Canterbury Institute of Management (CIM)
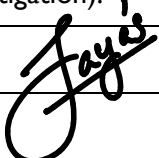## ASSESSMENT COVER SHEET

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 11*		
Due Date & Time	15/12/2024		

Course Lecturer/Tutor Name:	Assessment Word Count (if applicable):
Sharad Neupane	288

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:	12/12/24

Ayesh Jayasekara - CIM12137

System Implementation

There are multiple system development paradigms depending on how the development work is approached. Some of those are,

- Structured Development

- Object-Oriented Development

- Agile Development

Structured	Object-Oriented	Agile
Structured software development involves divide and conquer method to decompose the system to multiple submodules and develop them individually. These submodules may be derived based on logical grouping of business logic.	Logically reimagine the system in terms of real world objects and grouping of their attributes and functions. It usually results in dividing software components to loosely coupled packages that group classes of similar business interests.	A popular system development methodology that suites large software solution with complex & dynamic system requirements that are not clear in the project initiation. Breaks down the system by minimal viable subproducts and make go-to-market times quicker.

> **The object-oriented (OO) approach may be considered as an evolution in thinking about how to relate data structures and programs to one another, it is a "post-relational" approach that intends to facilitate modularity, flexibility, and hence, maintainability**

(Jones, 2018)

Testing & Quality Assurance

Any software solution must be thoroughly tested before releasing to stakeholders. While testing can be achieved in iterations by fraction of business logic (unit tests) and overall quality assurance should follow.

While unit tests focus on successful execution of individual business logic units, quality assurance focus on overal system usability and functions based on positive, negative & edge scenarios.

Installation

Once the development and testing phases are completed, then follows installation phase where an actual software solution is deployed and change over to new system. In the meantime, stakeholders must also be trained to leverage the new functions of the new system.

Bibliography

Jones, C. (2018). *Software Methodologies - A Quantitative Guide [First Edition]* [https://www.perlego.com/book/1486965 (visited 2024-12-12)]. CRC Press.