ASSIGNMENT COVER PAGE

Programme	Course Code and Title
UCSEW	CSE3033/N SOFTWARE ENGINEERING

Student's name / student's id	Lecturer's name
CHAN SEOW FEN / 0207368	TEE ENG HONG

Date issued	Submission Deadline	Indicative Weighting
22/1/2024	19/2/2024	30%

Assignment [1] title	Agile Project Case Study and Analysis

This assessment assesses the following course learning outcomes

# as in Course Guide	UOWM KDU Penang University College Learning Outcome
CLO2	Analyse the empirical nature of software engineering and the application of empirical methods in software engineering development
CLO3	Evaluate advanced software engineering techniques and processes in the development of a software artefact.

# as in Course Guide	University of Lincoln Learning Outcome
LO2	Analyse the empirical nature of software engineering and the application of empirical methods in software engineering development
LO4	Critique current software engineering processes in safety critical system

Student's declaration

I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.

Student's signature:

Submission Date: 19/2 /24

Assignment 1

1%	1%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mei Hwei Wu. " Maximizing the number of spanning trees on the (+2) graphs ", International Journal of Computer Mathematics, 1990 Publication	<1%
2	Submitted to University of Hertfordshire Student Paper	<1%
3	freezingblue.com Internet Source	<1%
4	dokumen.pub Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	Off		

Table of Contents

1.0 Software Development Methodology

1.1 Justification on Software Development Methodology Selection

Based on the requirements of developing such new software for the educational institution, which is an upgraded version to customer's existing system that consists of the features of managing student registration, course result, payment as well as tracking their study progress, reflecting their confidence on stating their demands accurately, it is strongly believed that the most suitable software development methodology is Agile software development methodology, which is Scrum, in specific.

For a brief understanding, Agile software development methodology is simply an approach of managing project that breaks down the big project into phases and focuses on collaborating and improving continuously. The cycle of making plan, executing the plan and evaluating the plan and outcome is followed by the team as aligning to the core of Agile methodology (Atlassian, 2024). In short, Agile software development methodology were presented for minimising issues confronted when utilising the traditional software development methodology such as waterfall development and parallel development. There are some development approaches under the category of Agile software development methodology, which included Extreme Programming (XP), Kanban and Scrum (Dipendra Ghimire and Charters, 2022), which is the methodology been selected on developing this educational institution managing software.

To justify the reason of selecting Agile methodology, specifically Scrum as the software development methodology in this case is due to its iterative approach. To Illustrate that, Scrum provide an iterative development process that its process of designing, developing and testing of software in a repeated cycle, or time-bound iterations known as sprints, as shown in *Figure 1.1*. There will be a potential shippable increment and improvement in the software after each sprint. This is due to the fact that feedback and insights from customer and stakeholders is gathered and take it as the consideration of revising in the next cycle, and thus such development process offers continuous improvement due to the iteration process allows the development team to have inspection and adaptation to their software which make it better and fulfils the customer's demands (Salimi, 2024). In short, Scrum is selected due to its iterative process of developing a software, which allow continuous improvement of the software that meets the requirement of developing an educational management system.

Furthermore, even though the customer requirements are clear enough and one of the highlights of Scrum is about its ability to handle vague requirements, however, it is still useful in this case as there is potential changes that customers might made to refine their requirements as this project involves an upgrade on the existing system. Hence, this is where Scrum methodology come in need due to its outstanding characteristics of iterative approach, flexibility and adaption in specific, it allows the development team flexibly to fulfil the changing requirements while maintaining effort on delivering values to the customer. To illustrate that, the changes on requirements that are uncovered are placed under a product backlog, which is a bunch of ordered list that list out the demands of customer in improving the product, that will be done in the next sprint (Scrum.org, 2024). In essence, one of the reasons for selecting Scrum as the development methodology is due to its characteristics of flexible and strong adaptability which address the issue of changing requirements that potentially happening on an education management system as there might be changes in their management.

Following that, as the requirements of customer involved the features on managing student's registration, course result, payment, as well as tracking the progress of study, it could be considered as a complex system. The reason is that database management is involved in the system, it rises the concern on data complexity, business logic complexity, functionalities integration, user-interface complexity, security, scalability and performance concerns, which greatly increase the complexity of the system. Hence, the introduce of Scrum

methodologies is to address the complexity issues of the requested educational institution management system. To illustrate that, Scrum development is considered good in handling complex project. This is due to its empirical process control. In short, the decision in development is made on continuously observation and experimentation, instead of having a detailed upfront planning. The core of empirical process control is based on three central ideas, which refers to transparency, inspection and adaptation (SCRUMstudy, 2024). Briefly, teams will be inspecting the progress and adapting the method continuously based on empirical evidence, and hence allowing the identification on the issue and addressing the complexity at the early stage, which is as soon as it is raised before it turns into a significant issue that is more difficult and time-consuming to address it (Scrum Alliance, 2024). In short, due to the high complexity of this educational institution management system, Scrum is selected as the software development methodology due to its empirical process control, which could effectively address the complexity by iterative feedback loop on empirical evidence.

In conclusion, based on the requirements of the customer of building an upgraded system to an existing educational institution management system, Scrum as one of the Agile software development methodology is selected due to its notable ability of continuously improving the product to meet customer's expectation by iterative approach, outstanding capability on addressing potential changing requirements due to its characteristics of flexible and highly adaptive, and lastly effectively addressing the high complexity of the requested system due to its empirical process control. Hence, it is the most suitable software development methodology that could address the need of customer after analysing the situation and demands of customer.
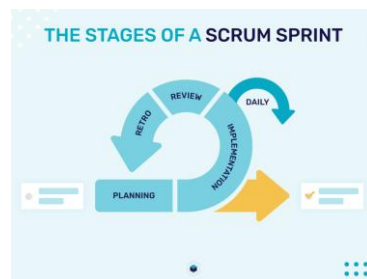

Figure 1.1 Illustration of Scrum Sprint (Czisch, 2022)

1.2 Activities Involved in the Design and the Software Development Process

The activities involved in the design and the development of the requested educational institution management system included 5 events, excluding the creation and organisation of product backlog, the preparation of sprint backlog, and product increment which three of them are referred as sprint artifacts. The main activities start with sprint planning, following by development sprints, daily stand-ups which is also referred as daily scrum, sprint review and lastly sprint retrospective (Scrum.org, 2022).

Before getting started on describing the 5 activities in the design and development process of the software, it is essential to introduce the three artifacts used in the proper implementation of the Scrum methodology as their preparation are included as the activities in the software development process but not considered as the main events. Firstly, the creation and organisation of the product backlog requires development teams to list down all the desired features of customer such as the payment and course result management as well as breaking down the whole task that needed to be completed with prioritising the customer demands and business values. Next, sprint backlog is used to focus the task for a single sprint which is the iteration cycle (Infinity, 2024). Lastly, product increment refers to the accomplished product backlog during a sprint, it can also be used to define the sum of every completed sprint backlog item and user stories (ScrumAlliance, 2024).

Next, moving forward to the main discussion, which is the main activities involved in the design and development process of the software. Firstly, it started with the sprint planning. This activity mainly focused on planning the task and scope of the work for this current sprint by selecting a subset of items from the product backlog during the meeting of the entire development team, hence it is the first activity and happens at the beginning for every sprint. After that, those tasks selected, and specific user stories will become the sprint backlog for that specific sprint. The meeting will be directed by the scrum master and the team responsible on selecting the task. Those user stories will always align with the objective of the software and also agreed by the scrum team in order to ensure the feasibility of implementing them during the sprint (Atlassian, 2019).

Following that, development sprint is the actual period of time when the scrum team collaborate and works together in accomplishing the increment, which is the tasks and features stated in the sprint backlog. Commonly, the sprint period is about 2 weeks, range between 1 week to a month according to the situation that if want to go for easy scoping or having adequate time for delivering valuable increment. It is generally advised to shorten the sprint period if the task is complex and consists of remarkable number of unknowns, which is suited in this case of developing the educational institution management system with high complexity. As for more details, the scope in the sprint can be renegotiated between the stakeholders and the development team during the sprint if there is necessary, which forms the core of the Scrum empirical nature (Atlassian, 2018).

In corresponding to that, daily scrum, also known as daily stand-ups as highlighting that should be a short and simple process, is a short meeting, which usually 15 minutes long as it is the guidelines, will be done in a daily manner and usually in the morning for development team members to share their progress, discuss the way to cope with obstacles and synchronise the progress of their works. Generally, each member will have to present their talks that consist of the answer of three question: What they did yesterday, what they are going to do today and if there are any difficulties (Atlassian, 2018). The reason of having daily scrum is mainly to keep track of everyone progress and ensure everyone is on the same track, aligned with the sprint objectives as well as planning out the goals for tomorrow (Scrum.org, 2024).

Next, sprint review is done at the end of the sprint. It is to have an informal session that gathers up the team members to have a view on a inspect or demo, which is the increment. The accomplished backlog items have to be presented by the development team to the stakeholders, which is the customer requesting the educational institution management system in this case and receive their feedback and insights. The stakeholders, or customer in this case will have to decide if this increment can pass or rejected after reviewing the demo. Based on current sprint, reworks on the product backlog will also be done by the customer to add in to the planning session of next sprint during the review meeting (Scrum.org, 2024).

Lastly, the sprint retrospective is where the meeting, or also known as a ceremony that only involving the development team takes place. It is mainly used to reflects on the outcome of that specific sprint, identifying the areas that can be improved, and most importantly adapts the processes accordingly. Having such meeting is necessary in enhancing the efficiency and the values delivered in the next sprint as it provides valuable insights as well as create better bonds within the development team members and increasing the proactivity of the teams by organising such ceremony (MacNeil, 2022).
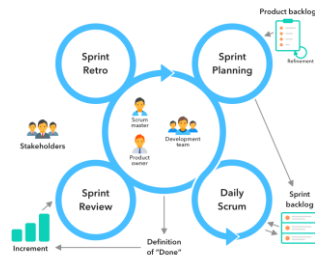
Figure 1.2 Diagram of Scrum Development Process Activities (Infinity, 2024)

1.3 Strengths and Weaknesses of Development Process

1.3.1 Strengths

One of the most well-known advantages of Scrum software development methodology is about its flexibility and adaptability as mentioned before. Due to its outstanding ability to adapt to customer's changing requirements or challenges that are unexpected during the development process, it reflects its flexibility and responsiveness that will bring to the development team. To illustrate further, as mentioned before, the features of managing the education institution affairs might be evolving and the demands will keep changing, Scrum flexibility approach allows development team to adapt and accommodate any changes in requirements seamlessly. Moreover, Scrum as an iterative approach could also address the changes of requirements quickly and effectively as customer is able to continuously provide feedback or discuss about the adjustments during the sprint review (Atlassian, 2024).

Moreover, the short delivery time of the product or software to the customer is also one of the highlights of Scrum development methodology. Generally, Scrum development often results in faster delivery of projects or features compared to other methodologies (Obi, 2019). The reason of such rapid development is achieved due to the nature of Scrum development, which is breaking down the huge project into a smaller part that is more manageable, then, the work is delivered in a short iteration where the longest period will not exceed one month. Hence, this resulted in the acceleration of software development process (Chandana, 2024). This is a huge advantage to this educational institution's management software as it serves as an upgraded version of the existing system. Thus, the customer, which is the education institution will be greatly benefited as they are able to provide the upgraded system sooner, which brings satisfaction to the administrator, staff as well as students.

Last but not least, Scrum methodology provides higher quality work due to its increased transparency. To illustrate that, team members are ensured to be productive and focused on their task by utilising Scrum methodology. The reason is that higher transparency is achieved among the stakeholders, which is the educational institution in this case, with development teams and scrum master. In corresponding to that, transparency is achieved by the visibility provided by Scrum methodology, that all changes are visible to everyone, regardless of the role, due to every small detail is presented to every member during the daily sprint. Hence, every member would have to show they are fully responsible for their work, which resulting in an environment with higher productivity, thus offering high quality project and software (Vasiliauskas, 2022). Additionally, trust and alignment among the team as well as stakeholders are fostered due to such transparency provided by sprint review. In essence, this ensures that the educational institution management software developed meets the expectations of the customer and enhances the overall user experience of the staff and students.

1.3.2 Weaknesses

In terms of Scrum development disadvantages, one of it obviously is about the dependency on customer availability. The reason is that as mentioned in the activities of design and developing software according to Scrum methodology, there is one of the main events known as sprint review, which requires the attendance of all development team

members and the customer to showcase the increment of the specific sprint (Zoho Corporation, 2024). Which means that, it requires the customer to attend that long meeting for every sprint. As mentioned before, the sprint period ranges from 1 week to 1 month. If the customer has to attend the meeting for every week, the customer availability can be a challenge. Particularly when their availability is limited to other responsibilities, such as managing the school affairs in this case. In order to ensure the success of the project, it is crucial for development team to have effective communication to manage customer expectations, which highly requires the involvement of the customer, as they have to discuss about their required features, offering feedback on each sprint increment, and validating the deliverables, hence, it will affect the project progress and efficiency if there is any constraint on customer availability.

Furthermore, another drawback of Scrum development is about it does not necessarily help in meet overall deadline. To illustrate that, Scrum does not guarantee the project will meet the overall deadline even though it facilitates in breaking large projects into smaller and manageable parts, leading to the acceleration of implementation (Vasiliauskas, 2022). The reason is that as for example in this case, after breaking the big project of educational institution management system, there will be multiple smaller parts being set for the delivers of incremental upgrades, however, it can be challenging for to manage the deadlines for larger part without diligent oversight.

Lastly, Scrum methodology lack of comprehensive documentation due to its emphasis on working software than documentation. To further illustrate that, there can be challenging in maintaining detailed and clear records of the design, features and implementation details regarding the educational institution management system due to the reason that Scrum approach advocate on rapid development and adaptability to evolving requirements. As for this case, the customer might want to develop another upgraded version of system based on this system developed, the comprehensive and detailed documentation is crucial for knowledge transfer, hence, it will be challenging to maintain such documentation by using Scrum development to develop the software (Deepak Mohan Kumar, 2023).

2.0 Design Review

Overall, the scope and the design of the user interface of the developed software as requested by the educational institution for managing student's registration, course result, payment, as well as tracking the study progress of students is considered a good design in terms of the golden rules of UI, Mandel's Golden Rules in specific. Briefly, it is a guidance meant to act as an assistance towards designer in designing user interface that is well designed, specifically to be user-friendly by enhancing the usability of the user interface It consist of three groups of rules, which included place user in control, reduce users' memory load and make the interface consistent (Mandel, 1997).

First of all, the developed system is having a good user interface as according to the third group of Mandel's Golden Rules, which is a good user interface should make the interface consistent. Under the third group, one of the rules stated that it is crucial to maintain consistency within and across products. To further explain that it requires the user interface to follows the rule of throughout the software workflows, it should constantly utilise the same patterns on design and having consistent sequences for a similar situation. To be specific, it refers to the proper and consistent use on terminology, typography and colours. In short, it is required to have consistent font colours, font styles and alignment of the content. The reason is that, presenting elements of user-interface that appears in the same way will greatly enhance the predictability of the system and hence, greatly increase the usability of the software (Mandel, 1997). For example, as shown in *Figure 2.1*, *Figure 2.2*, *Figure 2.3*, and *Figure 2.4*. The navigation bar on top of the system remains the same for every page. Moreover, the font style, font size and basic colour themes are consistent for similar pages.

Hence, in essence, the design of the developed software fulfils the rule of maintain consistency within and across products and hence enhance the predictability of the software, making user navigating the pages faster, thus resulting in enhanced usability, which made it a good user interface.
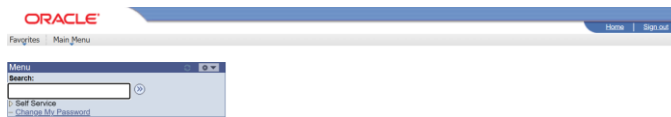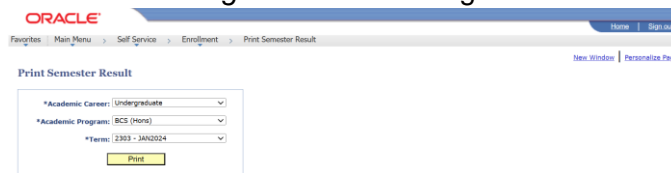

Figure 2.1 Home Page

Figure 2.2 Self Service Page

Figure 2.3 Print Semester Result Page
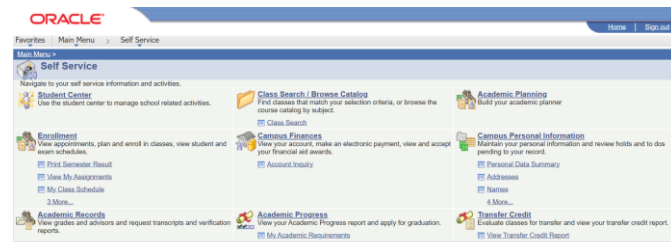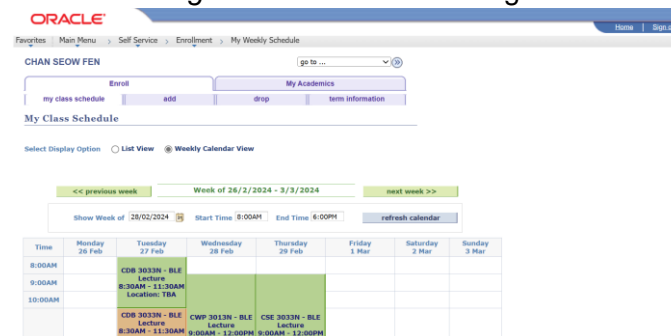

Figure 2.4 My Weekly Schedule Page

Moreover, the developed software is considered having a good user interface design according to the Mandel's Golden Rules, which is provide interface shortcuts, under the rule group of reduce users' memory load (Mandel, 1997). To illustrate that, as when a user uses an application more frequently, the more the demand from the user to have quicker approach to perform the task. For example, *Figure 2.5* shows that, the developed software considers a navigation button of "Favourites" that contains the services or pages that user recently used, which greatly enhance the convenience for the user who frequently utilises the system as they could easily navigate to the desired pages without the need of searching it from the main pages, by using that functionality. In essence, the developed system is a good design as it follows the guidelines on Mandel's Golden Rules, which is provide interface shortcuts.
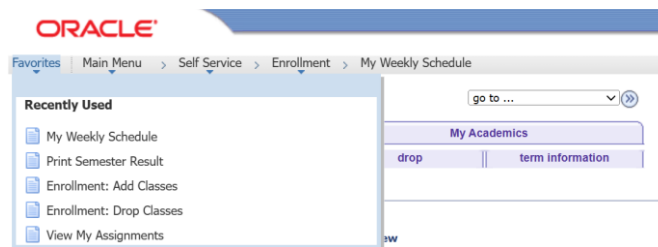
Figure 2.5 Favourites Button to Navigate to the Recently Used Services

Furthermore, the scope and user interface design of the developed software is perceived as a good design as according to the Mandel's Golden Rules, which is display descriptive messages and text that is under the rule of place users in control. To further elaborate that, it requires feedback offering from the software to the user after complete performing a group of actions in order to prevent user from confusing as they fully recognised their action have been done, which provide users with the sense of relief (Mandel, 1997). For

example, as shown in *Figure 2.6* and *Figure 2.7*, after the user attempts to change the password and clicked on the "Change Password" button after entering valid input, the software developed will prompt the alert for informing the user that new password has been saved and password has changed successfully. Hence, having such information prompted to the user can clear the confusion and doubts of user as well as make user feels reassure as knowing their attempt is successful. In essence, this design of the developed software has fulfilled the Mandel's Golden Rules of Interface Design, which is display descriptive messages and text, thus, made it a good design of user interface.
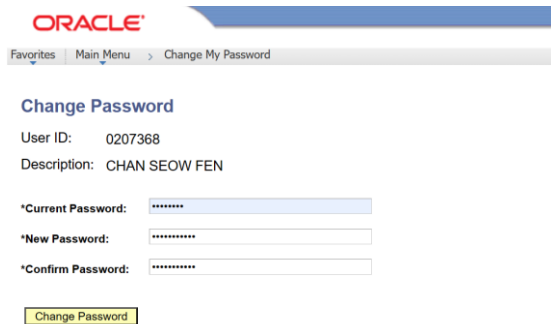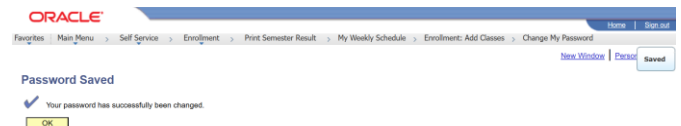

Figure 2.6 Change Password Page

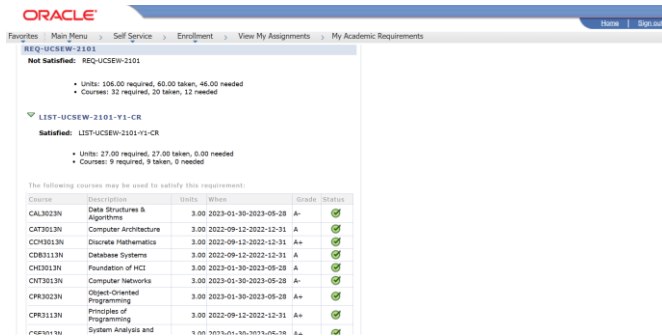*Figure 2.7* Page Informing Password Has Been Changed Successfully

      In addition, the software developed is perceived as a software with good scope and user interface design as according to the Mandel's Golden Rules, which again referred to display descriptive messages and text. As the previous case is more towards displaying the result of the interaction, in this case, it is more on the part of telling user about the error. To illustrate that, it is essential for a good user interface to be user-friendly on letting user understand the problem that faced and resolve it by informing the user about the errors when something is not right (Mandel, 1997). For example, in *Figure 2.8*, when user accidentally key in wrong credentials, either wrong id or password, or both, the software developed will immediately prompt the error to the user and informing the mistake which is invalid credentials. By having such implementation, it greatly enhances the usability of the system as user could understand what the error is they are facing and resolve it such as retype valid credentials again to login to the system. Hence, the system fulfilled the Mandel's Golden Rules which is display descriptive messages and text by imforming the credentials issues and prompt it to the user to clarify the situation, which made it a good design.

Figure 2.8 Prompt Error Message on Invalid Password or ID when User Mistyped the Credentials

      Lastly, it is strongly believed that the developed software is having a good scope and user interface design as based on Mandel's Golden Rules, which refers to user progressive disclosure that is under reduce users' memory load. To illustrate that, it requires the practice of prevent overloading the software or site with same level information. The first level that used to capture user attention should be reduce as lesser as possible as too much information

would lose user's attention. In short, it is essential to ensure that user would not be distracted by unnecessary or unwanted information. Hence, the concept is based on offering data-entry procedures that have been simplified and make the display readable in order to reduce the short-term memory load of user (Mandel, 1997). For example, as shown in *Figure 2.9* and *Figure 2.10*, the information displayed on the developed software is considered as minimalist, it only provides the necessary information to the user, with the minimal distractions. In addition, it also provides the option to expand or hide a certain section, to make user focus on their desired content. In essence, this option highly enhances the usability of the software as well as the user-friendliness of the program. Hence, the scope and the design of the user interface of developed software is considered as a good design as it adheres to the Mandel's Golden Rules, which is user progressive disclosure.

Figure 2.9 Page Showing the Expanded Information
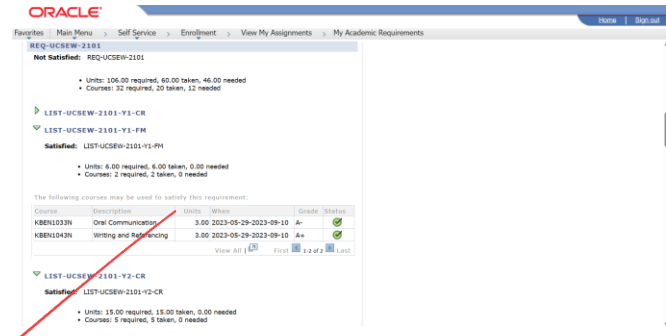
Figure 2.10 Page Hides the Information after User Choose to Hide It

3.0 Software Sustainability & Improvement

3.1 Review on Software Sustainability

According to the software developed using the Scrum approach, it is confident that the software's is sustainable in the long term, however, there are certain factors needed to be considered in order to fully avoid the risks of potential failures.

The software sustainability is not a concern even in the long term is due to the iterative nature of Scrum, as mentioned before, its excellent ability to handle the changing requirements of customer over time. As for a recap, Scrum approach is highly adaptative which provides improvement continuously, and also highly responsive to the customer requirements and feedback that will be continuously evolving in the future. Hence, it is more likely that the software product developed based on Scrum methodology will remains valuable and sustainable in the long run as it ensures that customers have chances and opportunities to participate and provide insightful feedback and direction toward the software development during the sprint review, which is the process that showcasing the sprint increments to the stakeholders (Bambazek, Groher and Seyff, 2022). Hence, in short, it is confident that the system developed will be sustainable due to the agility of Scrum development, especially about adaptability and flexibility.

However, one of the potential risks to the failure of software sustainability is on the documentation. As mentioned in the drawbacks of Scrum development, Agile software development process mostly does not consist of detailed and comprehensive documentation that recorded the important information such as the system implementation details, the minor issues on the system that might be evolved into a large threat to the system sustainability, and some of the decisions on the design. Such information is essential as without the system details, the software's maintainability and scalability will be a serious concern in the future

such as increased technical debt and the complexity of the system (Gilad David Maayan, 2023).

3.2 Suggestion on Improvement Area

3.2.1 Software Process Methodology

      In terms of the improvement on selected software development methodology, which is Scrum, its disadvantages have been discussed before, which included highly dependent on customer availability, might be challenging to meet deadline of overall design and lack of detailed documentation. Hence, this section will mainly focus on the suggestion to improve the area of weaknesses of the Scrum methodology.

      Firstly, in order to resolve the potential issues raised on high dependent on customer availability which is the progress of sprint is impacted, it is highly recommended to leverage online technology such as software for project management, collaborative workspace for sharing document, video conferencing platform for online meeting, which highly facilitates the virtual interaction that does not constrained by time or place, making it convenient for customer to join the sprint review, even though the customer is not available to attend the meeting physically (Atlassian, 2024). Furthermore, it is also essential to discuss with the customer about other representatives for attending the sprint review as to in charge for the decision making and hence avoid the efficiency and progress being delayed or impacted by unforeseen circumstances.

      Next, in order to better manage the overall deadline, it is recommended that to be more precise and prudent when cutting down the big project into smaller slice, it is crucial to estimate the efforts and time needed for those smaller tasks more accurately, as well as setting the time needed a little more than the expectation as there might be unforeseen circumstances that causes delays. Moreover, it is also important develop a clearer understanding of the capacity of the tasks so that development teams can understand better on the time constraints and avoid exceeding the due date. Lastly, it is essential for the scrum master to keep track on the due date of the larger project so that it can ensure the development team to meet overall project deadlines as the larger project would not fall behind the schedule for too far (Vasiliauskas, 2022).

      Lastly, to address the issue on documentation, it is recommended to facilitate the practices on doing documentation. To illustrate that, it is crucial for the development team to understand the importance of recognising the documentation value, which is important for maintaining system integrity, knowledge transfer as well as for the use of future maintenance and further enhancement. Even though Scrum highlighted on the software than documentation, there is still efforts should be made to record essential design concepts, user stories, and architecture diagrams throughout the process of developing the software. Such information is critical to ensure key information is accessible to customers, development team members as well as the future participants (Vinod Sukumaran, 2019).

3.2.2 Design Characteristics

      In terms of The Mandel's Golden Rules, there is a few criteria that does the developed software does not adhere to, hence it serves as an area of improvement on the design characteristics of the developed software. Firstly, it does not achieve the Mandel's Golden Rules, which is provide visual cues, under the rule of reduce users' memory load. To further elaborate that, the rule requires the system to provide proper feedback with graphical element to the user for every action done by user so that they know what is happening (Mandel, 1997). However, as shown in *Figure 3.1*, the developed system does not inform the user that he or she does not have the permission on modifying those information, it also does not directly retract the button for user to perform modification. instead, it leaves the button to be clickable but with a little shades of grey on the button from the user perspective, however, it is not obvious that the button could not be pressed. And after user interact with the button, it does now prompt any dialog or alert box for informing user about the issue or even a loading

graphical element informing the current status. Hence, it ends up making confusion to the user as they will be waiting for some time and only realised that the action that did does not work for them. Therefore, as the developed system design does not adhere to the Mandel's Golden Rules, it still demands an improvement on such area. *Figure 3.2* provides the example of provide visual cues with a red exclamation mark informing the error. Hence, the suggestion on improving the design in *Figure 3.1* is to prompt the message similar to *Figure 3.2*, include an graphical element showing there is an error and mention that the user does not have permission on modifying such information, and ask the user try to contact the administrator of the software to modify the information as demanded, as to provide visual clue with explanation to ensure user does not get confused.
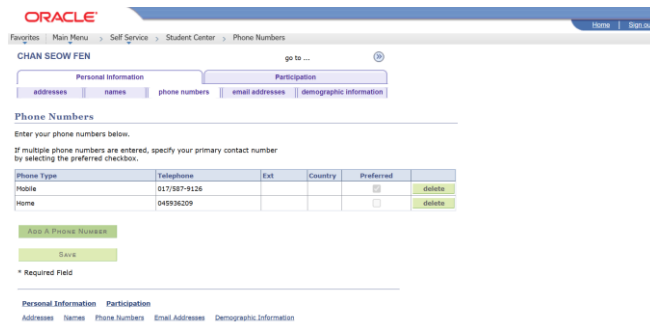
Figure 3.1 Modification on Phone Numbers Button is Not Responsive to User
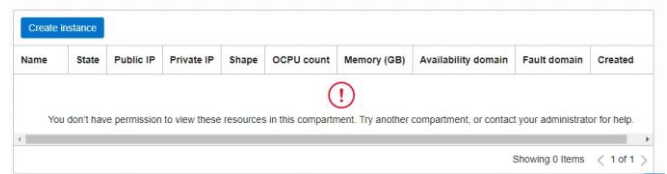
Figure 3.2 Example of Provide Immediate and Reversible Actions, and Feedback – Informing the User About Permission Issue (*Reddit - Dive into anything*, 2024)

Lastly, according to Mandel's Golden Rules, the system user interface should allow users to customise the interface, which is under the rule of place user in control. To illustrate that, it is required to provide user freedom in order to feel in control of the system, which helps in reassuring the user as some form of free is offered (Mandel, 1997). However, in *Figure 3.3*, the developed system does not include such features of allowing users to customise the services that appeared in the main page. The user does not have freedom to control the system, rather just adhering to the design of the software. Hence, there is an area of improvement in design characteristics as it does not fulfil the requirements of the Mandel's Golden Rules, which is allow users to customise the interface. *Figure 3.4* shows the example of allow users to customise the interface. Hence, the suggestion on improving the design in *Figure 3.3* is that include the function for user to select whether what services will appear on the main page and the others will be hid in order to enhance the control of user to the developed software, which is placing the users in control as according to Mandel's Golden Rules.

Figure 3.3 Does Not Provide User the Ability to Customise the Main Page
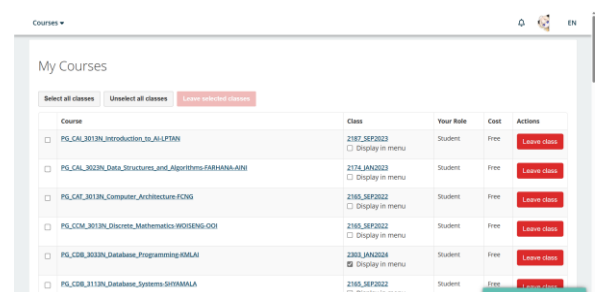
Figure 3.4 Example of Allow Users to Customise the Interface– Support the Option of Choosing What to Display on the Menu [Website: Open Learning]

4.0 Reference List

Atlassian   (2018) *What   is   Scrum?*, *Atlassian*.   Available   at:
https://www.atlassian.com/agile/scrum#continued (Accessed: 26 February 2024).

Atlassian   (2019) *Sprint   planning*, *Atlassian*.   Available   at:
https://www.atlassian.com/agile/scrum/sprint-
planning#:~:text=What%20is%20sprint%20planning%3F,with%20the%20whole%20scrum%
20team. (Accessed: 25 February 2024).

Atlassian (2024) *How to Manage Scrum Remote Teams | Atlassian*, *Atlassian*. Available at:
https://www.atlassian.com/agile/scrum/distributed-scrum (Accessed: 27 February 2024).

Atlassian (2024) *Three Pillars of Scrum: Understanding Scrum's Core Principles*, *Atlassian*.
Available   at: https://www.atlassian.com/agile/project-management/3-pillars-
scrum#:~:text=Adaptation%20in%20Scrum%20yields%20several,product%20quality%20and
%20user%20satisfaction. (Accessed: 26 February 2024).

Atlassian   (2024) *What   is   Agile? | Atlassian*, *Atlassian*.   Available   at:
https://www.atlassian.com/agile#:~:text=What%20is%20the%20Agile%20methodology,plann
ing%2C%20executing%2C%20and%20evaluating. (Accessed: 25 February 2024).

Bambazek, P., Groher, I. and Seyff, N. (2022) 'Sustainability in Agile Software Development:
A   Survey   Study   among   Practitioners'.   Available   at:
https://doi.org/10.1109/ict4s55073.2022.00013.

Chandana   (2024) *Scrum   Project   Management: Advantages   and
Disadvantages*, *Simplilearn.com*.   Simplilearn.   Available   at:
https://www.simplilearn.com/scrum-project-management-article (Accessed: 26 February
2024).

Czisch, C. (2022) *What Is A Scrum Sprint And How To Run It? - Vollcom Digital*, *Vollcom
Digital*. Vollcom Digital. Available at: https://www.vollcom-digital.com/blog/digital-
transformation/what-is-a-scrum-sprint-and-how-to-run-it/ (Accessed: 25 February 2024).

Deepak Mohan Kumar (2023) *What is Scrum? According to The Scrum Guide, the scrum is
'a lightweight framework that helps people, teams and organizations generate value through
adaptive solutions for complex problems' (Schwaber & Sutherland, 2020). Scrum is the most
widely used and popular agile framework and is based on the p*, *Linkedin.com*. Available at:
https://www.linkedin.com/pulse/what-scrum-when-use-advantages-disadvantages-deepak-
mohan-kumar (Accessed: 27 February 2024).

Dipendra Ghimire and Charters, S. (2022) 'The Impact of Agile Development Practices on
Project   Outcomes', *Software*,   1(3),   pp.   265–275.   Available   at:
https://doi.org/10.3390/software1030012.

Gilad David Maayan (2023) *Documentation in Agile: Challenges and Trends in 2023 -
DevOps.com*, *DevOps.com*.   Available   at: https://devops.com/documentation-in-agile-
challenges-and-trends-in-
2023/#:~:text=Balancing%20speed%20and%20detail%3A%20Agile,project%20evolves%20
and%20requirements%20change. (Accessed: 27 February 2024).

Infinity (2024) *Scrum Methodology: An Agile Framework for Project Management*, *Infinity*.
Available at: https://startinfinity.com/project-management-methodologies/scrum (Accessed:
25 February 2024).

MacNeil, C. (2022) *The Art of Sprint Retrospectives: Run an Effective Retro [2023] •Asana, Asana*. Asana. Available at: https://asana.com/resources/sprint-retrospective (Accessed: 26 February 2024).

Mandel, T. (1997) *Chapter 5: The Golden Rules of User Interface Design The Elements of User Interface Design © John Wiley & Sons, 1997*. Available at: https://theomandel.com/wp-content/uploads/2012/07/Mandel-GoldenRules.pdf (Accessed: 27 February 2024).

Obi, I. (2019) *Advantages of Delivering Products Using Scrum, Linkedin.com*. Available at: https://www.linkedin.com/pulse/advantages-delivering-products-using-scrum-anthony-obi (Accessed: 26 February 2024).

Reddit - Dive into anything (2024) *Reddit.com*. Available at: https://www.reddit.com/r/oraclecloud/comments/uif56o/oracle_cloud_account_ban_for_no_reason/ (Accessed: 27 February 2024).

Salimi, S. (2024) *Iterative Development, Agile Academy*. Available at: https://www.agile-academy.com/en/agile-dictionary/iterative-development/#:~:text=In%20iterative%20development%2C%20Scrum%20teams,next%20iteration%20of%20the%20product. (Accessed: 25 February 2024).

Scrum Alliance (2024) *Your Guide to Empirical Process Control in Scrum Teams | Scrum Alliance, Scrumalliance.org*. Available at: https://resources.scrumalliance.org/Article/empirical-process-control-scrum#:~:text=Empirical%20process%20control%20is%20a,and%20making%20adjustments%20as%20needed. (Accessed: 25 February 2024).

Scrum.org (2022) *The Purpose of the Five Scrum Events, Scrum.org*. Available at: https://www.scrum.org/resources/blog/purpose-five-scrum-events (Accessed: 27 February 2024).

Scrum.org (2024) *What is a Daily Scrum?, Scrum.org*. Available at: https://www.scrum.org/resources/what-is-a-daily-scrum (Accessed: 26 February 2024).

Scrum.org (2024) *What is a Product Backlog?, Scrum.org*. Available at: https://www.scrum.org/resources/what-is-a-product-backlog#:~:text=As%20described%20in%20the%20Scrum,in%20a%20Sprint%20Planning%20event. (Accessed: 25 February 2024).

Scrum.org (2024) *What is a Sprint Review?, Scrum.org*. Available at: https://www.scrum.org/resources/what-is-a-sprint-review (Accessed: 26 February 2024).

ScrumAlliance (2024) *What Is Scrum: A Guide to the Most Popular Agile Framework, Scrumalliance.org*. Available at: https://www.scrumalliance.org/about-scrum (Accessed: 26 February 2024).

SCRUMstudy (2024) *Empirical process control in Scrum project| SCRUMstudy, Scrumstudy.com*. Available at: https://www.scrumstudy.com/whyscrum/scrum-empirical-process-control (Accessed: 25 February 2024).

Vasiliauskas, V. (2022) *14 Scrum Advantages and Disadvantages in 2024, Teamhood*. Available at: https://teamhood.com/agile/scrum-advantages-disadvantages/ (Accessed: 26 February 2024).

Vinod Sukumaran (2019) *Agile Documentation: Balancing Speed and Knowledge Retention*, *Toptal Projects Blog*. Toptal. Available at: https://www.toptal.com/project-managers/agile/agile-documentation (Accessed: 27 February 2024).

Wu, L. and Wang, Z. (2020) *Understanding and managing the challenges of distributed scrum teams*. Available at: https://www.diva-portal.org/smash/get/diva2:1485315/FULLTEXT02.pdf.

Zoho Corporation (2024) *What is a sprint review? Definition, purpose, best practices | Zoho Sprints*, *Zoho*. Available at: https://www.zoho.com/sprints/sprint-reviews.html (Accessed: 27 February 2024).

Marking Rubric – Assignment 1

Section	Failed (0-49)	Third Class (50-59)	Second Class Lower (60-69)	Second Class Upper (70-79)	First Class (80-100)	Mark
Section 1: Software Development Methodology (35%) [CLO3]	No discussion or totally wrong description of software development methodologies for the software.	Part of software development methodology description is incorrect or software development methodology are described with very brief justification.	The software development methodology is described with a few areas of justifications are unclear or incorrect.	The description of the software development methodology for the software are correct with 1-2 areas of justifications are unclear or incorrect.	The description of the software development methodology for the software are correct and the justifications are detailed and clear.	Raw mark 95 /100 Section mark /35
Section 2: Design Review (35%) [CLO3]	No review of the scope and design aspects of the software or all review areas are incorrect.	The review of the scope and design aspects of the software are largely irrelevant with very limited evidence and explanations.	The review of the scope and design aspects of the software are given but a few areas of scopes or design metrics are unclear or incorrect.	The review of the scope and design aspects of the software are given with 1-2 areas of explanations are unclear or incorrect.	The review of the scope and design aspects of the software are given and the explanations are detailed and clear with full evidence.	Raw mark 90 /100 Section mark /35
Section 3: Software Sustainability & Improvement (30%) [CLO2]	No discussion on software sustainability & improvement or all discussion are incorrect.	Only part of software's sustainability & improvement is discussed, or software's sustainability & improvement are discussed but irrelevant with very limited evidence and explanations.	The software's sustainability & improvement area of the software are discussed but a few areas are unclear or incorrect, wrongly justified.	The software's sustainability & improvement area of the software are discussed with 1-2 areas are unclear or incorrect, wrongly justified.	The software's sustainability & improvement area of the software are discussed and the explanations are detailed and clear with full evidence.	Raw mark 95 /100 Section mark /30
					Total Score:	93.25 /100