

CSE3PM Assessment 3: Project Management Plan

Details and instructions

Assessment 2	<p>Part 1: Group Project management plan, Part 2: Reflections, Part 3: Final presentation</p> <p>This is a group Assignment and cannot be completed individually.</p>
Due date	<p>Reflection 1: 11th May 2025 11.59 pm (AEST/AEDT) Reflection 2: 25th May 2025 11.59 pm (AEST/AEDT) Presentation: Week 12, individual presentation times to be confirmed Project Plan (Jira/Confluence documentation): 8th Jun 2025 11.59 pm (AEST/AEDT)</p>
Weighting	<p>Two Reflections: 20% Presentation: 15% Project Plan: 25%</p>
Word count/length	<p>Equivalent word count is 2000/student (+/-10%; The words in tables or numbers are considered. The reference list is not included in the word count, but in-text citations are)</p>
SILOs	<p>Assessment 3 aligns to Subject Intended learning Outcomes (SILOs):</p> <ul style="list-style-type: none"> • Develop professional project design documentation to meet the requirements of software or cybersecurity industry standards. • Prepare effective and collaborative project team meetings through application of team dynamics knowledge. • Develop an IT project management framework that integrates security analysis and risk mitigation throughout the project's lifecycle. • Critically appraise, analyse, refine and document system requirements to solve realtime problems in software or cybersecurity systems.
Feedback	<p>Written feedback will be available within three weeks of the due date</p>

**Generative
AI tools**

Make sure that the final product is your own work, and not just copied from an AI generator. You can use the generated text as a learning tool for inspiration or guidance, but the final submitted assessment must be your own work.

Specific Instructions

Choose one project: We've presented two projects below: one focusing on core IT and the other on cybersecurity. The team will need to carefully assess their expertise and interests before deciding on which project to pursue.

Only the project plan: The assignment exclusively centres on the planning phase of a project, excluding execution. Our deliverables will solely consist of planning documentation, adhering strictly to the requirements outlined in the assignment specification.

Group Assessment: This assessment is a team-based project. All Part I documents must be submitted exclusively through Confluence/Jira, as it is a mandatory requirement.

Team Collaboration: Successful completion of the project requires active collaboration from the entire team.

Individual Contribution Evaluation: While the project is a collective effort, individual contributions will be assessed and factored into the evaluation.

Gradual Contribution: Contributions will be continuously assessed throughout the project sprints. It is essential to document these contributions in the Confluence space.

Impact on Assessment: The visibility and consistency of individual contributions during the sprints will significantly influence the overall assessment of each team member's performance.

Approach

Agile Methodology: The project will be managed using the Agile approach, which emphasizes iterative development and collaboration.

Sprint Duration: Each sprint will have a duration of 2 weeks to allow for focused work and progress tracking.

Sprint Timeline:

Sprint 1: 28th Apr 2025 to 9th May 2025

Sprint 2: 12th May 2025 to 23rd May 2025

Task Management Tools: The team must use JIRA to effectively manage and track tasks. JIRA will help define clear roles and responsibilities for each sprint, ensuring efficient progress monitoring.

Assignment Deliverables

Your IT company has been approached by a group of stakeholders interested in developing the "CityPulse" platform or "SafeNet" platform (based on the project your team selects). They are in the process of selecting a development partner and have asked your company to create a detailed project plan to demonstrate your capabilities and approach. To secure the project, your company needs to create a comprehensive project plan that includes all essential elements and present it to the stakeholders. This will be done in three parts,

Part 1 - Group Project Management Plan (25%) – 8th Jun 2025 11.59 pm (AEST/AEDT)

Your team of project managers is responsible for delivering the project efficiently. Here's a high-level overview of the tasks:

1. Project Charter:
 - Create a Project Charter that outlines the project's purpose, objectives, stakeholders, and high-level scope.
2. Scope Definition:
 - Clearly define the project scope, including deliverables, boundaries, and exclusions.
3. WBS and Gantt Chart:
 - Develop a Work Breakdown Structure (WBS) to break down the project into manageable tasks.
 - Create a Gantt chart to visualize project tasks and their timelines.
4. Activity List :
 - Compile an activity list detailing all project tasks.
5. Project Milestones:
 - Document key project milestones, indicating their significance and timelines.
6. Stakeholder Identification:
 - Identify all project stakeholders, including internal and external parties.
7. Risk Management:
 - Prepare a Risk Identification, Assessment, and Evaluation document.
 - Utilize a Risk Matrix and Risk Register to systematically evaluate and manage project risks.
8. Resource Allocation:
 - Document the necessary project resources, including human resources, technology, and any other requirements.
9. Success Factors and Criteria:
 - Identify the critical success factors that will contribute to the project's success.
 - Define project success criteria that will be used to evaluate project outcomes.
10. Communication Plan:
 - Utilize a provided template to develop a comprehensive Communication Plan that outlines how project information will be disseminated to stakeholders.

It's common in project management assessments for additional documents to receive additional marks, especially when those documents enhance the quality and completeness of the project management process. Additional documents can demonstrate a deeper understanding of project management principles and best practices. However, it's important to ensure that any additional documents are relevant, well-prepared, and directly contribute to the project's success.

These tasks are integral to effective project management and will help ensure the successful delivery of the project to the organization. Each task should be completed with careful consideration of project goals, stakeholders, and best practices in project management.

Part 2 - Group Reflection Report (20%):

There will be two group progress reflection reports due at the end of each sprint:

Group Progress Reflection 1: 11th May 2025 11.59 pm (AEST/AEDT)

Group Progress Reflection 2: 25th May 2025 11.59 pm (AEST/AEDT)

These reflections will account for a total of 20% of your overall marks. They should be documented on Confluence and will provide an opportunity to reflect on your team's work during each sprint.

A template will be provided to assist you in structuring these reflections. These reports should cover key aspects of your project progress, including achievements, challenges, lessons learned, and adjustments made during each sprint.

Part 3- Group Presentation (15%):

In Week 12, your team will deliver a group presentation based on the documentation prepared. During the presentation, you will summarize how your team plans to manage the project during the monitoring and execution phase. Additionally, you will outline the approach for handing over the project to the client.

A general summary of the requirements for the group presentation will be provided by your tutor in the workshop during Week 10.

This presentation will account for 15% of your overall marks and is an opportunity to showcase your project management skills, communication abilities, and understanding of the project's management and execution phases.

Project 1: CityPulse – Real-Time Public Transport and Traffic Data Visualizer for Melbourne

Introduction: Your company is currently in the process of competing for a major contract under the CityPulse initiative, launched by the Victorian Government. This initiative aims to enhance urban mobility, reduce congestion, and improve public access to real-time transport and traffic information across Melbourne.

The main goal of the initiative is developing CityPulse, a web-based platform that provides real-time visualizations of public transport movements and traffic conditions across Melbourne and its growing suburban regions. The platform is designed to be intuitive, scalable, and citizen-focused, turning complex data into meaningful insights that help commuters, city planners, and transport operators make better decisions.

CityPulse will be particularly valuable for suburban communities that have been heavily affected by rising traffic congestion and an increasing number of road accidents — both of which have led to longer commute times and a growing frustration among daily travellers.

Problem Statement: Melbourne's rapid population growth has led to increased traffic congestion and a higher incidence of traffic accidents, particularly affecting suburban communities. Commuters often face delays due to traffic congestion or public transport schedule changes, but reliable real-time data is rarely accessible in a user-friendly format. Existing transport apps typically focus on route planning without providing live insights into current road conditions or public transport statuses.

CityPulse aims to bridge this gap by offering a centralized platform that:

- Aggregates real-time data from Melbourne's public transport systems and traffic sensors.
- Visualizes this data in an interactive and user-friendly format.
- Helps users understand delays, road conditions, and live vehicle positions at a glance.

This platform is particularly beneficial for suburban commuters who experience significant delays due to traffic congestion and accidents, providing them with timely information to plan their journeys more effectively.

Platform Modules:

1. Data Integration Module:

- **Real-Time Data Feeds:** Connects to Public Transport Victoria's (PTV) GTFS Realtime feeds to collect live data on vehicle positions, trip updates, and service alerts for trains, trams, and buses.
- **Traffic Data Integration:** Incorporates traffic congestion data from sources such as VicRoads' Smarter Roads initiative, which utilizes traffic cameras and sensors across Melbourne.
- **Data Processing:** Normalizes and stores data in a structured format for quick access and display.

2. Visualization and Map Module:

- **Interactive Mapping:** Utilizes mapping tools (e.g., Leaflet, Mapbox, or Google Maps API) to display a live map of Melbourne.
- **Live Vehicle Tracking:** Shows real-time positions of buses, trams, and trains with status indicators (e.g., On time, Delayed).
- **Traffic Congestion Visualization:** Highlights areas of high traffic congestion, particularly in suburban regions experiencing increased delays.

3. User Dashboard Module:

- **Route and Stop Search:** Allows users to search for specific routes, stops, or locations within Melbourne.
- **Filters:** Enables filtering by vehicle type (e.g., Bus, Tram, Train) and traffic conditions.
- **Information Display:** Provides estimated arrival times, congestion alerts, and historical trends for selected routes.

Key Considerations:

1. **User Experience:** The interface must be intuitive for daily commuters, including those who may not be tech-savvy.
2. **Real-Time Accuracy:** Must display updates in near real-time with clear refresh indicators.
3. **Scalability:** The platform should support large datasets and potentially integrate additional data sources in the future.
4. **Performance Optimization:** Efficient data fetching and caching to prevent lag or overload.

Technology Stack:

- **Front-end:** React (preferred) or Angular, integrated with map APIs (Google Maps, Leaflet, or Mapbox).
- **Back-end:** Node.js or Python Flask to fetch and process real-time transport and traffic data.
- **Database:** PostgreSQL for storing user preferences, historical traffic logs, and transit metadata.
- **Hosting:** AWS, Azure, or Heroku for scalable cloud hosting.
- **APIs:** PTV's GTFS Realtime feeds for public transport data and VicRoads' traffic data APIs for road congestion information.
- **Authentication (Optional):** Firebase Auth or a custom login system for personalizing dashboards and alerts.

Project 2: SafeNet – A Parental Control System with Real-Time Web Threat Detection

Introduction:

Your company is currently in the planning and early development phase of SafeNet, a smart parental control system designed to help families monitor and protect children's online activity in real time. With growing concerns around cyberbullying, harmful content, online predators, and malware-infected websites, many parents are seeking a reliable, user-friendly solution to manage and secure their children's digital experience, across browsers, devices, and networks.

SafeNet is being designed to offer real-time web monitoring, intelligent threat detection, and flexible parental controls that can adapt to different household rules and child age groups. The platform will include a dashboard for parents to view alerts, set restrictions, and receive notifications if harmful content or risky behavior is detected.

Problem Statement

Children today spend a significant amount of time online, using devices for education, entertainment, and social interaction. However, this increased access exposes them to a wide range of digital threats, including explicit content, phishing websites, scams, online predators, and malware.

Existing parental control tools are often limited to basic content filtering or manual site blocking. They don't actively detect web threats in real time or provide context-aware alerts that help parents respond to new risks as they emerge. Parents need a solution that is smarter, more adaptive, and easier to use.

SafeNet aims to fill this gap by providing an integrated, cloud-connected platform that uses both predefined content rules and real-time web threat intelligence to keep children safe online, without requiring parents to be cybersecurity experts.

Platform Modules

1. Real-Time Monitoring and Threat Detection Module

- Tracks browser activity and detects visits to harmful or suspicious websites
- Cross-checks domains against known threat databases (e.g., Google Safe Browsing, PhishTank)
- Uses keyword matching and pattern analysis to identify inappropriate or risky content (e.g., violence, adult material, scams)
- Sends instant alerts when threats or blocked content are accessed

2. Parental Control Dashboard

- Parents can view real-time browsing activity and device usage

- Enables creation of rules per child profile (e.g., daily screen time limits, bedtime schedules, blocked categories)
- Offers "Pause Internet" and "Safe Browsing Mode" buttons for quick parental actions
- Generates weekly usage reports and alert summaries

3. User and Device Management Module

- Allows parents to create multiple child profiles
- Supports multiple devices (mobile, tablet, desktop) linked to the same account
- Provides QR code/device linking for easy setup and management
- Includes options for temporary override (e.g., for homework or special permissions)

4. Notification and Alert Module

- Real-time notifications via mobile app, email, or SMS for high-severity events
- Configurable alerts for non-urgent activities (e.g., extended screen time)
- Alert center within dashboard to review and mark issues as resolved

Key Considerations

1. **Privacy and Ethics:** Must collect only essential usage data, respecting family privacy and complying with child data protection regulations .
2. **User Experience:** Designed for non-technical parents with a clean, mobile-friendly interface.
3. **Security:** All communications and data storage must be encrypted and secure.
4. **Scalability:** Must support growing families and households with multiple children/devices.
5. **Flexibility:** Should allow content control and threat detection settings to adapt to different age groups.

Software Requirements

1. **Front-end:** React for web dashboard; React Native or Flutter for mobile app
2. **Back-end:** Node.js or Django for managing rules, logs, and alerts
3. **Database:** PostgreSQL to store user profiles, alert history, and device configurations
4. **Hosting:** AWS, Firebase, or Azure with scalable backend services and real-time data syncing
5. **Web Threat APIs:** Integration with Google Safe Browsing, OpenDNS, and PhishTank for live threat detection
6. **Authentication:** Firebase Auth, Auth0, or Keycloak for secure login and role-based access