



Ain Shams University
Faculty of Computer & Information Sciences
Information Systems Department

Testing as a Service (TaaS) in Cloud Computing Environment

A thesis submitted in partial fulfilment of the requirements for the degree of
Master of Science

By

Amira Ali Abdel Aal Elwaan

B.Sc. in Computer and Information Sciences,
Teaching Assistant at Information System Department,
Faculty of Computer and Information Sciences,
Ain Shams University

Under Supervision of

Prof. Nagwa Lotfy Badr

Professor,
Information System Department,
Faculty of Computer and Information Sciences,
Ain Shams University

Dr. Huda Amin Maghawry

Lecturer,
Information System Department,
Faculty of Computer and Information Sciences,
Ain Shams University

Cairo 2018

Acknowledgements

I would like to thank Allah, the Most Gracious and Merciful, who inspired me and blessed me. Without the help and inspiration of Allah, I would not be able to carry on this work.

It is my honour to thank Prof. Nagwa Badr for supervising this research work. Her endless patience, support and encouragement, especially in the hard times fostered me to have continuous progress. I would like also to thank Dr. Huda Amin Maghawry for supervising this work. Her continuous help and practical support and feedback, during all stages of my work, allowed me to have tangible research outcomes.

I would like to thank my great parents who have devoted themselves to support me in my whole life, especially my mother for her endless support and patience during this work, without her encouragement and support, this work would never be completed. Whatever I do, I will never manage to return her favour.

Abstract

Software testing is a fundamental step during software development life cycle to ensure the software quality and functionality. Software testing requires many resources, infrastructure, tools, and skilful testers. Recently, software systems become more complex and large in size. This leads to increase in the software testing cost, time, and effort as well as increasing need to find adequate testing techniques and approaches to test application under test (AUT) thoroughly. Cloud computing is an emerging technology that can be used in the software engineering field to overcome defects of the traditional testing approaches by using cloud computing resources. As a result, testing-as-a-service (TaaS) is introduced as a service model that conducts all testing activities in a fully automated manner.

There are various types of software testing. In this thesis, we focus on the following two types of software testing: i) GUI (Graphical User Interface) testing for mobile applications and ii) performance testing for web applications. GUI testing is a type of mobile applications testing performed to ensure the proper functionality of the GUI components of the AUT. Testing mobile applications gain much attention due to the widespread of smartphones and the tremendous number of mobile applications development. It is essential to test mobile applications before being released for the public use. Typically, GUI testing requires a lot of effort and time whether manual or automatic. Thus, migrating GUI testing of mobile applications to cloud environment and leveraging all cloud resources and infrastructure is considered the best solution

Performance testing is a type of software testing which is conducted to show how web application behaves under a certain workload. An efficient approach to carry out performance testing is introduced to exhaustively check the behaviour of the web application before launching it.

In this thesis, a system for software testing based on TaaS architecture is proposed. The proposed system performs GUI testing for Android-based applications and performance testing for web applications, where all testing activities including automatic test case generation and simultaneous test execution on multiple virtual nodes are carried out. Approaches for GUI testing for Android applications and performance testing for web application are introduced. The proposed testing approaches have high coverage and low test execution time due to simultaneous execution on multiple virtual nodes in a cloud based environment. Therefore, the proposed system reduces testing time and meets fast time-to market constraint of software applications. Moreover, the proposed system addresses many issues such as maximizing resource utilization, continuous monitoring to ensure system reliability and applying fault-tolerance approach to handle occurrence of any failure.

Publications

1. Amira Ali, Nagwa Badr "Performance Testing as a Service for Web Applications."
In Intelligent Computing and Information Systems (ICICIS), IEEE, 2015.
2. Amira Ali, Huda Amin Maghawry and Nagwa Badr "Automated Parallel GUI Testing as a Service for Mobile Applications" In Journal of Software: Evolution and Process,
Impact Factor: 1.167, 2018.

Contents

CHAPTER ONE: INTRODUCTION	1
1.1 GUI TESTING FOR MOBILE APPLICATIONS	1
1.2 PERFORMANCE TESTING FOR WEB APPLICATIONS.....	3
1.3 MIGRATION OF SOFTWARE TESTING TO CLOUD ENVIRONMENT	3
1.4 THESIS OUTLINE.....	5
CHAPTER TWO: RELATED WORK	6
2.1 TAAS	6
2.2 MOBILE APPLICATIONS TESTING IN THE CLOUD ENVIRONMENT ...	8
2.3 MODEL BASED TESTING FOR MOBILE APPLICATIONS TESTING.....	11
2.4 PERFORMANCE TAAS	12
CHAPTER THREE: THE PROPOSED ARCHITECTURE.....	13
3.1 OVERVIEW OF THE PROPOSED TAAS FRAMEWORK	13
3.2 THE SERVICE MANAGEMENT AND TESTING LAYER	17
3.2.1 The Approach of GUI Testing for Mobile Applications	23
3.2.1 The Approach of Performance Testing for Web Applications	33
CHAPTER FOUR: EXPERIMENTAL RESULTS	38
CHAPTER FIVE: CONCLUSION AND FUTURE WORK	45
APPENDIX A: APPLICATIONS UNDER TEST ACTIVITY DIAGRAM	46
REFERENCES.....	48

List of Figures

Figure 3. 1: The proposed TaaS Architecture	14
Figure 3 .2:The Proposed Architecture Component Diagram	16
Figure 3.3: Allocation Plan Generation Approach	20
Figure 3.4: Test Task Execution Module for Mobile Application GUI Testing	23
Figure 3.5: Automated GUI Test Case Generation Approach	26
Figure 3.6: GUI Testing Approach Activity Diagram.....	32
Figure 3.7: Automated Performance Test Case Generation Approach.....	34
Figure 3.8: Performance Test Script Generation Method.....	36
Figure 4.3: Overall Completion Time Comparison for Several Test Request.	43
Figure A.1: Simple Notepad Application Activity Diagram.	46
Figure A.2: Clock Application Activity Diagram.....	47

List of Tables

Table 3.1: VM Pool	19
Table 4.1: Comparison of GUI Test Case Generation Approaches for Simple Notepad Application.	40
Table 4.2: Comparison of GUI Test Case Generation Approaches for Clock Application	41
Table 4.3: Hit Value in case of FCFS and The Proposed Scheduler Approach	44

List of abbreviations

APK: Android Application Package

AUT: Application under Test

FCFS: First Come First Served

GUI: Graphical User Interface

IaaS: Infrastructure as a Service

LCS: Longest Common Subsequence

MBT: Model Based Testing

SUT: Software under Test

TaaS: Testing as a Service

UML: Unified Modelling Language

VM: Virtual Machine

Chapter One: Introduction

Software testing is an essential activity that is undertaken to ensure the software quality and functionality. Testing software requires enough infrastructure, resources, and budget to check the AUT [1]. There are various types of software testing such as: unit testing, integration testing, compatibility testing, security testing, performance testing, GUI testing, and regression testing [2]. This thesis focuses on two types of testing: GUI testing for mobile applications and performance testing for web applications.

1.1 GUI Testing for Mobile Applications

Testing mobile applications is defined as the process of testing the quality of the software applications developed for smartphones [3]. Nowadays, mobile applications have gained much popularity due to the continuous increase in the number of smartphone users. Thousands of mobile applications are developed daily and these applications are available in many stores giving users plenty of options to choose from them [4]. However, the majority of mobile applications are developed within restricted resources, budget and limited time due to time-to-market pressure. This severely affects the quality of the mobile applications development [5]. Therefore, testing mobile applications adequately within a constricted time and cost becomes a big challenge that will affect the continuity of the mobile applications

development [6]. Mobile applications testing requires much time, cost, resources and skilful tester and tools for writing and executing test cases [7], [8], [9].

GUI testing for mobile applications, as a consolidated type of testing, aims to ensure that GUI components of the mobile applications meet their functional requirements and achieve high quality standard. GUI test cases are composed of sequences of input events to ensure that GUI components conform to their required specifications. Model-Based Testing (MBT) is defined as a type of testing methodology that derives test cases from different models. Models has two types: system requirements models (black box testing) and models constructed from source code (white box testing) [10], [11].

GUI test cases generation requires accurate modelling to the user behaviours to imitate the interactions between users of mobile applications and the GUI controls. In this Thesis, test case generation approach based on UML activity diagram is introduced. The Unified Modelling Language (UML) is a widely used modelling language that effectively model different software artifacts [12]. UML activity diagram is used to model software workflows in terms of a stepwise activities and actions. Activity diagram becomes a good candidate for representing user behaviour which allows automated GUI mobile applications testing [13]. In practice, GUI testing takes too much time which may lead to delay-to-market for mobile applications. That's why it is essential to develop automated cost-effective testing techniques as an alternative. Due to the rapid growth of Android applications [14], the proposed system focuses on GUI testing of Android applications in a cloud-based environment. Besides,

Android is an open source platform which facilitates availability of Android applications for our experiments.

1.2 Performance Testing for Web Applications

Web-based applications refer to applications developed to run using internet browser and are accessed by users over a network. It is important to test web applications exhaustively before launching them to check their reliability [15]. Performance testing is defined as a type of testing that determine how a system behaves under different workload. The workload is the expected number of concurrent users that perform a specific transaction on AUT within a certain duration [16]. The purpose of performance testing for web application is to execute test scripts that simulate real user behaviour at different loads to determine how the AUT behaves in terms of throughput (i.e. Number of executed requests per second) and response time (i.e. time taken to execute certain request) [17], [18].

1.3 Migration of Software Testing to Cloud Environment

Software testing is a challenging activity for many software systems, especially for large-scale systems. The amount of tests cases can range from a few hundred to several thousands. This requires significant computing resources and long execution time for generating and executing test cases. That's why software testing is considered a labour-intensive process

that requires much cost, time, and resources in addition to skilful tester and tools for generating and executing test cases.

In the last few years, cloud computing has gained significant attention. Cloud computing is defined by the National Institute of Standards and Technology (NIST) as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [19]. Cloud computing is the next revolution of distributed computing paradigm which can support on-demand service sharing with higher level of flexibility and dynamic scalability to sharable computing resources which are configurable [20]. Cloud computing as an emerging technology can be used in the field of software engineering to provide cloud testing. Thus, migrating software testing to the cloud environment and leveraging cloud unlimited storage, resources (hardware and software) and scalable infrastructure is the best solution to overcome difficulties of the traditional testing strategies. [21] [22], [23], [24]. In the cloud based environment, there is no need to set up test environment resources as well as providing a readily on demand virtual test environment to meet fast time-to-market constraint of the software applications. As a result, Testing as a Service (TaaS) is introduced as a service model that automatically performs all testing activities including automatic test cases generation, test execution and test report generation. [25], [26], [27]. The AUT is submitted to TaaS platform. The TaaS platform automatically accomplishes all the test activities, generates detailed test report and submits the test report to the user. The proposed

system generates test cases with high coverage as well as leveraging cloud environment resources that allow simultaneous test execution on multiple virtual nodes.

1.4 Thesis Outline

Chapter 2 highlights the most relevant related work for software testing in a cloud-based environment. Section 2.1 survey the most recent researches related to TaaS. Section 2.2 discusses mobile applications testing in cloud based environment. Section 2.3 presents researches related to performance TaaS. Section 2.4 presents relevant researches related to the feasibility of using MBT (Model Based Testing) in mobile application testing.

Chapter 3 introduces the proposed system architecture and a detailed description of the functionality of its modules. Section 3.1 introduces overall architecture of the proposed system and defines its layers. Then section 3.2 describes service management and testing layer and discusses in detail functionality of each module. As well as introducing testing approaches to carry out performance testing and GUI testing in sections 3.2.1 and 3.2.2 respectively.

Chapter 4 presents experimental results of the proposed framework. Finally, Chapter 5 presents conclusion and proposes some ideas for future work.

Chapter Two: Related Work

Recently, several researches concerned with studying software testing in a cloud-based environment from different perspectives. In this chapter, the recent state of the art researches related to software testing in the cloud environment are discussed.

2.1 TaaS

There are many researchers provide insightful discussion to TaaS in terms of definition, benefits, challenges and advantages of cloud testing over traditional testing as in [18], [22-24], [26], [28-31]. While [32] and [33] focus on defining the decision factors for migration testing activities to the cloud and whether migration will be beneficial for organization or not.

In addition, there are many studies introduce a prototype architecture to TaaS. For example, Lian yu et al. [34] presented a prototype implementation to TaaS which is composed of 5 layers: 1) test tenant and test service contributor layer, 2) test task management layer, 3) test resource management layer, 4) test layer and 5) database layer. Besides, it addresses many issues such as: clustering tenants' requests according to the similarity in test environment required, scheduling clustered tasks and monitoring resources. However, the techniques used to carry out each test activity involved in the testing process are not discussed. All testing activities such as: test case generation, test execution and result visualization are provided as a services. The test service contributors