Research Issues for Software Testing in the Cloud

Leah Muthoni Riungu, Ossi Taipale, Kari Smolander
Software Engineering Laboratory
Lappeenranta University of Technology
Lappeenranta, Finland
leah.riungu@lut.fi, ossi.taipale@lut.fi, kari.smolander@lut.fi

Abstract

Cloud computing is causing a paradigm shift in the provision and use of computing services; away from the traditional desktop form to online services. This implies that the manner in which these computing services are tested should also change. This paper discusses the research issues that cloud computing imposes on software testing. These issues were gathered from interviews with industry practitioners from eleven software organizations. The interviews were analyzed using qualitative grounded theory method. Findings of the study were compared with existing literature. The research issues were categorized according to application, management, legal and financial issues. The issues are discussed with the intention of soliciting academic research on software testing in the cloud. By addressing these issues, researchers can offer reliable recommendation for testing vendors and customers.

1. Introduction

Cloud computing is the latest major trend in the IT industry. Market research and analysis suggests that in 2008, the worldwide expenditure on cloud computing was $16bn and it would increase to $42bn/year by 2012 [1]. Leading companies such as Google and IBM have a vested interest in cloud computing, demonstrated by efforts to strengthen their cloud computing departments [2]. However, these computing giants are not the only ones interested in cloud computing; the general masses of people also want to understand it. As it can be seen in Figure 1, the quest for knowledge about cloud computing has continually grown since 2007.

Software testing is a time-consuming process. It is important for ensuring the validity and functionality of a software product. Software products and services are constantly evolving due to changing technologies and market requirements. Cloud computing is one such technology that is changing the way software is produced and consumed – mainly from the traditional desktop form to online software services. As software products and services continue to experience this shift, it implies that the methods, tools and concepts to test them should also change [4]. Therefore, it is important to highlight cloud computing research issues from a testing perspective.

We found two comprehensive reviews of academic research in cloud computing. The first one presented a detailed evaluation of peer-reviewed academic research on cloud computing [15]. It considered the technical aspects of implementing clouds and highlighted the academic research agenda for cloud computing. Some of the issues discussed include security, legal, privacy and interoperability problems. In addition to these issues, the second review [29], discussed the research issues for enterprise cloud computing – mainly focusing on the organizational changes and financial implications brought about by cloud computing. Both reviews comprehensively demonstrated the need for
research in cloud computing from various viewpoints. However, little has previously been written about research issues for software testing in cloud computing, hence the focus of our paper.

In our earlier study [5], we acknowledged that cloud computing was going to be a central platform for online testing and service delivery and also listed some research issues. Testing in the cloud is seen as an arena of cloud computing that is easy to break into, where a number of commercial players such as IBM, Skytap and Utest have already made their entry [26]. In addition, the cloud is said to avail a wide variety of testing environments at a cheaper cost [27]. In this paper, our objective is to discuss the research issues of testing in the cloud. Therefore, we solicit an academic research agenda that puts testing within the fast growing cloud computing context and hope that our discussion would be a resource to researchers interested this new area.

Section 2 presents a background on cloud computing and examples of testing in the cloud. The research process is described in section 3, followed by results in section 4. Section 5 ends the paper with discussion and conclusions.

2. Cloud computing

In an attempt to provide an integrated and harmonic description of cloud computing, Vaquero, et al. [6] carefully considered already existing descriptions and proposed a definition with an emphasis on scalability, pay-per-use utility model and virtualization i.e. servers, storage space, network and/or operating systems hosted in the cloud. Major companies such as Sun Microsystems [7] and IBM [8] have also attempted to define cloud computing, paying attention to the benefits it brings to the enterprise. The US National Institute of Standards and Technology (NIST) has developed a definition that encompasses various views on cloud computing. NIST defines cloud computing as; “A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, application and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [9].

The definition further describes cloud computing as having; (i) five essential characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service; (ii) three service models: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS); and (iii) four deployment models: private clouds, community clouds, public clouds and hybrid clouds.

The essential characteristics can help in distinguishing whether a particular offer is truly cloud computing or not. These essential characteristics of the cloud are:

• On-demand self-service: computing services such as servers, storage and virtual machines can be acquired automatically if and when needed by a user without human interaction with the cloud service providers.
• Broad network access: computing services can be accessed over a network using different devices e.g. laptops and mobile phones.
• Resource pooling: computing services are pooled to be accessed and used by many users through a multi-tenant approach that enables the services to be allocated as per the demand of the users.
• Rapid elasticity: computing services are unlimited and can quickly be scaled in and out as required by the customers.
• Measured service: appropriate metrics like storage, active user accounts, bandwidth etc are used to measure the usage of the computing services. This provides transparency of the utilized service to the cloud service provider and customer.

These characteristics hold true for all clouds. There are many cloud providers and each of them provides cloud services at different layers of the cloud which are referred to as service models in the NIST definition. The service models include:

a) Software as a Service (SaaS): Customers are able to make use of applications that are running in cloud environments, usually by means of a web browser. The customer does not have the rights to control or manage the cloud infrastructure that is running the software. An example of a SaaS model in cloud computing is Google docs, where the user can edit and manage documents online [10].

b) Platform as a Service (PaaS): In this case, customers are provided with programming and execution environments through which they can run and access applications of their own choice. Similar to the SaaS model, customers cannot control the underlying cloud infrastructure but have control over the applications they create and to a certain degree, configuration settings of the hosting environment. An example of a PaaS is Google app engine [11].

c) Infrastructure as a Service (IaaS): This is where computing services such as storage, processing and networks are provided by the IaaS provider for the customers to deploy and run their applications. IaaS gives a customer the flexibility to control and run
software over the computing environment. A popular example is Amazon’s EC2 [12].

In addition to the three service models mentioned in the NIST definition, another proposal for a cloud computing stack includes two additional service layers namely Human as a Service (HaaS) and supporting services [13]. These layers could also be viewed as service models. Human as a Service (HaaS) incorporates crowdsourcing so that a crowd of people can use the cloud technology from different geographical places and work together to complete a task requiring effort from a large group of people. A good example of crowdsourcing is uTest, which provides software testing solutions to its customers through on-demand access to a community of professional testers [14]. Another form of HaaS is Information Aggregation Services (IAS) that deals with generating a unified figure that represents a popular opinion of the crowd [13].

The service models in the NIST definition as described above are deployed in the clouds. Different types of clouds exist and are referred to as deployment models in the NIST definition. These models include:

- Private cloud: The cloud infrastructure operates mainly to serve one organization only and may be managed by the organization itself or an external cloud provider.
- Community cloud: Several organizations share the cloud infrastructure and provide services to a specific community that has similar needs.
- Public cloud: The cloud infrastructure is available for use by anyone and is usually owned by a large organization.
- Hybrid cloud: This is a combination of two or all of the above mentioned clouds.

Cloud computing is evolving and the NIST definition above is expected to change over time as new breakthroughs are made [9]. We share a similar opinion as Sriram and Khajeh-Hosseini [15], that the NIST definition seems to encompass majority of cloud computing features. Thus, we are in favor of this definition for our current and future research. In addition to the NIST service models, we agree with the additional Human as a Service (HaaS) model introduced in [13]. Although HaaS may not necessarily be viewed as a service model by the NIST team, we believe that cloud computing will also enable several services to be delivered through crowdsourced efforts.

Cloud computing seems to be driving a shift towards availing everything-as-a-service (XaaS) [16]. In that sense, it seems to extend service oriented architecture. Software testing has gradually matured into an offering that can be provided as a service by independent testing organizations. Following from the SaaS, PaaS and IaaS service models, software testing can also use cloud computing capabilities to be provided as a cloud service i.e. software testing as a service (STaaS). We elaborate more on STaaS in the next section.

2.1 Testing in the cloud

Software testing as a service (STaaS) is defined as “a model of software testing used to test an application as a service provided to customers across the internet” [17]. It enables daily operation, maintenance and testing support through web-based browsers, testing frameworks and servers.

Testing in the cloud or cloud testing can have three facets (1) the system or application under test is accessible online. This might be SaaS software or non-SaaS software. In addition, this includes testing at different test levels e.g. performance testing; (2) testing infrastructure and platforms are hosted across different deployment models of the cloud i.e. public, community, private or hybrid clouds; (3) testing of the cloud itself. Cloud environments should be tested and measured for their performance, availability, security and scalability in order to support efficient delivery of services [18].

**Figure 2: Facets of testing in the cloud**

Examples of testing services that are already being deployed in the cloud are presented below.

Cloud computing enables large-scale performance testing. The authors in [19] present a case study where they tested a Network Management System (NMS) for a Voice over IP (VoIP) telephony switching system. They preferred cloud-based testing against running a simulator or using real elements because it provided an affordable and scalable implementation. The test was run on Amazon’s cloud, and the results revealed improvement in the software that had otherwise not been achieved through other means. The total cost for
running the entire test was $120 – further illustrating the cost effectiveness of cloud testing.

Cloud9 is a web service that enables parallel symbolic execution of computer clusters operating on public cloud infrastructures such as Amazon EC2 as well as on clusters running cloud software like Eucalyptus [20]. Users of the Cloud9 testing service are charged according to test goal specifications that they provide, so that the total cost of service is proportional to the program size.

Systems that are highly dependable should be tested thoroughly for parallel and distributed processing as well as fault tolerance capabilities. To address software testing problems associated with such systems, D-cloud is a large-scale software testing environment that uses cloud computing technology i.e. Eucalyptus for cloud management and QEMU - an open source machine emulator - for virtualization [21]. D-cloud allows automatic configuration, testing with fault injection along the test case descriptions.

The York Extensible Testing Infrastructure (YETI) is an automated random testing tool with the ability to test programs written in different programming languages [22]. However, YETI faces performance and security issues. After running it in the cloud, results showed that cloud computing helped to solve the performance issue when testing large programs. The security problem was solved by the ability to run tests on clean virtual machines.

Another testing service in the cloud is the autonomic self-testing (AST) and test support as-a-service (TSaaS) that makes use of processing capabilities provided by the cloud to improve self-testing processes [23]. With regard to virtualization capabilities in the cloud, an example is the virtualization-aware automated testing service (VATS) [24], which enhances testing services in the cloud by executing tests and managing virtual infrastructure. In addition, the remote network labs (RNL), is an on-demand network cloud that enables users to build virtual test laboratories [25].

As a summary, cloud computing seems to support software testing by availing computing power and virtualization capabilities that would otherwise be impossible or too expensive to attain. However, as it continues to create new opportunities for providers and customers of cloud services, it also brings along issues that need to be addressed.

3. Research Process

We selected the grounded theory [28] approach so as to discover the research issues for software testing in the cloud. Seaman [36] recommends grounded theory as a way to identify new theories and concepts – making it a valid choice for research in new areas without established theories. Grounded theory as an approach aims at the creation of a theory from the collected and analyzed data.

In our study, we interviewed eleven organizations. The selection of interviewees was guided by theoretical sampling. This means that the selected organizations represent polar points and bring variation to the interview data hence strengthening the findings of the study [28]. Snowball sampling was also used, in which one interviewee was a referral from another interviewee. Table 1 shows details about the interviewees and organizations. Theme-based interviews were conducted with each interviewee, lasting about 40 to 60 minutes each. All interviews were recorded and transcribed for analysis. Interview questions are available at http://www2.it.lut.fi/project/MASTO/.

We analyzed the data following the three step coding procedures of the grounded theory – open coding, axial coding and selective coding [28].

<table>
<thead>
<tr>
<th>Organization</th>
<th>Business</th>
<th>Provider/Customer</th>
<th>Interviewee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Develops accounting software for small businesses</td>
<td>Customer</td>
<td>Software manager</td>
</tr>
<tr>
<td>2</td>
<td>Information, logistics and mail communication</td>
<td>Customer</td>
<td>Quality and processes manager</td>
</tr>
<tr>
<td>3</td>
<td>Service development in banking</td>
<td>Customer</td>
<td>Program manager</td>
</tr>
<tr>
<td>4</td>
<td>Develops software for the energy market</td>
<td>Customer</td>
<td>Chief Technology Officer</td>
</tr>
<tr>
<td>5</td>
<td>Develops systems for work time data collection</td>
<td>Customer</td>
<td>Project manager/test engineer</td>
</tr>
<tr>
<td>6</td>
<td>Testing services and consultancy</td>
<td>Provider</td>
<td>Testing and methodologies director</td>
</tr>
<tr>
<td>7</td>
<td>Performance testing services</td>
<td>Provider</td>
<td>Performance testing unit leader</td>
</tr>
<tr>
<td>8</td>
<td>Functional testing and test management</td>
<td>Provider</td>
<td>Functional testing unit leader</td>
</tr>
<tr>
<td>9</td>
<td>Testing and quality services</td>
<td>Provider</td>
<td>Vice President</td>
</tr>
<tr>
<td>10</td>
<td>Testing services</td>
<td>Provider</td>
<td>Chief Executive Officer</td>
</tr>
<tr>
<td>11</td>
<td>Testing services</td>
<td>Provider</td>
<td>Testing manager/quality adviser</td>
</tr>
</tbody>
</table>
coding, we deduced the initial categories guided by a predefined interview question i.e. “In your opinion, is there a specific area that you think should be the focus of STaaS research?”

In axial coding, we examined the categories to identify similarities, relations and causal conditions among them. The result of axial coding was grouping of the initial categories into three categories, which have various relationships with each other, as shown in Figure 3.

The objective of selective coding is to define a central theory, which in essence, should be related to other categories [28]. Sometimes, a central category cannot be deduced because each category tells part of the theory. Should that happen, as in our case, a conceptual idea related to all categories should be developed [28]. In this paper we concentrate only on the categorization of the issues, which is mainly the result of axial coding.

**Figure 3: Categories of research issues**

4. Results

We discuss the research results from three perspectives; application issues, management issues, and legal and financial issues. In support of our results we also compare them with issues gathered from literature.

4.1. Application issues

4.1.1. Applications suitable for online software testing. Testing vendors and customers interested in testing in the cloud would want to be aware of the type of applications that can readily be tested in the cloud. This would help them to better prepare for the migration of testing to the cloud.

“It’s good to try to find the exact business areas that STaaS would be best approached...how you actually do it, and implement it” (Testing and methodologies manager, organization 6)

“...there is growing business where you have web services and web based applications...how to handle testing them over the internet.” (Quality and process manager, organization 2)

Parveen and Tilley in [30] raised a vital research question: “When is it safe to migrate testing to the cloud?” Based on their experience in working with JUnit test cases for different environments such as Hadoop and Spring framework, they addressed the question using two viewpoints: the characteristics of an application under test and the types of testing done on the application. Some characteristics to consider are test case dependency, the operating environment within which to carry out testing and the ability of an application’s interface to be programmed. The types of testing identified to be appropriate for testing in the cloud are unit testing, high volume automated testing and performance testing.

The application under test might be SaaS software that uses the cloud as its production environment or it could be non-SaaS software. How does this nature of an application affect the way it is tested? For example, in the case of non-SaaS software, researchers would investigate how switching between the external production environment and the cloud testing environment affects the overall testing work, the bottlenecks involved and how to tackle them.

Organizations such as banks, financial institutions and organizations dealing with mission critical systems may be reluctant to leverage cloud computing capabilities for their testing needs. However, due to the data-intensive nature of their systems, they stand a chance to benefit from cloud computing infrastructure. What is the solution for these organizations? Could private clouds offer the security and reliability required for such systems?

4.1.2. Readymade online performance testing package for any customer. SOASTA, a cloud testing provider, has for almost two years been providing performance testing of web applications in the cloud [31]. However, many other applications as well as the cloud itself need to be tested for various performance attributes e.g. response time, speed and throughput. Cloud computing provides tremendous computing power for performance testing e.g. by enabling simulation of load on demand [32].

This brings up the issue of developing a performance testing package that suits for different customer needs. “I want to create a ready package that I can ship to any customer, maybe this can be a forest industry customer or a telecom customer...all of those that have IT systems, and test them.” (Performance testing unit leader, organization 7)
An interesting research issue would be to evaluate how and whether certain testing infrastructure in the cloud helps to meet a specific performance attribute. A suitable approach to address this issue is by running a test bed which would encourage researchers to explore different aspects of performance testing, reflect on the experiences encountered in the process and thereafter give recommendations based on the results.

A one-size-fits-all performance testing package may sound ideal. However, due to different nature and characteristics of applications under test, a more realistic and practical approach might be to come up with performance testing packages addressing similar types of applications, for example, performance testing of a web application is different from performance testing of a mobile application.

4.1.3. Quality checks for applications that have been tested in the cloud. The current day’s advances in technology have influenced a high demand for quality of service by end users. It is not enough to carry out testing in the cloud – we need to guarantee the reliability of the testing process as well as that of the tested application.

“I think that if somebody provides software as a service, the service provider should look into how they deliver this service, and how they can take care of the quality.” (Performance testing unit leader, organization 7)

This highlights an important research issue of finding ways to validate the quality of cloud tested applications at all levels. A possible approach would be for researchers to develop a testing methodology to verify and validate the quality of overall testing in the cloud. Quality is sometimes a highly subjective attribute, varying due to different end-user expectations.

End users have quality of service requirements that service providers ought to meet. Another approach would be to incorporate a strong emphasis on the user experience as a way to validate the quality of the application. This brings a vital issue for researchers – to understand and possibly define the success factors and metrics that determine the quality of online testing.

4.1.4. Harmonization of the test processes across multiple players. With the growing number of cloud providers, customers have a wide range of choice from where to get cloud services. Assuming a customer chooses to acquire the services from more than one cloud testing provider, it would be efficient to have some methods, tools and facilities for monitoring and managing the software testing processes from the different providers in an “all-in-one” fashion.

“...how is the system under test made accessible to the tester and how are the tools and facilities for testing made available to different players participating in the testing process.” (CEO, organization 10)

In addition, due to the elastic nature of the cloud, a unified monitoring system would keep track of resource availability and give notifications for appropriate actions to be taken. In doing so, it would be expected that a high level of interoperability exists between the cloud provider and customer systems.

“I would say the methods, processes, tools and facilities for managing and controlling the test assignment, are needed.” (CEO, organization 10)

Therefore, the need for research on cloud computing standards is imperative. This would provide knowledge on how to build clouds that are highly interoperable and interactive at all levels, hence achieving the flexibility of users’ needs and enhanced performance of cloud services.

4.1.5. Online testing solutions for e-business applications. E-business solutions have been in use long before the cloud computing hype. Implementing e-business applications relies heavily on XML-based standards and specifications.

“Our customers need to verify that their applications are working, they have to test that they have implemented the XML right, schema relation is correct and such kind of issues.” (Program manager, organization 3)

Testing applications based on standards is said to be easier because the variables to be tested can be determined beforehand [5]. This brings up the idea of leveraging on cloud computing capabilities to enhance testing and business-to-business integration of e-business systems.

E-business providers are constantly looking for ways to improve their systems. A point for research is how cloud computing enhances e-business models, the risks involved and how to curb them.

4.2. Management issues

4.2.1. Pool of testers. Cloud service providers frequently promise up to 24/7 availability of the cloud services. This means that they must take up the responsibility of ensuring a continuous service. While cloud computing provides the means to deliver the services, human effort is still required to complete a task e.g. performing a test. Thus, vendors of testing services in the cloud would have to find ways to avail
testers round the clock so as to compete effectively. This would not be a very easy thing to do especially for small and medium sized companies.

“...the problem lies in how you create big enough pool [of testers] to get [the promised] reaction time.” (Testing and methodologies manager, Organization 6)

Crowdsourcing is an interesting dynamic for testing in the cloud. There is potential for research to investigate how crowdsourcing supports testing in the cloud and the different models of crowdsourcing that could be practical for testing in the cloud. One crowdsourcing model is that of availing a global pool of testers for on-demand testing needs from customers, for example by earlier mentioned uTest [14]. Another crowdsourcing model is testing of specific software by a community of users or a group of interest. The feedback obtained from the community would then be used to improve the tested application.

Incorporating crowdsourcing or rather, human as a service (HaaS) within testing in the cloud obviously raises social issues such as trust and communication. Hence, another potential research issue is to investigate the role and impact of trust in collaborative testing activities in the cloud.

4.2.2. Effects on the customer’s business. Some important issues come up when an organization wishes to adopt a cloud approach to acquiring testing services. What would be the business or technical factors to consider? When is a customer safe to migrate testing to the cloud?

“What we are interested in, is how our testing affects to customers’ business. To tell them what is the value of testing for their business.” (Testing manager, organization 11)

Migrating testing to the cloud is not straightforward. It introduces changes to already established ways of working. This might encounter employee opposition. Research could propose strategies that support an efficient and productive transition.

The type of cloud to use should be considered carefully. “Enterprises need to have a better understanding of the way different types of cloud work, how they impact businesses and which testing approaches should be used for them. They need to adopt an end-to-end testing approach, starting from requirements to deployment, because each stage has different testing requirements” [33]. Research could support cloud testing vendors in finding ways to deliver testing services that result in great customer experiences, hence, enhancing the quality of service.

4.3. Legal and financial issues

4.3.1. Test data. The management of test data is a delicate issue. In order for effective testing to take place, some testing tasks depend highly on the actual customer or production data. In some cases, due to rules and regulations, customers are prohibited from supplying confidential or production data to third parties.

“...sometimes a customer has confidential data and some NDAs have to be signed by the specific people...so I don’t know if it’s enough to globally sign NDAs that everything’s confidential.” (Performance testing unit leader, organization 6)

“...the real data quite often is controlled by legislation. For example, granting someone overseas remote access to data about your real end-customers or real end-users in order to test your product. There may be quite serious complications.” (CEO, organization 10)

A solution to this problem may be the development of new models or algorithms that would generate almost “identical” test data to facilitate productive testing results. In addition, it would be necessary to show that generated test data produces the same quality of test results that would have been achieved if real data had been used.

Another viable option would be the use of private clouds but the trade-off in costs would have to be justified.

4.3.2. Pricing models and service descriptions for testing services. In general, pricing models for online computing services need to be well elaborated so that customers are well informed and able to estimate costs. In order to achieve transparent pricing models, different factors and metrics should be considered while calculating the value of a cloud based testing service.

“I would say that the biggest obstacle right now is, pricing models, service descriptions and, metrics, that would cover the, the quality of service... How do I package it, how do I price it and how do I agree with my customer, for example the service level and service quality targets and so on.” (Vice president, organization 9)

5. Discussion and conclusions

This paper provided an overview of cloud computing and discussed various research issues within the context of testing in the cloud. We presented the
research issues in four categories – expressing potential subjects for research.

Cloud computing provides large business and technical benefits to software testing. As it becomes more common practice and availing a myriad of cloud solutions, services and applications, we observed that organizations seem to be on the lookout for ways to enhance the testing process. There is an anticipation of an increase in testing solutions in the cloud, providing flexibility and cost benefits. We believe cloud computing promises a lot of potential for testing. Additionally, vendors will provide testing services through crowdsourcing as well as testing platforms and infrastructure hosted in the cloud.

As the shift to the cloud continues to grow, it will also increase the need for testing in and of the cloud. The problems associated with migration to the cloud are many: security, reliability, performance, scalability and manageability among others. Organizations looking to shift their systems or applications to the cloud need to understand the problems and risks involved in doing so and take the appropriate precautions. Therefore, comprehensive testing becomes necessary as a means to address these problems and risks.

There is a need for pilot projects to see how delivery of testing in the cloud will work in practice. An example is discussed in where Sogeti started a “proof of concept” with IBM Development and Test Cloud to experiment with test tooling and infrastructure in the cloud [34]. Another effort is Open Cirrus (TM) - an open cloud-computing research testbed aimed at supporting research in various aspects of cloud computing e.g. design and management of services [35]. The testbed attempts to encourage collaboration among a community of interest to share and exchange knowledge.

Our current work-in-progress is looking at how cloud computing, service oriented architecture, open source development technologies and crowdsourcing affect an organization’s testing process. In general, we are studying how different software organizations adopt to new methods and concepts, specifically within their testing activities. We also plan to explore how cloud software development and testing will affect quality requirements in the future – addressing the interdependency of cloud software development, cloud testing and overall quality assurance.

The research issues for testing in the cloud are many. We believe different research approaches and methods will suit different problems and scenarios. Suitable research approach could be action research, with survey and the grounded theory method among others. Due to the industry-specific problems of cloud computing, we recommend research approaches that encourage a high collaboration between the industry and researchers, e.g. action research, design research and case studies research. This interplay would enable real-life problems to be addressed in a scientific and methodological manner.

Cloud computing is growing and there is need for academic research to address different research issues associated with it. We hope that this paper can act as starting point for gathering academic research for software testing in the cloud.

5. References


