updated Gantt Chart 9](#_Toc147696958)

[4.2 Apendix 10](#_Toc147696959)

# Introduction

## Background

The goal of this project is the development of a data analysis and visualization tool (DAVT) specifically for the analysis and visualisation of a dataset of ~400k New York Restaurant inspection Results hosted on Kaggle.com.

## Scope

This project will produce a DAVT with a graphical user interface (GUI) that the user will perform both data analysis and data visualisation tasks with. The tasks they can perform will be limited to the following set:

1. The user can retrieve all inspection details for a specified period.

2. The user can plot violations distributed on a per-suburb basis.

3. The user can retrieve all violations containing a specified keyword for a specified period.

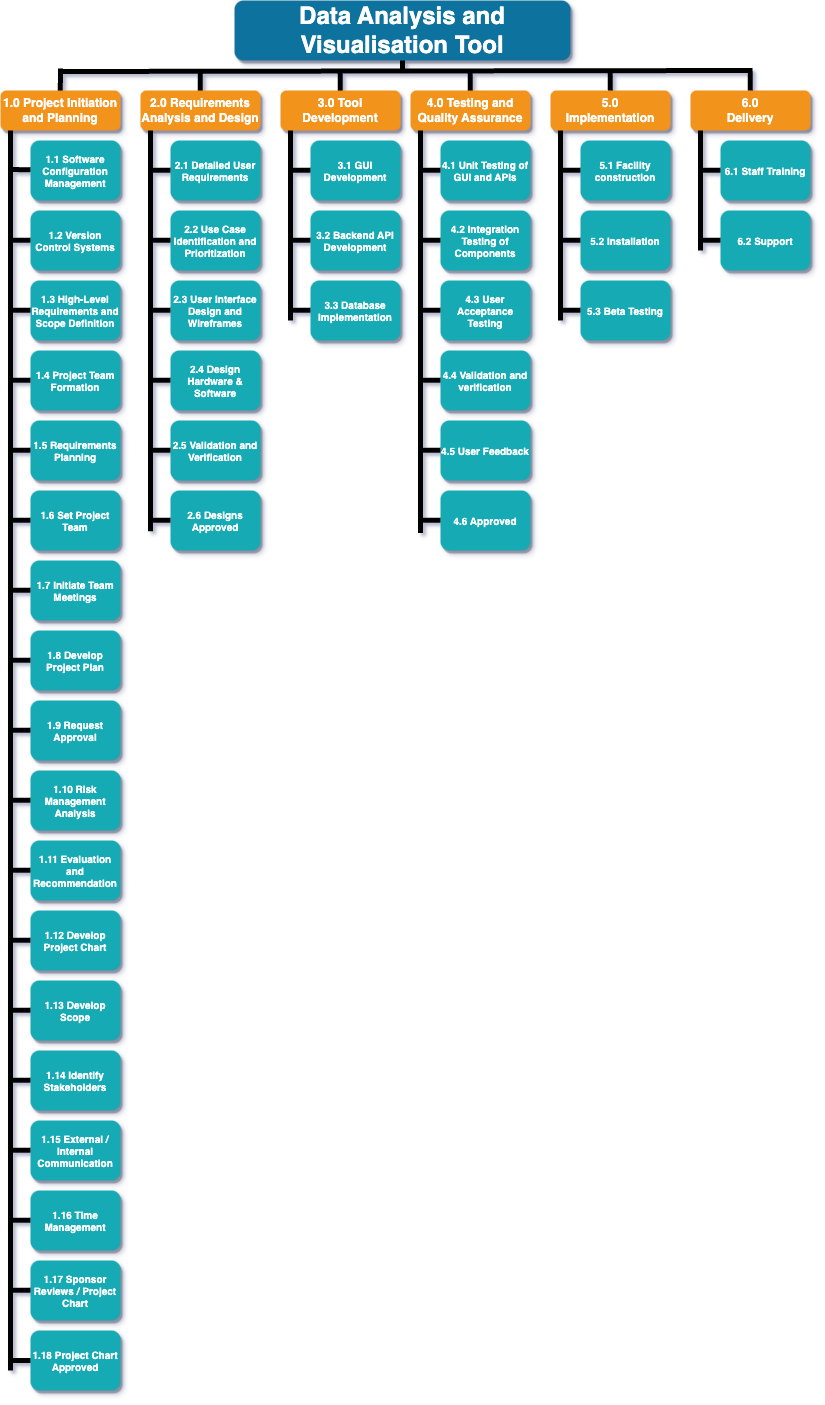
4. The user can map violations pertaining to animals distributed over time and suburbs.

5. The user can pull the 100 places with the best improvement over the last year per boro.

## Document contents

This document contains an introduction to the problem space, the objective of this project and a breakdown of all the tasks required for the completion of this project, detailed in the form of a Work Breakdown Structure, Activity Definition and Estimation and a Gantt chart.

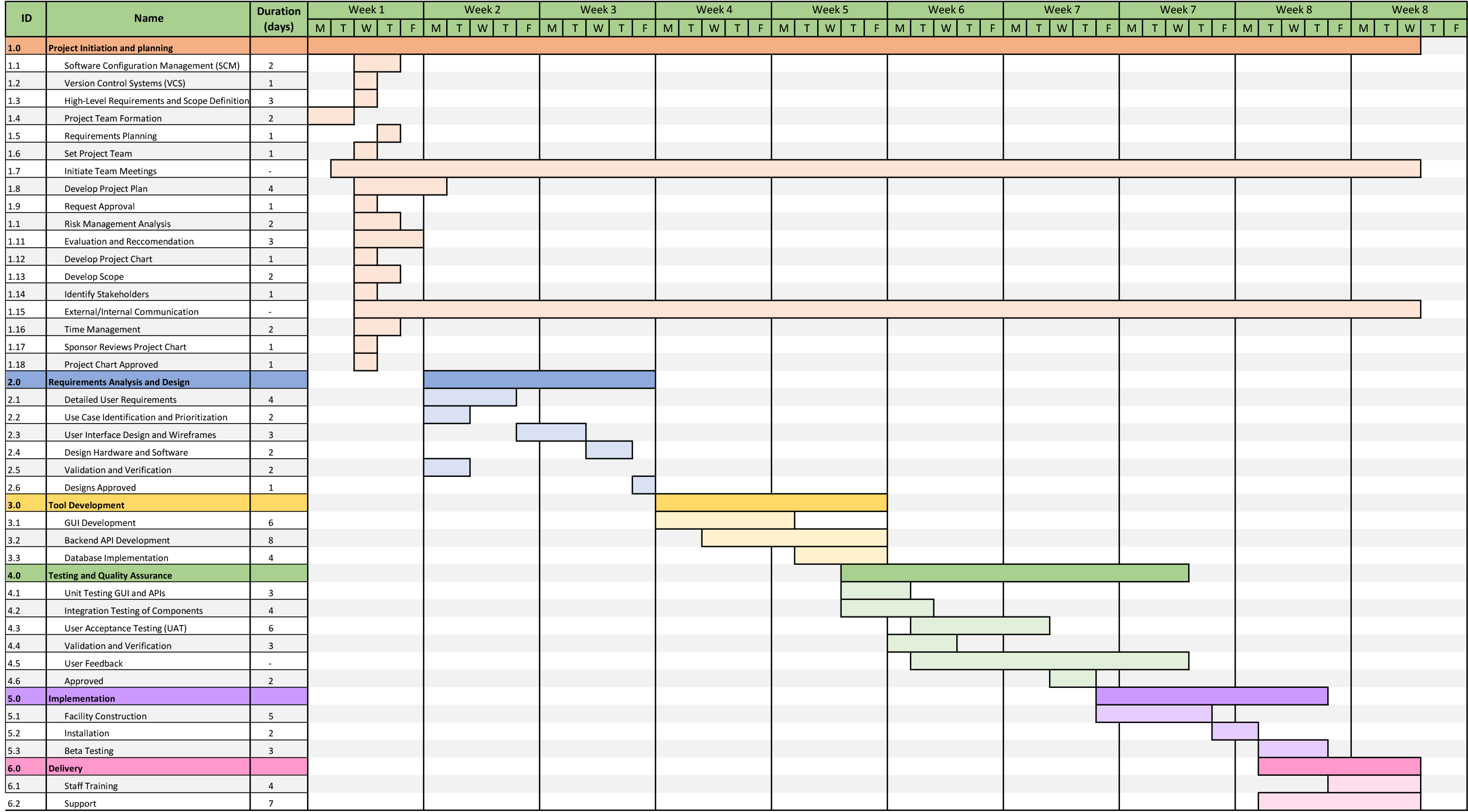
# Work Breakdown Structure



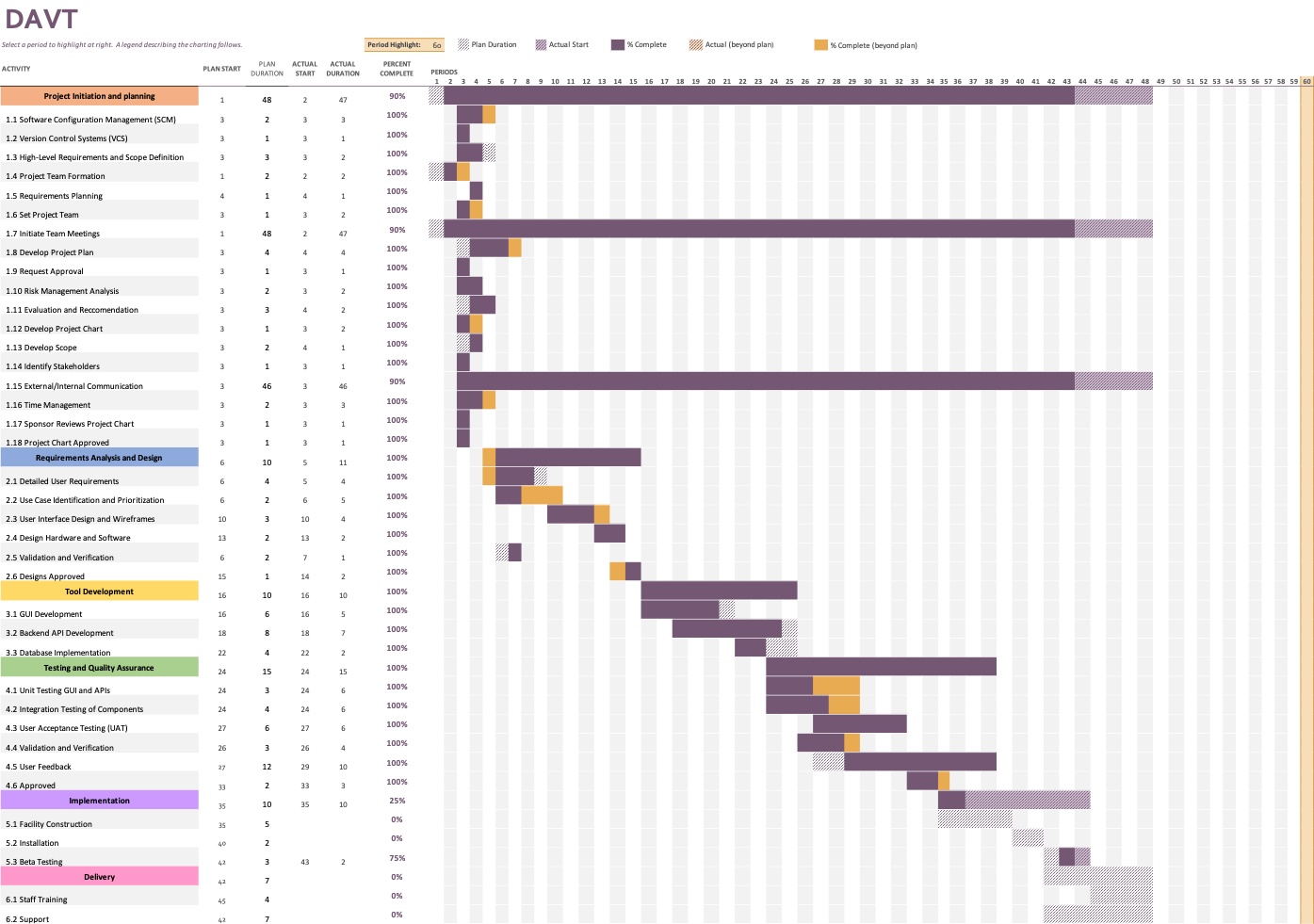
# Activity Definition & Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| ***Id*** | ***Name*** | ***Description*** | ***Estimated Duration*** |
| **1.0 Project Initiation & Planning** | |  |  |
| *1.1* | Software Configuration Management (SCM) | *Define and implement a process for managing software configurations.* | *2* |
| *1.2* | Version Control Systems (VCS) | *Set up a Version Control System (e.g., GitHub) for source code management* | *1* |
| *1.3* | High-Level Requirements and Scope Definition | 1. *Gather high-level requirements.* 2. *Define and document the scope of the project.* | *3* |
| *1.4* | Project Team Formation | 1. *Identify roles and responsibilities for project team members.* 2. *Assemble the project team.* | *2* |
| *1.5* | Requirements Planning | 1. *Plan the requirements gathering process.* 2. *Identify stakeholders and users to be involved* | *1* |
| *1.6* | Set Project Team | 1. *Assign specific roles and responsibilities to each team member.* 2. *Ensure everyone is clear on their tasks.* | *1* |
| *1.7* | Initiate Team Meetings | *Schedule and hold initial team meetings to discuss project goals and expectations.* | *Ongoing* |
| *1.8* | Develop Project Plan | *Create a detailed project plan outlining tasks, dependencies, and timelines.* | *4* |
| *1.9* | Request Approval | *Present the project plan to stakeholders for approval* | *1* |
| *1.10* | Risk Management Analysis | 1. *Identify potential project risks and their impacts.* 2. *Develop strategies to mitigate or manage risks.* | *2* |
| *1.11* | Evaluation and Recommendation | 1. *Evaluate potential tools, technologies, and methodologies for the project.* 2. *Make recommendations based on evaluations.* | *3* |
| *1.12* | Develop Project Chart | *Create a visual representation of the project's timeline and milestones.* | *1* |
| *1.13* | Develop Scope | 1. *Further define and clarify the project scope.* 2. *Ensure alignment with stakeholder expectations.* | *2* |
| *1.14* | Identify Stakeholders | 1. *List all individuals, groups, or organizations affected by the project.* 2. *Determine their roles and influence on the project.* | *1* |
| *1.15* | External/Internal Communication | 1. *Establish a communication plan for stakeholders and team members.* 2. *Determine frequency, channels, and content of communication.* | *Ongoing* |
| *1.16* | Time Management | 1. *Plan and allocate time for each project phase & task.* 2. *Develop a project schedule.* | *2* |
| *1.17* | Sponsor Reviews Project Chart | *Present the project chart to the sponsor for review/feedback* | *1* |
| *1.18* | Project Chart Approved | *Obtain sponsor approval for the project chart and plan* | *1* |
| **2.0 Requirements Analysis & Design** | |  |  |
| *2.1* | Detailed User Requirements | 1. *Conduct interviews/surveys to gather detailed user requirements.* 2. *Create user stories and use cases.* | *4* |
| *2.2* | Use Case Identification and Prioritization | 1. *Identify key use cases for the tool based on user requirements.* 2. *Prioritize use cases based on their importance and impact.* | *2* |
| *2.3* | User Interface Design and Wireframes | 1. *Design the graphical user interface (GUI) for the tool.* 2. *Create wireframes to visualize layout and interactions.* | *3* |
| *2.4* | Design Hardware & Software | 1. *Define the hardware & software architecture for the tool.* 2. *Identify technology stacks, frameworks, and libraries.* | *2* |
| *2.5* | Validation and Verification | 1. *Review and validate the design against user requirements.* 2. *Verify that the design aligns with project goals.* | *2* |
| *2.6* | Designs Approved | *Obtain stakeholder approval for the user interface and software architecture* | *1* |
| **3.0 Tool Development** | |  |  |
| *3.1* | GUI Development | 1. *Develop the graphical user interface.* 2. *Implement period selection/plotting/search/analysis interfaces.* | *6* |
| *3.2* | Backend API Development | 1. *Design and implement APIs to handle data retrieval and processing.* 2. *Develop APIs for inspection details, violation distribution, keyword search, and animal-related analysis.* | *8* |
| *3.3* | Database Implementation | 1. *Set up the database schema for storing inspection data.* 2. *Implement CRUD (Create, Read, Update, Delete) operations.* | *4* |
| **4.0 Testing and Quality Assurance** | |  |  |
| *4.1* | Unit Testing of GUI and APIs | 1. *Develop and execute unit tests for GUI components and APIs.* 2. *Ensure individual components function as intended.* | *3* |
| *4.2* | Integration Testing of Components | 1. *Test the integration between GUI, APIs, and database.* 2. *Validate that components work together seamlessly.* | *4* |
| *4.3* | User Acceptance Testing (UAT) | 1. *Collaborate with users to perform UAT on the tool.* 2. *Address feedback and ensure user satisfaction.* | *6* |
| *4.4* | Validation and Verification | 1. *Review and validate the developed tool against requirements.* 2. *Verify that the tool meets user needs and expectations.* | *3* |
| *4.5* | User Feedback | 1. *Gather user feedback during UAT and testing phases* 2. *Incorporate feedback to improve the tool.* | *Ongoing* |
| *4.6* | Approved | 1. *Obtain stakeholder and user approval for the tested and refined tool* 2. *Ensure the tool is ready for deployment.* | *2* |
| **5.0 Implementation** | |  |  |
| *5.1* | Facility Construction | 1. *Setup the required infrastructure and environment for tool deployment* 2. *Ensure the necessary hardware, software, and resources are in place* | *5* |
| *5.2* | Installation | 1. Hardware Setup and Assembly 2. Network Configuration and Integration 3. Equipment Calibration and Testing | *2* |
| *5.3* | Beta Testing | 1. Unit Testing of Components 2. Integration Testing 3. Performance Testing 4. Quality Control and Defect Resolution | *3* |
| **6.0 Delivery** | |  |  |
| *6.1* | Staff Training | 1. End-User Training 2. Technical Documentation Creation 3. User Manuals and Guides. 4. Training Materials Preparation | *4* |
| *6.2* | Support | 1. System Monitoring and Troubleshooting Setup. 2. User Support Setup. 3. Knowledge Transfer and Documentation Handover 4. Final Stakeholder Review and Approval | *7* |

# Gantt Chart



# Updated Gantt Chart



# Apendix

Due to time constraints, after finalising the application's functionality, we could not implement the styles (custom colour palette, font, shapes, etc) planned in our original Software Design Document. As such, our application uses the default native styles of the wxPython library.