Project Governance Portfolio (PGP) Guidelines

Important Note: Generative AI tools are NOT allowed to be used in this unit's assignments unless it is specified in their instructions and guidelines.

Summary

Applying appropriate project governance to the IE project is critical to achieving the required learning outcomes for the unit (readiness for work) and is assessed in each of the IE projects deliverables.

Project Governance Portfolio is marked as part of all project deliverables. It is expected that teams keep their project governance up to date at all times. It is expected that as a team you come up with the appropriate folder structure and naming convention for your team folder and documents. An example has been provided in your PGP Folder.

Objective/Aims

- To effectively apply project governance techniques and tools explained in the IE seminar, pre-readings and studios and prescribed for the IE project.
- Demonstrate your understanding of Kanban boards and the ability to use leankit effectively.
- Demonstrate the application of appropriate Agile methodologies to your project.
- Conduct effective project meetings (stand up meetings, Industry Mentor meetings etc.)
- Maintain a complete project repository with all design and build artefacts included, Industry Mentor Mahara and Leankit as per the unit specifications, in a professional manner.

Assessment

Your Project Governance (including, the structure and content of your folder, LeanKit, Retrospective, Industry Mentor Mahara (e-Portfolio) will be assessed as part of each iteration by your studio mentors. You will receive feedback on it and it will contribute to your mark for the iteration.

Project Governance Portfolio Guidelines

To provide a clear understanding of the project's evolution, we require development artifacts such as data models and a detailed description of the chosen development approach. Software packages, languages, frameworks etc only include if they add value. Marks will be deducted for useless additions.

Typical artefacts in the project governance are as follows and **must** contain: (The PGP structure in your team's folder, is a suggestion)

- **Team Information Folder**
  - A word document that contains photos of each team member
  - Link to each version of iteration build and its credentials
  - Link to leankit
  - Team social activities photos

- **Team Meeting Folder**
  - Documents that contain **iteration planning meeting agenda** (during A&D phase, your team will discuss the end users goals and needs, opportunities for the product, technologies involved).
  - Video records of your stand ups
  - Weekly team meetings
  - Industry Mentor minutes
  - Backlog grooming meeting
  - Feedback noted from mentors.

- **Design and Analysis Artifacts Folder which must contain mandatory artefacts such as**
  - Documentation on how you have used AI in your design (for example what prompts did you use, what were the results produced and how the results were changed). **Please see the section below of the Design artefacts folder.**
  - Persona profile (can have more than one persona from iteration 2)
    - Must have basic understanding of the target audience through the development of well-defined personas and updated with each iteration
    - Accurately identifies major needs and expectations with evidence of user research via news articles references.
    - Representing real people with backgrounds, goals, and values.
  - Design Thinking artefacts, Lotus Blossom and Empathy map
  - Conceptual ERD diagram
  - Conceptual architect diagram
  - User story mapping, identified use cases
  - Customer journey mapping
  - Ethic canvas
  - Lo-fi prototypes/high level prototypes
  - Site map and

    - any other design documentation to see how the team moved from their data and information collection to their understanding of what was needed.

- **Usability Testing Video folder must contain with basic understanding:**
    - At Least 10 - 15 usability testing videos for each iteration
    - Report on all the findings from the testing videos should be documented in the leankit with the expected outcome, fix and who will be responsible for the fix.
    - Demonstrated good coverage of testing was achieved at various stages.
    - Evidence of team members provided adequate support to testers.
    - Testing feedback was generally applied to refine user stories and improve the final product.

- **Iteration Build Folder**
    - The development artefacts to help understanding how you go there
    - Documentation of pair programming observations
    - Links to your github and code repository (include any pair programming notes)
    - First Cut at Technology to be used: Development approach, with any first-cut data models; Software packages, languages, frameworks etc.

- **Testing Folder**
    - Testing is well-defined and executed, demonstrating a good understanding of testing methodologies.
    - Testing documentation provided is generally adequate and meaningful, demonstrating a reasonable understanding of the rationale behind testing decisions with test cases.
    - Thoughtful consideration is given to what is tested and how it will be used.
    - Evidence of effective testing is visible within the final build

- **System Architecture Folder**
    - May include from a **potential sponsor's** point of view **(non IT)** of the proposed system (diagram would be best).
    - Should be a very high level at this point. To also include potential data sources and how they might be used (may include APIs).

- **Risks Folder (if required)**
    - Documents most project risks with reasonable accuracy
    - Develops generally effective mitigation strategies for identified risks.

- **Retrospective Folder**
    - **Suggested retrospective template: https://miro.com/blog/7-retrospective-templates-love-use-miro/**

- **Data Governance Folder**
    - Includes a data management plan report which is an ongoing live document.

    - The report should have details on what data you plan on using, how the data supports your proposed product. **Please see below for further information.**

*The expectation is that teams with **MDS/MAI students** will do more in this section than other teams. (Potential modelling; Hindsight to Insight to Foresight; etc).*

- **Security Aspects folder which must contain;**
    - Security Plan which must specify specific security aspects and all implemented security measures for this product (updated as necessary and please see further details on Security Plan below).
    - For MCS students only, Vulnerability Assessment Report for each iteration (please see further details below).
- **Handover Documents folder**
    - Refer to Handover Package Guidelines
    - Live documents ie Product Document, Support Document and Maintenance Document will be graded at Week 11 however will be reviewed throughout the studios.

Data Management Plan Report Guidelines

The data management plan is a live document and is **to be submitted during the iteration analysis and design**. The Monash teaching staff will view your data management plan and all feedback must be implemented as you submit the iteration build.

Your data management plan is how information will be updated/transposed into your application. You must define the process of your data. This can be in a word document form and is a live document that will be reviewed in each iteration review and to be uploaded in the Data Governance folder in the PGP.

This information will contain details about the data sources and their use, and their update frequency both at source and into your system. Any specific modelling or analysis for this iteration will be described. How they are being used in **THIS** iteration, including the process for updating data within your application.

Modelling or specific analysis will also be described here. These may be business requirements in the form of an ER model and/or more specific data modelling carried out by students with data science expertise.

The data management plan should:

- The live **data management plan report** should also contain information about the **process of how you wrangled, cleansed, transposed, stored and archived your data.**

- It should include all the **Open Data Source(s)** details and how they are updated into the system.
- **Must have Ethical, Legal and Privacy Issues headings** which outlines the ethics in Artificial Intelligence (AI)
    - Demonstrates some understanding of ethical considerations that are generally well-addressed, including data management and some potential algorithmic biases.
    - Must communicate ethical, legal and privacy issues with regards to your open data.
- Demonstrates a strong critical thinking and analysis of the **data modelling techniques** employed and critical thinking and analysis in the application of data analysis techniques.
- Demonstrates to show screenshots of your process and any database relationship diagrams of your structure.
- Demonstrates a good understanding of the problem domain and well-defined and meaningful user needs based on the open data sets
- Demonstrates a good understanding of how and why the requirements are needed, considering future needs and potential challenges.

Open Data Source Table example:

Example - Data Sources (Open datasets)					
Names	**Physical access (e.g. API, CSV)**	**Frequency of source updates**	**Frequency of iteration system updates**	**Granularity**	**Copyright / licensing details**
Bridges in Australia https://services2.arcgis	API CSV downloaded	Every 2 weeks	Every 3 months	Aggregate number of bridges per LGA	http://creativecommons.org/licenses/by/3.0/ - Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made.

Design Artefacts Folder:

The purpose of the AI design document is to document the AI prompts used at each iteration for the areas using AI tools (co-pilot only). This document serves as a record of co-pilot's usage, demonstrating your understanding of AI's potential and limitations within the design process. It will also allow for a more comprehensive evaluation of your project's development.

If you have utilized co-pilot for Problem Statement, User Stories and Acceptance Criteria an "AI Design Report" is required for each area. The AI design report is a live document and is to be submitted via the PGP Folder in Design Folder.

This documentation will specifically address the following areas:

1. **Problem Statement:** How co-pilot was used to refine and improve the project's core problem statement.
2. **User Story Development:** How co-pilot was used to generate, refine, and prioritize user stories.
3. **Acceptance Criteria Definition:** How co-pilot was used to formulate and refine the acceptance criteria for project deliverables.

Each AI design document is should be structured as follows:

1. Prompt Text: Write down the exact text of the prompt you used, include context and parametre and provide the result.
2. Write another prompt to compare results and to improve the result.
    - Provide examples of both successful and unsuccessful outputs to illustrate the prompt's effectiveness.
3. Results and Evaluation: Document the results generated by the AI and evaluate the changes made to the prompt and reasons behind those changes to track improvements over time. and revise in your own words.

Tips for effective AI Design documentation:

- Be Specific and Clear: Ensure that the prompt and its context are clearly defined to avoid ambiguity.
- Use Natural Language: Write prompts in a conversational tone to make them easy to understand and replicate.

Security Plan Report Guidelines:

The **security plan report** guidelines includes:

The security plan is a live document and is **to be submitted during the iteration analysis and design**. Your security plan would cover the security aspects relevant to your application where you would be expected to analyse the security risks relevant to your system application, produce security measures and policies to be adopted to ensure secure operation of the application. Take note to refrain from generic security aspects as should consider aspects relevant to your application.

System Security Awareness

This section covers foundational security practices, including general security awareness, authorization controls, threat prevention, data protection, risk assessment, and the establishment of security measures and policies**.**

Risk	Risk Description	Risk Rating	Recommendation
1	Broken Authentication	High	Adopt a Strong password /passphrases policy - All the passwords must follow a strict policy of at least 9 characters consisting of alpha-numeric letters and operational processes to change it, frequency etc. Management of password/passphrases, what is your process for changing the authentication.
2	Open ports and services	Low	Ensure any unwanted ports and services are closed.

Ethical, Legal, Security and Privacy issues

You must explain how your product will protect user information and follow privacy rules. Also, you need to show how you will keep your product safe from hackers and other threats.

Risk Analysis

This section would capture all the possible problems that could happen with your product. You must document how likely these problems are to happen (risk) and the impact if they did happen. You need to make a list of these problems and a mitigation plan.

If you were to be attacked you are expected to provide a root cause analysis of the possible incident and document this in your PGP folder and in your LeanKit.

Document Widespread Security Attacks - guidelines that are applicable to your product. Guidelines can be found in Moodle. (for MCS Only students)

Vulnerable Assessment (MCS students only)

For each iteration as part of the iteration build submission, each MCS student must submit the following to their PGP:

Penetration Tests & Findings

a) Conduct penetration testing on your own system.
b) You are to perform one or more attacks.
c) Document your findings with relevant screenshots and descriptions.

Warning: Brute force attacks are strictly prohibited.

Result Evaluation, Impact Assessment & Reinforcement

a) Analyse the results of your penetration tests.
b) Describe the vulnerabilities discovered.
c) Produce an impact assessment of the identified vulnerabilities.
d) List and implement recommended mitigation strategies.

Note: Should no vulnerabilities be detected or exploited during your evaluation, provide proactive security recommendations based on your observations. For instance, if a system features a form without a reCAPTCHA, it would be prudent to suggest its integration to mitigate potential spam and protect against automated bot-related threats.

Formalise and Document and Structure your Work *in report format*

Security Drill Structure:

1. Pen Test Findings
   Screenshots & descriptions of penetration test attacks, findings, and vulnerabilities discovered.

2. Impact Assessment
   A table of identified and evaluated vulnerabilities of their potential effects or consequences on the confidentiality, integrity, and availability of the system or data.

3. Recommendations
   Suggestions to address and rectify the identified vulnerabilities, best practices to adopt to prevent future vulnerabilities or threats. Priority levels for each recommendation based on the potential impact and ease of implementation.

Submission Format:

Your mentors (as your studio managers). To be uploaded to Project Governance Folder > Security Sub Folder. **This is due at the time of the build submission of each iteration.**