**Project Plan**

**AJJ BNB**

**Student Names:**

**Ashley Pergoliti (s5311775)**

**Jonas Sajonas (s5284977)**

**Jodie Thomson (s118338)**

Table of Contents

[1.0 Introduction 3](#_Toc144563369)

[1.1 Background 3](#_Toc144563370)

[1.2 Scope scope and outline of the project management document 3](#_Toc144563371)

[1.3 Document contents 4](#_Toc144563372)

[2.0 Work Breakdown Structure 5](#_Toc144563373)

[3.0 Activity Definition & Estimation 5](#_Toc144563374)

[adjust total times for each section as things over lap 5](#_Toc144563375)

[4.0 Gantt Chart add copy here as well as sep attach Talk about the critical path in intro here 7](#_Toc144563376)

# Introduction

**Should contain an overview of the project (from a project management/component perspective)**

The project proposal to AJJ BNB is contained within this document and the accompanying software design document. The proposal is to create a software solution that extracts and analyses the public Sydney Airbnb data set hosted on kaggle.com. The solution will enable the company to become data-driven and make informed insights and decisions on the Airbnb spaces and trends to compete in a challenging market.

## Background

In today's data-driven business landscape, utilising gathered data means gaining the edge in the competition with rivals. AJJ BNB is at the forefront of the accommodation rivalry in Sydney, they will benefit the most by staying ahead of the competition. Sydney is a sought-after destination for travellers, tourists, and businesses alike. At the same time, Airbnb is a popular choice for accommodation and a lucrative investment choice for property owners. Kaggle.com provides access to Sydney Airbnb Open Data. The data provided includes detailed listings, pricing, calendar variances, reviews, and more.

The proposed software aims to provide a user-friendly, quick access point to specific information and trends about the Sydney Airbnb market. Users can receive charted or listed data tailored to their specific requirements in as few steps as possible.

## Scope scope and outline of the project management document

The scope of this document covers the project management aspect of the proposed project. The supporting documentation included has been developed to plan out and structure the project tasks as well as allocate time estimates required to produce individual deliverables as well as the duration of the entire project. The project management documents are presented as deliverables themselves and are then used to staff and fund the project as well as provide phase timing to the client.

The included Work Breakdown Structure covers all the deliverables for the project and lays them out in sections including project management, software design, software implementation, testing and executive summaries. Each Section has a group of sub sections shown beneath.

The activity definition uses the deliverable structure from the WBS to provide a time estimate for each task. This is then used to create the Gantt chart

The Gantt chart shows a visual representation of the project’s timeline. The time estimates for each task are highlighted as well as showing where tasks are occurring concurrently. The Gantt chart shows the overall time commitments as well as the time required to perform each part. The Gantt chart allows us the find the critical path to the project identifying which tasks must be completed before others can commence

## Document contents

This document serves as an overview of the project plan. The Software Design Document contains detailed software information that thoroughly examines the software solution. A higher-resolution Gantt chart is also available for further clarity.

**Contents of this Document:**

* Background
* Scope
* Work Breakdown Structure
* Activity Definition and Estimation
* Gantt Chart

# 2.0 Work Breakdown Structure

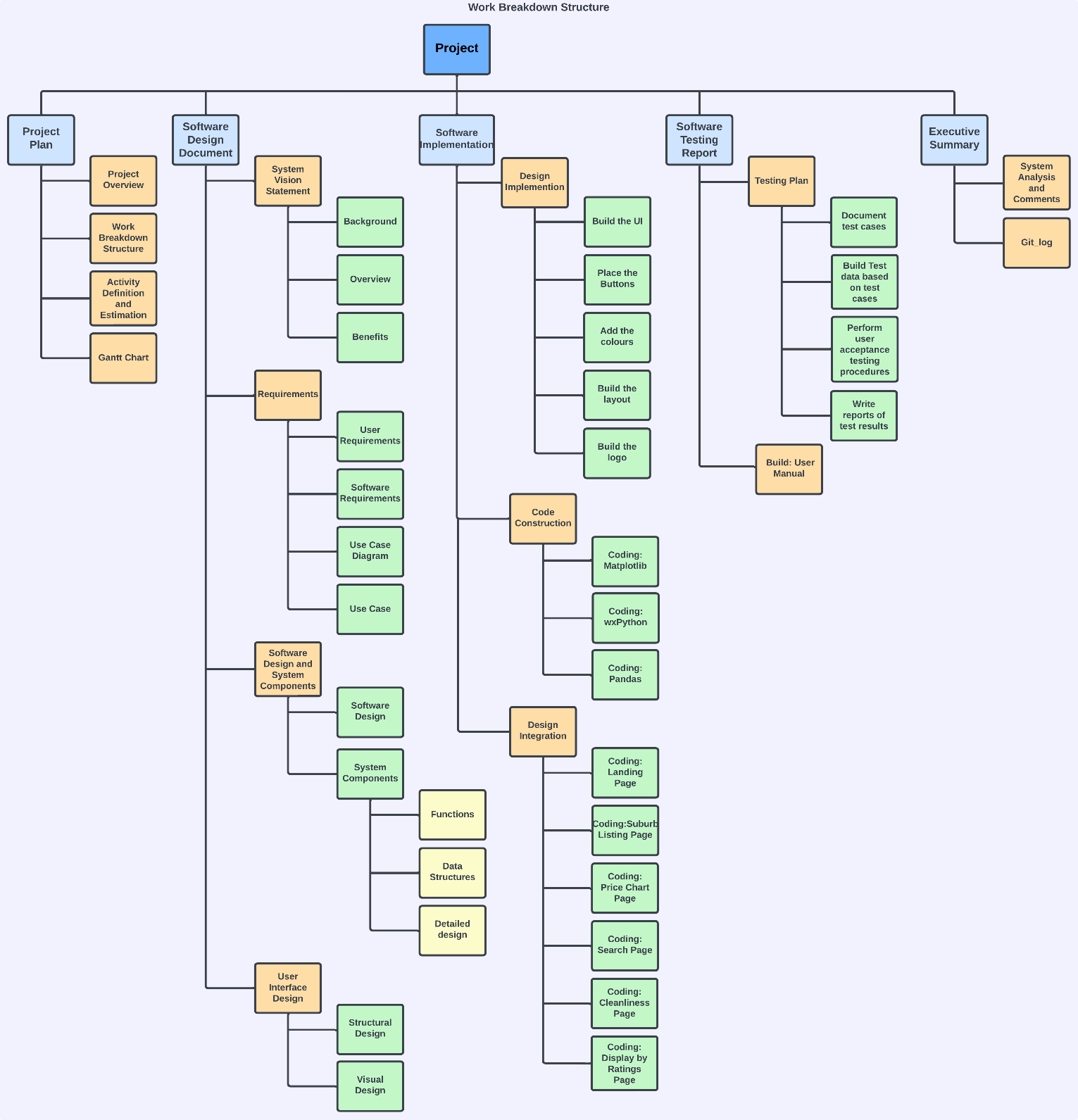
This section uses a deliverable-phased approach to show the Work Breakdown Structure (WBS). It visualizes the project's scope, tasks, and progress for each phase and its unique deliverables. Each level 1 element in this structure represents the top-level definition, while level 2 elements represent individual deliverables.

Figure 1:Work Breakdown Structure

# 3. Activity Definition & Estimation

This section presents a detailed breakdown of project activities derived from the phases outlined in the Work Breakdown Structure (WBS). Each activity is assigned a unique identifier, an activity name, comprehensive details, and an estimated time required for its completion.

Table 1: Activity Definition and Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| **Gantt Number** | **Item** | **Activity Details** | **Time Estimation** |
| **1** | **Project Plan** |  |  |
| 1.1 | Project Overview | Outline of the project | 1 days |
| 1.2 | Work Breakdown Structure | Formulate the WBS hierarchy | 2 days |
| 1.3 | Activity Definition and Estimation | Describe the activities and duration estimates | 2 days |
| 1.4 | Gantt Chart | Display schedule and progress of tasks | 2 days |
| **2** | **Software Design Document** | **? Round up to weeks?** |  |
| **2.1** | **System Vision Statement** | **Declare the purpose, goals and value the system aims to provide** |  |
| 2.1.1 | Background | !Why the software solution is being made??? | 2 hours |
| 2.1.2 | Overview | Scope of the system | 2 hours |
| 2.1.3 | Benefits | Benefits of the software to the company | 2 hours |
| **2.2** | **Requirements** |  |  |
| 2.2.1 | User Requirements | Description of what the user expects from the system | 1 day |
| 2.2.2 | Software Requirements | Description of what the software can deliver | 1 day |
| 2.2.3 | Use Case Diagram | Visual representation of interactions between actors and system | 4 hours |
| 2.2.4 | Use Cases | Description of how users interact with the system | 1 day |
| **2.3** | **Software Design and System Components** |  |  |
| 2.3.1 | Software Design | Diagram/hierarchy that shows | 1 day |
| 2.3.2 | System Components | The design of the different aspects of the app |  |
| 2.3.2.1 | Functions | List functions required | 2 days |
| 2.3.2.2 | Data Structures | Layout of …. Diagrams? | 2 days |
| 2.3.2.3 | Detailed design | pseudocode | 5 days |
| **2.4** | **User Interface Design** |  |  |
| 2.4.1 | Structural Design | Inc hierarchy | 1 day |
| 2.4.2 | Visual Design | Inc wireframes | 4 days |
| **3** | **Software Implementation** | **The implementation of the software** |  |
| **3.1** | **Design Implementation** |  |  |
| 3.1.1 | Build the User Interface | User interface design | 3 days |
| 3.1.2 | Place the Buttons |  |  |
| 3.1.3 | Add the Colours |  |  |
| 3.1.4 | Build the Layout |  |  |
| 3.1.5 | Build the Logo |  |  |
| **3.2** | **Code Construction** | **The construction of all aspects of the code** |  |
| 3.2.1 | Coding: Matplotlib | The code for creating the graph | 3 days |
| 3.2.2 | Coding: wxPython | The code for creating the UI design of the app | 3 days |
| 3.2.3 | Coding: Pandas | The code for interacting with Excel documents | 3 days |
| **3.3** | **Design Integration** | **The construction of all aspects of the code** |  |
| 3.3.1 | Coding: Landing Page |  |  |
| 3.3.2 | Coding: Suburn Listing Page |  |  |
| 3.3.3 | Coding: Price Chart Page |  |  |
| 3.3.4 | Coding: Search Page |  |  |
| 3.3.5 | Coding: Cleanliness Page |  |  |
| 3.3.6 | Coding: Display by Ratings Page |  |  |
| **4** | **Software Testing Report** |  |  |
| **4.1** | Testing Plan |  |  |
| 4.1.1 | Document Test Cases | Design testing | 5 days |
| 4.1.2 | Build Test Data based on test Cases |  | ???? |
| 4.1.3 | Perform user acceptance testing procedures | Create the Manual for users | 7 days |
| 4.1.4 | Write Reports of Test Results |  |  |
| 4.2 | Build: User Manual |  |  |
| **5** | **Executive Summary** |  |  |
| 5.1 | Analysis and Comments | Project report analysis and | 1day |
| 5.2 | Git\_log | Summary of the Git log | 1hr |
|  |  |  |  |

# 4.0 Gantt Chart

**add copy here as well as sep attach**  
This section is dedicated to the Gantt Chart, a visual tool that depicts the timeline, tasks, dependencies, and progress of the project. It is a resource management and allocation tracker and a project scheduling tool. It also gives insights into project phases needing more attention and time allocation. We also defined the Critical Path, a component that tracks the longest series of tasks within the project. Because each delay in activities along this path immediately extends the project's timeframe, this path represents the most significant period necessary for project completion.

[GANTT CHART]

The Critical Path is delineated through the following sequence of tasks: [List the specific tasks or path].

Based on this critical path, the estimated project completion time is [Estimated Time], providing crucial insight into the project's minimum duration for completion.

*This section should contain your Gantt chart. The items in the Gantt chart should match the activity definition from section 3. You should also submit your Gantt chart file separately.*