Canterbury Institute of Management (CIM)

ASSESSMENT COVER SHEET

1. Personal Details

Student ID	**Given Name(s)**	**Surname**	**Email Address**
1. CIM12137	Ayesh Chathuranga	Jayasekara	cim12137@ciom.edu.au

Campus	Darwin Campus
Course Title and Code	MSIT401 System Development Methodologies
Assessment Title	Reflective Journal - *Week 10*
Due Date & Time	08/12/2024

Course Lecturer/Tutor Name: Sharad Neupane	Assessment Word Count (if applicable): 498

2. Student Declaration

By signing and submitting this coversheet, I/we declare that:

- ✓ This assessment submission is my/our own work unless otherwise acknowledged (including the use of generative AI tools) and is in accordance with the Institute's ***Academic Integrity and Honesty Policy*** available on the website.
- ✓ No part of this assessment has been submitted previously for advanced standing or academic credit in this or any other course.
- ✓ I/we certify that we have not given a copy or have shown a copy of this assessment item to another student enrolled in the course, other than members of this group.
- ✓ I/we are aware that the Lecturer/Tutor of this assessment may, for the purpose of assessing this assessment task communicate a copy of this assessment task to a plagiarism checking service to detect possible breaches of academic integrity, for example, plagiarism, recycling, cheating, contract cheating, or unauthorised use of generative AI (which may then retain a copy of the item on its database for the purpose of future investigation).

Signature:	Jay's	Date:	07/12/24

MSIT401 System Development Methodologies
Reflective Journal - *Week 10*

Ayesh Jayasekara - CIM12137

System Architecture

As with any implementation, system architecture is the blueprint of any computerized system either hardware or software. It is therefore important that all the aspects are critically evaluated when deciding on a system design architecture which defines how the system would work, integrate and evolve over a period of time.

> **Sostware architecture, unlike hardware architecture, can be regarded as a blueprint for the system to be developed. As such, it also provides a checklist of activities to be performed as part of the development. Such an architecture can detail some of the deliverables to be produced during the development, deliverables that are required for the proper functioning of the system.**
>
> (Yadin, 2016)

The architectural choices often depend on,

- **Organizational culture:** How the target stakeholders would consume the system
- **Resources availability:** What are the tangible and intangible resources available to accommodate design, implementation, execution & maintenance of system
- **Total cost of ownership:** Cost benifts and saving options such as cloud computing or open source software components
- **Scalability:** Future-proofed capacity to evolve over time with growing business growth and demand
- **Various integration requirements:** The flexibility to interact and integrate with existing and potential new systems
- **Existing system infrastructure:** Technical gap between existing systems such as legacy systems that are much harder to replace or refactor
- **Transactional or Batch processing requirements:** Business requirements of on the fly processing Vs. batch processing to save and optimize resources

- **Security:** How new system would face external pressure from perpetration and ensure confidentiality, availability & integrity

While the actual system architecture itself must be evaluated based on above aspects, it is important that various deployment architectures are also taken into consideration, such as,

- **Cloud Computing Vs. On-premise:** Cloud computing offers the benifit of pay-as-you-go to cut down initial capital investment whereas on-premise implementation is expensive. However, not all systems can be implemented in the cloud. (Banking Systems for example has legal obligations to *NOT* to store any Personally Identifiable Information outside legislative premises)
- **Client Server:** This is the most common design choice before the era of distributed computing and edge computing concepts. Organizations can opt in to shed processing loads to clients to save processing costs etc.
- **Fat/Thin Clients:** While fat clients with all business logic built into one module is easy to implement comparatively to thin clients with multiple lossely-couple modules, these comes with its own advantages and disadvantages

Making Architectural Design Choices

As a responsible architectural design maker, all of these aspects must carefully be evaluated before settling for design choices. These are the most critical choices be made in the pre-implementation phase that would directly impact success or failure of any implementation.

A design document must then be drafted with extensive elaboration and justifications based on which the decision were made for future reference. In real life scenarios, this usually involves leadership level collaboration and presenting various options before mutually agreeing to the final system architecture.

Bibliography

Yadin, A. (2016). *Computer Systems Architecture [First Edition]* [https://ereader.perlego.com/1/book/1640743/1 (visited 2024-12-07)]. Chapman and Hall.