

**AGILE DEVELOPMENT**

**Module Title**

**GeekUp – Valuation Mobile App**

**Assignment Title**

**Autumn 2024**

**Examination Cycle**

**THI HAN NAING**

**Candidate Name**

**00189889**

**Candidate No**

**KMD Institute (Yangon)**

**Centre Name**

**13/July/2024**

**Submission Date:**

**Important Notes:**

* Please refer to the Assignment Presentation Requirements for advice on how to set out your

assignment. These can be found on the NCC Education *Campus*. Scroll down the left hand

side of the screen until you reach Personal Support. Click on this, and then on Policies and

Advice. You will find the Assignment Presentation Requirements under the Advice section.

* You must familiarize yourself with the NCC Education Academic Dishonesty and Plagiarism

Policy and ensure that you acknowledge all the sources which you use in your work. The

policy is available on *Campus*. Follow the instructions above, but click on Policies rather

than Advice.

* You must complete the **‘Statement and Confirmation of Own Work’**. The form is

available on the Policies section of *Campus*. Scroll down the left hand side until you reach

Personal Support. Click on this and then click on Policies and Advice.

* Please make a note of the recommended word count. You could lose marks if you write 10%

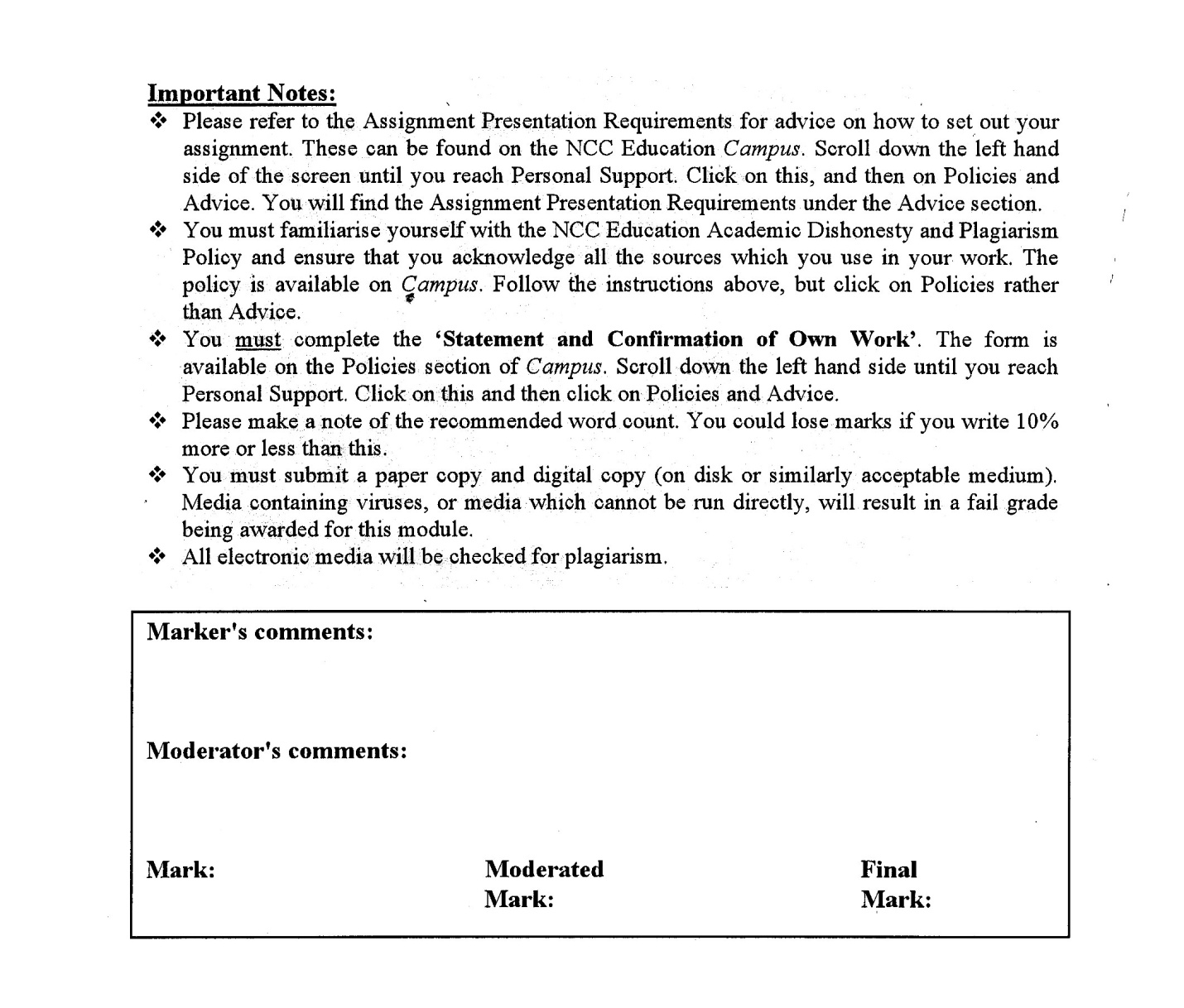
more or less than this.

* You must submit a paper copy and digital copy (on disk or similarly acceptable medium).

Media containing viruses, or media which cannot be run directly, will result in a fail grade

being awarded for this module.

* All electronic media will be checked for plagiarism.



****

A signed copy of this form must be submitted with every assignment.

If the statement is missing your work may not be marked.

For Level 7 assignments, please use the separate Level 7 Candidate Statement of Own Work form instead of this one.

Student Declaration

I confirm the following details:

|  |  |
| --- | --- |
| **Candidate Name:** | **THI HAN NAING** |
| **Candidate ID Number:** | **00189889** |
| **Qualification:** | **Level 5 Diploma in Computing** |
| **Unit:** | **Agile Development** |
| **Centre:** | **KMD Institute (Yangon)** |
| **Word Count:** | **5247** |
| I have read and understood both NCC Education’s *Academic Misconduct Policy* and the *Referencing and Bibliographies* document. To the best of my knowledge my work has been accurately referenced and all sources cited correctly.  I confirm that I have not exceeded the stipulated word limit by more than 10%.  I confirm that this is my own work and that I have not colluded or plagiarised any part of it. | |
| **Candidate Signature:** | **Thihannaing** |
| **Date:** | **13/July/2024** |

**Table of Contents**

Contents

[Introduction 4](#_Toc171775639)

[Task 1 6](#_Toc171775640)

[Comparison of DSDM over Waterfall 6](#_Toc171775641)

[Advantages comparison of DSDM and Waterfall 6](#_Toc171775642)

[Disadvantages comparison of DSDM and Waterfall 6](#_Toc171775643)

[Main Phases of DSDM 7](#_Toc171775644)

[Task 2 11](#_Toc171775645)

[Agile Principles 11](#_Toc171775646)

[Task 3 14](#_Toc171775647)

[DSDM Products 14](#_Toc171775648)

[Task 4 17](#_Toc171775649)

[Five User Stories 17](#_Toc171775650)

[Task 5 20](#_Toc171775651)

[A) Sequence diagram demonstrating the flow of a customer purchasing item from the app 20](#_Toc171775652)

[B) Class diagram of system being proposed in the scenario 21](#_Toc171775653)

[Task 6 24](#_Toc171775654)

[Prototyping 24](#_Toc171775655)

[Task 7 28](#_Toc171775656)

[Retrospective 28](#_Toc171775657)

[A) The Term of Timeboxing 28](#_Toc171775658)

[B) Another agile approach might have been preferred for this project 30](#_Toc171775659)

[References 32](#_Toc171775660)

[References 32](#_Toc171775661)

[Candidate Checklist 34](#_Toc171775662)

# Introduction

The assignment is about software developing a peer-to-peer marketplace system for GeekUp company with DSDM methodology. The assignment includes Introducing DSDM and Agile practices in Task 1, DSDM Principles in Task 2, DSDM Products in Task 3, Requirements/ User Stories in Task 4, Modelling which includes Sequence Diagram and Class Diagram in Task 5, Prototyping in Task 6, and Retrospective in Task 7. By following documentation guidelines from this assignment, Disrupt Digital software house can be able to develop the application software which would meets GeekUp’s requirements.

Task (1)

Word Count (1090)

# Task 1

## Comparison of DSDM over Waterfall

### Advantages comparison of DSDM and Waterfall

Advantages of using DSDM

1. Its iterative nature helps with developing the future improvement of the project

For example, in this scenario, the company wants to develop their application in the future. In this case, DSDM (agile methodology) is the best to choose.

1. Use a few groups of people can lead to lower the cost and reduce the conflicts between the groups

For example, since this is the start up company and will use some limited amount of funds, using fewer people would be better. And whenever conflict happens the problems can be solved in a minimum time.

1. Project steps can be worked in parallel which means project can be done faster.

For example, for a start up company, developing the application faster would get more chance in the market. And also, functions or adding new features can be done faster and easier in DSDM.

Advantages of using Waterfall

1. It can predict the cost of the overall project.

For example, the company wants to predict the exact amount of money they want to use in the early steps of the project.

1. Suitable for the projects where the requirements can be exactly defined.

For example, the company knows their requirements (functions, UI, etc.) so that using waterfall method will help the project to implement faster and easier than waterfall used projects.

### Disadvantages comparison of DSDM and Waterfall

Disadvantages of using DSDM

1. Since it has infinite nature, the shareholders and owners cannot predict the final cost

For example, the company will need more funding in the future if they continue to develop their application to become a better one.

1. Documentation problems always occur in agile related methods. Since it is iterative, the documentation will also need to be updated whenever a change occurs.

For example, whenever the owners want to read the project book (documentation book), it would not be written clearly and would be unfinished one.

Disadvantages of using Waterfall

1. Project will take too long since it cannot be worked in parallel and needs to wait for the previous stream to finish completely. And also, whenever an update is needed, the whole system needed to re-designed and implement.

For example, this would affect the company’s plans to be delayed. And since the company will add more features in the future to get a position in the market, using waterfall will not be a good choice for the company.

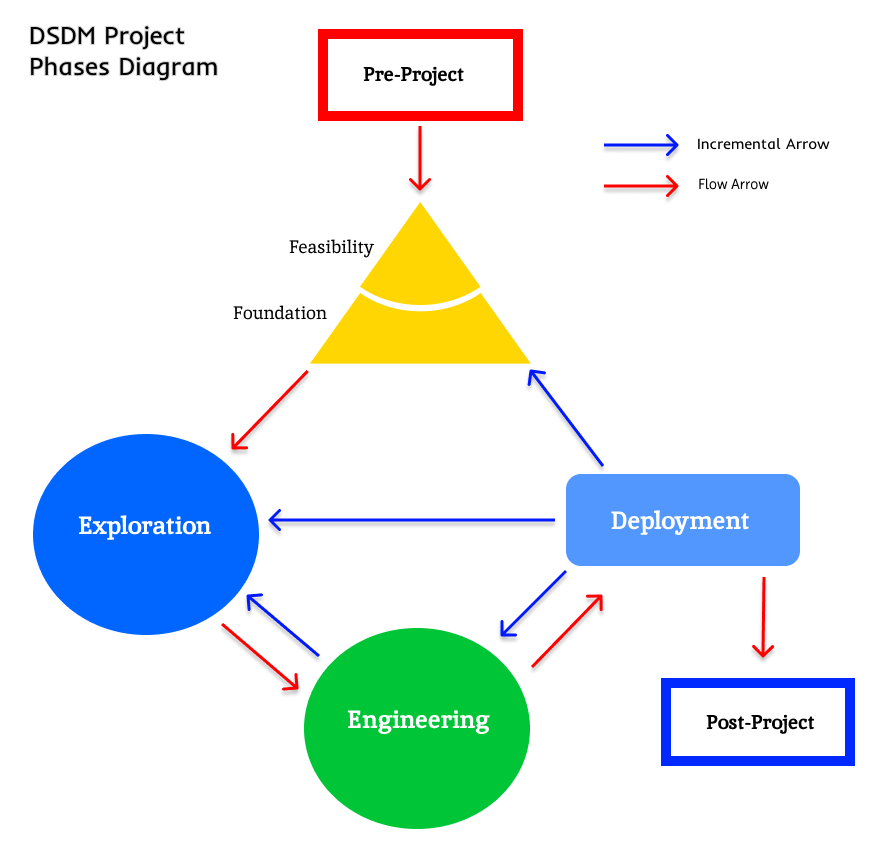
1. A lot of people will be needed and the cost will be high. Normally waterfall methodology used project can able to predict the cost but it is high since a lot of people will be included.

For example, waterfall needs lot of human resources and most of the funds will be for their salary. And many employee numbers can lead to more possible conflicts between them.

(Anon., 2023) (Anon., n.d.) (Beyond, 2023) (Paredes, 2024) (Anon., n.d.)

## Main Phases of DSDM

Illustration for all main phases of the DSDM life cycle



**Phases of DSDM**

1. Pre-Project: Identify the project needs, scope. Recruit project members. Establish clear business case and securing the approvals and funding.

For example, from the scenario, Billy and Sarah Hampson will need to secure their funding and find recruit the project manager, team leaders, technical coordinators, etc.

1. Feasibility: analyze the project is practical and achievable. Make feasibility reports and assess if the project is viable when deploying.

For example, project members will need to define risk managements, analyze and evaluate technical (e.g. Software and hardware capabilities) and financial feasibilities (e.g. budget stay within funding limits). Ensuring and evaluate the project’s key features, functions and other requirements will really needed in real life or not. In this stage the stakeholders’ involvements will also be needed.

1. Foundations: Define the project scope and constraints from the researches got from feasibility and pre-project phases. Make foundation for the project or ensure the infrastructure of the project.

For example, the team will define the scope using the requirements described in the scenario. They will need to brainstorm about the constraints and choose tools (e.g. software, programming languages) they will use for the project. If foundation plans are technically or financially impossible, they will need to make feasibility plans again.

1. Exploration: the team will need to create time-box planning based on MoSCoW prioritization. They will also need to make prototypes and receive feedbacks or agreements from stakeholders.

For example, projects members will discuss about their project plans based on MoSCoW and make documentations or prototypes which will be reviewed by stakeholders or owners to require their agreements. After all the requirement is clearly defined, project will be able to go to Engineering stage.

1. Engineering: develop the project in incremental way. While developing it, members will need to follow the features in order of priority. Need continuous testing and reviewing after developing new function is complete.

For example, project members will develop the project in parallel and they will able to test and review after each feature is finished. They will also need to make risk managements in this stage. If the project cannot be developed in technical way, it will need to go back to foundation stage or previous stage.

1. Deployment: after all the development is finished, overall testing is made before deploying. And discuss with the owners whether they like it or not. And deployment plan will be made after the stakeholders discussed.

For example, after all the project development is finished and tested. The system will be tested and demonstrated in front of stakeholders. If it satisfies the stakeholders needs, the owners and stakeholders will make a deployment plan. And the system will be deployed in real life as a beta version. During this time, risk analysis will also If the system really suitable with the requirements, an official version will be deployed. If not, they will need to go back to the engineering stage or respective stage.

1. Post-Project: Evaluate the project and lessons learned during the project will be documented. Analyze the project’s impacts and document the benefits from the project. Also ensure the maintenance and establish support mechanisms and user support.

For example, after the system is deployed and used. It will need maintenance like updating new features and fixing bugs or upgrading securities. Lessons, problem solving methods, financial benefits will be documented as a project book. Also, user manual and project documentations will be handed to owners/clients in this final stage.

(Anon., n.d.) (Islam, n.d.)

Task (2)

Word Count (603)

# Task 2

## Agile Principles

1. **Focus on business need**

Before developing a project, study about related business is must for better quality product. To achieve project members must need to study business requirements, budget capabilities, environmental constraints, clients’ requirements, and technical capabilities.

For example, from the scenario given by Geek Up, they clearly requested their most wanted functions and design. The project members will need to brain storm about what they got from business strategies. They will need to make analytics by learning what the market trends are aiming, deciding the functions mentioned by the owner are really necessary or not, can technology will be able to make requirements possible, and planning the system will able to develop into better version in future when the owners want to expand their business.

1. **Deliver on time**

DSDM is an iterative methodology and iterating a lot can delay the project. To make the project to be delivered on the exact time, time box plan will be needed. Project members will need to follow the time box plans and must finish their tasks by time box plan.

For example, there will be a time box plan including at least two timeboxes for this project. Those time boxes may be able to develop in parallel. Project members will need to develop functions and will need to combine those projects depend on timebox plan. For example, if account registration functions are finished, they will be needed to combine with other functions and then tested. Those development and testing will be needed to finished within the time box plan.

1. **Collaborate**

Collaboration is needed in every team when making something to finish successfully. Not only team members will need to collaborate but also the stakeholders, the owners will need to collaborate during a project. Project members roles will be clearly defined based on their skill and levels. There will be workshops, regular meetings during the project.

For example, there will be workshops during this project where the stakeholders can share their knowledge which will help the other members. Meetings where owners can review the product, can share their opinions, requirements. There will also be regular meeting during the project members where they can share their task accomplishments, troubles, and their opinions.

1. **Develop iteratively**

DSDM follows the iterative nature. This iterative nature can help a better-quality product, new knowledge and experiences, and can evolve the product with better solutions. Team will need to prepare for the iteration since the early stages. Depending on user involvement, team will also need to make back up plans and prepare easy iteration plans. And they will need to develop the system depending on those plans and make sure the system can be iterative or not.

For example, team members will make the system iteratively. So that whenever the owners add more features in future or when the system got an error, they can easily refine the system. Project iteration plans will also be added in the time box plans whether they will be needed or not.

1. **Demonstrate control**

There should be demonstrations during the project. Team will need to demonstrate their prototypes, development status, results to all the stakeholders. This will help with the iteration of the project and fasten the project.

For example, there will be demonstrations to the owners after prototype is finished and time box 1 and time box 2 are done. If the owners dislike the design or a function or something else, the team can change the system in iterative way. this demonstration plans will also be included in the time box plans.

(Anon., n.d.) (Petit, 2024)

Task (3)

Word Count (621)

# Task 3

## DSDM Products

**Business Foundation**

This is one of the important products of DSDM for current project. It is a fundamental aspect of a project and must be understood clearly from the beginning before further development processes. This includes market analysis, target audience, competitive landscape, and expected benefits. The documentation details the business workflow, processes, and provide the clear understanding of project’s aims and objectives. This document is crucial and can be also used as a guiding document through the project lifecycle for all the member while maintain focus and consistency of the project.

Type: Milestone Product

Role involved: Analyst, Sponsors and Owners, Business Visionary

Approver: GeekUp owners (business owners)

**Feasibility Assessment**

This is necessary in DSDM and also suitable for this project. It mainly includes technical, operational and financial feasibilities. It also ensures that the project will give profits and viable in real life and in future. The technical aspect reviews whether the used technologies and resources can support the project and in future too. The operations aspect includes workflows, ability to penetrate the market, potential problems and risk managements. Financial aspect includes that the project may increase business profits by overcoming the cost that are used in project and other expenses. After making feasibility assessment, owners and stakeholders will be able to know the profit of a product and can be able to decide that the project must continue or not.

Type: Milestone Product

Role involved: Analyst, Project Manager, and Business sponsor/owners (Billy and Sarah Hampson)

Approver: Business Visionary, and Business sponsor

**Prioritized Requirements List**

After requirements are defined, team will break down the features and prioritize the most critical functions. This is a dynamic and list the most necessary requirement for the project, based on business urgency. It also prioritizes the business values and key functions and features which would improve business. From the scenario given above, important key values, functions and features may include user account creation, item listing, payment integration, search and notification, and reminders, and those will be ranked by importance. PRL is regularly reviewed and updated to meet changes in business priorities through some project stages.

Type: Evolutionary Product

Role involved: Business Visionary, Analyst

Approver: Project Manager and Technical Coordinator

**Timebox plan**

Timebox plan includes the detail tasks, activities, timelines and deliverables. There are at least two timebox plans for every project and each timebox last two to four weeks depending on requirements. It typically has planning, designing, development and review phases. Those timebox can be developed in parallel in some cases. For the current project, after breaking the project into time boxes, team can develop in parallel by not wasting time and also focus on delivering the final product quality. There are also iterative plans, backup plans and risk managing in each timebox. After each time box is finished, deployment is focused.

Type: Milestone Product

Role involved: Project Manager, Team Leaders, Developers.

Approver: Project Manager and Technical Coordinator

**Deployment Plan**

After the project is developed with its requirements, aims and objectives, team will need to implement the tasks from deployment plan. It is already schedule before exploration and engineering stages of DSDM. This includes deployment schedules, resource requirements, risk managements and backup plans. Deployment plan also includes user training and support plans to users use the system while maintaining security. Deployment plan is also incremented and iterated through development stage of the project. And also include the stakeholders’ involvements the sooner the project is deployed. After the development and testing is finished, sometimes beta version is released first to fix bugs. Later the final version is released with the permissions from the owners and stakeholders.

Type: Evolutionary Product

Role involved: Technical Coordinator, Developers, Analyst.

Approver: Project Manager and Technical Coordinator.

(Anon., n.d.) (Anon., n.d.)

Task (4)

Word Count (530)

# Task 4

## Five User Stories

1. **Registration and login**

As a user, I want to login via Google and my social media accounts, so that I can quickly access platform and reduce forgotten password.

Acceptance criteria –

The user can create an account using Google or Facebook.

The user can login using their Google or Facebook.

The user can use other social media accounts to use the app in future.

1. **Search functions**

As a user, I want to perform up to three searches before prompting to create an account/login, so that I can explore app’s features and listings before creating an account.

Acceptance criteria –

The user can perform three searches without creating account.

1. **Save and Notify searches**

**As** a user, **I want** to save searches and receive notification based on my preferences (daily, weekly, every Friday or when new item is listed), **so that** I can stay updated without checking the app constantly.

Acceptance criteria –

The user can save searches

The user can customize notification reminders

The user can get notification depending on their likes and favored taste.

1. **Machine learning for listing**

As a seller, I want the app to use machine learning to scan and identify my items quickly, so that I can prove my items are real and can list my items efficiently.

Acceptance criteria –

Sellers can be proof for their authentications

Sellers can register items faster and list efficiently

1. **Smooth Performance**

As a user, I want the app to be responsive and easy to use with devices at least five years old without lagging, so that I can have smooth and enjoyable experience while using the app even with the old device.

Acceptance criteria –

The user can use the app without crushing on minimal 5 years old device

The user can use the app on any devices

The user can enjoy using the app without disrupting or lagging

(REHKOPF, n.d.) (Anon., n.d.)

**Difference between User stories and Epics**

A user story is a smallest unit and a simple description of a specific feature or functionality written from the perspective of an end user or customer. Its purpose is to deliver the expectations to users and outlines what they want and why they want. User story are manageable, and can be complete within a single iteration or sprint. These stories are typically written by product owner, product manager or program manager. (Anon., n.d.)

For example,as a user, I want the app to be easy to use and visually shot on point, so that I can use the app without troubles and done my jobs efficiently.

An Epic is a larger body of work that can be broken down into multiple user stories. Epics are wide in scope and larger than user stories and take longer to complete. It represents a set of features and often take multiple iterations or sprint. (REHKOPF, n.d.)

For example –

User Account Management (An epic)

As a user, I want to create account using google, so that I can manage my account better. (a user story)

As a user, I want to reset my password, so that if I forget I can able to access my account. (a user story)

Task (5)

# Task 5

## Sequence diagram demonstrating the flow of a customer purchasing item from the app

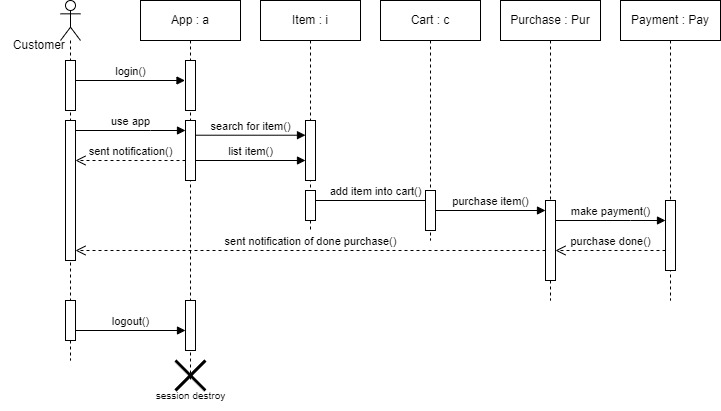


Fig: Sequence diagram of the flow of a customer purchasing

The sequence diagram above shows the purchase item process of the customers from the app. At first customer must need to login to the account to use the app. Customer can able to search item, list item, and can get notifications. When customer search for item and found the favored item, customer can add item into cart and then make purchase for its item. The customer can then make payment for the purchased item. After the payment is finished, purchase procedure is done and success process message will be sent back to the customer.

(Anon., n.d.)

## Class diagram of system being proposed in the scenario

**Initial Class Diagram**

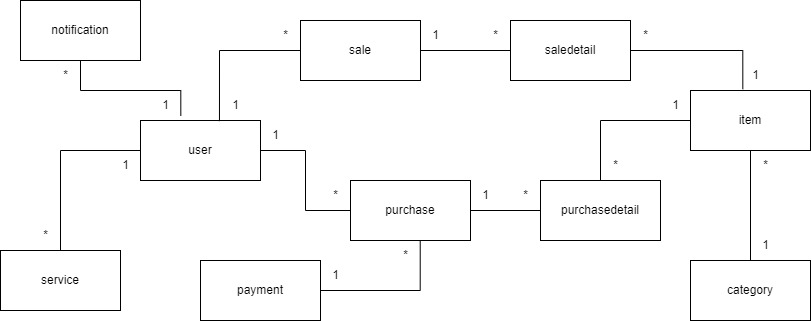


Fig: Initial class diagram of system related to the scenario

**Detailed class diagram**

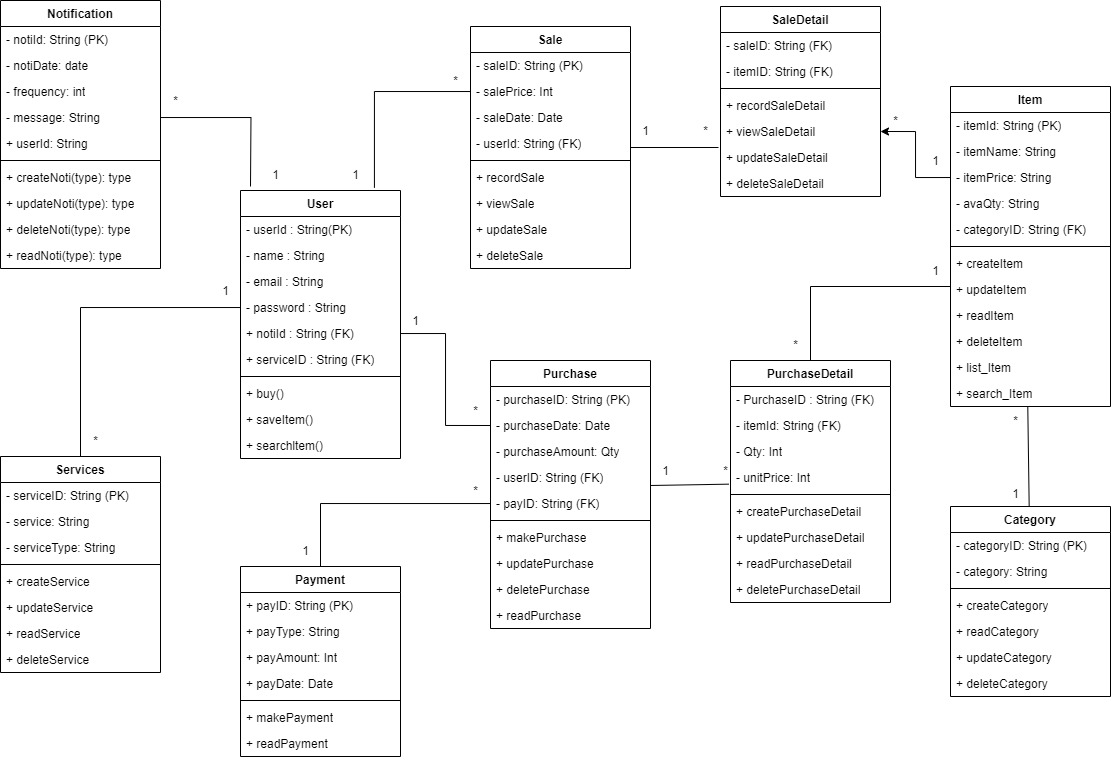


Fig: Detailed class diagram of system related to the scenario

Initial class diagram and detailed class diagram are described based on the scenario. In initial class diagram, only entity names and their relationship are included, however in detailed class diagram entities, attributes, methods and their relationships are included. In these class diagrams, there are ten entities: notification, service, user, purchase, purchasedetail , sale, saledetail, payment, item, and category. Notification and service tables are in connected with user table in many to one relationship. User table also has relation with sale and purchase tables which are connected in one to many relationships. Saledetail and purchasedetail are dummy table since purchase and item has many to many relationship and sale and item also has many to many relationship. Item also has relation with the category table in one to many relationship. In addition, payment table has one to many relationship with purchase table. All these tables has related method depending on the permissions. For example, item table has list item and search item method as an extra and payment table has only make payment and read payment. User can do search item, buy and save item. And most of the other tables has CRUD functions.

Task (6)

Word Count (1004)

# Task 6

## Prototyping

In software development field, to reduce risks, and uncertainties there are several techniques that are used. Among those techniques Proofs of Concept and Architectural Spike are mostly used in Agile based developments. Proofs of Concept may not be explicitly mentioned in the frameworks of Agile but it is used in some developments which use Scrum and Kanban. Architectural Spike is often used in Scrum, SAFe (Scaled Agile Framework) and XP (Extreme Programming). Both Proofs of Concept and Architectural Spike are used when teams need to reduce risks and validate certain aspects of a project so that they can reduce risk of wasting time, resources and uncertainties. However, they have different focuses like POCs focus on validating concepts. And spikes focus on technical feasibility and architectural decisions.

**Architectural Spike**

The word of Spike from Architectural Spike comes from the Extreme Programming (XP) practice which is a framework of Agile methodology. An Architectural Spike focus on time-box investigations on a feasibility of system’s architecture or technical requirements in order to reduce technical risks and increase the reliability of user story’s estimations. The goal is to explore, evaluate and validate the possible and potential solutions by identifying of risks and having plans for mitigation of risks in early development of software life cycles. This method is valuable in a project where uncertainties are high.

Purpose of an Architectural Spike

1. To determine the architectural patterns and different technologies while doing feasibility studies whether the decisions are fit to the solutions, flows and designs of overall system.
2. It is also used to mitigate risks by identifying potential issues and limitations of the chosen approach e.g. extreme programming.
3. Another reason to use it to make confirmation of decisions. To reduce the likelihood of redesigning which may reduce cost.

(Cohn, n.d.) (Anon., n.d.) (Sisictester, 2022)

Example in the GeekUp Scenario:

GeekUp requirements has many functions and features and applying architectural spikes will be needed since some implementing for some functions may have low possible rates. The team will need to identify and evaluate about the possibility of those features in their respective timeboxes. For example, spike could be conducted to evaluate the feasibility of machine learning. They will need to evaluate about different frameworks, and the effectiveness of machine learning in the real-life app. They will also need to identify and have mitigation plans for the possible risks. Another example is that the possibility of performance and responsiveness. The team will need to brainstorm about the methods, their limitations, cost and effectiveness, and time requirements, etc. Giving responsiveness may be capable but providing good performance in any device with the life span of 5 years old may be uncertain since software technologies and hardware requirements are increasing year by year. For the situation like this, architectural spike will need to be applied and potential problems like bottlenecks situations will be identify. After this, GeekUp will be able to make decisions on the architecture which supports a smooth and responsive user experience. Most of other requirements of GeekUp are technically possible in these days and only Machine Learning and Performance requirement has uncertainties.

**Proofs of Concept**

Proofs of Concept is a small-scale implementation targets for gathering of evidence to support the feasibility of a project. POC tests ensure the resource allocations by having better plans. Therefore, it can enhance trust among stakeholders and investors. Developing POC can also estimate the return on investment when the project ideas that lack planning and testing. Proofs of Concept is usually taken before committing of full-scale development by providing a demonstration of an idea’s viability.

Purpose of POC

1. Making feasibility demonstrations to support moving forward with the idea.
2. To get feedback from the stakeholders and to make better refinements.
3. Like the architectural spikes, one of the purposes of POC is to reduce risks and uncertainties.

(Malsam, 2023) (Anon., n.d.) (Anon., n.d.)

Example in the GeekUp Scenario

Proofs of Concept could be created to validate the search function, Google sign-in integrations, and save searches. For search function, the company will provide 3 free searches without creating an account nor login. By applying POC here can improve the search features and can be able to identify incidents or problems during search process. The team will need to ensure that after three searches the system transition must change into account creation seamlessly. For the Google sign-in, the team will need to ensure that they can use Google sign-in functions, and will need to test a lot to ensure the system can work with that function. Additionally, this POC can be extended since they will use media accounts for the login process. The team will need to create the POC for saving of searches too. There will be each POC for these. For example, saving of searching will need to be tested in different database with different input and many data. Then they will have to make sure fetching data process can work seamlessly without any errors.

**Differences of Architectural Spikes and Proofs of Concept**

Architectural Spike focus explorations, evaluations and validations on technical approaches or architectures. They will help with identifying potential issues and to guide decision making.

Proofs of Concept are more practical and focus on demonstrating the feasibilities. They will provide prototypes so that stakeholders can interact, give feedbacks and validate that system will meet their needs.

**Conclusion**

Both architectural spikes and proofs of concept are crucial in the development of GeekUp app. By using spikes, GeekUp can ensure technical approaches, mitigating of risks and confirmation on architectural decisions. Conducting proofs of concept provide more practical validations, feedbacks and feasibility of specific features. Since GeekUp requirements have lots of complexities and intend to being a cutting-edge application. So, it is crucial for the team to include both Architectural Spikes and Proofs of Concept to reduce uncertainties and ability for delivering a high-quality product that will meets user needs and expectations. Last but not least these development techniques can ensure smooth and successful development within the six-month timeframe.

Task (7)

Word Count (1048)

# Task 7

## Retrospective

### The Term of Timeboxing

Timeboxing in a project is a project management technique to allocate tasks or phases of the project. Timeboxing approach helps with time managing and effective since it focusses on completing tasks within the deadlines. Timeboxing describes activities with start date and end date for each task and has specific durations. There are also expected outputs for design, coding and testing. The project team will have to finish tasks within their timeline to get those expected outputs. Using time-box will also increase management skill and discipline for each member.

Timeboxing may come with many time boxes or whole timebox plan. Current time box is a whole-time box plan represents for the software development of GeekUp’s project. This timebox plan is designed after the team has finished adjusting. The following timebox will include its name, whole start date and end date, tasks, durations, specific task start & end dates, effort percentages and person involvements.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Timebox Name** | **GeekUp Software Development Time Box** | | | | |
| **Start Date** | | 6 May, 2024 | | | |
| **End Date** | | 6 November, 2024 | | | |
| **Task** | **Duration**  **(By Day)** | **Start Date**  **(2024)** | **End Date**  **(2024)** | **Effort Percent (%)** | **Person Involvement** |
| Requirement Gathering | 15 | 6 May | 24 May | 5 | All team members |
| Analysis | 20 | 27 May | 21 June | 15 | Analysts |
| Feasibility studies | 10 | 24 June | 5 July | 5 | PM, Team leaders |
| Designs | 10 | 8 July | 19 July | 15 | PM, Team leaders |
| Prototypes | 10 | 22 July | 2 August | 5 | UI/UX |
| Development  (Both frontend & backend) | 50 | 5 August | 11  October | 30 | Developers |
| Testing | 10 | 14 October | 25 October | 10 | Testers |
| Deployment  (Beta version) | 5 | 28 October | 1 November | 3 | GeekUp & PM |
| Bug Fixing | 2 | 4  November | 5  November | 2 | Developer |
| Deployment (Final version) | 1 | 6 November | - | 10 | GeekUp & PM |
| Key Deliverables (Output)  **Design**  Diagrams (Use-Case, Class, ER, Sequence, DFD, Gantt Chart)  Prototypes (Low-level & High-level)  **Coding**  Functions (Searching, Google Sig-in, Notification)  Machine Learning (Scanning, listing, identifying)  CRUD (accounts, items, purchasing, selling)  Responsive Insurance  **Testing**  Unit, Integration, Load, Accessibility, Acceptance | | | | | |

Reading this time will help team members to know their respective tasks. The persons who will involve in this time box are GeekUp' founders, Project Manager, Team Leaders, UI/UX designers, Developers, and Tester. There will be specific tasks related to them. For the prioritization, the tasks with high effort percentage must be done. To have the expected outcomes all team will need to follow this time and work together to finish within the given timeline. The person in charge and persons involving are also described. According to this time-box, GeekUp’s founders will also need to involve in deployment stage.

The expected outcomes are described in the timebox. Estimate effort requirement for each outcome will be described. For the design diagrams will need 70%, and prototypes will need 30%. For the coding all the function will need 30% of the effort and machine learning will take 50%. CRUD will require 15% and for the responsive on any device will need 5%. For the testing 20% of effort will take for each.

Iterations will have in each task and reviewing steps will be involved after prototype, development and after testing. Some tasks can be work in parallel when the iterations take longer than framed timeline. For example, designing for frontend or prototyping can be developed in parallel with other coding stages. And most of the testing can be worked in parallel with other tasks like unit testing. This parallel development will be performed only if some tasks cannot be able to finish within their deadlines. And for the documentation, all the teams will make report for their tasks while implementing their tasks. At the final, project manager will make final documentation, user manual and presentations which will be delivered to founders later.

### Another agile approach might have been preferred for this project

There are various agile approaches and frameworks to develop the project. Most popular agile frameworks are Scrum, XP, DSDM, Kanban, Lean and Crystal. This project use DSDM framework and the other suitable agile approach for the current project is Scrum. Both DSDM and Scrum use timebox plans and both are similar. Like other agile approaches, Scrum also has iterative nature, collaboration, emphasis and adaptability to changing requirements. GeekUp wants to make cutting-edge application which will includes huge functions like machine learning and responsiveness of the application for any device with 5 years old age. By using Scrum those functions and features can be possible to make since Scrum prioritize user’s requirements and user’s involvements.

Scrum mainly prioritizes the functional requirements in their sprints. Each sprint typically takes 2-4 weeks, and have client involvements to reduce risks and time consuming. Another advantage of Scrum is that there are Daily stand-up meetings (Daily Scrum) to ensure about continuous communications and to reduce conflicts and risks. At the end of each sprint, there is a Sprint Review which means team will have to present working plans to stakeholders, and receive feedback for necessary adjustments. This is followed by a Sprint Retrospective where team reviews the areas for improvement.

Scrum emphasis on client involvement and feedback ensuring that the product will be able to deliver within the time limitation while maintaining the quality. The iterative nature of Scrum can make continuous improvement and incremented by each sprint. This make the app responsive and at the same time able to create advanced features like machine learning for identifying and various purpose use of items. Scrum also has testing stages in each sprint and make sure delivering items has high quality and meeting project deadlines.

Overall, Scrum is a popular framework of Agile like DSDM and known for its continuous improvement, and adaptability. The application required from GeekUp is a cutting-edge software which mean the application must be advanced and most recent technical features. Like using DSDM, Scrum will also able develop this app within the timeline. In addition, Scrum may be able to produce better result since it focuses on users’ requirement. Even though users’ involvement is necessary in Scrum, unlike other methodologies which will the delay the project due to user’s involvement, Scrum can able to deliver the product while insuring the quality of the product within given time. So overall, replacing DSDM with Scrum will able to make the project successful within six-month timeframe.

(Anon., n.d.) (Anon., n.d.)

**References List**

# References

# References

Anon., 2023. [Online]   
Available at: https://www.indeed.com/career-advice/career-development/dynamic-systems-development-method

Anon., n.d. [Online]   
Available at: https://airfocus.com/glossary/what-is-dynamic-systems-development-method/

Anon., n.d. [Online]   
Available at: https://www.lucidchart.com/blog/pros-and-cons-of-waterfall-methodology

Anon., n.d. [Online]   
Available at: https://www.agilebusiness.org/dsdm-project-framework/process.html

Anon., n.d. [Online]   
Available at: https://www.agilebusiness.org/dsdm-project-framework/principles.html

Anon., n.d. [Online]   
Available at: https://www.agilebusiness.org/dsdm-project-framework/product.html

Anon., n.d. [Online]   
Available at: https://www.brainscape.com/flashcards/chapter-8-dsdm-products-5984453/packs/8987147

Anon., n.d. [Online]   
Available at: https://www.lucidchart.com/pages/uml-sequence-diagram

Anon., n.d. [Online]   
Available at: https://pm.stackexchange.com/questions/14613/what-does-architectural-spike-mean

Anon., n.d. [Online]   
Available at: https://www.atlassian.com/work-management/project-management/proof-of-concept

Anon., n.d. [Online]   
Available at: https://www.nashtechglobal.com/our-thinking/insights/how-to-run-a-successful-proof-of-concept/#:~:text=What%20Is%20a%20Proof%20of,get%20products%20to%20market%20faster.

Anon., n.d. [Online]   
Available at: https://www.atlassian.com/agile/scrum

Anon., n.d. [Online]   
Available at: https://www.scrum.org/resources/what-scrum-module

Anon., n.d. *simpliaxis.* [Online]   
Available at: https://www.simpliaxis.com/resources/epic-vs-user-story

Anon., n.d. *visual paradigm.* [Online]   
Available at: https://www.visual-paradigm.com/guide/agile-software-development/what-is-user-story/#:~:text=A%20user%20story%20is%20a,simplified%20description%20of%20a%20requirement.

Beyond, O., 2023. [Online]   
Available at: https://www.one-beyond.com/pros-cons-waterfall-software-development/

Cohn, M., n.d. [Online]   
Available at: https://www.mountaingoatsoftware.com/blog/spikes#:~:text=In%20agile%20projects%2C%20a%20spike,approach%20to%20certain%20user%20stories.

Islam, M. N., n.d. [Online]   
Available at: https://nazrul.me/2011/11/22/dsdm-overview-phases-of-dsdm-life-cycle/

Malsam, W., 2023. [Online]   
Available at: https://www.projectmanager.com/blog/proof-of-concept-definition

Paredes, R., 2024. [Online]   
Available at: https://safetyculture.com/topics/waterfall-methodology/

Petit, C., 2024. [Online]   
Available at: https://www.appvizer.com/magazine/operations/project-management/dsdm

REHKOPF, M., n.d. *atlassian.* [Online]   
Available at: https://www.atlassian.com/agile/project-management/user-stories

REHKOPF, M., n.d. *atlassian.* [Online]   
Available at: https://www.atlassian.com/agile/project-management/epics

Sisictester, M., 2022. [Online]   
Available at: https://www.lambdatest.com/blog/agile-architecture-spikes-in-shift-left-bdd/

# Candidate Checklist



Please use the following checklist to ensure that your work is ready for submission.

Have you read the NCC Education documents 'What is Academic Misconduct? Guidance for Candidates' and 'Avoiding Plagiarism and Collusion: Guidance for Candidates' and ensured that you have acknowledge all the sources that you have used in your work?



Have you completed the 'Statement and Confirmation of Own Work' form and attached it to your assignment? **You must do this**.



Have you ensured that your work has not gone over or under the recommended word count by more than 10%?



Have you ensured that your work does not contain viruses and can be run directly?