Chapter 3

SYSTEM DEVELOPMENT METHODOLOGY

3.1. Introduction

Software development methodology refers to the structured processes involved while working on a project. The methodology's main goal is to provide a systematic approach to software development. Planning is very important when it comes to executing a project. It can save time and money at the same time. Software development is a complex process that involves proper planning, sufficient resources, a budget, and skilled developers to complete the project. It can even lead to software failure if proper planning for software development is chosen. As a result, it is important to have excellent software planning for the project.

To achieve and ensure the development process of the software and its related products, a software development process is described as a collection of processes, actions, activities, and controls Young (2013). There are several software development approaches. Some of them have shown effective results and are used extensively in developing software by developers. Some of the notable approaches are Waterfall method, Incremental method, Prototyping, Agile, and Rational Unified Process (RUP). These approaches consist of seven different phases which is known as the Software Development Life Cycle (SDLC). In this chapter, we'll be discussing the approach that'll be taken for implementing the project.

3.2. Methodology Choice and Justification

The system development methodology is a standardized process to perform all the necessary steps to analyze, design, implement, and maintain a system efficiently. There are many system development approaches that can be implemented for the successful planning of the project. According to the system requirements, Agile methodology seems to be the suitable software development approach that can be used for our project.

In Agile methodology, the project is broken down into small pieces of user functionalities. These functionalities can be organized by prioritizing their importance and setting a delivering time limit to complete them. They can be divided into 2–4-week cycles known as iterations or sprints. Before each cycle, the goal of the sprint needs to be specified. By making an analysis of the requirements, the functionalities will be divided and developed sequentially by giving highest priority to the most important ones. In this way, the development process will be easier and smooth for the developer to develop.

Agile methodology seems to be best suited for our project because of its phases. It is easier to implement the agile methodology as well. The use of sprints to complete tasks in Agile methodology makes it more suitable for the implementation of the project since it’ll help in saving time, money and dividing the tasks according to their priority. It has been seen from previous projects that by embracing the Agile Methodology, organizations can experience numerous benefits. They include more adaptability, a shorter time to market, higher customer satisfaction, better software quality, and lower project risks. Due to its endless benefits, it can be concluded that Agile methodology is the best suited for my project.

3.3. Phases of the Agile Methodology

Agile methodology follows a cyclical and iterative approach to implement the development of the software. It typically consists of six phases. The phases include the planning, Designing , development, testing, deployment and reviewing. The flow can be viewed from the figure below.

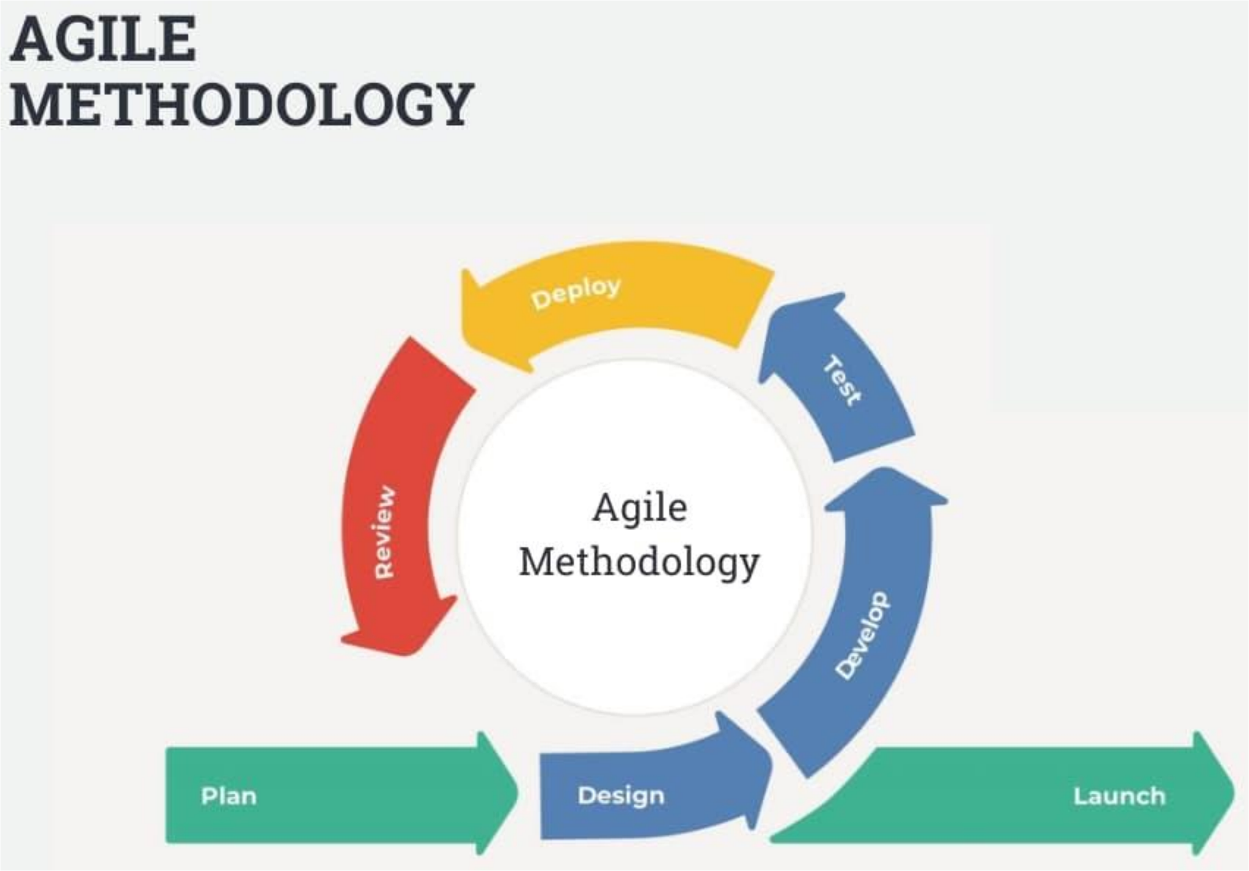

Figure 3.1: Workflows of Agile Methodology (source: interqualitybg.com)

3.3.1. Planning

Planning is the stage of the system where requirements set by the stakeholder are reviewed by the developers. The developer team reaches a conclusion with the client on how the software/system should look like. After that, the developer divides the various tasks that are needed to develop the system. In this stage, the priority and division of the tasks is done. A detailed planning for the iterations or sprints is made during the planning stage. The items for the development can be updated and viewed in the backlog. The backlog is updated during this phase as well. For each sprint and iteration, a goal is set to be completed by the end of that sprint. This includes the timeline for completing the tasks, the percentage of tasks that need to be completed and any other constraints or dependencies. During this phase, communication with the stakeholder is done profusely to meet the goals set for each sprint.

3.3.2. Design

Design is the stage where the system blueprint is designed based on the analysis of the requirements and results of the analysis. The prototype of the system is developed in the design stage. A detailed description of different features and operations of the system is carried out in this phase as well. The prototype will demonstrate the basic idea of how the application should look like. Developing the prototype is a key task that is to be completed in the design phase. It is also important to develop the prototype based on the requirements set by the stakeholders but also based on the flexibility of the developer. So, developing the prototype, validating the results of the analysis pinned during the planning stage are to be done during this phase.

3.3.3. Development

Development is the stage where the software development is carried out according to the plan by the developer. Developers will start developing the system using the programming language that has been chosen for developing the system. Different modules are created for developing the system in this phase. For example, the integration of database with the backend, the front-end of the system to interact with services or logic that has been developed by the back end. The software development process is carried out based on predetermined requirements and procedures during the previous phases.

3.3.4. Testing

In this stage, the software that has been developed is tested or checked. The testing of the software is very important to avoid software failure. It has been seen in the case of many software companies that due to the lack of proper testing, it has failed during the launch of the software. That is why this phase plays an essential role to avoid software failure and to ensure software success. Software quality is assured and maintained in this phase.

3.3.5. Deployment

Deployment is the stage where the software is ready to be placed in the market after getting clearance from the software quality control team. It comes after the testing phase since the quality and percentage of failure/success are determined in the testing phase. If the system developed matches the requirements, the software will be ready for deployment.

3.3.6. Review

Review is the stage where we get user feedback whether it's about bugs that weren't found during testing or about adding new features if needed. Review plays an essential role in software launching. User feedback is the most important part of the software development cycle since the system is basically built for the user. It is important to ensure a smooth user experience for the user to ensure excellent software review. From this stage, it could be a new software development life cycle to fix bugs, define an iterative development plan, or update features in future releases.

3.4. Technology Used Description

This section will briefly explain the required tools and technology used for the development of the proposed system which is the faculty of computing staff publication dashboard.

Table 3.1: Technology used

Technology	**Purpose**	**Type**
Windows	Operating System	Operating System
Google Drive	Documentation	Storage
Draw.io	Design	UML Diagram Editor
Figma		UI and UX Design
Visual Studio Code	Software Development	IDE

ReactJs	Stack	Javascript library
Material Ui		UI component library
Tailwind CSS		CSS framework
Django		Python framework
MongoDb		Database
Amazon Elastic Compute Cloud (Amazon EC2)	Network	Cloud Hosting

3.5. System Requirement Analysis

This section includes hardware and software, which are essential for designing, developing, and testing the system. The system will work properly and be compatible if the appropriate hardware and software tools are used.

3.5.1. Hardware Requirements

We will need a computer or laptop for documentation and, most significantly, for the development of the system.

Component	**Requirements**
Processor	Ryzen 7 3750H
RAM	8GB

Operating System	Windows 7 / 8 / 10 / 11 (32-bit or 64-bits)
HDD	100GB
Minimum Available Disk Space	20GB
Internet Connection	Active Internet Connection

3.5.2. Software Requirements

Software Requirements must be met for developing the system.

3.5.2.1. Vs Code

Visual Studio Code is a source-code editor that will be used as the main platform for coding the application in this project.

3.5.2.2. Microsoft Word

Microsoft Word is the ideal tool for documentation. It gives users powerful features to help them write more efficiently and effectively. The Microsoft Word software is required to complete the project’s documentation.

3.6. Chapter Summary

This chapter goes through the system methodology used to develop this project, which is the Agile method. This chapter also explores more into the Agile phases and how they work. Following that, this chapter explains the technology utilized in the development of the project, as well as the hardware's minimal requirements and a list of required software.