CHAPTER

6

APPLYING SYSTEMATIC AGILE DEVELOPMENT FOR HEALTHCARE SYSTEM

Mohd Adham Isa, Rohayanti Hassan, and Shahliza Abd. Halim

6.1 INTRODUCTION

The development of a large-scale healthcare system requires several requirements. Such requirements include software and system requirements, user experience, architecture, and database design. With a large number of requirements, the development of those systems becomes more complex, adding to the complexity of task alignment, high output expectations, and cost. Furthermore, developing a completely new system that inherits from the previous version makes system development more difficult. Many issues must be addressed, including database integration and migration, design consistency, and requirements traceability. If these issues are not handled properly, they may have an impact on the entire development process. As a result, selecting the appropriate software process model is critical to meet the expectations of the healthcare domain and its stakeholders (Aoyama, 1998).

The Agile development process is one of the development methodologies that promotes an iterative and incremental process where requirements and solutions evolve through collaboration between cross-functional teams (Mirza & Datta, 2019). It promotes adaptive planning,

evolutionary development, and delivery, and encourages rapid and flexible response to change. Due to the nature of the healthcare system's volatile requirements, adapting this methodology to it is extremely difficult. To meet these challenges, the Agile process, as well as its development environment, requires the use of appropriate and appropriate tools. There are numerous Agile support tools on the market today, with JIRA being one of the most powerful. JIRA tools support nearly every aspect of the Agile lifecycle, and a new plug-in supports DevOps features, which are particularly useful in an Agile environment.

This chapter is organised by firstly presenting related work in Section 6.2. Section 6.3 describes the execution planning while Section 6.4 explains the discussion of the project and finally, Section 6.5 of this chapter provides the conclusions drawn on the study.

6.2 CONCEPTS IN SOFWARE PROCESS MODEL

This section includes a review of the software process model as well as a description of the Agile development process in detail. In addition, a JIRA tool will be examined to determine its strengths and weaknesses.

6.2.1 Software Process Model

A variety of software process models are currently available for use in software development. Software process models like waterfall, V-Model, iteration, and Agile encourage a continuous improvement process that starts with requirement analysis and ends with maintenance. The process model's main goals are to systematically develop software using a proper procedure, a suitable method, and tools, and to track work progress at each phase to ensure that the assigned tasks are completed. The process model's main goals are to systematically develop software with a proper procedure, using a suitable method and tools, and to monitor work progress at each phase to ensure the assigned tasks are completed. The following are the fundamental requirements for software process models:

(1) The process should clearly outline which specific artifacts need to be created at each phase. For example, during the initial planning phase, it's important to develop project charters, scope documents, and stakeholder registers to fully grasp the project's objectives and key stakeholders. This detailed approach helps establish a structured process and guarantees the production of all required documentation to support each project phase. The process must specify what artifacts should be created for each phase.
(2) The process must closely match the project objectives and main goals to ensure consistency and successful results. It is essential that each step in the process directly supports the project's specific objectives and overall goals. This alignment helps in using resources efficiently and effectively, ultimately leading to the successful completion of the project.
(3) The process should have a separate lifecycle, enabling tasks to be linked to data from various sources both before and after their completion. This means that each stage of the process should be self-contained and identifiable by the specific data inputs that influenced its execution, ensuring transparency and accountability throughout each step.

6.2.2 Agile Development Process

The Agile process model encourages cross-functional team members, and software is frequently released in the form of software versions in order to gather quick feedback and gather requirements changes if necessary (Looks et al., 2021). Here are the fundamentals description of the Agile process model:

(1) Agile is a process methodology that encourages cross-functional collaboration, and the feedback from each sprint is critical information for the next sprint's improvement. This approach is built on two foundational pillars: