MONASH University
Information Technology

FIT5057- Project Management
Semester 2 - 2024
Assignment Three – Team Assignment

Due Dates:

- **Team presentation and Agile Artefacts (10%):** Team presentation in the form of a Stand-up Meeting to be conducted during **Weeks 12** Applied sessions**.**
- **Individual submission (90%):** Deliverables 1-3 as an individual assignment through Moodle before the deadline - Thursday 31 October 2024.

Value: 25% of overall unit assessment

Mode of Submission: Online via Moodle

LEARNING OBJECTIVES

The objectives of Assignment 3 are designed to deepen students' understanding and application of:

- Adaptive Project Management: Utilising Agile and Scrum methodologies to manage and adapt to evolving project requirements dynamically.
- Scrum Master Roles: Enhancing knowledge and skills pertinent to the roles and responsibilities of a Scrum Master within IT project management.
- Team Collaboration: Fostering advanced collaborative skills necessary for effective teamwork and project success in real-time environments.

THE CONTEXT

This assignment involves a series of deliverables that will guide you through the establishment of agile project foundations, detailed sprint planning and allocation and execution of the initial sprints. Each deliverable is designed to integrate the theoretical underpinnings of agile project management with practical application, allowing students to experience first-hand the dynamic environment of agile project execution.

- Transition from Planning to Action: Assignment 3 focuses on moving from the detailed project plans developed in Assignment 2 into an active development phase using agile methodologies.
- Focus on Agile Execution: You will learn to apply agile principles like sprints, continuous improvement, and regular evaluation for effective project execution.
- Practical Application of Agile Principles: The assignment includes deliverables that require students to:
    - Refine goals and requirements using agile approaches.
    - Create user personas to understand user needs.
    - Develop a high-level agile project timeline.
    - Break down the project into manageable sprints.
- **Educational Objectives:** This assignment fosters practical project management skills and prepares students to manage real-world projects within agile frameworks.

- **Emphasis on User and Organisational Needs:** Agile methodologies will help you develop solutions that are highly responsive to user needs and contribute to the stakeholder's organisational goals.
- **Bridging Theory and Practice:** This assignment helps students apply their theoretical knowledge of agile project management to practical project scenarios, preparing them for future project management roles.

Note: A 'Writing Guide' is provided to help you structure your responses effectively. Using this writing guide will ensure you address all key aspects of the questions and demonstrate your understanding of agile concepts.

INSTRUCTIONS

1. The workshop topics will address different components of the assignment (see Specification below). You should actively participate and engage in classes and discuss emerging concepts as a team. Where required, discuss your progress with your tutor, who can provide support and advice.
2. There are two assignment requirements:
    a. Group demonstration (Team stand-up meeting and Agile artefacts) due Week 12 during Applied session. Short extensions are **not** available for this task and **ALL** team members must be present, unless they have explicit permission from the CE (Roisin McNaney).
    b. Individual submission (all deliverables) due Week 14 (31 October 2024, 11:55 PM).
3. Submissions (individual) must take the form of **a single** PDF document.
    a. Individual submission should be named: <<A3>>_<<Group #>>_<<Student ID>>.pdf.
        - <<A3>> is Assignment 3.
        - <<Group #>> is the group number that will be assigned to your group when you enrol yourself in groups through the Moodle portal.
        - <<Student ID>> is your unique Monash University student number.
        - For example, a file name could be A3_0301_12345678.pdf, where the group number is '0301' and the student ID is '12345678'.

PLAGIARISM DECLARATION

When uploading your assignment on Moodle, you are required to accept the Student Statement (which includes a declaration that you have not plagiarised during the preparation of your assignment solution). You are required to ACCEPT the Student Statement; otherwise, you will not be able to submit your assignment electronically and your assignment will NOT be assessed.

NOTE: Marks may be deducted for any paragraphs or tables containing text which have been inserted into the document as screenshots or images. Any sections created in another word document or spreadsheet must be copied and pasted as text.

MONASH University
Information Technology

GENERATIVE AI

AI & Generative AI tools may be used SELECTIVELY within this assessment. Where used, AI must be used responsibly, clearly documented, and appropriately acknowledged (see Learn HQ).

Any work submitted for a mark must:

- represent a sincere demonstration of your human efforts, skills, and subject knowledge that you will be accountable for.
- adhere to the guidelines for AI use set for the assessment task.
- reflect the university's commitment to academic integrity and ethical behaviour.

Inappropriate AI use and/or AI use without acknowledgement will be considered a breach of academic integrity.

LATE SUBMISSION

1. Submission must be made by the due date. Unless an extension or special consideration has been granted, or otherwise specified in the learning management system, students who submit an assessment task after the due date will receive a late-submission penalty of 5 percent of the available marks in that task. A further penalty of 5 percent of the available marks will be applied for each additional day (24-hour period), or part thereof, the assessment task is overdue.
2. If you cannot complete an assessment (due to exceptional circumstances beyond your control), you may be eligible for a short extension or special consideration. A short extension is two calendar days and is available once for each eligible assessment. The first short extension for a unit will be granted without a reason given. All subsequent extensions require a reason when the application is submitted. Special consideration is a longer extension (for more than two days). Eligible students must supply a reason and supporting documentation.
3. For more details of the Special Consideration procedure and rules, and how to apply, visit: https://www.monash.edu/students/admin/exams/changes/special-consideration

ASSIGNMENT TEAMS

WHAT IS A TEAM PROJECT?

A "Team Project" is NOT the same as a traditional "Group Project".

In a "Team Project" the members of a team work on closely related problems (in this case projects within a portfolio of maximum 4 projects). Students perform some work as a team, in this case, the (a) Group presentation (Team stand-up meeting and Agile artefacts). Students also submit other work as an individual, in this case, (b) the "Main Submission" (that is, all other components of the assignment).

Even though the "Main Submission" is an individual piece of work, you are expected to work alongside the other members of your team in the unit's applied sessions as a number of elements of the individual "Main submission" depend upon decisions that you make collectively as a team (e.g. how to navigate your projects towards the success of your portfolio).

If, after reading the above, you are still unclear about the difference between a "Team Project" and a traditional "Group Project" then contact your tutor and/or Chief Examiner (Roisin.McNaney@monash.edu).

MONASH University
Information Technology

ASSIGNMENT SUMMARY

To formulate a successful portfolio, your team and you (as a Scrum Master for your individual project) will complete the following deliverables[1]. Discussion of the deliverables will be the focus of Workshops and Applied Sessions in weeks 9 to 12 of the semester[2]:

- Individual Submission:
  - Deliverable 1 (30%): Agile Project Foundations [*Relevant content: workshop 10; workshop 11; applied 10*]
  - Deliverable 2 (40%): Agile Planning and Sprint Allocation [*Relevant content: workshop 11, applied 11]*
  - Deliverable 3 (10%): Agile Reflection and Professional Development [*Relevant content: workshop 11; applied 9; applied 10; applied 11]*
- Team Presentation and Agile Artefacts (10%) [*Relevant content: workshop 12; applied 12; Recorded Deep dive]*
- Completeness and correctness of statements; Referencing (10%)

ASSIGNMENT TASKS

Building on the groundwork established in Assignment 2, students will now act as Scrum Team, leading their teams through the execution of their individual projects developed in the previous assignment. This stage introduces the practical challenges and decision-making processes typical in agile project environments, requiring students to iteratively develop and refine their project outputs based on real-time feedback and evolving project needs.

This shift to a practical, hands-on approach in Assignment 3 ensures that students not only learn about the theoretical aspects of IT project management but also gain valuable experience in applying these principles in a controlled yet dynamic and responsive learning environment.

Project 1: Eidolon's Veil

Project Overview:

Eidolon's Veil is a surreal exploration and puzzle adventure game that transports players into a dreamlike world filled with forgotten memories, cryptic symbols, and hidden truths. The protagonist can transform into different forms—such as a shadow, a wisp of light, or an ancient tree—each with unique abilities that help solve symbolic puzzles. The game's visual style features hand-painted landscapes transitioning between reality and abstraction, evoking strong emotional themes such as love, loss, and identity.

Project 2: Cipher Protocol

Project Overview:

Cipher Protocol is a cyberpunk stealth-action game set in a dystopian metropolis ruled by mega-corporations. Players assume the role of a rogue hacker, infiltrating secure facilities, manipulating

[1] Note: Your work in each deliverable may influence the decisions you made in previous deliverables. Be prepared to revisit previous deliverables if necessary.

[2] Some topics were briefly discussed during weeks 1 - 4 of the teaching period. There may be slight variation in the weekly content following the weekly feedback collected.

surveillance systems, and uncovering corporate secrets. The game features dynamic level design with vertical exploration and morally complex choices that impact the city's power structure. Players must navigate their way through a world of corporate control, rebel factions, or total anarchy, with each decision shaping the future of the city.

Project 3: Retro Rivals

Project Overview:

Retro Rivals brings back the nostalgia of '80s arcade racing, complete with vibrant retro visuals and an electrifying synthwave soundtrack. Players can choose from a selection of classic cars, from muscle cars to neon-lit speedsters, and compete in time-attack challenges. Tracks feature hidden shortcuts and ghost racers to beat, blending modern mechanics with the iconic arcade racing experience.

Project 4: Stellar Nexus

Project Overview:

Stellar Nexus is a galactic exploration adventure that invites players to captain their starship and explore a procedurally generated galaxy. Players will uncover ancient ruins, alien artefacts, and cosmic anomalies, while engaging in diplomacy, trade, or conflict with alien civilisations. Each decision shapes the galaxy's fate, allowing for non-linear exploration and dynamic storytelling based on player choices.

DELIVERABLE 1: Agile Project Foundations

Key Focus: Setting a strong foundation for iterative development by refining your understanding of the problem, the users, and laying groundwork for an agile workflow.

Task 1.1 Project Vision and Initial Requirements for your individual project

- Refine your project's Vision: Using the writing guide provided, develop a clear project vision statement. Describe what your project aims to achieve, including its scope, elaborate on the value your project offers, and identify the stakeholders who will benefit the most.
- Problem Statement: Clearly define the problem your project will solve, focusing on the benefits for the game players or end users.
- High-Level Requirements: Identify and list a minimum of 6 essential features your product (game) must include to meet user needs. Use user-friendly language, avoiding technical jargon, and focus on the benefits to the game players. These features will eventually be included in your product backlog (e.g., "Allow players to easily transform into different forms to solve puzzles" instead of "Implement a transformation system for character avatars").

Note: Task 1.1 may have similarity and overlap with some of your Assignment 1 and 2 deliverables, however, it should not be a repetition of your previous work. Use the formative feedback you have received across your other assignments to refine your project vision.

Task 1.2 Scrum Persona

Personas represent the various types of users who will interact with your game or product. Use the writing guide provided in Moodle for this task to develop two personas that will help you better understand user needs, behaviours, and goals. These personas should be directly related to the type of player or user who will engage with the project you're developing.
Include the following details for each persona:

- Name and Photo: Give each persona a realistic name and include a representative image.
- Description: Provide key information about the persona's background (e.g., age, occupation, interests, skill level).
- Goals/Needs: Define the persona's primary goals and what they want to achieve when interacting with your game.
- Frustrations/Pain Points: Identify any challenges or frustrations the persona might encounter while playing the game or using the product.

Using empathy to guide your decision-making, write a short reflection (50-100 words) on how your persona might benefit from one of the features in the game

Task 1.3 Journey Mapping

Map Out a Critical User Journey:

To ensure your journey map is focused on user-centred design, select one of the personas you developed in **Task 1.2** and use it as the basis for mapping a critical user journey. This will help you understand how this specific user type interacts with your game or project.

Choose a critical journey that is essential to the success of your project. The journey typically begins when the player becomes aware of a need (e.g., solving a puzzle, completing a level) and concludes when they either make a decision or accomplish their goal. By linking your journey map to the persona, you will be able to empathise with the user, identify their pain points, and propose improvements that make the experience more tailored and engaging.

Using the provided writing guide, map out this journey and include the following elements:

1. Stages:

- Stage 1. Awareness & Consideration: Describe how the user first becomes aware of the game or product and what factors contribute to their decision to engage.
- Stage 2. First Interaction: Detail the user's first experience with the game mechanics or core product features.
- Stage 3. Engagement & Problem Solving: Explain how the user interacts with the core features and what challenges they face.
- Stage 4. Decision Point: Outline the factors influencing whether the user continues to engage with the product or leaves.

2. Steps: Detail how the user progresses through each stage, from becoming aware to deciding whether to continue using the product. Break down these steps into clear sprint tasks that can be tackled iteratively.

3. Touchpoints: List all interactions the user has with the system, interface, or other people. Identify which interactions can be refined in future sprints.

4. Pain Points: Identify frustrations or inefficiencies the user might face at each stage. Each pain point should translate into a user story or task that can be prioritised in the backlog for future sprints.

5. Ideal Journey Improvements: Propose changes or new features that address the identified pain points. Each improvement should be associated with specific sprints for future development.

Note:

In real-world Agile projects, the journey map is often revisited and refined after each sprint based on user feedback. However, for this assignment, you are required to submit one version of the journey map as part of your deliverables.

MONASH University
Information Technology

Deliverable 1: structure & criteria

Deliverable 1	Task	criteria	Weight
Agile Project Foundations	**Task 1.1**	Develop a project vision document and high-level requirements (<350 words) that includes the following: ● Project's scope that describes what you are going to create. ● Describe the core problem your project will address and the overall solution you propose ● List a minimum of 6 essential features and functionalities your system must have to achieve the refined business vision.	10%
	Task 1.2	Develop two scrum personas (< 500 words in total) for your project that includes the following: ● Name and Photo ● Description ● Goals/Needs ● Frustration and pain points	10%
	Task 1.3	Journey Mapping (<400 words) ● Complete a focused journey map for a critical journey within your project. ● Clearly identify user pain points and potential areas of friction. ● Propose transformational improvements that align with your project goals.	10%

DELIVERABLE 2: Agile Planning and Sprint Allocation

Key Focus: Translating your project vision into actionable sprints, prioritising impactful features, and demonstrating adaptability within an agile framework.

Task 2.1 Develop a comprehensive product backlog incorporating five major epics specific to your project. Utilise the writing guide provided in Moodle and include the following essential categories in your product backlog:

i. Define Your Epics:

- Identify 5 major epics that represent the core functionalities of your project. Break down each epic into features (smaller epics) as needed.
- Describe each epic in detail and clearly explain how it contributes to the project's goals.
- Link each epic to stakeholder needs or business value to ensure alignment with the project vision.

ii. Break Down Features & User Stories:

For each epic, create a minimum of 2 specific features, written in user story format: "As a [user type], I want to [action], so that [benefit]."

Apply the INVEST criteria when writing user stories:

- Independent: The story should be self-contained.
- Negotiable: The story should allow room for discussion and flexibility.
- Valuable: It should provide value to the stakeholders.
- Estimable: The story should be estimable in terms of effort.
- Small: It should be small enough to complete within one sprint.
- Testable: There should be clear acceptance criteria for when the story is complete.

iii. Prioritise Strategically: Apply an agile prioritisation method (such as MoSCoW, Kano, or Weighted Shortest Job First (WSJF)) to your backlog. Ensure that Must Have features are included in the early sprints to address critical stakeholder needs and pain points. Prioritise stories based on business value, risk, and dependencies.

iv. Estimate Effort:

- Assign story points to each user story using techniques like Planning Poker or relative estimation. Be sure to estimate effort based on complexity, risk, and uncertainty.
- Ensure that effort estimates are realistic, considering the team's capacity and experience.

Task 2.2 Strategic Sprint Allocation

Your team faces the challenge of planning a project over three months, divided into five sprints. You've already developed a prioritised list of five major epics (Task 2.1). Now it's time to strategically allocate these epics to sprints, delivering value incrementally while addressing critical pain points.

a. Total Duration: 3 months (12 weeks)

- Number of Sprints: 5
- Average Sprint Length: 2.4 weeks per sprint

b. Steps for Sprint Allocation:

- Break Down Large Epics:
  - Review the five epics you defined in Task 2.1 and determine if any epics need to be broken down into smaller sub-epics or features to be distributed across multiple sprints.
  - Example: If you have an epic like “Level Design” in Eidolon’s Veil, it might be broken down into “Initial Environment Design” (Sprint 1) and “Advanced Puzzle Layouts” (Sprint 3).
- Prioritise & Allocate Epics:
  - Allocate epics or sub-epics to each sprint. Use an agile prioritisation method like MoSCoW, Kano, or Weighted Shortest Job First (WSJF) to ensure that the most critical features (e.g., “Must Haves”) are addressed early.
  - Deliver Incremental Value: Ensure that each sprint delivers a usable, functional piece of the project. This could be a feature that stakeholders can use, test, or provide feedback on at the end of each sprint.
  - Example: For Retro Rivals, Sprint 1 might focus on developing basic racing mechanics and core car physics, while Sprint 2 focuses on implementing the synthwave soundtrack and retro visuals.
- Resolve Pain Points & Dependencies:
  - Focus on resolving key pain points early in the project, especially those that stakeholders have prioritised. If any features depend on others being completed first, make sure to allocate them to earlier sprints.
  - Example: For Cipher Protocol, you might need to complete the hacking mechanics in Sprint 1 before working on more advanced stealth systems in Sprint 2.
- Theme-Based Sprints (Optional):
  - If applicable, organise sprints around broader themes such as “Core Gameplay Mechanics,” “User Experience Enhancements,” or “Narrative Development.” This helps create a structured progression in the project.

- Example: For Stellar Nexus, Sprint 1 could be focused on procedural galaxy generation (core mechanics), while Sprint 3 might focus on diplomacy systems (narrative enhancements).

c. Adaptability:

- Create a Buffer: Ensure that each sprint plan includes some buffer time to accommodate unexpected challenges or scope changes. This will allow your project to remain flexible and adaptable to new requirements.
- Adapt to Changes: Be prepared to adjust your sprint allocations mid-project if new requirements or critical issues arise. Justify how you would manage scope changes, explaining which features might be delayed or reprioritised.

d. Justify & Explain Trade-offs:

- Provide a detailed explanation for why you allocated specific epics to particular sprints. This should include any trade-offs you made, such as prioritising core functionality over non-essential features or delaying a less critical feature for a future sprint.
- Example: If you delayed implementing a minor feature in Retro Rivals to prioritise core racing mechanics, explain how this decision improves overall project outcomes.

e. Critically Analyse Your Plan:

- Identify any gaps or missing information in your sprint allocation plan and explain how you would address them during the project.
- Example: If you're unsure how long a particular feature might take to develop (e.g., advanced hacking systems in Cipher Protocol), explain how you would gather more information and adjust your plan as necessary.

Task 2.3 Sprint 1 Execution Plan

For Sprint 1 of your project, you will develop a detailed execution plan that outlines the key tasks, user stories, acceptance criteria, and team capacity. This plan will ensure that Sprint 1 is focused on delivering real value and meeting key stakeholder needs.

Your Sprint 1 Execution Plan should include the following components:

a. Objective:

- Clearly communicate the main goal of Sprint 1, linking it to critical stakeholder pain points and your project vision. The objective should focus on delivering value that stakeholders can see and provide feedback on.
- Example: "The goal of Sprint 1 is to implement the basic form transformation mechanics for Eidolon's Veil so players can start interacting with the environment."

b. Sprint Scope:

Provide a comprehensive breakdown of all features, user stories, and acceptance criteria that will be included in Sprint 1. Ensure that the scope is clear, manageable, and focused on achieving the sprint objective.

- User Stories: For each epic or feature, include specific user stories that will be addressed in Sprint 1.
- Acceptance Criteria: Provide clear and measurable acceptance criteria for each user story, ensuring that each task is testable and achievable.
- Dependencies: Identify any dependencies between tasks or user stories.

Example for Cipher Protocol:

User Story: "As a hacker, I want to disable security cameras so that I can infiltrate secure areas."

Acceptance Criteria:

1.The player can disable security cameras through the hacking mechanic.

2.Cameras visually shut down and stop tracking the player.

3.The hacking mechanic works without bugs and is tested by the team.

c. Definition of Done (DoD):

- Define the Definition of Done (DoD) for Sprint 1. This should include the criteria for determining when each task, feature, or user story is considered complete. The DoD should cover quality checks, functionality testing, and stakeholder approval.
- Example: For the hacking feature in Cipher Protocol, the DoD could include:

  1.Hacking functionality is fully implemented.

  2.The feature passes all test cases.

  3.Stakeholders have approved the functionality after the sprint review.

d. Capacity Alignment:

Assess your team's available capacity and ensure that the sprint scope is aligned with the team's ability to deliver. Take into account the complexity of tasks, team size, and any foreseeable risks. Include a buffer of time to handle unexpected delays or issues.

- Example: If your team velocity is 20 story points per sprint, allocate tasks that total around 18 story points, leaving a 10% buffer for contingencies.

e. Sprint Review & Retrospective:

Plan how you will conduct the Sprint 1 Review to gather feedback from stakeholders. Also, outline the Sprint Retrospective to reflect on the sprint's successes and challenges, identifying improvements for future sprints.

- Sprint Review Example: "We will demonstrate the form transformation mechanics in Eidolon's Veil to stakeholders and gather feedback on functionality and user experience."

- Sprint Retrospective Example: "The team will discuss what went well during Sprint 1, what challenges were faced, and what can be improved in future sprints."

Note: Please refer to the additional resources for more information about sprint and accepting criteria.

Deliverable 2: Structure & criteria

Deliverable 2	Task	Criteria	Weight
Agile Planning and Sprint Allocation	**Task 2.1**	Define 5 major epics with clear descriptions of their alignment to project goals. (<500 words) ● Break down epics into features and user stories using the format: "As a… I want to… so that…". ● Apply the INVEST criteria (Independent, Negotiable, Valuable, Estimable, Small, Testable) to ensure quality user stories. ● Clearly explain how each epic aligns with the project's overall goals ● Apply an agile prioritisation method (e.g, MoSCoW, Kano) to prioritise features. ● Assign story points for effort estimation and ensure realistic capacity planning e.g. using techniques like Planning Poker	20%
	Task 2.2	Allocate epics to 5 sprints with a focus on incremental value delivery. (<300 words) ● Prioritise resolving key pain points and consider dependencies between epics. ● Break down large epics into smaller tasks, ensuring that each sprint delivers usable functionality. ● Include justification for allocation decisions, explain trade-offs, and critically analyse missing information and how to handle potential changes. ● Ensure adaptability by creating buffer time and outlining how to adjust if requirements change mid-project.	10%
	Task 2.3	Define a clear Sprint 1 objective linked to solving stakeholder pain points and delivering project value. (<300 words) ● Provide a comprehensive scope of features, user stories, and acceptance criteria to be completed. ● Establish a clear and measurable Definition of Done (DoD) to validate when the sprint's work is complete. ● Realistically assess team capacity and leave a buffer for contingencies. ● Outline plans for the Sprint Review to gather feedback from stakeholders and the Sprint Retrospective to reflect on the sprint and improve future work.	10%

DELIVERABLE 3: Agile Reflection and Professional Development

Task 3.1: Reflection on Agile Game (LEGO/Agile Activity Experience)

Learning from the Game: Reflect on your experience participating in the LEGO/Agile game[3] or similar Agile activities. Focus on the following:

- Agile Principles in Action: How did the game help you understand core Agile principles like iteration, collaboration, and flexibility? What key lessons did you learn about working in Agile teams?
- Role of the Scrum Master: During the game, how did the role of the Scrum Master or other roles help facilitate teamwork, problem-solving, and continuous improvement? How do these experiences compare with theoretical knowledge of Agile roles?
- Challenges and Solutions: What challenges did your team face during the Agile game (e.g., communication breakdowns, time constraints), and how did you work together to solve these challenges using Agile practices?
- Example: "During the LEGO game, I learned how frequent iterations and quick feedback loops allow for continuous improvement. Our team struggled with communication early on, but after our Scrum Master facilitated a retrospective, we adjusted our workflow and became more efficient."

Task 3.2: Personal Reflection and Career Development in Agile Roles

Your Experience with Agile Roles:

Based on your experience with the LEGO/Agile game in Week 11 Applied Class, reflect on the Agile concepts and your personal fit or suitability for Agile roles such as Scrum Master or Project Manager. Consider the following:

- Role Preference: Did the Agile game help you identify a preference for any specific role (e.g., Scrum Master, team member, Product Owner)? What qualities or skills did you use or discover during the activity that align with this role?
- Skills Development: Identify 2-3 Agile-related skills (e.g., facilitation, time management, collaboration) that you developed or improved during the game. How can these skills help you in a future Agile role?
- Future Career Path: Reflect on how this experience impacts your career strategy. Would you pursue a role as a Scrum Master or Project Manager, and what additional skills or experiences do you need to develop to reach your goal?
- Example: "Through the game, I found that I enjoyed facilitating discussions and ensuring that the team remained focused on the sprint goal, which is why I'm now considering pursuing a Scrum

[3] Planned during week 11 applied classes

Master role. I realised that my communication and problem-solving skills are strengths, but I need to work on managing large teams effectively."

Deliverable 3: structure & criteria

Deliverable 3	Task	Criteria	Weight
Agile Reflection and Professional Development (Word limit <500 for both tasks)	**Task 3.1**	Reflect on key learnings from the LEGO/Agile game. ● Discuss how Agile principles were applied during the game. Analyse the role of the Scrum Master or other Agile roles in facilitating teamwork. ● Reflect on how challenges were addressed through Agile practices. ● Provide examples of how Agile concepts were applied in the game.	5%
	Task 3.2	Reflect on your role preference and skills you discovered during the Agile game. ● Identify 2-3 Agile-related skills you developed or improved. ● Reflect on how this experience informs your future career path as a Scrum Master or Project Manager. ● Discuss how you plan to develop the skills needed to achieve your career goals.	5%

TEAM PRESENTATION AND AGILE ARTEFACTS (week 12 demonstrations)

The daily scrum (also called the daily coordination meeting) allows the team to coordinate work. This is part of the daily agile practice, ensuring all team members are aligned and any impediments are quickly addressed and typically done by answering three questions:

- What did I do yesterday?
- What do I plan to do today?
- Is there anything blocking my work?

In each team stand up meeting, one of the team members should assume the role of scrum master and facilitate the meeting. The other team members could assume the role of developer/security specialist/data analyst etc. The facilitator should make sure that everyone gets time to speak and all team members are heard. If there are any risks or blockers, the facilitator should make sure that the product owner (role played by your tutors) is aware and necessary steps are taken.

Team members will demonstrate task progress updates to their tutors on the team's Scrum/Kanban board, transitioning from "To Do" to "In Progress," "Blocked," or "Done". Utilise either a physical Scrum board with post-it notes or digital tools like Jira, Asana, or Trello to demonstrate tracking and collaboration.

Scrum or Kanban Board Usage: Teams could utilise and customise agile boards continuously to track and present project progress, updating statuses and tasks as they evolve through the sprint. This can be done as live documents/artefacts of their project's progress.

- Accomplishments: What they completed since the last stand-up.
- Current Focus: What tasks they are working on today.
- Blockers: Any challenges or obstacles they are facing.

"Show Your Work": the team uses their Scrum/Kanban board effectively during their demonstration in week 12, referring to it to visually explain their progress, decision-making rationale, and any adjustments made in response to feedback.
Essential Steps:

- Project Selection: As a team, agree on which of the four projects in your team's portfolio you will focus on for this demonstration. Only one project (one Scrum/Kanban) is required during the demonstration.
- Requirement Review: Revisit your refined project objectives and requirements (for the project you selected in the previous step) from Deliverable 1. Ensure the team has a shared understanding of the project's core goals. These goals will go on to your backlog list. Document all requirements, a list of things that will help your product owner, ideas/general needs, USE THE TERMS YOUR PRODUCT OWNER UNDERSTANDS.
- Epic Definition: Brainstorm and list major epics that encapsulate the essential areas of functionality for your chosen project. Ensure these epics align with the key objectives. Refer to the Common Features provided.
- User Story Generation: Within each epic, develop user stories focusing on the 'Must Have' features. Consider using the format: "As a [user type], I want to [action], so that [benefit]."

- Kanban Setup (Choose One):
    1. Physical Board: If possible, utilise a whiteboard or similar surface with sticky notes to create basic Kanban columns (e.g. "Backlog," "To Do," "In Progress," "Done").
    2. Digital Tool: Select a Kanban tool (Trello, Asana, etc). Create the essential columns and set up your board layout.
    3. Transfer to Kanban: Place your epics and prioritised user stories into the appropriate columns of your Kanban board.

Demonstration Format

Each team will have a total of 5 minutes to present their project updates during the stand-up meetings and demonstrate how they have utilised Agile methodologies throughout the project lifecycle.

Week 12: Stand-Up Meeting Demonstration

Scope:

- Each team will present their stand-up meeting which focuses on the development and prioritisation of the project backlog on a Trello Board or similar tool. Present your initial project backlog on Trello (or similar). It should include:

- At least 3 clearly defined epics aligned to project goals.
- 6-10 user stories (using the "As a... I want to...so that..." format) focused on "Must Have" features.

- Teams must demonstrate how they have defined and prioritised requirements using the MoSCoW method.
- The demonstration should include a visual presentation of their Trello Board, showing the breakdown of tasks into Epics, User Stories, and the prioritisation of these tasks.

Evaluation Criteria:

- Clarity of Presentation: How well the team communicates their process and project updates.
- Organization of the Backlog: The structure and clarity of the Trello Board or project management tool.
- Prioritisation Methodology: Effectiveness and appropriateness of using MoSCoW for task prioritisation.
- Engagement: Level of team participation and ability to answer questions from the tutor.

Team Presentation structure & requirements

Deliverable	Requirements	Weight
Stand-up Meetings	Perform a daily scrum (also called the daily coordination meeting) in presence of your tutor (role playing as your Product Owner)	10%

Completeness and correctness of statements; clarity of expressions (weight 5%)

The individual report should contain clearly structured information that is focused on identifying and addressing all requirements of the deliverable with clear supporting information. The report should be well presented and include key elements of a succinct report such as a title or cover page, a table of contents, page numbers, appendices (if applicable), etc.

All sections of the report and the analyses and discussions are expected to demonstrate originality, creativity, and sophisticated thinking. The statements show an excellent understanding of the foundation of the unit and a strong capability to transform the theories into intellectual contributions.

Use of references (weight 5%)

Students are expected to search for relevant extra readings and draw on suitable literature from academic publications as well as practitioner outlets. All sources of information must be fully and appropriately acknowledged using in-text citation and reference list. Use at least 6 references, 3 of which must be from peer reviewed academic journals or conference publications.

The reference section should use the American Psychological Association (APA) style of referencing.

(APA 7th style: https://guides.lib.monash.edu/apa-7)