Chapter 3

by Muhammad Darlen Sava

Submission date: 01-Jul-2023 06:02AM (UTC+0800)

Submission ID: 2124924448

File name: Ch_3.docx (100.01K)

Word count: 948

Character count: 5061

CHAPTER 3

SYSTEM DEVELOPMENT METHODOLOGY

3.1 Introduction

One of the most critical aspects of developing a system. There are a lot of methods that have their benefits and drawbacks. This chapter will discuss the chosen process and the justification for it. The phases of the selected methodology will also look in-depth, and a brief description of the technology used in the system's development and its system requirements will follow.

3.2 Methodology Choice and Justification

For developing any system, there are several methodologies or what is usually called The Software Development Life Cycle (SDLC) to choose from. Some popular ones are Agile, Lean, Waterfall, Iterative, Spiral, and DevOps. They all differ in one way but still have the same goal: to aid development quickly and cost-effectively. Agile, specifically Agile Scrum, will be chosen as the methodology for developing the Student Academic Management System.

Agile methodology was chosen for several reasons, but the methodology will be explained in short. Agile methodology is one of the most popular methodologies used in development nowadays. The Agile SDLC development approach emphasizes teamwork, customer satisfaction, and development across several short cycles, or sprints, instead of a top-down process with a single set of steps (mondayblog, 2022).

In agile methodology, the development of the system will be separated into different iterations, usually called “sprints.” In these sprints, a task or function of the system has to be done based on the time set. Because of this, the development will have a rapid delivery of the system since each sprint, there is something to show to the stakeholders. It is also flexible because the stakeholders can add requirements anytime based on their feedback for each iteration/sprint. Thus, the satisfaction of stakeholders can also be seen for each iteration.

3.3 Phases of Chosen Methodology

In this part of the chapter, the phases of Agile methodology will be defined and explained one by one. The phases are as follows: Requirements, Design, Development, Testing, Deployment, and Feedback. The depiction of these phases is as in the picture below.

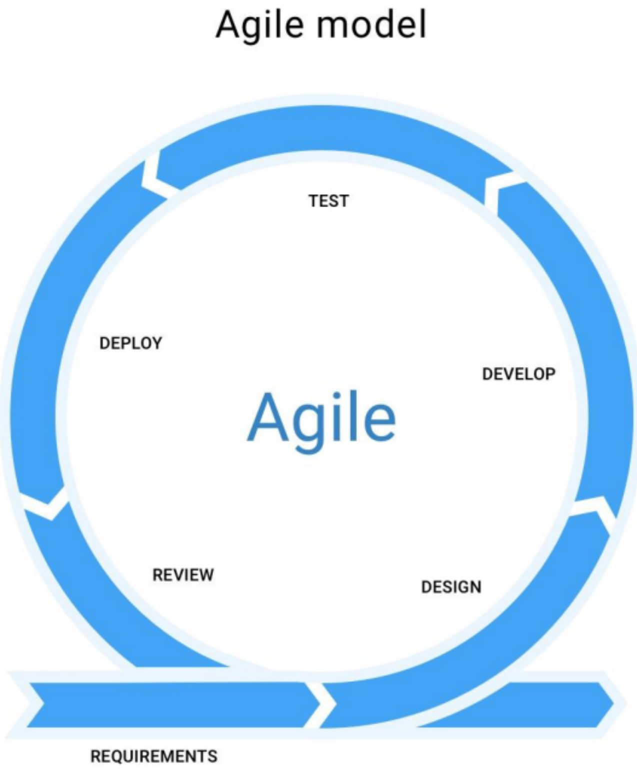

Figure 3.1: Agile methodology model

3.3.1 Requirements

In this phase, the developer must assemble a list of requirements from the stakeholders. This requirement can be acquired by interviewing or discussing it with the stakeholders. The stakeholders will then be responsible for the backlog and user stories; meanwhile, the developer must ensure they have enough information and continue to the next step. This phase is crucial because it can determine the future of the project cost in time or money.

3.3.2 Design

The focus in this phase is to define more in-depth details of the project from the requirements that have been set. These details include the project scope, objective, what tools will be used in the project, and the system requirement, just some examples. Discussing furthermore with the stakeholders is also considered reasonable at this phase.

3.3.3 Development

In the development phase, the developer must transform the requirements and design into a functional system from the previous two phases. This phase is the foundation of the Agile methodology. Thus, it is where most of the developer's attention is drawn.

3.3.4 Testing

As the name suggests, testing is the phase where the product/system is tested. During this phase, it is not possible to eliminate all bugs that reside in the system but still enough to make the system go into the deployment stage.

3.3.5 Deployment

After testing is done and the bugs that have been encountered patched, then the deployment of the system can be done. Even though the system has been deployed, changes or fixes regarding newly found bugs can still be done.

3.3.6 Feedback

After the product/system has been delivered to the stakeholders and is available to several users, collecting their feedback will be the last step of agile methodology. From the feedback gained, the developer can work to improve the system.

3.4 Technology Used Description

In this part, the development tools and technology used for the proposed system will be discussed briefly.

Table 3.1: Technology used and description

Technology	Description
Windows 10	Operating System
Google Drive	Documentation
Figma	Mockup interface
Draw.io	Design diagram
Visual Studio Code	Code Editor
Laravel	PHP framework
PHP	Programming Language
Amazon S3	Cloud Storage
Amazon EC2	Web Hosting
Vue.js	Javascript framework
Tailwind	CSS framework

PostgreSQL	Relational Database
pgAdmin	Database management tools

3.5 System Requirement Analysis

System requirements analysis will help specify the required software and hardware to run the proposed system with the user in mind. Subchapter 3.5.1 will show the software requirements, consisting of OS and web browser choices. Subchapter 3.5.2 shows the hardware requirements, consisting of each OS and its required processor, RAM and Hard Drive.

3.5.1 Software Requirements

1. Operating system: Windows 8 or later, macOS Sierra 10.12 or later, and 64-bit Ubuntu 14.04+(Linux).
2. Web Browser: Mozilla Firefox version 48 or later, Internet Explorer version 11 or later, Google Chrome version 54 or later

3.5.2 Hardware Requirements

1. Processor: Windows or Linux with Pentium 4 or newer processor that supports SSE2 or Mac computer with an Intel x86 or Apple silicon processor.
2. RAM: Windows or Linux with 512MB of RAM / 2GB of RAM for the 64-bit version or Mac computer with 512 MB of RAM.
3. Windows, Linux, or Mac with 200MB of hard drive space.

3.6 Chapter Summary

The chosen methodology and the justification for choosing it is the main key point in this chapter. But there is also an in-depth look at what the methodology has to offer in the form of its phases. The technology used is also described briefly with its description. Last but not least, the system requirement analysis is also stated, which includes both software and hardware requirements to run the system.

Chapter 3

ORIGINALITY REPORT

8%	3%	0%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to American University in the Emirates Student Paper	2%
2	Submitted to Queen Mary and Westfield College Student Paper	2%
3	Submitted to Southern Cross University Student Paper	1%
4	Submitted to American Public University System Student Paper	1%
5	su-plus.strathmore.edu Internet Source	1%
6	support.kaspersky.com Internet Source	1%

Exclude quotes	Off	Exclude matches	< 1%
Exclude bibliography	On		