Name: \_\_\_\_\_\_\_Lam Phan\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

This is an individual assignment.

1) Use Figure 2.1 – The map of the Volere Process. For each of the 9 activities in the Volere Process, list the inputs and outputs. Then, using your own words, describe the purpose of each activity. Two of the processes will not have specific definitions from the book, but you will be familiar with them from your Software Engineering class.

1.Project Blastoff: Determines the scope of the work to be studied. Describe what we are trying to achieve and get the project launched.

Input: Customer needs, strategic plan for product.

Output: major risks, initial costs, divide the work, project

2.Trawl for Requirements: gather the requirements for one business use case at a time.

Input: work, project, Domain Knowledge reusable requirements, wants and needs of the stake holders.

Output: potential requirement, requirement for experiment.

3. Prototype the Requirements: quick representation of a potential product, this present the user with some kind of simulation of the requirements.

Input: requirement for experiment

Output: potential requirement

4.Write the Requirements: the requirements are written in form for communicate purpose between stakeholders and the analysts. The team can ensure the required project is built only by writing them down.

Input: Requirement Template, Potential Requirements

Output: Formalized Requirement

5.Quality Gateway: test the prior to being included in Requirements Specification, and test each requirement for completeness, correctness, measurability, absence of ambiguity and several other attributes before allowing the requirement to be added to the specification. If the requirements are rejected then they are sent back the stakeholders who will take them back to the trawling activity for clarification and further explanation. If the requirements are accepted then they will be sent to Requirements Specification.

Input: Formalized Requirement, strategic plan for product.

Output: Requirements rejected or accepted.

6.Review the Specification: checks that there are no missing requirements, that all the requirements are consistent with one another, or any conflicts between the requirements have been resolved. The review confirms that the specification is really complete and suitable so that we can move on to the next stage of development. We could reassess the costs and risks of the project.

Input: Requirement Specification

Output: Risks and costs, missing requirements, review specifications

7.Design and Build: Once we have the requirements, and the architecture we can start to design and build the product.

Input: Requirements Specification, Architecture

Output: Product

8.Product use and Evolution: When the product is build, we can test the product and see if we can do anything to make the product better.

Input: Product

Output: New needs

9.Requirements Reuse: before starting on a new requirement project, we go through the specifications written for previous projects and look for potentially reusable material. We might find dozens of requirements we can reuse without alteration or are suitable for the new project.

Input: Reuse Library

Output: Domain Knowledge Reusable Requirement

2) What is the different between traditional and iterative requirement approaches?

Traditional approach is less effective and having more risk than iterative requirement approach. Traditional requirement focuses more on the documentation part while iterative having less focus on documentation. In terms of cost, iterative requirement approach is more efficient. In iterative requirements, we would have customer always in loop of development cycle (the Volere process for example), it means that more customer interaction than traditional approach which could help the product becomes better.

3) Consider Rabbit, Horse, and Elephant projects. For each, give project characteristics and an example of the types of requirements produced by each type of project.

Rabbit Projects are iterative and are the most agile of projects but has short lives. The projects include small number of stakeholders. The projects are those where circumstances allow the highest degree of agility. They gather requirements in small units and then implement the solution piecemeal, using the implementation to get feedback from the stakeholders. However, the feedback should not be used to find out what the stakeholders wanted in the first place. Rabbit projects usually include a small number of stakeholders. Example of the types of requirements produced: Functional Requirements.

Horse (fast, strong, medium life and dependable) Projects are the halfway house of agility they use as much agility as possible, but have constraints imposed by the project and the organization. Horse projects use iterative approach to requirements gathering that is the requirements for one unit of work are gather and then designers start to work on those requirements. The advantage is that while the requirements analysts are gathering the requirements for one business use case, the developers are busy building a solution for the requirements from the previous business use case. The sections on trawling, writing and the quality gateway will be of great interest to horse projects. Example of types of requirements produced: functional requirements.

Elephant (solid, strong, long life, and long memory) Projects are the least agile but are large, dependable and have long memories. Aspirations toward agility may be limited by the organization of the project. There would be a lot of stakeholders perform their daily duties Example of types of requirements produced: functional and non-functional requirements.