

SWE30010 – Managing IT Projects

Learning Summary Report

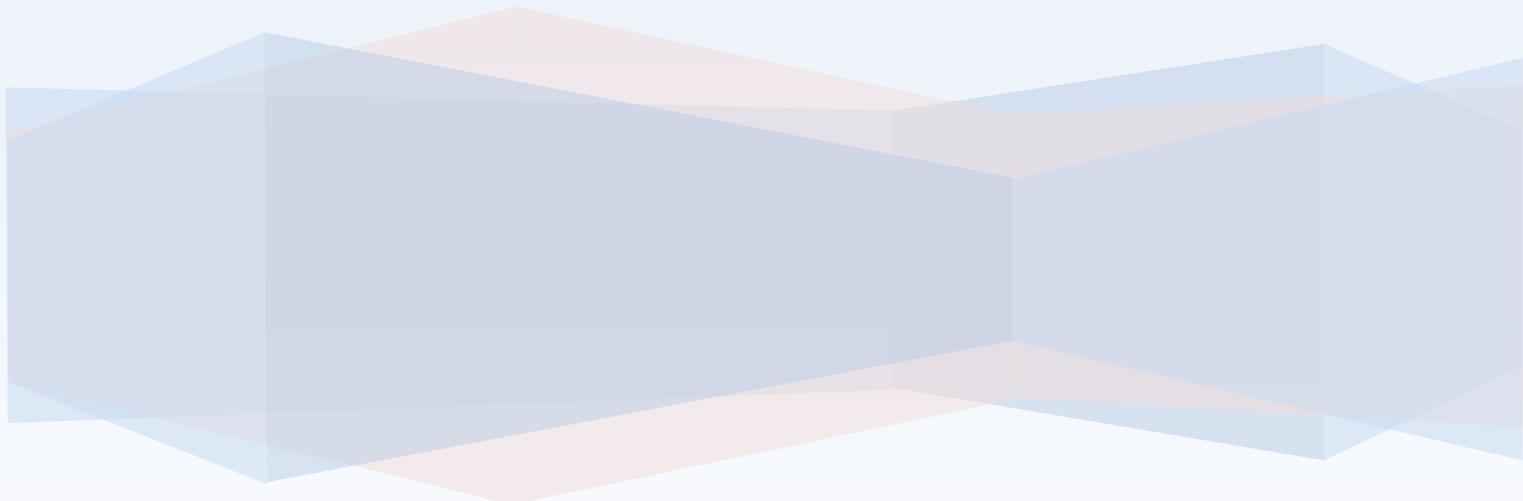
TRAC DUC ANH LUONG (103488117)

Portfolio Submission Due

All Grades: Week 13 Tue (9th April 2024), 08:59 PM

Portfolio Interview Dates

Distinction / High Distinction: Week 12 Fri (5th April 2024), 08:00 AM – 08:15 AM



Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

	Pass (D)	Credit (C)	Distinction (B)	High Distinction (A)
Self-Assessment (please tick)				✓

Self-assessment Statement

	Included (please tick)
Learning Summary Report	✓
All Pass Tasks are Complete on Canvas	✓

Minimum Pass Checklist

	Included (please tick)
All Credit Tasks are Complete on Canvas	✓

Minimum Credit Checklist, in addition to Pass Checklist

	Included (please tick)
Interview booked	✓
All Distinction Tasks are Complete on Canvas	✓
Other pieces (please specify)	

Minimum Distinction Checklist, in addition to Credit Checklist

	Included (please tick)
Software Project Document [Plan, Design, QA] meet HD criteria and standards	✓
Research Article / Essay meets HD criteria and standards	✓
Other pieces (please specify)	

Minimum High Distinction Checklist, in addition to Distinction Checklist

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person.

Signature: _____ Trac Duc Anh Luong _____

Portfolio Overview

This portfolio includes work that demonstrates that I have achieve all Unit Learning Outcomes for SWE30010 Managing IT Projects to a **High Distinction** level.

Overview of pieces included

This section list out the pieces that I have included in my portfolio submission.

Group Tasks

Task	Task Name	Link
02P	Scope and Product Backlog	https://drive.google.com/file/d/1wCph4mzYcYpV-y_l6plrl2IBEk1315nv/view?usp=drive_link
04P	Solution Direction and Design	https://drive.google.com/file/d/19T4gOyLViBhA4TaICSVFW8jBLckUcNe8/view?usp=drive_link
06P	Software Quality and Definition of Done	https://drive.google.com/file/d/1p6aPewz6pMKmsYLgUkVbcb23ADPA7OKX/view?usp=drive_link
07P	Project Proposal	https://drive.google.com/file/d/1OqxLitR_0-R7fz_dn8FDlmid4S_qRRvd/view?usp=drive_link
08P	Sprint Planning Meeting	https://drive.google.com/file/d/1W_CEM7Uj_mjA7OZKaOTk8bQITj1VUdvL/view?usp=drive_link
09P	Project Set Up	https://drive.google.com/file/d/1Nelj8rJp9x1pjxZkgvNmaKv4RwaGokBq/view?usp=drive_link
10P	Sprint 1 Mid-Point Progress Reporting	https://drive.google.com/file/d/1-9_B3K8ujZJI4CEBQAro3nfkvmWw6nu1/view?usp=drive_link
11P	Sprint 1 End-of-Sprint Progress Reporting	https://drive.google.com/file/d/1DXcM4K7jYvLSL5PRmDTGJWX7EHSmOS0D/view?usp=drive_link
12P	Sprint Review	https://drive.google.com/file/d/1RowiYIYamgqlKqoJZP8taoDLqQG-_rg4/view?usp=drive_link
13P	Sprint Retrospective	https://drive.google.com/file/d/1LYa_sTcG__-IoETYeDQh1frTFF8o12om/view?usp=drive_link
14P	Software Design	https://drive.google.com/file/d/11vNJB-aeuvepkG9JipgQN7HyL1k3dV8G/view?usp=drive_link

Individual Tasks

Task	Task Name
01P	Scope and Product Backlog
03P	Solution Direction and Design
05P	Software Quality and Definition of Done
15P	Peer Review
61C	Estimation Method
62C	Estimation Accuracy
71D	Quality Definition
72D	Quality Planning
73D	Quality Review
81HD	Software Project Planning, Design, and Quality Management

82HD	Software Development Methodology Comparison
------	---

01P - Scope and Product Backlog

Project Proposal: Let's Shop E-commerce Website

Synopsis

Let's Shop: Powering Your Tech Dreams in Hanoi's Heart

Hanoi's vibrant streets are filled with digital energy, and its residents seek a seamless way to embrace the latest technologies. Let's Shop steps into the market, a tailor-made e-commerce platform catering specifically to local demand.

Local Gems, Global Choices: Leaving generic products behind, Let's Shop features a diverse selection of electronics, reflecting Hanoi's uniqueness. We handpick international household names and local favourites, from smartphones to home appliances.

Effortless Shopping: Our user-friendly website prioritises simplicity, guiding customers through a smooth shopping experience. Customers can easily browse, compare, and add items to their carts. Secure payment gateways guarantee peace of mind at checkout.

Convenience Delivered: We know our customers' time is precious. They can track their order every step of the way and choose between speedy home delivery or convenient store pickup for added flexibility.

Data-Driven Decisions: Let's Shop goes beyond simple selling. Our integrated analytics dashboards reveal valuable insights into customer trends and preferences, allowing us to refine and personalise their shopping journey constantly.

Beyond Products: We foster a vibrant community with product ratings and reviews, encouraging customers to share their experiences and make informed purchasing decisions.

Fostering Hanoi's Core: While Let's Shop strives to be a comprehensive solution, some aspects fall outside our current focus. We currently facilitate online transactions, leaving inventory management to individual sellers. International shipping and a mobile app are yet to be options but remain future possibilities.

Let's Shop is more than just an online store; it bridges local needs with global possibilities. We aim to become the ultimate destination for Hanoi residents seeking a convenient, reliable, personalised electronics shopping experience.

Background

The rise of digital technology has transformed the retail landscape worldwide, and Vietnam is no exception. In Hanoi, the capital city known for its bustling markets and vibrant commerce, there is a growing demand for a localised online platform that caters to the electronics market.

The proposed project aims to develop an e-commerce website dedicated to selling electronics specifically for the residents of Hanoi. This platform will provide a wide range of electronic products and cater to local needs and preferences.

Scope

Objectives

The primary objective of this project is to develop a user-friendly e-commerce website dedicated to selling electronics to the residents of Hanoi. The platform will be designed to cater to the specific needs and preferences of the local market, offering a wide range of electronic products from both international and local brands.

Specifications

- User Roles:** The website will have two roles: Customer and Admin. The Admin Panel will be separated from the customer's user interface.

- **User Experience:** Provide an intuitive user interface that ensures the shopping experience and simple navigation.
- **Product Range:** Offers a wide range of electronic products from both global and local brands.
- **Product Placement:** Items added to the cart can be checked out and create orders.
- **Secure Payment:** Integrate safe payment gateways from established third-party providers.
- **Order Delivery:** Monitor the order status and email the user of changes.
- **Data Analytics:** Integrate statistic charts and graphs into the admin panel, providing insightful information about the customer's trends and pain points.
- **User Group:** Customise the product offerings to the local market in Hanoi, Vietnam and provide a 'store pickup' option for customers who prefer to collect their purchases from a physical location.
- **Ratings and review:** Upon receiving the order, the customers can provide feedback through ratings and reviews for each purchased product.

Out of scope

While we aim to provide a comprehensive e-commerce solution, certain aspects are beyond the scope of this project:

- We will not be managing the physical inventory of products. The platform will serve as a medium for online transactions, and the responsibility of inventory management lies with the respective sellers.
- The platform will not offer international shipping. At this stage, the service is localised to Hanoi.
- We will not be developing a mobile application for the platform. The focus is creating a responsive website that works well on various devices.

Deliverables and schedule

Initial Release Schedule of the Product backlog items

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule (Sprint 1 2 3 ...)
F1	Design database schema	None	7	Sprint 1
F2	Design system architecture	F1	9	Sprint 1
F3	Set up tech stack and environment	F2	8	Sprint 1
F4	Design rough wireframes	F2	6	Sprint 1
F5	Design an interactive prototype	F4	8	Sprint 1
F6	Set up SMTP server for email delivery and OTP mechanism	F3	7	Sprint 2
F7	Set up user roles, develop user registration and login authentication	F3	9	Sprint 2
F8	Develop admin screens for product, category, and order monitoring	F3	8	Sprint 3

F9	Develop user screens for product browsing and filtering, shopping cart, and order placement	F3	9	Sprint 3
F10	Integrate secure payment for transaction	F3	10	Sprint 4
F11	Create Admin dashboard with data analytics providing customer insights and trends	F3	8	Sprint 5
F12	Develop customer ratings and reviews for purchased items	F9	7	Sprint 5
F13	Deploy app into production	F10, F11, F12	10	Sprint 6
F14	Monitor app performance, gather feedback and reviews for future improvements	F13	9	Sprint 7

Deliverables

- Online electronic store website (fully functional application)
- Source code (all binaries, config files)
- Social media page
- Comprehensive admin manual (online & downloadable)
- Detailed system training program for admin
- Ongoing maintenance and support

Schedule

- **Sprint 1:** January 22 - February 5
- **Sprint 2:** February 6 - March 1
- **Sprint 3:** March 2 - March 29
- **Sprint 4:** March 30 - April 26
- **Sprint 5:** April 27 - May 24
- **Sprint 6:** May 25 - June 21
- **Sprint 7:** June 22 - July 19
- **Sprint 8:** July 20 - August 9

03P - Solution Direction and Design

Project Proposal : Let's Shop Ecommerce Website

Solution Direction

We propose using OutSystems, a low-code web and mobile application development platform for this project. Some alternative tech stacks that can be used are LAMP, MERN, WordPress, and Shopify.

Tech stack	Description	Reason for discard
MERN	Stands for MongoDB, Express.js, React.js, Node.js,	1. Not as secure as other stacks like LAMP, which can be a concern for e-commerce

	is a popular choice for building web applications using JavaScript.	websites that handle sensitive customer information. 2. Not the best choice for large-scale e-commerce websites that require high levels of scalability.
LAMP	Stands for Linux, Apache, MySQL, PHP, is a popular choice for building web applications using PHP.	1. Can be slow and inefficient when handling large amounts of data, which can be a concern for e-commerce websites that handle many transactions. 2. Not be the best choice for e-commerce websites that require high levels of security, as it is more vulnerable to attacks than other stacks like MEAN or MERN
WordPress	A popular content management system that can be used to build e-commerce websites.	1. Not be the best choice for e-commerce websites that require high levels of scalability and security. 2. Can be slow and inefficient when handling large amounts of data, which can be a concern for e-commerce websites that handle many transactions. 3. Is more vulnerable to attacks than other stacks like MEAN or MERN.
Shopify	A popular e-commerce platform that provides a range of features and functionalities.	1. Not the best choice for e-commerce websites that require high levels of customisation and flexibility. Shopify is a closed platform that limits the ability to customise the website beyond the available templates and themes. 2. Shopify charges a transaction fee for each sale made through the platform, which can concern businesses that handle many transactions.

Figure 1: Alternative tech stack explanation and reason for discarding

OutSystems, on the other hand, provides a low-code platform that can help businesses achieve faster development time, higher productivity, better performance, easier maintenance, and scalability. Here is how OutSystems fits in a KoST analysis of our Let's Shop E-commerce Website project:

Criteria	Analysis
Knowledge	1. Problem domain: Retail industry, electronic products that the store sells. This would include knowledge of the target audience, the products, and the competition. 2. Solution domain: Knowledge of e-commerce platforms, web development, and digital marketing.
Skills	1. Experience: This project will significantly benefit web development, e-commerce platforms, and digital marketing.

	<p>2. Other skills: Other skills that would be useful include knowledge of user experience design, search engine optimisation, and content creation.</p>
Technology	<p>1. Existing solutions: Many solutions exist for e-commerce websites, including Shopify, WooCommerce, Magento, and BigCommerce. These platforms provide a range of features and functionalities that can be customised to meet the business's specific needs.</p> <p>2. OutSystems: OutSystems is a low-code platform that enables rapid application development and delivery. It provides a visual development environment that allows developers to build applications without writing code. OutSystems can help businesses achieve faster development time, higher productivity, better performance, easier maintenance, and scalability.</p>

Figure 2: KoST analysis for the project

OutSystems is a low-code platform that can develop custom applications quickly and efficiently. It offers a range of pre-built templates and accelerators that can be customised to meet your specific needs and requirements. OutSystems provides a wide range of features, making it an ideal choice for developing a highly scalable e-commerce website that sells electronic products from a local retail store in Hanoi. Here are some reasons why:

- **Rapid development:** With OutSystems, you can create custom apps more quickly and effectively, which will shorten the time it takes for your e-commerce website to go live.
- **Scalability:** OutSystems' high scalability design enables you to add new features and functionalities as your company grows.
- **Customizability:** OutSystems offers a selection of accelerators and pre-built templates that can be altered to satisfy your unique specifications.
- **Integration:** OutSystems makes combining your e-commerce website with your current data sources and systems simple by integrating with various platforms and systems.
- **Security:** To guarantee that your e-commerce website is safe and secure from online attacks, OutSystems offers strong security measures.

In summary, OutSystems is an ideal choice for developing a highly scalable e-commerce website that sells electronic products from a local retail store in Hanoi. Its rapid development, scalability, customizability, integration, and security features make it a powerful platform for building custom applications.

High-level design

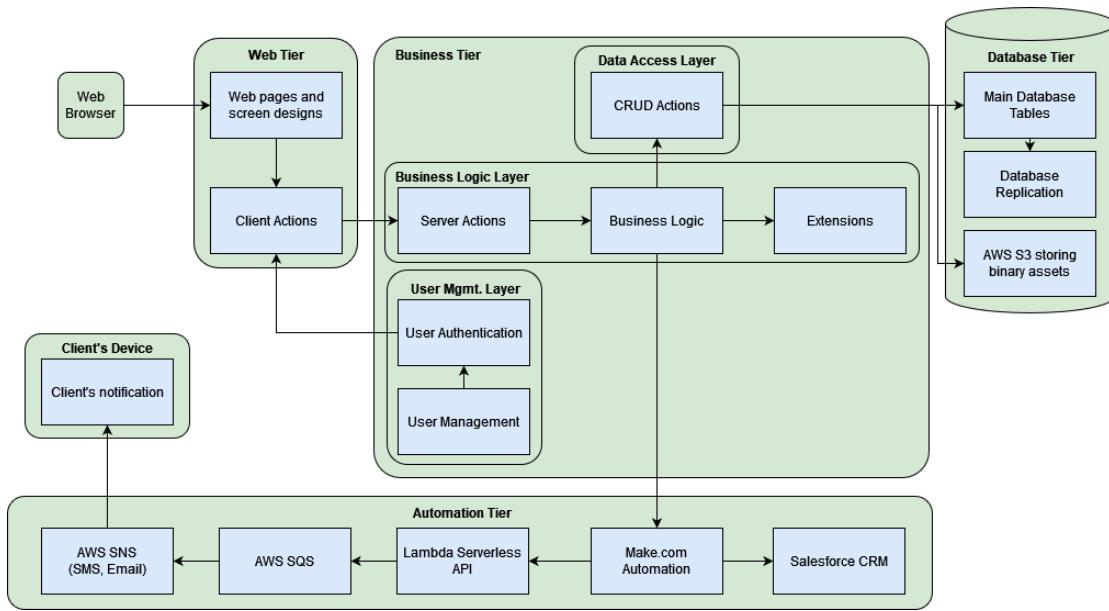


Figure 3: High-level Design Architecture Diagram

1. Web Tier: Includes the UI components of the project.

a. Web pages and screen designs: Include the project's interface, the client interface for shopping

and order placement and the admin interface for order management and data analytics.

b. Client Actions: These are the screen-level functions in OutSystems used to handle client-side events and screen interactions.

2. Business Tier: Includes the server action and serves as a bridge for the client to handle multiple services and access the Data and Automation Tier.

a. Business Logic Layer: Handles the core business logic.

i. Server Actions: runs on the server side. It contains server-side logic and can perform.

Server actions are typically used to implement business logic and are called from client, screen, or server actions.

ii. Business Logic: Deals with different services, including data retrieval, manipulation, input/output with Excel and CSV files, emailing, customer service automation, and validation tasks.

iii. Extensions: We will use extension assets primarily for input handling, Excel input/output, and PDF compilation.

b. Data Access Layer: Directly interacts with the database for data retrieval and manipulations.

i. CRUD Actions: Create, Read, Update, and Delete actions to records stored in the database tables.

c. User Management Layer

i. User Authentication: Using OTP mechanism and session for granting user access.

ii. User Management: Manage how different roles can interact with the system,

following the least-privilege principle for various roles.

d. SMTP Server

3. Database Tier: Stores critical data of the system.

a. Database Tables: Relational database table with schema storing product details, orders, user

accounts, and references to assets in AWS S3.

b. AWS S3: Stores binary assets of the system, including images, videos, and imported Excel files.

4. Automation Tier: Includes external services the system interacts with for customer management.

a. Workato: Automation platform, servers as a skeleton to trigger other services, handles input/output and provides easy integration with the OutSystems platform.

b. Salesforce CRM: A service for managing and maintaining customer relationship management.

c. Lambda Function: Serverless API to handle SNS messaging.

d. AWS SQS: message queuing to decouple and scale our serverless API.

e. AWS SNS: managed messaging service for communication, allowing messaging directly to users

05P - Software Quality and Definition of Done

Project Proposal : Let's Shop Ecommerce Website

Quality Management

Software Quality Definition:

In the context of this project, quality is defined as:

1. Delivering a website that consistently meets customer expectations for functionality, usability, reliability, performance, security, and maintainability.
2. Ensuring the website effectively supports business goals and objectives.

Quality metrics and acceptable measurements:

1. Functional Suitability:

- a. Number of errors found in testing: $\leq 5\%$ of total test cases
- b. Number of defects per KLOC: ≤ 5 defects
- c. User task completion rate: $\geq 95\%$

2. Performance Efficiency:

- a. Average page load time: ≤ 3 seconds
- b. Checkout completion time: ≤ 2 minutes
- c. Uptime: $\geq 99.9\%$

3. Usability:

- a. System Usability Scale (SUS) score: ≥ 70
- b. Time to learn basic tasks: ≤ 5 minutes
- c. Customer satisfaction rating: ≥ 4 out of 5

4. Reliability:

- a. Mean Time Between Failures (MTBF): $\geq 10,000$ hours

- b. Number of critical bugs: ≤ 0 per month

5. Security:

- a. Vulnerability assessment passes: 100%
- b. OWASP Top 10 compliance: 100%

6. Maintainability:

- a. Average time to fix bugs: ≤ 1 day
- b. Code coverage by unit tests: ≥ 80%

Definition of Done (DoD):

The following Definition of Done ensures the software is functionally correct and meets the usability, reliability, performance, and security standards, aligning with the overall project goals and user expectations.

1. All functional requirements were implemented and passed testing.
2. User acceptance testing (UAT) completed.
3. Usability testing was conducted with an average user satisfaction score of 4 out of 5.
4. Reliability tests passed, and the mean time between failure (MTBF) exceeds 10,000 hours.
5. Performance tests conducted with critical operations responding within 2 seconds.
6. A security assessment was performed, and zero critical vulnerabilities were identified.
7. Documentation updated and reviewed.
8. Code review completed with no unresolved issues.
9. Regression testing was executed and passed.
10. Continuous integration builds completed.

15P – Peer Review

SWE30010 Managing IT Projects

Self and Peer Review Assessment Form [Sprint #1]

Date: March 22, 2024

Your Team: Team 1 – The Beavers

Your Name: Trac Duc Anh Luong

Use the instructions (see below) to fill in scores for each category A to J.

Team Members (Name)	A	B	C	D	E	F	G	H	I	J	Total
Trac Duc Anh Luong (Self)	5	5	4	5	5	5	5	5	5	5	49
Minh Nghia Nguyen	5	5	5	4	5	5	5	5	4	5	48
Anh Duc Nguyen	5	4	4	5	5	5	5	5	5	5	48
Tran Dat Dinh	5	4	5	5	5	4	5	5	4	5	47
Gia Minh Nguyen	4	4	5	4	5	5	5	5	4	5	46
Cong Anh Nguyen	3	4	5	4	5	5	5	5	5	5	46

Your Reasoning / Justification (You must write a paragraph about each team member below. Incomplete reviews will not be accepted.)

Name, student number	Comments (complete sentences required)

Trac Duc Anh Luong 103488117 (Self)	As the group leader, I proved myself productive with high-quality work produced throughout the sprint planning and executing sprint 1. I always tried to take the initiative to communicate with teammates to gather task requirements and clear all confusion. The work was always delivered one week before our tutor's early feedback deadline. I took the initiative to prepare the contents and host the group meetings with positive influence while working with teammates. As I put much effort into our project, I could help my peers with many tasks and have their trust and respect.
Minh Nghia Nguyen 103806269	Nghia was productive throughout the sprint and produced high-quality work. He had good verbal and written communication skills in mail and messages, taking initiative in brainstorming tasks and being responsible for UI and data-related tasks. As Nghia took the role of the Scrum Master, his efficiency in completing deadlines was well respected by other members. Nghia was always on time in meetings, with a good attitude, and tried to contribute. Finally, Nghia always strived to help other members.
Anh Duc Nguyen 103488489	Duc delivered high-quality work, with some used as templates for the group to follow. His communication skills were excellent, as his ideas were well-communicated among group members. Duc took the initiative as he joined meetings early and was responsible for monitoring the group's meetings and screen-recorded daily meeting videos. Duc was well respected by team members and the leader, putting effort into contributing to the group's work and being responsible.
Tran Dat Dinh 103487143	Dat is good with catching up with deadlines. His quality of work can be significantly improved with better writing skills. Thanks to his communication and business analytics skills, the group members could understand and appreciate his ideas. Dat showed initiative by showing up early for meetings and preparing notes. Since Dat was responsible and tried to contribute to the group's work, he was well-liked by both the team and the leader. His contributions also resolved choke points in our tasks and made the teamwork easier.
Gia Minh Nguyen 103487156	Minh produced decent work, as his contributions in group submission received good feedback from our lecturer. His thoughts were effectively conveyed to the other group members, demonstrating his strong communication abilities. Minh showed initiative by being early for meetings, preparing keynotes, and contributing considerably. Due to his effort in contributing to the group's work and his responsibility, Minh was respected by both the team and the leader.
Cong Anh Nguyen 103792960	Cong Anh made contributions to the analysis of user stories and requirements. His effort contributed to the ideas and requirements so that the team could quickly move to the development phase and avoid scope creep and misunderstanding. However, further outputs and requirement analysis can be done on his part. Cong Anh's initiative was great, as he researched customization methods and performance improvement techniques in low code. Overall, the team was satisfied with Cong Anh's contributions and performance.

Self and Peer Assessment Form

The main purpose of this form (on Sheet 2) is for all Group members, including yourself, to reflect on its interactions, but it may also be helpful in resolving disputes over the relative contributions of Group members.

Using the spreadsheet Self and Peer Assessment Form

1. List the members of your Project Group
2. Enter a score between 0 and 5, for categories A to J for all members of the group including yourself.
3. You will be asked to take a newly completed form to Group meetings with your supervisor: your supervisor will tell you which meetings.

S. Winger-Haunty (1990). University of Wisconsin-Stout Modified by Pheroza Daruwalla and Ian Knowd, 1994

A. Quantity of Work

- 0 - Did nothing - uninvolved
- 1 - Does enough to get by
- 2 - Occasionally exceeds standards- needs improvement
- 3- Satisfactory. Does more than what is required
- 4 - Very industrious. High Quality. Consistent
- 5. Always exceeds productivity standards. Outstanding

B. Quality of Work

- 0 - Careless. Makes frequent mistakes. Assignment suffers.
- 1 - Mistakes frequent enough to question results.
- 2 - Work is basically correct.
- 3 - Accurate when and where it really counts. Satisfactory.
- 4 - Almost always accurate in all areas of contribution
- 5 - Outstanding. Perfect quality. No mistakes.

C. Communication Skills

- 0 - Blunt, discourteous, does not listen, antagonistic, distant, aloof.
- 1 - Sometimes tactless. Approachable and friendly once known by others.
- 2 - Agreeable and pleasant. Warm, friendly , sociable, listens.
- 3 - Always very polite and willing to help. Very sociable and outgoing. Listens and understands.
- 4 - Courteous and very pleasant. Excellent at establishing good will.
- 5 - Inspiring to others. Artful listener. Really understanding.

D. Initiative

- 0 - Displays no self starting characteristics. Acts without purpose.
- 1 - Puts forth little effort. Requires prodding - sets no speed records.
- 2 - Puts in minimal effort to get task completed.
- 3- Strives hard. Desire to achieve.
- 4 - High desire to achieve. Always puts in a solid days work.
- 5 - Sets high goals. Self starter with high motivation. Constantly goes beyond call of duty.

E. Efficiency

- 0 - Work is invariably late.
 - 1 - Work occasionally completed on schedule.
 - 2 - Work usually complete on schedule. Some contribution to minor problem solving.
 - 3 - Work always complete on schedule.
 - 4 - Work complete. Consistent in defining and resolving major problems.
 - 5 - Work invariably done ahead of schedule. Imaginative.
- Can be counted on to make major contributions.

F. Personal Relations

- 0 - A very disruptive influence
- 1 - Is source of some friction
- 2 - Causes no problems
- 3 - Satisfactory, harmonious
- 4 - Is a positive factor
- 5 - Respected by others. Presence adds to environmental stability

G. Group Meeting Attendance

- 0 - Never attended any meetings. Showed no interest.
- 1 - Occasionally attended. Would commit and then not show.
- 2 - Sometimes uncooperative in planning schedule. Hard to get in touch with.
- 3 - Would attend. Usually late
- 4 - Could be counted on to attend.
- 5 - Never missed a meeting. Always on time

H. Attitude and Enthusiasm

- 0 - Poor disposition, uninvolving, indifferent
- 1 - Unenthusiastic, blasé
- 2 - Half hearted
- 3 - Positive demeanour
- 4 - Positive attitude and spirited.
- 5 - Exuberant and eager. Positive influence. Inspiring to others. Team builder.

I. Effort

- 0 - Puts forth no effort. Expects others to carry the load.
- 1 - Puts forth some effort.
- 2 - Displays enough effort to get by.
- 3 - Solid contributions
- 4 - Strives very hard. Energetic.
- 5 - Self starter. Consistently goes beyond call of duty.

J. Dependability

- 0 - Uninvolved. Unreliable
- 1 - Unsteady, but tries somewhat.
- 2 - Occasionally would come through. Inconsistent.
- 3 - Needs some improvement. Suitable.
- 4 - Very trustworthy. Could be counted on to take responsibility.
- 5 - Always responsible. Kept the group together and in the right direction. Steady influence

61C – Estimation Method

Backlog item selected: Design database schema (Sprint 1)

Task	Estimated Effort	Justification
Identify data requirements	60 minutes	This task involves a thorough understanding of the data stored in the database, what implication of business they will be used for, and the user personas who will be using it (customers and site admins). The task requires gathering information from stakeholders and reviewing any existing documentation. The estimated effort is based on the time necessary to collect and analyse all data requirements, as complex systems will have transaction handling, and diverse stakeholders will result in more time allocation.

Define entities and relationships	90 minutes	This task involves identifying the entities (tables) needed to store the data and their relationships. For OutSystems, we must also identify static entities and their records, which resemble the Enumeration (Enum) data type in high-code programming languages. Delete rules, auto-indexing, and unique keys are other considerations when working with an OutSystems database. The entities will be represented using a data model diagram. The complexity of an extensive schema or intricate relationship will require more time to define.
Normalise the schema	60 minutes	This task ensures that the database schema is efficiently defined with minimal redundancy. The task includes removing duplicate data records and columns, splitting tables into smaller tables, removing many-to-many relationships, and creating foreign keys (FKs). The effort required depends on the normalisation level and the data's complexity. A denormalised schema with more redundancy requires additional effort to normalise effectively.
Document the schema	30 minutes	This task involves documenting the database schema clearly and concisely so that other team members and future maintenance teams can easily understand it. We aim to create a Database definition spreadsheet with the entity diagram and description of the tables and their rows, columns, and relationships. The level of detail our documentation requires will determine the effort needed for this task.

Figure 1: WBS

Total estimated effort: 4 hours

Reasoning:

The estimated effort for each task is based on the task's complexity and the team's experience. The following factors were considered when making these estimates:

1. **Volume of Data:** The large amount of data to be stored can profoundly impact the effort required for our database design task. Handling large volumes of data from concurrent users may require other considerations, such as partitioning, sharding, or multiple modules to simulate a distributed database system.
2. **Data Integrity Requirements:** Enforcing constraints, validation rules, and custom data types will be required to ensure the data integrity of the database. Creating additional database exceptions for data validation and maintaining consistency will take a reasonable amount of time.
3. **Integration with External Systems:** To work with third-party APIs, we require extra effort to design and implement data exchange mechanisms and protocols, handle data conversion, and ensure system compatibility.
4. **Data Access Patterns:** After reading documentation and stakeholder requirements, we need to apply our understanding of how users and applications will access and manipulate data. Creating specific access patterns, such as frequent read operations or complex querying requirements, requires additional effort to maximise performance.
5. **Regulatory Compliance:** Working with sensitive customer data requires the development team to comply with industry-specific standards like HIPAA or GDPR and may impose additional requirements with encryption, auditing, or anonymisation.
6. **Scalability Requirements:** As editing the database schema in production will be costly and time-consuming, getting the scalability requirements upfront, which includes vertical scaling and cache setting for each table, is compulsory.

7. **Collaboration and Communication Overhead:** It may take more time and effort to collaborate with stakeholders, communicate design decisions, and take comments into account. It may be necessary to have repeated talks and adjustments to ensure alignment between the database schema design and stakeholders' expectations.
8. **Tooling and Technology Stack:** Efficiency in database design and development can be influenced by the team's familiarity with the database management system (DBMS) and related tools. The projected time required for tasks like documentation, performance optimisation, and schema normalisation may vary depending on the learning curve related to new technologies or tools. As OutSystems is a relatively new tool, the needed time for our development team may increase.

62C – Estimation Accuracy

Evidence of the effort

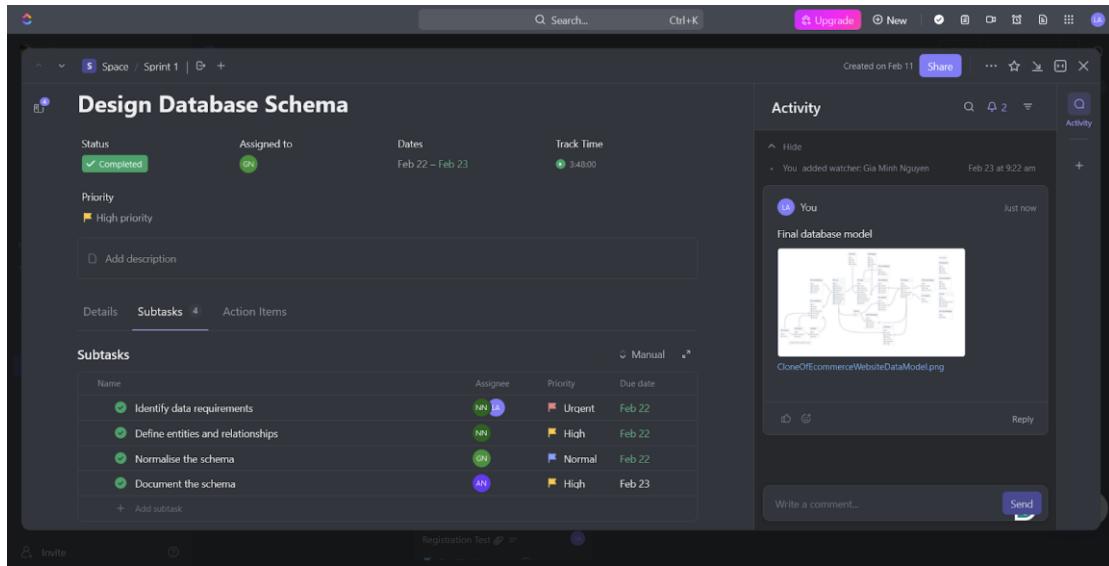


Figure 1: Time tracked for backlog item using ClickUp

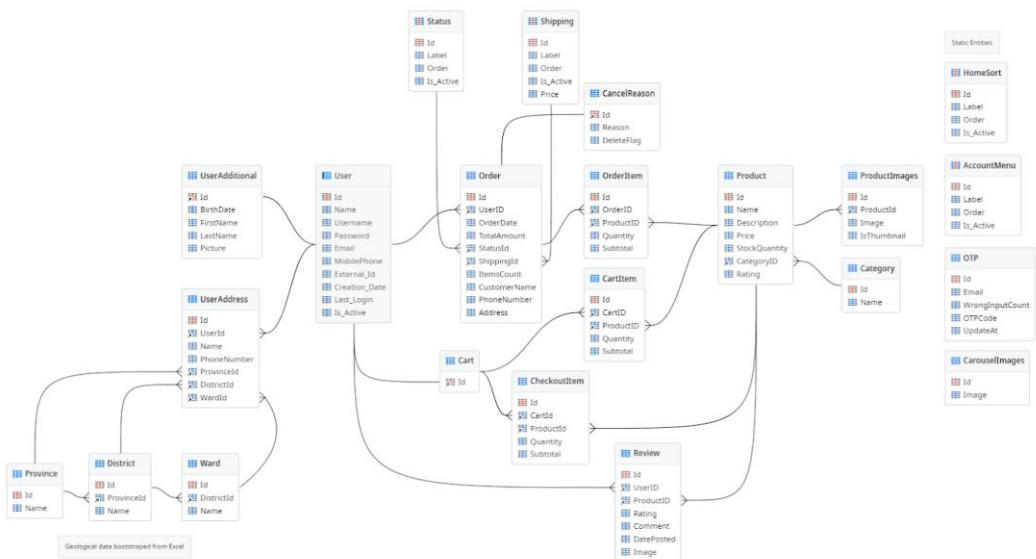


Figure 2: Final database model

Reflection and Analysis

Overview

The purpose of this document is to reflect and analyse the accuracy of effort estimation for the “Design Database Schema” backlog item in sprint 1 compared to the actual time spent tracked using ClickUp (a project management software similar to Jira). In Credit Task 61C, I estimated the team effort required for this task to be 4 hours. Upon completing the task in Sprint 1, the actual time tracked was 3 hours and 48 minutes. My initial estimation was only **6.7% higher** than the actual effort spent, falling within the 10% acceptable range this task requirement set. The percentage indicates a relatively accurate estimation beforehand compared to the actual work.

Reasons for achieving accuracy

1. **Detailed and well-planned WBS:** By creating a detailed Work Breakdown Structure (WBS), the selected task was systematically broken down into more minor, manageable small components. This approach facilitated an accurate assessment of time for each smaller sub-task. Each subtask of the database schema design process was outlined to mitigate the chance of oversight or underestimation of time.
2. **Multifacet factor consideration:** The estimation task considered various factors that could affect the complexity and time of each task and sub-task. Some examples include complexities associated with different data types and media files, security measures implementation, and schema documentation were all taken into consideration. By acknowledging these factors beforehand, the estimation process incorporated extra time to accommodate difficulties, ensuring the effort required is accurately reflected in the estimation effort and time.
3. **Realistic estimation approach:** Instead of relying on arbitrary data or “pure guessing” approaches, the method implemented was practical. Time and effort estimates were carried out using the team’s collective experience, data analysis, and insightful understanding of the team’s strengths and weaknesses. The implemented approach considered the team’s familiarity with the database schema and best practices in database development. The accuracy of the projections and estimates increased overall by leveraging data and insights.
4. **Proactive communication and clarification:** Actively seeking clarification and communicating effectively throughout the working process contributed heavily to the accurate estimation. Engaging in early conversations with stakeholders to clearly understand their specific data usage helped avoid additional out-of-scope requirements in the development process. The team also exchanged clarifying questions about the data stored, its sources, and additional complexity to ensure our understanding of the system. Teamwork and collaboration have leveraged individual expertise and collective knowledge as we received constructive feedback for potential oversights or further refinement.

Maintaining accuracy in future tasks

From this success, continuous improvement will be implemented, with my detailed plan to refine future task effort estimations.

1. **Time tracking for sub-tasks:** Track time for smaller tasks within the WBS can provide more accurate data for future estimation.
2. **Improve complexity assessment:** Improve the skills and ability to identify and consider potential complexities impacting the actual effort.
3. **Active collaboration and communication:** The key to teamwork estimation is to discuss with team members during estimation meetings for different perspectives and collective insights.
4. **Historical data analysis:** Leveraging past project records and task complexion times to create informed decisions for future tasks.
5. **Utilising project management and estimation tools:** With the help of existing project management tools like Jira and ClickUp, these tasks can be executed

automatically or manually with high accuracy and auto-generated reports when a sprint is completed.

Conclusion

The "Design Database Schema" task estimation within the permitted range shows a respectable degree of accuracy. I am confident that implementing the suggested changes will enhance my estimation abilities and the team's effectiveness even more and reliably complete tasks within the allotted time in subsequent tasks and sprints.

71D – Quality Definition

Selection

1. **Sprint backlog item:** Set up user roles, develop user registration and log-in authentication
2. **Characteristic:** Security
3. **Sub-characteristic:** Integrity

Justification

1. **Security:** The selected sprint backlog item interacts directly with sensitive user data, including passwords, personal information, and other potential information specific to our team's application. Ensuring the security of these data is instrumental for multiple reasons:
 - a. **Compliance with regulations:** Industry standards like GDPR and HIPAA denote specific security measures to protect user data. With potential legal repercussions from these regulators, it is best to ensure compliance by implementing robust security.
 - b. **Protecting user privacy:** Upon the user's agreement to the privacy and security policy when registering for our website, our team is entrusted with the user data and is responsible for handling them well. Strong security measures can enhance the user experience and trust in our system.
 - c. **Mitigating security risks:** Unauthorised access and data breaches can lead to enormous revenue losses, reputational damage, and legal repercussions. It is mandatory to follow best practices when developing security measures and minimising future risks.
 - d. **Ensuring other functional requirements:** Compromised user data and unauthorised breaches can disrupt critical system functionalities like order placement and payment gateway, impacting overall system integrity.

2. Integrity:

- a. **Data accuracy and consistency:** The system's correct operation depends on the integrity of the user data. Modified or inaccurate data might result in errors, malfunctions, and a decline in trust.
- b. **Preventing unauthorised modifications:** Malicious individuals or unforeseen mistakes may alter user data. Maintaining data accuracy and preventing unwanted modifications is achieved through data integrity.
- c. **Supporting audibility and traceability:** Proper recording and logging of user actions and modifications to data is essential for auditing and looking into possible security events.

Metrics and threshold values

Our goals include required fields, password strength, OTP verification, unauthorised access prevention, role-based access rights, and activity logging.

1. Registration:

Metric 1: Percentage of successful registrations with all required fields completed

- Threshold: **99.5%** (accounting for minor edge cases or user errors)

- Justification: Ensures all essential information is captured for user accounts. The validation will be executed on the client's side. Therefore, there can be minor edge cases or client-side exceptions.

Metric 2: Percentage of registrations with strong passwords as defined by our policy

- Threshold: **95%**
- Justification: Protects against weak password attacks and data breaches. Passwords are matched against regex patterns, including length, lowercase, uppercase, numbers, and special characters. However, there can be edge cases where the password matches all regex requirements but is still predictable, for example, "@Aaaaaaa1".

Metric 3: Percentage of registrations where email OTP verification is successful

- Threshold: **99.9%** (high threshold due to the security importance of email verification)
- Justification: Prevents unauthorised account creation and validates email addresses. Actions that call to the SMTP server for sending email are server-side. Therefore, there are little to no errors exposed by the client side. The only server-side exception can be email timeout due to many concurrent users.

2. Login:

Metric 1: Number of successful unauthorised login attempts

- Threshold: **0 (zero tolerance)**
- Justification: Prevents unauthorised access and potential security breaches. This metric is the most critical metric for login and security measures. Exposing admin privileges to unauthorised users is detrimental. Therefore, this metric has zero tolerance.

Metric 2: Percentage of login attempts with correct user role authorisation

- Threshold: **100%** (high threshold for ensuring appropriate access control)
- Justification: Ensures users only access functions and data authorised for their roles. The application assigns the user role upon systems events - log in and on application ready, which all screens and server actions will later be checked based on the privileges set in each user session.

3. Logging:

Metric 1: Percentage of registration activities successfully logged

- Threshold: **99.9%** (high threshold for complete activity auditing)
- Justification: Enables tracking user actions and identifying potential suspicious activity. The IP address of users who try to perform DDOS or spam the registration action will be blacklisted.

Metric 2: Percentage of login activities successfully logged

- Threshold: **99.9%** (high threshold for complete activity auditing)
- Justification: Provides essential data for security analysis and incident response. IP addresses of users who tried to force break into the system will be blacklisted.

Metric 3: Timeliness of log data availability for analysis

- Threshold: **15 minutes** (adjustable based on system criticality and response needs)
- Justification: Ensures quick access to log data for investigation and potential remediation. The threshold can be in a shorter time interval when the number of concurrent users is medium to low, with less logging and faster retrieval time (Excel or CSV export).

72D – Quality Planning

From the quality requirements defined in 71D, here are the detailed plan and checklist items for testing and validation to ensure the “Set up user roles, develop user registration and log-in” backlog item satisfies the requirements.

1. Registration

The screenshot shows a ClickUp task card for 'Registration Test'. The card details include:

- Status: TO BE CONFIRMED
- Assigned to: UX
- Dates: 4 days ago – Yesterday
- Priority: Normal priority
- Sprint points: 2

Metric details:

- Metric 1: Percentage of successful registrations with all required fields completed
- Metric 2: Percentage of registrations with strong passwords as defined by our policy
- Metric 3: Percentage of registrations where email OTP verification is successful

The activity feed on the right shows two screenshots of a web application interface. The first screenshot shows a registration form with several fields highlighted in red, indicating they are required. The second screenshot shows a success message after a registration attempt.

Figure 1: ClickUp evidence for Registration Test

Metric 1: Percentage of successful registrations with all required fields completed (**99.5%**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Are the required fields highlighted in red when missing upon clicking “Register”?	<input type="checkbox"/>	<input type="checkbox"/>
2	Warning “Email expected!” appears when the user enters the wrong email format.	<input type="checkbox"/>	<input type="checkbox"/>
3	Warning “Email already exists” appears when the user enters an email address that has already been registered.	<input type="checkbox"/>	<input type="checkbox"/>
4	Warning “Invalid phone number” appears when the user enters the wrong phone number format (Vietnamese phone number).	<input type="checkbox"/>	<input type="checkbox"/>
5	Warning “You have to be 18 or older to register” appears when the user enters a date of birth less than 18 years from today.	<input type="checkbox"/>	<input type="checkbox"/>
6	Warning “Birthdate cannot be in the future” appears when the user enters a date of birth in the future from today.	<input type="checkbox"/>	<input type="checkbox"/>

Metric 2: Percentage of registrations with strong passwords as defined by our policy (95%)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Password must contain at least 8 characters.	<input type="checkbox"/>	<input type="checkbox"/>
2	Password must contain at least 1 uppercase character.	<input type="checkbox"/>	<input type="checkbox"/>
3	Password must contain at least 1 lowercase character.	<input type="checkbox"/>	<input type="checkbox"/>
4	Password must contain at least 1 number.	<input type="checkbox"/>	<input type="checkbox"/>
5	Password must contain at least 1 special character.	<input type="checkbox"/>	<input type="checkbox"/>
6	Warning “Password does not meet requirement” appears when one of the above requirements is unmet.	<input type="checkbox"/>	<input type="checkbox"/>
7	The tooltip updates accordingly based on password input changes.	<input type="checkbox"/>	<input type="checkbox"/>

Metric 3: Percentage of registrations where email OTP verification is successful (99.9%)

No.	Task	Check the appropriate column	
		Yes	N/A
1	OTP sends to a valid email.	<input type="checkbox"/>	<input type="checkbox"/>
2	OTP code is a 6-digit number.	<input type="checkbox"/>	<input type="checkbox"/>
3	A popup to enter OTP appears with a 120-second countdown.	<input type="checkbox"/>	<input type="checkbox"/>
4	Entering the wrong OTP code more than 2 times disables the submit button.	<input type="checkbox"/>	<input type="checkbox"/>
5	When the countdown ends, the submit button is disabled, and the “Resend OTP” link appears.	<input type="checkbox"/>	<input type="checkbox"/>
6	Entering the correct OTP on the 1st try before the countdown ends creates a user account.	<input type="checkbox"/>	<input type="checkbox"/>

7	Entering the correct OTP on the 2nd try before the countdown ends creates a user account.	<input type="checkbox"/>	<input type="checkbox"/>
---	---	--------------------------	--------------------------

2. Login

Figure 2: ClickUp evidence for Login Test

Metric 1: Number of successful unauthorised login attempts (**0 - zero tolerance**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Fail to log in with incorrect email.	<input type="checkbox"/>	<input type="checkbox"/>
2	Fail to log in with correct email and incorrect password.	<input type="checkbox"/>	<input type="checkbox"/>
3	Error message “Invalid username or password” appears in no. 1 & 2 scenarios.	<input type="checkbox"/>	<input type="checkbox"/>
4	Login successfully with correct username and correct password.	<input type="checkbox"/>	<input type="checkbox"/>

Metric 2: Percentage of login attempts with correct user role authorisation (**100%**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	POST request from <i>ActionDoLogin</i> using Admin credentials returns a response containing rolesInfo of Admin.	<input type="checkbox"/>	<input type="checkbox"/>
2	POST request from <i>ActionDoLogin</i> using Customer credentials returns a response containing rolesInfo of Customer.	<input type="checkbox"/>	<input type="checkbox"/>
3	Upon logging in, the Admin user is re-directed to the Admin Dashboard.	<input type="checkbox"/>	<input type="checkbox"/>
4	Upon logging in, the Customer user is re-directed to the Home page.	<input type="checkbox"/>	<input type="checkbox"/>
5	Customer's browser local storage has <i>\$OS_Users\$EcommerceWebsite\$ClientVars\$IsAdmin</i> key set to <i>false</i> .	<input type="checkbox"/>	<input type="checkbox"/>
6	Admin's browser local storage has <i>\$OS_Users\$EcommerceWebsite\$ClientVars\$IsAdmin</i> key set to <i>true</i> .	<input type="checkbox"/>	<input type="checkbox"/>
7	Customer accessing Admin's screens through URL is re-directed to the InvalidPermissions screen.	<input type="checkbox"/>	<input type="checkbox"/>
8	Admin accessing Customer's screens through URL is re-directed to the InvalidPermissions screen.	<input type="checkbox"/>	<input type="checkbox"/>

3. Logging

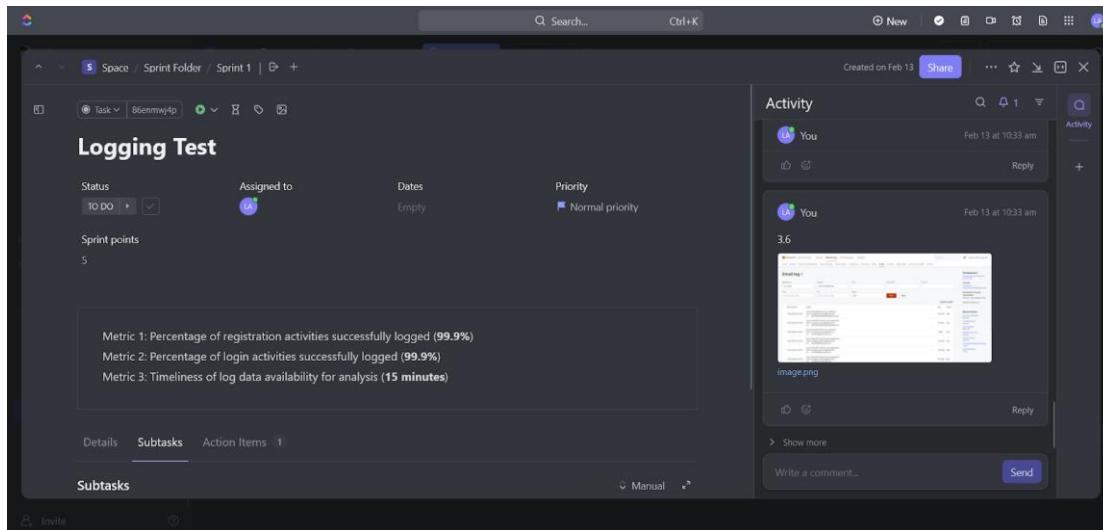


Figure 3: ClickUp evidence for Logging Test

Metric 1: Percentage of registration activities successfully logged (**99.9%**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Register using action <i>GetOtpsByEmail</i> is logged to the Screen Requests tab on OutSystems Service Center.	<input type="checkbox"/>	<input type="checkbox"/>

Metric 2: Percentage of login activities successfully logged (**99.9%**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Failed logins are logged to the Error tab on OutSystems Service Center.	<input type="checkbox"/>	<input type="checkbox"/>
2	Successful logins are logged to the Screen Requests tab on OutSystems Service Center.	<input type="checkbox"/>	<input type="checkbox"/>

Metric 3: Timeliness of log data availability for analysis (**15 minutes**)

No.	Task	Check the appropriate column	
		Yes	N/A
1	Error log ready within 3 minutes	<input type="checkbox"/>	<input type="checkbox"/>
2	General log ready within 3 minutes	<input type="checkbox"/>	<input type="checkbox"/>
3	Screen requests log ready within 2 minutes	<input type="checkbox"/>	<input type="checkbox"/>
4	Extension log ready within 3 minutes	<input type="checkbox"/>	<input type="checkbox"/>
5	Timer log ready within 2 minutes	<input type="checkbox"/>	<input type="checkbox"/>
6	Email log ready within 2 minutes	<input type="checkbox"/>	<input type="checkbox"/>

Figure 4: ClickUp task board for all 3 testing tasks

73D – Quality Review

Introduction

In task 72D, we have introduced the planning for our quality assurance activities and checklist items. For this task, each item will be conducted with evidence and comparison to proposed metrics to re-evaluate our plan. This report will cover the criteria that worked well, those that needed adjustment, and our approach for future backlog items to adhere to the quality requirements. Our group utilised ClickUp, a recent project management software, to document the QA activities and checklist. Other alternatives that were considered were Jira and Trello.

Reflection

Overall quality

The sprint backlog item achieved high quality, with zero unauthorised login attempts and robust security measures implemented. 40 out of 41 checklist items were successfully tested and marked, with only one exception of the log taking longer to export from the server in the Timer tab.

Quality achieved

1. Registration

Metric 1: 100% successful registration with all required fields. Additional formatting and logic checks for email, phone number, and age.

Metric 2: 95% successful registration with strong passwords. (there are edge cases where the tested passwords meet all requirements but remain predictable using the online password strength evaluator).

Metric 3: 100% successful OTP verifications.

2. Login

Metric 1: 0 successful unauthorised login attempts (zero tolerance criteria achieved)

Metric 2: 100% login attempts with correct user role authorisation (achieved the desired threshold).

3. Logging

Metric 1: 100% successful registration activities were logged (achieved the desired threshold).

Metric 2: 100% successful login activities were logged (achieved the desired threshold).

Metric 3: Log data availability within 20 minutes (exceeding the 15-minute threshold). Only the Timer logs took longer due to configuration issues.

Quality satisfaction:

The sprint backlog items satisfy most quality definitions while only falling short in some areas:

1. **Registration password strength:** The high percentage of strong passwords showcases effective security measures. However, edge cases signal further checking and stricter regular expression patterns.
2. **Logging timeliness:** While most log tabs are readily available, the total exceeded the 15-minute threshold for Timer logs, which requires future optimisations.

Positive outcomes

1. **Zero unauthorised login attempts:** This outcome highlights the effectiveness of security measures in preventing breaches from external threats.
2. **Comprehensive testing:** All quality assurance activities were implemented, ensuring thorough evaluation of the backlog item and satisfying the definition of done in the project proposal.

3. **Detailed documentation:** Clickup checklist, assignees, time log, comments, and visual evidence from screenshots demonstrated completed task and facilitated collaboration.

Challenges encountered

1. **Edge cases in password strength:** While the validation rules are working, unpredictable passwords require additional checks.
2. **Timer log availability:** Configuration issues and personal environment resources caused delays in the Timer log accessibility, requiring optimisation for system analysis.
3. **Minor fluctuation from registration metrics:** Edge cases, user errors, and client-side exceptions affected the desired thresholds, which requires further research and planning.

Potential enhancement for future backlog items

1. **Enhance password validation:** Implement mechanisms to identify and flag predictable passwords despite meeting character requirements. This can include extra regular expressions for matching consecutive letters and patterns.
2. **Optimise timer log configuration:** Refine configurations to ensure timer logs are available within the 15-minute threshold. We planned to cut down the use of unnecessary timers, keep the delayed order checks and monthly reports to admin emails only, and cut down retention time for data from 2 months to 1 month.
3. **Address minor registration issues:** Further refine validation and user guidance to minimise edge cases and user errors. We planned to implement our Vietnamese address API for future references and accurate shipping information. This can quickly be developed and integrated with OutSystems' bootstrap Excel data function.
4. **Continuous monitoring and improvement:** Regularly review metrics, identify potential issues, and implement proactive solutions.

Conclusion

Given that the sprint backlog item met essential quality standards, there is room for improvement. We may further improve this implementation's overall quality and security by taking care of the password strength, availability of the timer log, and minor registration concerns. In the upcoming sprints, we will guarantee that we constantly meet the quality criteria through continuous monitoring and improvement procedures.

Evidence of quality assurance conducted

The screenshot shows a Clickup interface. On the left, a task titled "Registration Test" is displayed with the following details:

- Status: Completed
- Assigned to: [User icon]
- Dates: Feb 23 – Feb 24
- Track Time: 1:12:00
- Priority: Normal priority

Below the task details, there is a box containing three metrics:

- Metric 1: Percentage of successful registrations with all required fields completed
- Metric 2: Percentage of registrations with strong passwords as defined by our policy
- Metric 3: Percentage of registrations where email OTP verification is successful

At the bottom of the task card, there are tabs for Details, Subtasks, Action Items, and Custom Fields. The Activity tab is selected on the right side of the screen, showing two messages from the user:

- Message 1.1: Attached a screenshot of a registration form. The screenshot shows a user profile picture and several input fields for registration information.
- Message 1.2: A message placeholder "Write a comment..." with a "Send" button.

Figure 1: Registration test overview

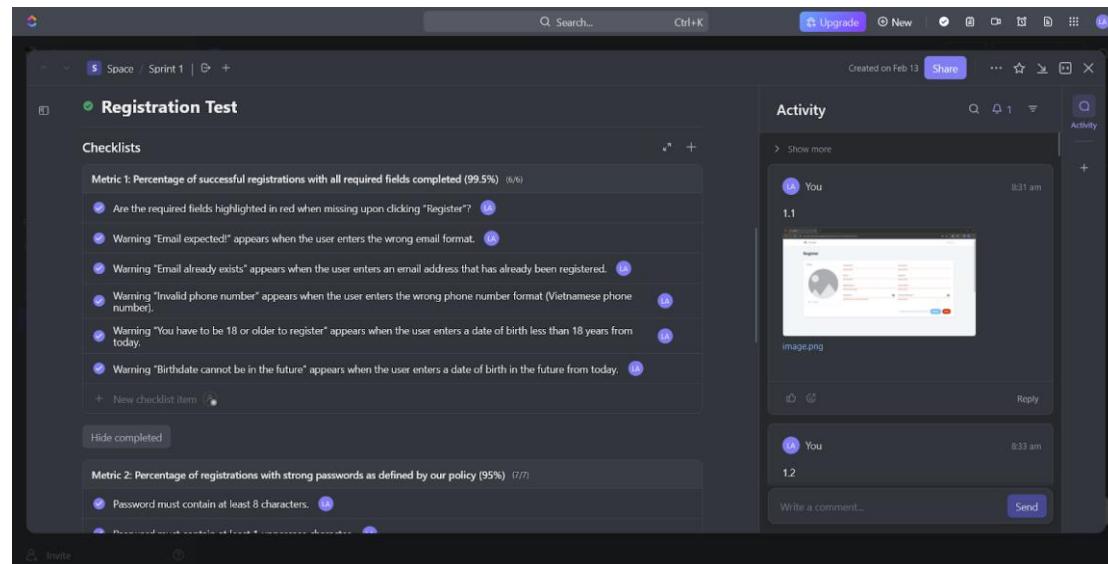


Figure 2: Checklist for metric 1 in registration test

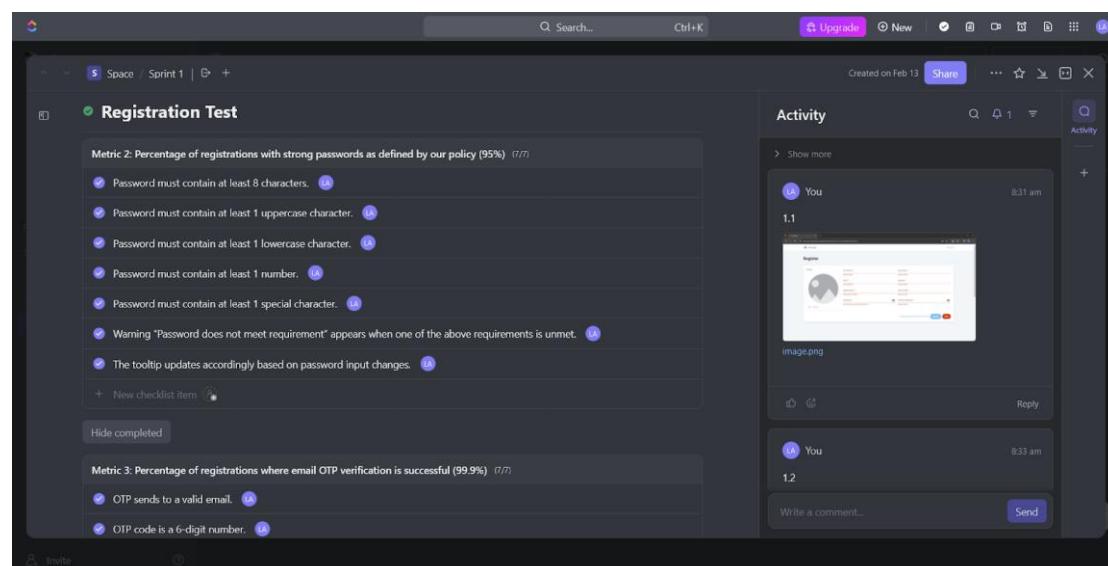


Figure 3: Checklist for metric 2 in registration test

Registration Test

Metric 3: Percentage of registrations where email OTP verification is successful (99.9%) 0/0

- OTP sends to a valid email. 1
- OTP code is a 6-digit number. 1
- A popup to enter OTP appears with a 120-second countdown. 1
- Entering the wrong OTP code more than 2 times disables the submit button. 1
- When the countdown ends, the submit button is disabled, and the "Resend OTP" link appears. 1
- Entering the correct OTP on the 1st try before the countdown ends creates a user account. 1
- Entering the correct OTP on the 2nd try before the countdown ends creates a user account. 1

+ New checklist item 1

Hide completed + Add checklist Use Template

Activity

You 8:31 am 1.1

image.png

You 8:33 am 1.2

Write a comment... Send

Figure 4: Checklist for metric 3 in registration test

Registration Test

image.png 8:59 am image.png 8:56 am image.png 8:49 am image.png 8:48 am

image.png 8:47 am image.png 8:47 am image.png 8:43 am image.png 8:42 am

image.png 8:41 am image.png 8:41 am image.png 8:40 am image.png 8:40 am

Activity

- You created this task 8:24 am
- You changed status from To Do to In Progress 8:24 am
- You added 6 items to checklist Metric 1: Percentage of successful registrations with all required fields completed (99.5%) 8:27 am
- You checked Are the required fields highlighted in red when missing upon clicking "Register" in Metric 1: Percentage of successful registrations with all required fields completed (99.5%) 8:27 am
- You checked Warning "Email expected" appears when the user enters the wrong email format. in Metric 1: Percentage of successful registrations with all required fields completed (99.5%) 8:27 am
- You added 7 items to checklist Metric 2: Percentage of registrations with strong passwords as defined by our policy (99%) 8:28 am
- You added 7 items to checklist Metric 3: Percentage of registrations where email OTP verification is successful (99.9%) 8:28 am
- You assigned to: You 8:29 am
- You added watcher: You 8:29 am

Write a comment... Send

Figure 5: Registration test screenshot evidence according to checklist items

The screenshot shows a Jira task titled "Login Test". The task details are as follows:

- Status:** Completed
- Assigned to:** [User icon]
- Dates:** Feb 23 – Feb 24
- Track Time:** 0:04:00
- Priority:** High priority

Under the "Activity" tab, there is a log of events:

- You checked Customer accessing Admin's screens through URL is re-directed to the invalidPermissions screen. in Metric 2: Percentage of login attempts with correct user role authorisation (100%)
- You checked Admin accessing Customer's screens through URL is re-directed to the invalidPermissions screen. in Metric 2: Percentage of login attempts with correct user role authorisation (100%)
- You set priority to High
- You uploaded a file: LoginCommand.txt

At the bottom, there is a comment input field: "Write a comment..." and a "Send" button.

Figure 6: Login test overview

The screenshot shows the "Action Items" tab for the "Login Test" task. It displays a checklist for "Metric 1: Number of successful unauthorised login attempts (0 - zero tolerance)". The checklist items are:

- Fail to log in with incorrect email. [Status: In Progress]
- Fail to log in with correct email and incorrect password. [Status: In Progress]
- Error message "Invalid username or password" appears in no. 1 & 2 scenarios. [Status: In Progress]
- Login successfully with correct username and correct password. [Status: In Progress]

At the bottom of the checklist, there is a "Hide completed" button. To the right, there is an "Activity" panel showing a message from "You" at 10:00 am: "2.8" followed by an image file named "image.png".

Figure 7: Checklist for metric 1 in login test

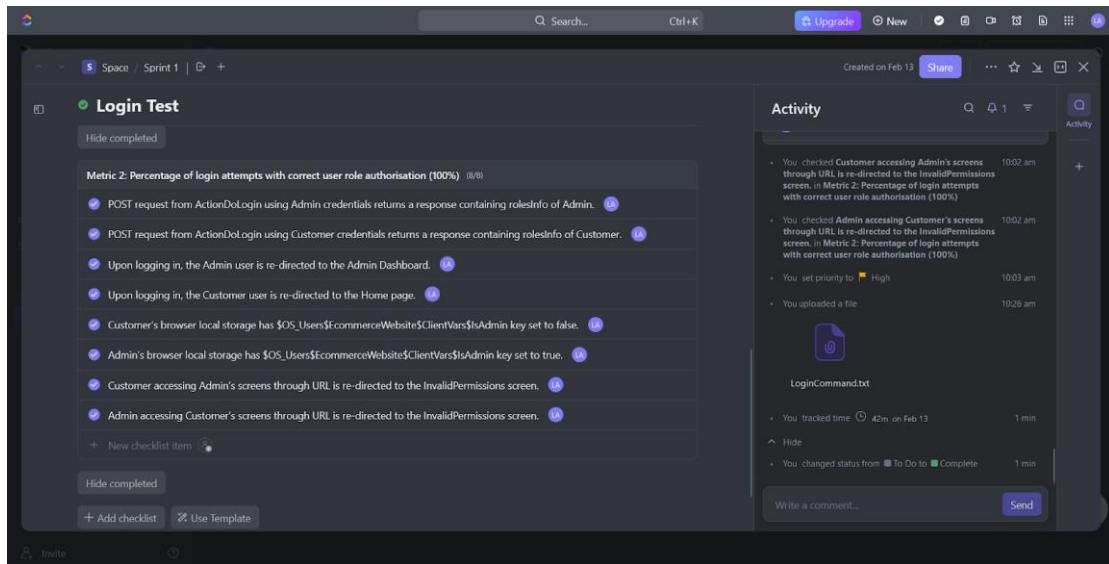


Figure 8: Checklist for metric 2 in login test

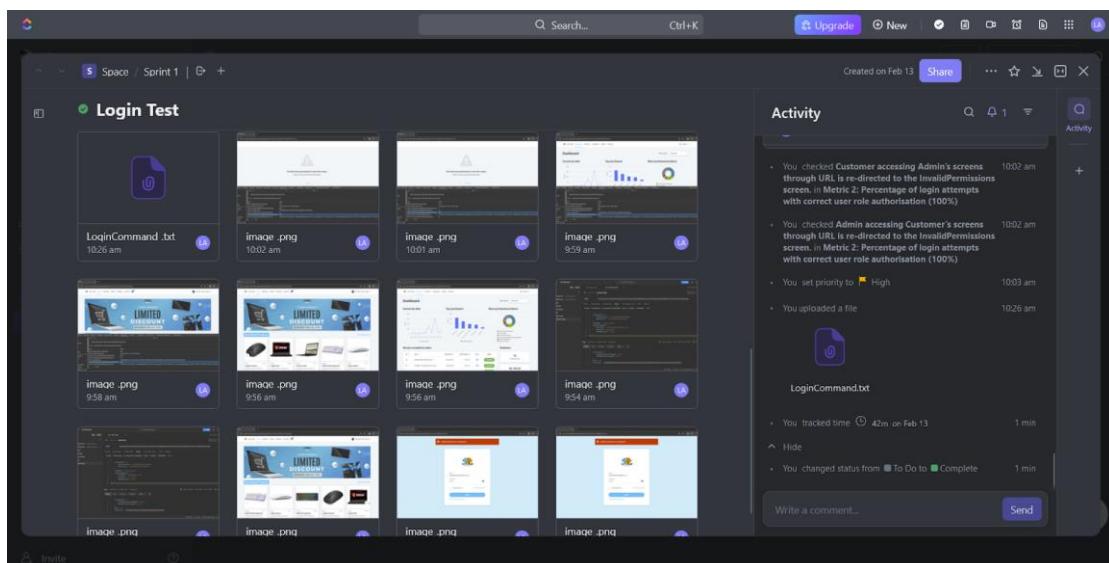


Figure 9: Login test screenshot and bash script evidence according to checklist items

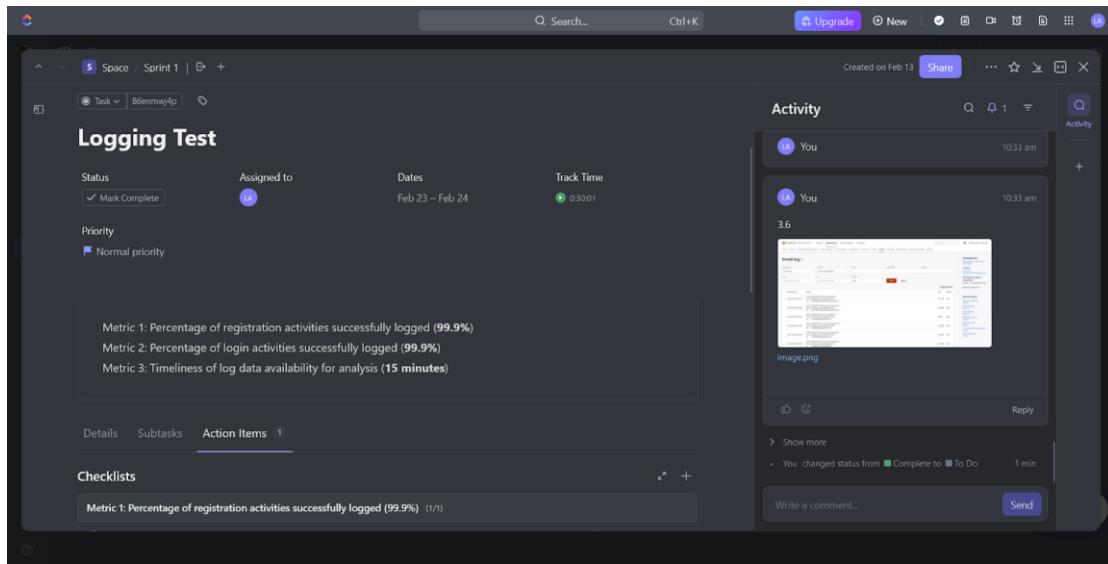


Figure 10: Logging test overview

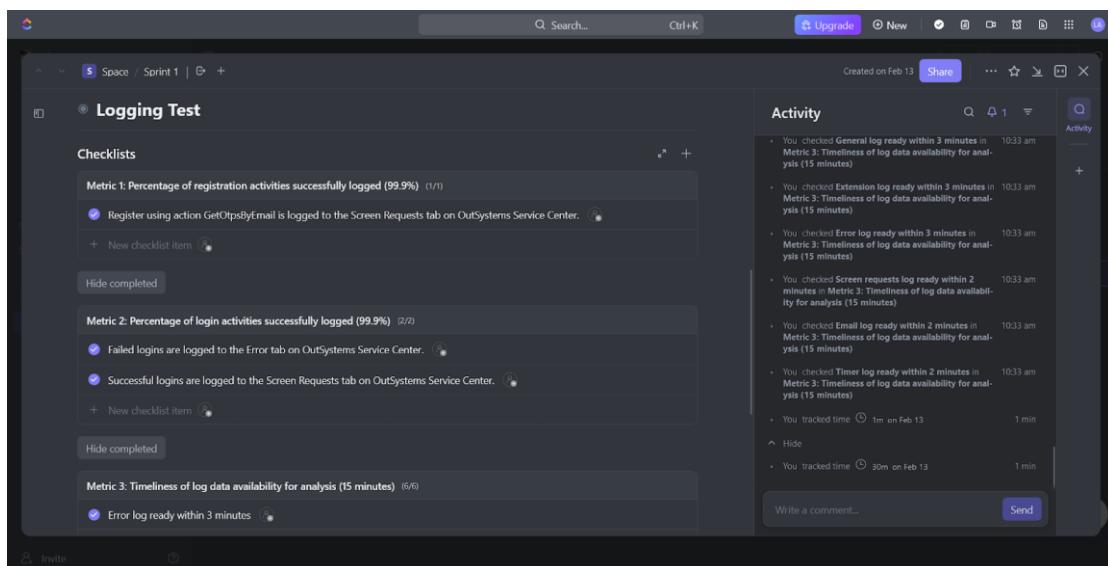


Figure 11: Checklist for metrics 1 and 2 in logging test

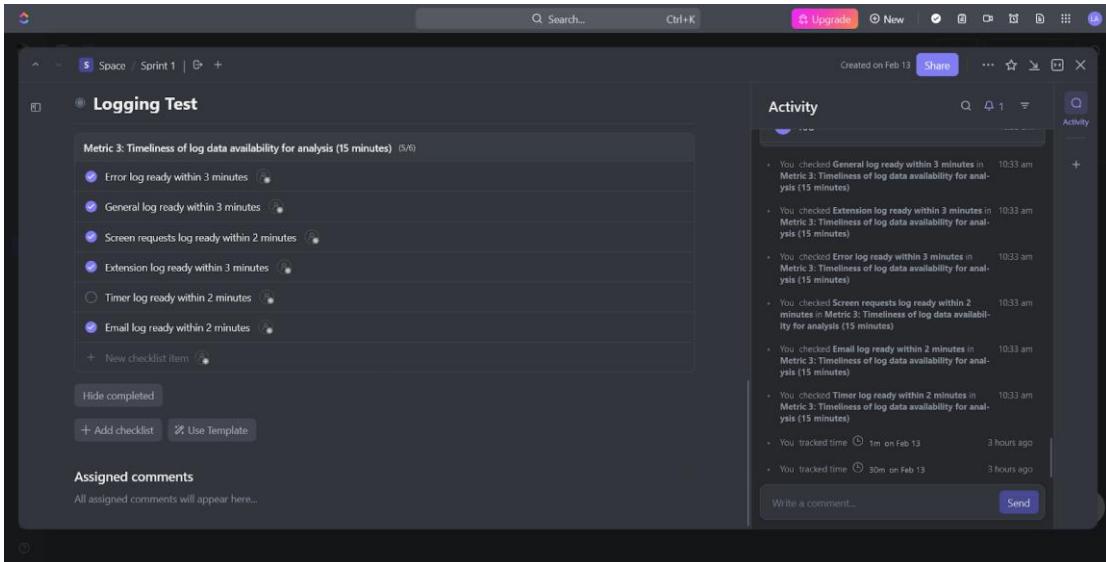


Figure 12: Checklist for metric 3 in logging test

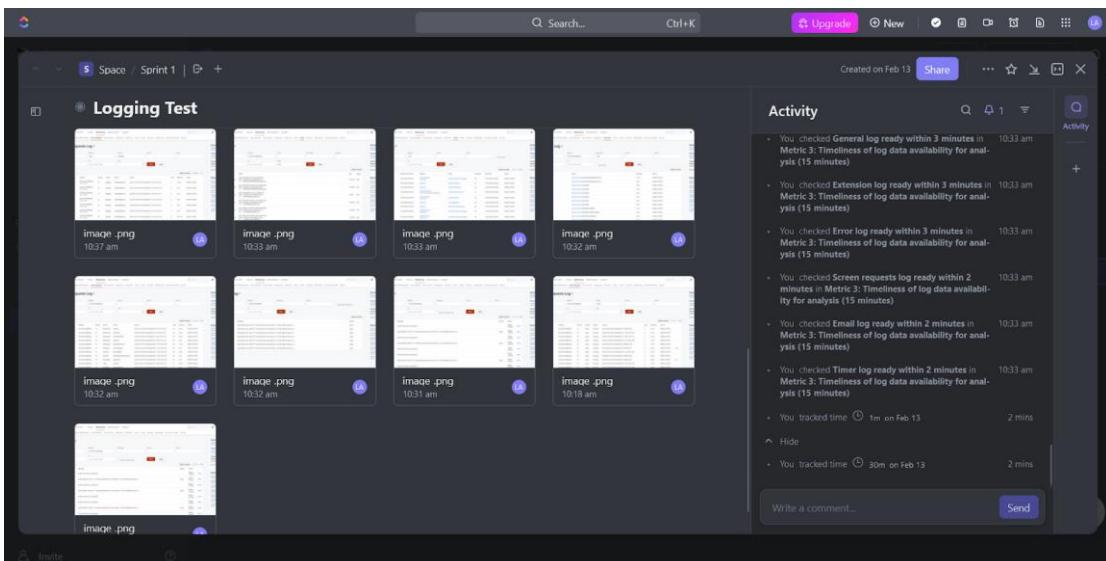


Figure 13: Logging test screenshot evidence according to checklist items

81HD - Software Project Planning, Design and Quality Management

Project description

Project Name: Speakeasy-OMS

Company: Speakeasy Cocktail Bar

Software: Order Management System (Speakeeasy-OMS)

Synopsis

Speakeasy is a popular cocktail bar in Hanoi. We offer various cocktails and drinks with different variations from all over the world, with the choice to add different touches and customisations for our customers.

The bar currently uses physical queues and notes to take orders from customers. It is time-consuming and labour-costly for the bartenders and waiters to remember orders and manually enter the order information into Excel spreadsheets. It is also difficult for new

customers who do not have a favourite drink in mind and need to ask the waiter or bartender for advice.

We would like a web app that lets customers look up drinks based on their preferences and make orders, as well as customised dashboards, order management and data analytics for the managers to monitor.

The web app must have a responsive user interface that is easy to use and a separate admin panel.

Background

Speakeasy is a well-established cocktail bar in Hanoi, known for its extensive menu of international drinks and customizability options. Currently, the bar relies on a manual system of queues, written notes, and Excel spreadsheets to manage orders. This process proves inefficient, requiring significant staff time to handle order taking, memorisation, and data entry. Furthermore, new customers often struggle to navigate the vast drink selection without staff assistance. To address these limitations, Speakeeasy seeks a web application to streamline operations. This app will empower customers to browse drinks based on preferences, place orders directly, and personalise their selections. Additionally, Speakeeasy desires a dedicated management interface for functionalities like order monitoring, data analysis, and customised dashboards. A user-friendly responsive interface and a separate admin panel are key priorities for this web application.

Backlog items list

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule (Sprint 1 2 3 ...)
F1	Design database schema	None	9	Sprint 1
F2	Design UI wireframes and interactive prototypes for the app	None	8	Sprint 1
F3	Setup development environment and modules	F1	7	Sprint 1
F4	Setup user roles (Customer & Admin), Registration and Login Screens	F3	7	Sprint 1
F5	Setup API connector in IS (Integration Service) module	F3	8	Sprint 1
F6	Create customer screens for searching and filtering drinks using API connector	F5	9	Sprint 1
F7	Add order mechanism and QR code for payment integration	F6	7	Sprint 2
F8	Create customer screens for favourite and drink history	F7	7	Sprint 2
F9	Create admin dashboard for analytics and Excel export	F8	8	Sprint 2
F10	Finish documentation: readme, basic design, detail design, etc.	F9	8	Sprint 2
F11	Deploy app into production	F10	10	Sprint 2
F12	Monitor app performance, gather feedback and reviews for future improvements	F11	9	Sprint 2

*Figure 1: Backlog items and release schedules***Deliverables**

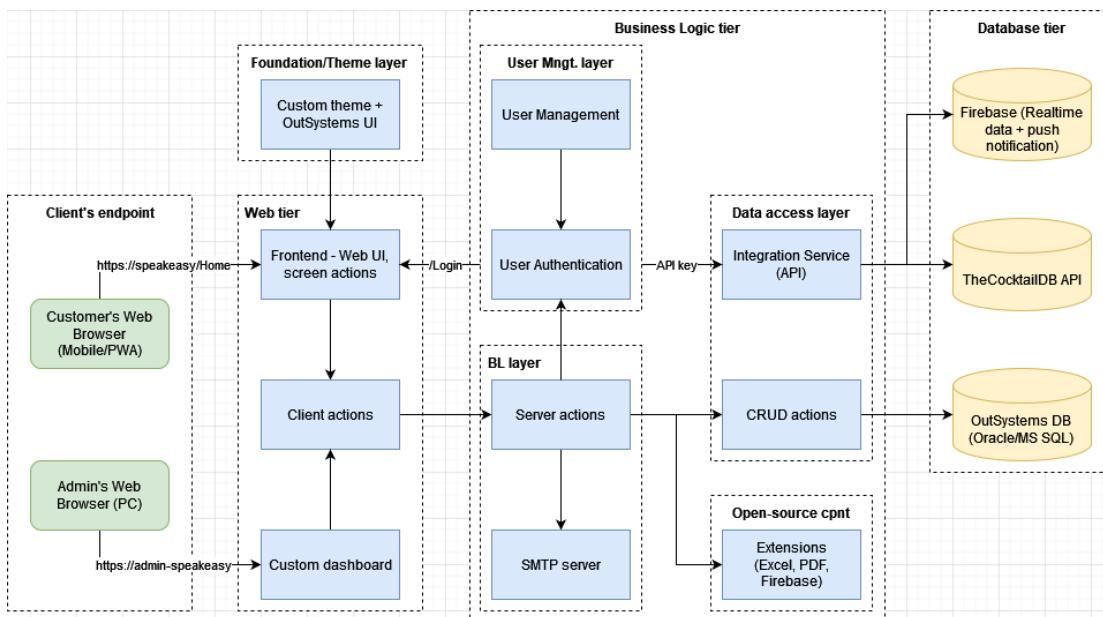
1. Online cocktail order website (fully functional application)
2. Source code (all binaries, config files)
3. System documentation: readme, basic design, detail design, config instructions, etc.
4. Comprehensive admin manual (online & downloadable)
5. Detailed system training program for admin
6. Ongoing maintenance and support

Release schedule

The project will implement **2-week sprints** from April 1 to April 26. The details are as follows:

Sprint 1: April 1 – April 12

Sprint 2: April 15 - April 26

Software design*Figure 2: Software high-level architecture*

In Figure 2, we can see the high-level implementation and interactions between different components of the Speakeasy-OMS software, following a 3-tier architecture.

1. Web tier: The UI components of the system
 - a. Frontend - Web UI, screen actions: Include web pages, compiled HTML, CSS, JS, and screen actions that call to client and server actions. This is the customer's entry to the web page.
 - b. Client actions: Perform system operations on the client side, call to server actions
 - c. Custom dashboard: Entry to the admin's panel, including reports, statistics, and data export functions.
2. Business Logic tier: Server actions that validate data and communicate with the data access layer.
 - a. User Management layer

- i. User Management: Provision of how users with different roles can access the screens following the least-privilege rule.
- ii. User Authentication: Validate OTPs and API keys sent from the client for login and accessing the Integration Service module.
- b. Business Logic layer
 - i. Server actions: Contains server-side logic and business logic.
 - ii. SMTP server: Send emails for OTP and customer service.
- c. Data Access layer
- i. Integration Service (API): Consumer for API to TheCocktailDB and Firebase extension in OutSystems.
- ii. CRUD actions: Performs basic CRUD actions on the OutSystems database.
- d. Open-source components
- i. Extensions: Can implement from open-source Forge or self-custom extensions from the development team. Mostly used for PDF and Excel.
- 3. Database tier: Data storage for the system can be accessed from the Data Access layer in the BL tier.
 - a. Firebase: Allows real-time data and push notifications.
 - b. TheCocktailDB API: Public API that allows querying of data on drinks. The app will use a paid API key for upgraded features and performances.
 - c. OutSystems DB: This database stores customer and order (invoice) information.

WBS and time estimation

For this section, we will provide the detailed plan and estimation for Sprint 1.

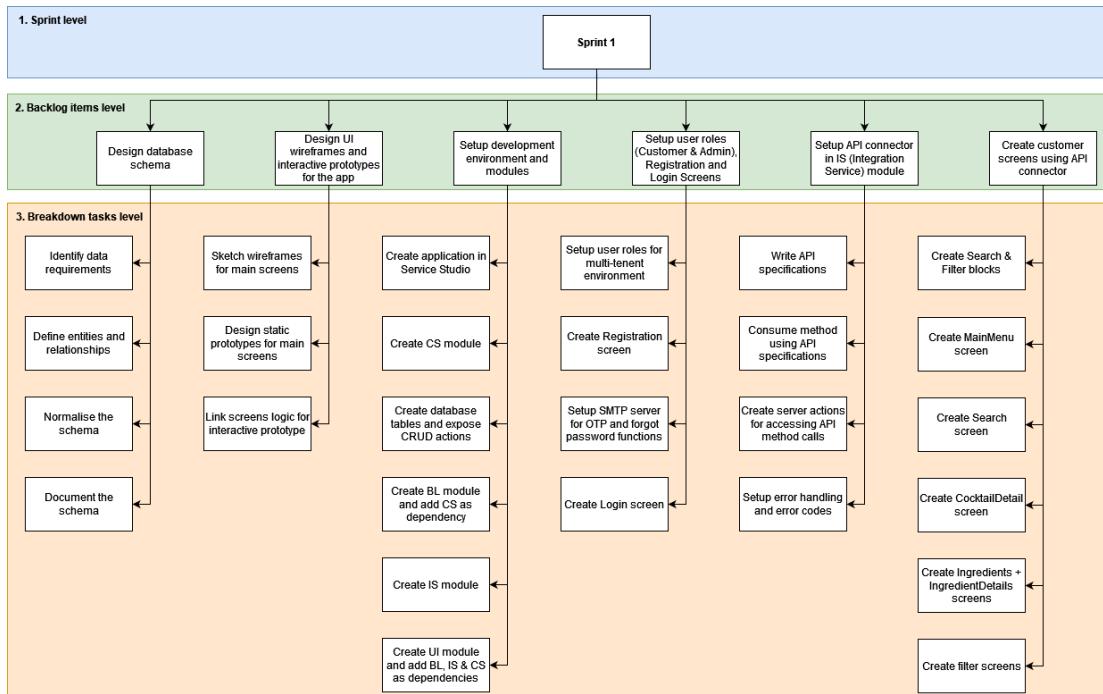


Figure 3: WBS tree structure

Notes:

1. Color annotation for Figure 3:
 - a. Cyan: Sprint level

- b. Green: Backlog items level
 - c. Orange: Breakdown tasks level
2. For planning, I will assume that the project development has 6 SWE30010 students, including my team members from Group 1 and me:
- a. Trac Duc Anh Luong (Self, PM)
 - b. Minh Nghia Nguyen (Scrum member)
 - c. Anh Duc Nguyen (Scrum member)
 - d. Gia Minh Nguyen (Scrum member)
 - e. Cong Anh Nguyen (Scrum member)
 - f. Tran Dat Dinh (Scrum member)

No.	Backlog item	Task	Assignee	Estimate
1	Design database schema	Identify data requirements	Trac Duc Anh Luong	2h
		Define entities and relationships	Minh Nghia Nguyen	2h
		Normalise the schema	Anh Duc Nguyen	2h
		Document the schema (Entity Design)	Gia Minh Nguyen	4h
2	Design UI wireframes and interactive prototypes for the app	Sketch wireframes for main screens	Cong Anh Nguyen	3h
		Design static prototypes for main screens	Tran Dat Dinh	6h
		Link screens logic for interactive prototype	Trac Duc Anh Luong	4h
3	Setup development environment and modules	Create the application in Service Studio	Minh Nghia Nguyen	1h
		Create CS module	Anh Duc Nguyen	1h
		Create database tables and expose CRUD actions	Gia Minh Nguyen	2h
		Create BL module and add CS as a dependency	Cong Anh Nguyen	1h
		Create IS module	Tran Dat Dinh	1h

		Create UI module and add BL, IS & CS as dependencies	Trac Duc Anh Luong	2h
4	Setup user roles (Customer & Admin), Registration and Login Screens	Setup user roles for multi-tenant environment	Minh Nghia Nguyen	2h
		Create Registration screen	Anh Duc Nguyen	6h
		Setup SMTP server for OTP and forgot password functions	Gia Minh Nguyen	3h
		Create Login screen	Cong Anh Nguyen	8h
5	Setup API connector in IS (Integration Service) module	Write API specifications	Tran Dat Dinh	4h
		Consume method using API specifications	Trac Duc Anh Luong	4h
		Create server actions for accessing API method calls	Minh Nghia Nguyen	3h
		Setup error handling and error codes	Anh Duc Nguyen	4h
6	Create customer screens using API connector	Create Search & Filter blocks	Gia Minh Nguyen	5h
		Create MainMenu screen	Cong Anh Nguyen	3h
		Create CocktailDetail screen	Tran Dat Dinh	4h
		Create Ingredients & IngredientDetails screens	Trac Duc Anh Luong	6h
		Create Search screen	Minh Nghia Nguyen	8h
		Create filter screens	Anh Duc Nguyen	4h

Figure 4: WBS time/effort estimation

The task dependency is described in the table figure above. Some items can be done asynchronously; however, the logical sequence of tasks must be followed. The total estimated time for the project is 96 hours, matching the task requirement as follows:

- **6** SWE30010 students
- **8** hours of work per student

- 2-week sprint

→ **Calculation:**

Total estimated time = [No. of students] x [hours of work per student] x [number of weeks per sprint]

$$= 6 \times 8 \times 2 = 96 \text{ (hours)}$$

Definition of done

The Speakeasy-OMS project will follow the ISO/IEC 25010 quality standard to clearly define the DoDs.

No.	Characteristic	Sub-characteristic	Implementation
1	Functional sustainability	Functional completeness	Every backlog item and breakdown task in Sprint 1 is completed and tested by pre-defined test cases.
2		Functional correctness	No logical error after test cases are passed. Software degradation is closely monitored during the maintenance phase.
3		Functional appropriateness	Conducted usability tests must produce at least 4 out of 5 regarding user satisfaction.
4	Performance efficiency	Time behaviour	The average load time of the website should be under 2 seconds.
5		Resource utilisation	The server's RAM and CPU usage should not exceed 80% during heavy load.
6		Capacity	The system should be able to handle heavy traffic with 1000+ monthly users and handle concurrency effectively.
7	Compatibility	Co-existence	As OutSystems is a multi-tenant environment, effective access control is a must to prevent users from accessing unauthorised resources.
8		Interoperability	The admin panel will analyse all data generated by the system's order and customer's preferences.
9	Interaction capability	Appropriateness recognisability	Customer feedback on the experience of ordering in the bar should be more straightforward compared to when they had to wait in queues previously.
10		Learnability	The UI should have straightforward navigation and instructions made easy for first-time users.

11		Operability	The admin should be able to analyse the data with ease.
12		User error protection	All client and server actions must have error handling and rollback transactions if necessary.
13		User engagement	The UI and UX must be coherent and easy to use for the user.
14		Inclusivity	Customers can use the product from anywhere with localisation and drinks customised to their preferences.
15		User assistance	Guides and feedback hubs should be available for customers.
16		Self-descriptiveness	The usage and interactions of the web page should be obvious enough for any user.
17	Reliability	Faultlessness	99% of all edge cases covered, with 0 edge cases for payment transactions to be allowed.
18		Availability	The application has minimal downtime and is accessible to customers and admins most of the time.
19		Fault tolerance	The application can handle minor errors gracefully without complete system failure.
20		Recoverability	Implement data backup and recovery procedures to restore lost or corrupted information.
21	Security	Confidentiality	User data (account details, order history) is stored securely with appropriate access controls.
22		Integrity	Implement data validation techniques to prevent accidental or malicious data corruption.
23		Non-repudiation	Implement logging mechanisms to record user activity (e.g., order placement, admin actions).
24		Accountability	Logs record user activity with timestamps and identifying information.
25		Authenticity	Implement secure authentication mechanisms for user logins (e.g., strong passwords, two-factor authentication).

26		Resistance	Implement security measures to prevent common attacks like SQL injection, cross-site scripting (XSS).
27	Maintainability	Modularity	The application is designed with well-defined, independent modules with clear interfaces.
28		Reusability	Consider opportunities to create reusable components within the application (blocks).
29		Analysability	Use clear naming conventions, comments, and architecture diagrams for better code readability. AI Mentor should not report any architecture or convention issues.
30		Modifiability	The application's design and code should be adaptable to accommodate future changes and growth.
31		Testability	The application is designed to be easily testable at different levels (unit, integration, system).
32	Flexibility	Adaptability	Document any assumptions or dependencies regarding hardware, software, or operating systems.
33		Scalability	The application should be able to handle an increase in users and order volume without significant performance degradation.
34		Installability	As this is a web application, any device operating system should be able to access it as long as an internet connection is established.
35		Replaceability	The focus of this project is likely to be building a new system, not replacing an existing one.
36	Safety	Operational constraint	Implement mechanisms to handle unexpected user inputs or errors gracefully.
37		Risk identification	Identify potential threats like unauthorised access, SQL injection, or data leaks.
38		Fail-safe	This involves features like automatic data backups, error logging, and restarting specific services in case of issues.
39		Hazard warning	The application should provide clear error messages and notifications to users in case of

			unexpected problems (no internet, new version updated, etc.).
40		Safe integration	Validate the security practices of any third-party components used.

Figure 5: DoD based on ISO/IEC 25010 characteristics

This DoD ensures the Speakeasy-OMS meets quality standards based on ISO25010 characteristics. Each backlog item should be considered "Done" only when it fulfils all the criteria mentioned above for its specific functionalities.

[82HD - Software Development Methodology Comparison](#)

(Next page)

Scrum Triumphs Over Waterfall: A Case Study of Agile Success in the “Let’s Shop” Software Project

Trac Duc Anh Luong

*Tutorial Session: Friday 10:00 am
Swinburne University of Technology
Hanoi, Vietnam
103488117@student.swin.edu.au*

Abstract— This essay compares a traditional software process model to an agile methodology, particularly Water and Scrum, through the insights of the “Let’s Shop” project, an e-commerce web application. The analysis seeks to comprehend the practical implications of choosing between the two given software development methodologies. Ultimately, the development team and stakeholders agreed that agile approaches are more effective in projects and add value quickly than traditional waterfall techniques.

Index terms—Project management, Agile, Scrum, Waterfall

I. INTRODUCTION

“Project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements” [1]. The rapid development of technology and stakeholder expectations require flexible software development approaches for their projects. The decision between traditional methodologies and agile frameworks plays a pivotal role that can profoundly influence the project outcomes. Through thoroughly examining Let’s Shop’s context, problem statements, and outcomes, the aim is to assess suitability for projects with similar scope and scale. The study analyses how each method interacted with the project’s dynamic requirements, adaptability, stakeholder involvement, development pace, and delivery.

II. BACKGROUND

There is an increasing need for a localised online platform that serves the electronics sector in Hanoi, the capital city recognized for its lively commerce and crowded markets. Let’s Shop is an e-commerce website dedicated to this demand. The primary objective of this project is to develop a user-friendly interface for the online store, expanding the reach of the existing physical store in Hanoi.

Specifications include Admin and Customer user roles, user experience, product placement, secure payment, order delivery, data analytics, ratings, and reviews. Managing inventory, international shipping, and the mobile version of the app are out of the scope of the Let’s Shop project. The development team is a small group of six members, including the product owner, data engineer, frontend engineer, backend engineer, scrum master, and tester. Throughout the software development lifecycle of the project, each phase was documented with evidence and data that will be analysed in this essay.

III. DISCUSSION

A. Definitions and characteristics

Scrum is an agile project management approach emphasising short, targeted sprints (usually 2-4 weeks) that facilitate rapid delivery and iterative improvement. Teams comprise developers who produce the product, a scrum master who facilitates processes, and a product owner who prioritizes features. While sprint planning and retrospectives direct development and adaptation, daily stand-up sessions guarantee openness. Because Scrum values adaptability, prompt feedback, and transparent communication, it is perfect for intricate projects with changing specifications. Its high pace and strict organization can be demanding, too, and to prevent burnout with polled workloads and fluctuating productivity, a committed team and constant adaptation are needed [2].

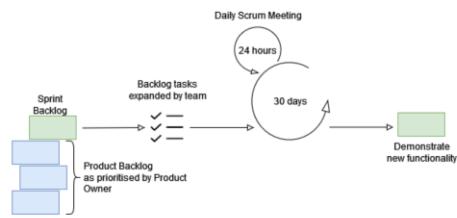


Figure 1: Scrum methodology

The Waterfall technique is a traditional approach to project management that consists of a series of steps that must be finished to proceed to the next phase, such as requirement collecting, design, development, and testing. This structure is appropriate for clearly defined projects with steady requirements since it provides unambiguous planning and control. Its rigidity, however, may not be able to adapt to changes or unforeseen problems, which could cause delays and expensive rework. In addition, incomplete phase integration and early feedback might lead to final products that are not aligned. Waterfall offers predictability, but projects that need constant modification or are dynamic may not be the best fit for their rigidity [3].

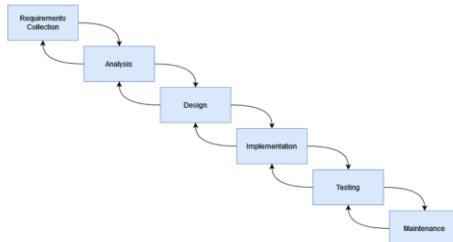


Figure 2: Waterfall model

B. Case study

At the forming stage, the “Let’s Shop” project adopted the Scrum methodology as the primary software development lifecycle by our stakeholders’ decision. We had 7 sprints, each taking up a 2-week interval or 10 working days. As the scale of the project is relatively small, we closely work with our stakeholders through weekly meetings for project refinement and quality approval. As the project requirements changed due to new information or expectations from the stakeholders within the reasonable scope, we needed a project management approach that allowed flexibility and adjustments. There were 6 members in the development team with minimal prior knowledge of software development lifecycles. Therefore, continuous learning, research, and open communication are instrumental to the project’s successful outcome.

C. Methodology

The researcher analysed the initial project requirements of the stakeholders and collected the primary data for this essay. Data acquiring means include surveys, interviews, testing, and team documentation during the development phase and reflection. The pros of collecting these primary data are that the researcher has full authority of data analysing techniques, charts, and graphs design to give fellow readers and the development team new insights

on how the choice of agile and scrum over a traditional model like waterfall was correct. Data-based research also eliminates any matter of personal biases. The cons of this method is that data acquisition can be time-consuming. Therefore, the source is not always real-time, especially when the case study is complicated, and it is impossible to acquire preliminary data.

D. Requirement analysis and survey results

1) Requirement analysis:

Scrum was chosen for the “Let’s Shop” project after carefully analysing how well it fits the project’s particular characteristics, which included a small team size, changing needs, and a critical focus on features. There were several strong arguments supporting this choice. First and foremost, Scrum became the preferred method because of its built-in flexibility and iterative process, which contrasted sharply with the Waterfall model’s rigidity. Because of its adaptability, the team responded quickly to changing stakeholder expectations and incorporated innovative ideas into each sprint, creating a culture of responsiveness and continuous improvement [4].

Furthermore, the Scrum framework’s frequent feedback loops enabled quick improvement and modification. In addition to the fast development pace, these iterative cycles of review and adaption ensured that the project stayed precisely aligned with changing stakeholder priorities and objectives. Moreover, implementing brief, targeted sprints was a tactical facilitator for prompt stakeholder involvement and value delivery. Scrum promoted stakeholder satisfaction by quickly delivering tangible functionalities and allowing for necessary error corrections by breaking the project into manageable pieces, handling each piece at a time. Stakeholder involvement in waterfall projects usually occurs early, during requirements collection and project planning. After these phases are finished, stakeholders frequently don’t have many chances to provide valuable feedback until the final deliverables are created [5].

Finally, Scrum’s focus on team empowerment and self-organization was crucial in helping the project team develop a drive, responsibility, and ownership culture. This culture of cooperation encouraged team members to take initiative and work toward common goals, increasing the efficiency and effectiveness of the project. Scrum’s execution in the “Let’s Shop” project is a convincing example of how well it works for similar projects with small, cross-functional teams, changing needs, a need for quick feedback, and a laser-like focus on essential features. Scrum’s ability to adapt, its iterative development process, and its firm dedication to continuous learning come together in these situations to produce a climate conducive to long-term success and innovation [6].

2) Survey results:

The researcher has conducted surveys in which the respondents are the development team members. This section will provide the survey results and key takeaways.

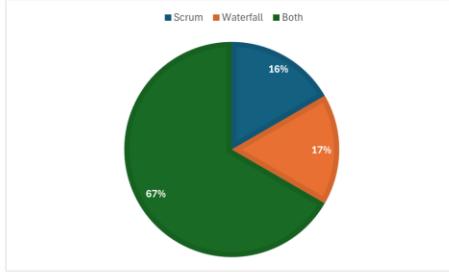


Figure 3: Knowledge of methodologies

In Figure 3, we can see that upon investigating the team's capabilities, we found that despite limited prior experience, they had taken the time to learn about project management techniques. Interestingly, a sizable percentage of the team (67%) showed knowledge of both Waterfall and Scrum processes. This result suggests that the group is flexible enough to be able to work through a variety of project requirements. Having Scrum knowledge, in particular, demonstrates their ability to adapt quickly to changes by taking advantage of the methodology's built-in flexibility.

Furthermore, the team's extent of knowledge in both approaches offers excellent chances for cooperation and knowledge sharing. This diversity promotes a dynamic learning environment where participants may effectively confront obstacles by building on each other's strengths. It also allows for the cooperation of ideas and best practices. Overall, the team's expertise in project management approaches highlights their potential for ongoing innovation and improvement through collaborative learning and their capacity to manage various project types.

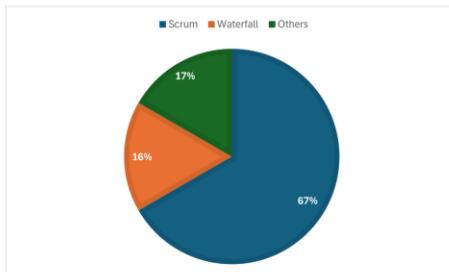


Figure 4: Implementation strategies

In Figure 4, the provided data reveals a clear preference within our team for Scrum as the project management methodology, with 2/3 of the team advocating for it, compared to 16% for Waterfall and 17% for others. The overwhelming support for Scrum suggests a belief in its suitability for the project. This might be due to several factors. Scrum's iterative nature aligns well with projects of unknown or evolving scope, allowing flexibility and adaptation. Additionally, its focus on collaboration and self-organization could appeal to a team of diverse skill sets and preferences.

The significantly smaller 16% supporting Waterfall indicates a less favourable view of its rigid, linear approach. This could be because the project demands a more dynamic and responsive methodology, something Scrum readily offers. The 17% advocating for "others" highlights that not everyone is entirely convinced by either Scrum or Waterfall. Exploring the methodologies and reasons behind their preference could reveal valuable insights into potential concerns or areas Waterfall addresses.

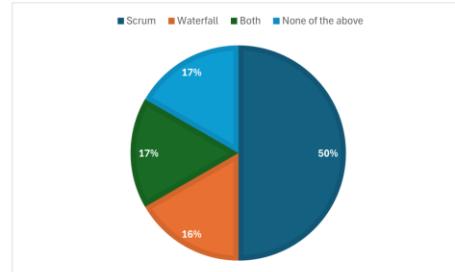


Figure 5: Meeting business requirements and customer satisfaction

In Figure 5, we can draw insights that when asked to vote for the project management framework that is most likely to meet the business requirements of the Let's Shop project and fulfil customer satisfaction, 50% of respondents advocated for Scrum and 17% advocated for both Scrum and Waterfall. Waterfall demonstrated that the team believed it could meet customer and business needs. Because Scrum is an iterative process, it facilitates constant feedback and adaptation, allowing the product to change in response to customer and market demands. This aligns with meeting business requirements and ensuring customer satisfaction through data-driven decisions and sprint-by-sprint course changes.

Although 17% of participants preferred a blend of Scrum and Waterfall, demonstrating an appreciation of the advantages of both approaches, Waterfall received less support (16%), which may indicate limitations in meeting

the competing goals of meeting business requirements and satisfying customers. The rigorous, sequential process of Waterfall may make it more difficult for the team to adapt quickly to changes in the market and client preferences. Its lack of adaptability may make adjusting to changing conditions difficult, which could jeopardize the project's success.

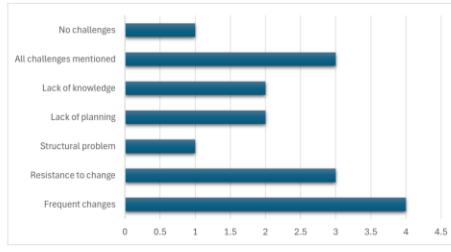


Figure 6: Challenges faced

In Figure 6, the respondents were asked to identify their challenges when approaching software project management with Scrum as their first framework. While applying Scrum, the most significant challenge was the frequent changes. The second biggest challenge was resistance to change, which also falls under the same category as the first. This fundamental knowledge shows that the team's adaptation to changes is the most important for the initial exposure of an SDLC model or migration from a traditional approach like Waterfall to Agile and Scrum.

Furthermore, since the research's primary objective required comparing the two project techniques, 67% of respondents believed the Scrum methodology was more effective at completing projects than the waterfall methodology. 84% of the respondents said the Scrum method contributed to real project improvements. 84% of them stated that they plan to keep adopting the Scrum approach for project management after considering its advantages. Additionally, of the pool of participants in this study, over 67% said they would advocate for mid-size and big businesses to adopt Scrum and Agile.

According to the "Let's Shop" project analysis, Scrum was the ideal choice because it fits its core values and has a lot of team support. Its adaptability, iterative process, and emphasis on stakeholder involvement helped it to produce value, adapt to changes, and promote a collaborative environment. Although there were difficulties, such as adjusting to frequent changes, the team's expertise and commitment indicate they can overcome them. Additional investigation into the preferences of "others" and ongoing education can refine the strategy for even higher success.

IV. CONCLUSION

The "Let's Shop" project analysis provides compelling evidence for selecting Scrum as the project management approach. This choice is in line with the particular requirements of the project, which include its small, flexible team, changing requirements, and emphasis on essential features. Such projects benefit from Scrum's natural flexibility, iterative development, and focus on continual learning.

The survey results further support this decision. Despite their lack of prior expertise, the team's solid understanding of both Scrum and Waterfall highlights their adaptability and potential for collaboration. Furthermore, the team's resounding endorsement of Scrum indicates confidence in its applicability.

Nonetheless, the difficulties encountered throughout implementation, especially adjusting to frequent changes, highlight the significance of ongoing education and team support. Examining the "others" group and the justification for their choice may yield insightful information for future development.

The "Let's Shop" project is an excellent example of how Scrum works well for projects with comparable needs. The team's ability to learn and adapt continuously and having the appropriate support network will be essential to their success in the future. The information acquired gives them important insights for maximizing their strategy and guaranteeing the project's long-term success as they traverse the opportunities and difficulties that lie ahead.

REFERENCES

- [1] P. M. Institute, *A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition and The Standard for Project Management*, 7th ed. Project Management Institute, 2021.
- [2] D. Cricic, B. Lalic, D. Gracanin, N. Tasic, M. Delic, and N. Medić, "Agile vs. Traditional Approach in Project Management: Strategies, Challenges and Reasons to Introduce Agile", *Procedia Manufacturing*, vol. 39, pp. 1407–1414, 2019, doi: 10.1016/j.promfg.2020.01.314.
- [3] S. Khan and S. Mahadik, "A Comparative Study of Agile and Waterfall Software Development Methodologies", pp. 399–402, 2022, doi: 10.48175/IJARSCT-5696.
- [4] L. Morris, M. Ma, and P. C. Wu, *Agile innovation a revolutionary approach to accelerate success, inspire engagement, and ignite creativity*. Hoboken, NJ Wiley, 2014.
- [5] K. Chari and M. Agrawal, "Impact of incorrect and new requirements on waterfall software project outcomes", *Empirical Software Engineering*, vol. 23, no. 1, pp. 165–185, Feb. 2018, doi: 10.1007/s10664-017-9506-4.
- [6] A. Alami and O. Krancher, "How Scrum adds value to achieving software quality?", *Empirical Software Engineering*, vol. 27, 2022, doi: 10.1007/s10664-022-10208-4.

Coverage of the Intended Learning Outcomes

ILO 1 – Understand and apply project management approaches

Demonstrate an appreciation of the significance of systematic approaches to project management in achieving successful IT projects, including software development and the pertinent agile approaches.

The following pieces of work have demonstrated my ability to cover this ILO:

- 01P – Planned the project with scrum backlog items and release schedule based on the defined background and scope of the project.

- 02P – Reviewed the project planning with teammates to find the optimal solution and contribute responses to the task.
- 03P – Decided the tech stack for the project and designed the project's architecture from a bird-eye perspective, illustrating the interactions between different tiers and layers.
- 04P – Reviewed the architecture design with teammates for optimal solutions, improved add-ons, and agreed upon the chosen tech stack.
- 05P – Defined quality assurance characteristics and definition of done for the project using the ISO 25010 model.
- 06P – Reviewed QA characteristics and metrics to finalise the final quality standards for the project.
- 07P – Compiled all previous work into a single project proposal and presented it to the stakeholders for approval.
- 08P – Created a WBS for accurate effort estimation and equal workload allocation among team members.
- 12P – Reviewed backlog items completed with the stakeholders and quality management strategies applied.
- 13P – Internally reviewed the Sprint within work members based on how backlog items met quality assurance standards.
- 81HD – Similar to 07P but worked as an independent project manager instead of a team member.
- 82HD – Deeply discussed the application of project management frameworks to a project I have worked on.

ILO 2 – Analyse risks and apply management strategies for risk management

Apply cognitive skills to analyse the risks to the completion of projects on time and within budget and identify appropriate management strategies for specific risks.

The following pieces of work have demonstrated my ability to cover this ILO:

- 03P, 04P – Alternative tech stacks and KoST analysis were considered to assess the team's capabilities for the project.
- 05P, 06P – Apply quality standards to reduce software malfunctions and degradation risks.
- 07P – Presented to the stakeholders how risks had been considered and management approaches.
- 61C – Different risks affecting the task execution and time allocation were considered for the task to be marked as done.
- 62C – Comparison of estimation and actual effort for the selected backlog item.
- 71D – Defined Quality Metrics so that software risks will be minimized.
- 72D – Compiled testing checklists for different metrics defined in 71D.
- 73D – Reviewed the quality characteristics according to the checklist after testing.
- 82HD – Discussed how potential risks can be mitigated or exposed based on different project management frameworks.

ILO 3 – Discuss human factors and human management

Discuss human factors in IT projects, team building and management.

The following pieces of work have demonstrated my ability to cover this ILO:

- 12P – Reviewed Sprint 1 with the stakeholders and discussed the team dynamics and improvement points.
- 13P – Reviewed Sprint 1 internally within the team, discussed the group velocity in the ten working days period and future improvement considerations.
- 15P – Peer-reviewed based on pre-defined criteria and metrics.
- 82HD – Discussed how the team's understanding and adaptability to different project management frameworks can impact the final outputs and project successes.

ILO 4 – Apply project management tools

Discuss project management tool options and use select tools effectively for project management.

The following pieces of work have demonstrated my ability to cover this ILO:

- 09P – Proposed using ClickUp for task board, work allocation, progress checking, and evidence storage.
- 10P, 11P – Implemented ClickUp functionalities and Excel spreadsheet to record the burndown progress chart during the two working weeks for Sprint 1.

ILO 5 – Communication and leadership

Communicate effectively as a professional and function as an effective leader or member of a diverse tea.

The following pieces of work have demonstrated my ability to cover this ILO:

- 02P, 04P, 06P, 07P, 08P, 09P, 10P, 11P, 12P, 13P, 14P

As appointed as the leader of Group 1 – The Beavers since week 2, I have played this role effectively by communicating with my teammates directly in class and indirectly through group chat messages, ClickUp task board and ClickUp comments. The group always met deadlines, submitted early for recommendations from our tutor, and received good feedback for all the group tasks mentioned above. The group also have good relationships and respect for each other and the leader.

Reflection

The most important things I learnt:

Throughout this course, I have had the opportunity to explore various topics related to project management. Here are key learning points that I have concluded:

Project Planning & Software Design: The course highly emphasised the importance of implementing a systematic approach to project planning and how incorporating project frameworks can assist the task. I have also learned to design software architecture using UML diagrams, illustrating how different components can be abstract and their interaction with other parts of the system. This allowed me to carry out the planning steps in future projects effectively.

ClickUp Experience: Gaining hands-on experience with ClickUp provided a practical understanding of project management tools. I can now navigate similar platforms to manage tasks efficiently, collaborate with teams, track progress, and evaluate the velocity using generated graphs and charts.

Risk Assessment & Mitigation: The focus on risk management was invaluable. I learned to identify potential risks that could impact schedules and progress relating to software requirements, technical depth of team members, release timeline, and how to develop strategies to mitigate them. This proactive approach will be essential in ensuring project success.

Effort vs. Estimation: Evaluating task estimations against actual effort was eye-opening. I understand the importance of accurate estimation to ensure realistic project timelines and human resource allocation.

Human Factors Analysis: The exploration of human factors and their impact on project success was impactful. I recognise the importance of team building and clear communication both verbally in tutorial classes and through email and text messages. It is also crucial to support other team members and ask for help and feedback on your work for cross-checking.

Enhanced Leadership Skills: This unit provided opportunities to develop leadership skills through real-world project scenarios. I was allowed to lead my group for the group tasks and development phases. With the experience of interacting with each team member and learning their strength, I am better equipped to lead different teams, delegate tasks effectively, and navigate challenges that arise during project execution.

These points have exceeded my expectations of what I would learn from the course.

The things that helped me most were:

Practical application with real projects: Throughout the course, my team and I have participated in all activities related to software project development, from defining scope, planning backlog items, breaking down tasks, coding, and quality control testing. These phases have helped me enhance my knowledge of project development.

Feedback and improvement: Feedback is essential internally in the team among members and from our tutor. The team implemented a strategy to complete tasks early and ask for feedback in class so that we can improve our work promptly before submission deadlines.

Comprehensive learning schedule: The scheduling of the unit was logical and comprehensive, as we learned the fundamentals of project management with corresponding assignments to implement theory and cross-check our knowledge.

Portfolio development: The tasks in the portfolio were logical, especially the C and above tasks, as they built the knowledge and steps on top of each other. For example, the D tasks first asked us to define quality assurance metrics, then planning, execution, and review in a sequence.

Tutor availability: As I have mentioned in the feedback and improvement point, our tutor, Dr Dung, was knowledgeable and supportive, as she cleared our confusion in tasks that were somewhat vague in the requirement description and gave us direct feedback so that we could work on improving for better outputs in the subsequent tasks.

I found the following topics particularly challenging:

With my hands-on experience working as a software developer, many concepts and definitions were already covered in my work before I enrolled in the unit. Therefore, I had little challenges while studying the unit, and the learning path was mainly enjoyable while expanding on my current understanding.

However, the concept of quality assurance still needed to be improved, as in the project planning phase, we had to base our metrics on the ISO 25010 model. Many considerations need to be made regarding success rate or performance thresholds.

I found the following topics particularly interesting:

Software design: I was excited about the content of our second week: software and architecture design. I have learned a lot about abstractions of software components and how they can be represented using existing diagramming tools.

Sprint planning and estimating: Using the work breakdown structure (WBS), I effectively estimated the time for task completion, comparing them with the actual effort spent for future planning accuracy.

Agile and Waterfall comparison: Task 82HD gave me an interesting topic to work on: comparing the effectiveness of 2 popular software management frameworks in a project I have worked on. By doing surveys with team members and analysing the project results, I got

a deeper understanding of how implementing these methodologies can affect the outcome of a software project.

I feel I learnt these topics, concepts, and/or tools really well:

Work breakdown structure: Analysing and breaking down tasks in our proposed project has helped me understand how a sizable backlog item can be broken down into sub-tasks and bottom-tasks for ease of management and workload allocation based on each member's strength and expertise.

Analysing burndown charts: Throughout Sprint 1, we documented our progress using the task board and burndown chart. The team's velocity was clearly shown with the correlation between estimated burndown and actual remaining effort, as well as how the team underestimated or overestimated our abilities to adjust our estimation in future sprints and projects.

Compiling quality assurance checklist: The quality assurance review activity was beneficial for identifying software defects, mitigating future degradation, and how a business requirement is broken down into more minor test cases and exceptional edge cases. After carefully executing and collecting evidence for each test case, I have gained experience in quality assurance and control.

I still need to work on the following areas:

ISO Standards: Even though the team and I have fulfilled the quality requirements listed in the ISO25010 quality model. There are other quality standards that I still need to explore, and I want to do further research to enhance my knowledge and incorporate it into future projects.

Backlog item selection: Upon completion of Sprint 1, my team completed all backlog items and fulfilled the quality standards. However, as the leader, I believe the backlog item should be broken further down in the planning phase. The evidence was that we completed three backlog items in the first week but needed to spend the whole second week doing F4. The estimation can be more distributed among different items in my proposed list.

This unit will help me in the future:

As I am working as a software developer, I have already had the chance to apply many theories to work regarding software project planning and execution. In the future, once I take on the role of a project manager or product owner, these experiences will be invaluable as I will have detailed and broad perspectives of the project I will be working on. Fundamental principles regarding backlog planning, software design, quality control, and human interaction are valid and directly applicable to my work in the software field.

If I did this unit again I would do the following things differently:

The unit has offered a great learning experience for me. However, if given the chance, I would do a few things differently for improved results.

Early preparation and research regarding challenging topics: The unit proposed many challenging topics (ex., ISO standards).

Further integration of software testing tools and frameworks: My testing experience for quality assurance mainly relates to unit tests and manual testing. I want to explore other tools that can automate the testing process and performance testing using stress and heavy loads.

Utilisation of office hours from the lecturer: To achieve this, I would like to read the material before class and prepare question lists to clear any assumptions or unclear points. This will be a proactive learning approach that I can apply to this and future units.

Extending beyond the material presented in the unit:

Lesson Learned from HD Task: The unit provided valuable exposure to both Agile and Waterfall methodologies. Doing the 82HD research task particularly highlighted the benefits of Agile in terms of producing functional outputs more quickly and gathering feedback from

stakeholders throughout the development process. This approach stands in contrast to Waterfall, where the customer might have to wait until the later stages of development to see a working product. Understanding the strengths and weaknesses of both methodologies equips me to advocate for the most suitable approach based on project requirements and stakeholder needs.

Detailed Design of the software project: For the software design task (14P), I have guided the team for creating industry standard documentations, using the Detail Design structure that industry companies usually used, with clear guidelines and evidence below. I have also created templates and structures for the team to follow after they finished developing their screens/functions.

The screenshot shows a ClickUp workspace titled 'The Beavers'. On the left, there's a sidebar with navigation links like Home, Inbox, Docs, Dashboards, Clips, Timesheets, More, Favorites, Spaces, and a Space named 'Space'. The main area shows a checklist under 'Required Documents' with the following items:

- 1. Entity Design
 - a. ERD already exist
 - b. Explanation of each table attribute name, data types, length, is mandatory, delete rules, descriptions
- 2. README
 - a. Overview of what the project does
 - b. Links to important documents
 - c. Flow between screens (link, data)
 - d. Overview of what the screen does
 - e. Rate the complexity (give reasons) - easy, medium, difficult
- 3. Detail Design
 - a. Introducing an overview of the screen
 - b. Screenshots of the screen
 - c. Circle and explain each item
 - i. Data types
 - ii. length
 - iii. Get data from which table, which field
 - d. Processing details
 - i. The OnClick events
 - ii. Validation
 - iii. Screen transition

Guiding teammates to using project management tool:

As powerful as ClickUp was, it was a relatively new software management tool that not many people have come across. I have guided the team how to create tasks, checklists, time estimates, and the guideline to using the tool for the most accurate summary report (ex. Only track your time in sub-tasks instead of parent tasks). After the training session, all team members were comfortable using the tool and reported that ClickUp has aided them well throughout Sprint 1.

Thorough test cases and effective execution of quality assurance process: All features and screens were thoroughly tested with test cases and evidence provided in the ClickUp tasks. The detailed test cases went beyond the general description of quality assurance criteria that had been defined in previous assignments.

Record of change and improvement in Portfolio

Task	Task Name	Change
15P	Peer Review	After thorough considerations, I have updated the peer review task, where team members are assessed in a fairer way, with each having clear notes and reasons of why they were given the review score. Overall, the team performance was good with future improvements to consider.