Chapter 8
Software Development Methodology for Cloud Computing and Its Impact

Chhabi Rani Panigrahi
C. V. Raman College of Engineering, India

Rajib Mall
Indian Institute of Technology Kharagpur, India

Bibudhendu Pati
C. V. Raman College of Engineering, India

ABSTRACT

This chapter emphasizes mainly on the software development methodology basically agile methods of software development in cloud computing platforms and its impact on software development processes. This chapter also covers the benefits of agile development methodology in cloud computing platform. Along with this all traditional software development phases are analyzed to discuss the differences between the traditional software development processes and software development in cloud computing environment. This chapter also includes a brief description of programming models such as MapReduce, BSPCloud, and Dryad etc. available in the literature to handle big data in SaaS cloud. Finally, we highlight the challenges and future scope of software development process in cloud computing environment.

INTRODUCTION

Software development process models are continuously evolving. They are affected by the type of the developed products, the nature of the project, and several other environmental factors. It is known that a particular software process development model cannot work well for all types of projects. The different constraints such as time, budget, and resources play also a significant role in deciding the best way to proceed in developing particular software. In this chapter, the main focus is given on how software

DOI: 10.4018/978-1-7998-3016-0.ch008

projects can be developed in cloud computing platform and the impacts of cloud computing on software development.

In terms of software product models, the widely known object oriented model is currently competed by possible alternatives. For cloud computing, service oriented model and service oriented architecture (SOA) seems as a possible successor of the object oriented paradigm. In SOA, design abstractions are built around the concept of services. Systems are decomposed based on the number and type of services they are offered to clients or users. Services are expected to be designed, implemented and deployed in very agile manners that can allow different types of users to call and use such services with the least amount of possible efforts. Some of the popular object orientated (OO) concepts such as abstraction and encapsulation can still be used and applied to good SOA design and such concepts are seen now as good software design principles rather than OO design principles. In abstraction, it is always important to decompose the system to the right level and number of services in the right granularity. Each service should have the right and relevant attributes, methods, associations, etc. that can help minimize its coupling with other services. Similarly, to apply encapsulation, services should be offered in ways that can relief the clients or users from any type of commitment or dependency in the implementation details of the service. This makes it easy to change the implementation details of such services with the least possible impact on clients. On the service side itself, encapsulation plays also an important role in separating the service representation or interface from its detail implementation. This can make it easy to update and change such service without impacting its interface or service clients. In SOA, focus is on Web services rather than Web applications. SOA helps in isolating service provider from user and provides generic services that are not intended for specific users and that are themselves unaware of the nature of use in the client side.

The rest of the chapter is organized as follows. First, the traditional software development process is revisited. Next, the cloud computing platform along with the benefits of software development with cloud platform is identified. Then the software process model for cloud platform is discussed. Subsequently, impacts of cloud computing platform on software development processes are identified and then different programming models for cloud computing platform are presented. Finally, future scope and challenges of software development with cloud platform are highlighted.

BACKGROUND

In this section, traditional method of software development is summarized. Subsequently, the cloud computing platform has been described and the benefits of software development on cloud computing platform are also highlighted.

Traditional Software Development Life Cycle

Software Development Life Cycle (SDLC) is a process followed in software industry to produce high quality software (Roger et al., 2014). The SDLC is aimed at meeting customer expectations, complete the project within times and cost budgets. Traditional software development requires different hardware technologies and the development process involves different stakeholders. The quality and success of a software project depends on various factors such as time, budget, efficiency, maintainability, dependability, and usability of the product (Farooq & Quadri, 2012). Since 1968, software engineering evolves