Moti Begna
SENG 5852
3/20/2023

Research Paper Outline

Paragraph 1

Motivation

The main goal of this paper is to answer the question, “does Cloud Computing improve software quality, performance and reliability?” My reason for deciding upon this topic is two-fold:

1. My current position greatly involves utilizing Cloud Computing concepts and technologies (hereinafter referred to as CC).
2. I believe CC to be one of the most essential components that any company (small or large) can find useful in implementing within their web infrastructure.

Through extensive research and analysis, I hope to not only prove that CC does indeed improve software quality, performance, and reliability, but to also take away new and potentially innovative concepts that I can then turn around and use at work. As mentioned before, one of my reasons for choosing this topic was because I utilize CC in my day to day, and so I would like to present possible new implementations of it on my team with the implications being that doing so would be beneficial in enhancing our current web infrastructure.

Overall Thesis

Through the research and analysis conducted for this paper, I will prove that while there are certainly some pitfalls that must be looked out for when implementing CC, the overall value that it has in increasing various software quality principles makes it a vital tool that any business can use.

Paragraph 2

Paragraph Thesis

Before diving into the quality implications that CC has, it is important to first understand why CC was introduced—its history and what problems it aims to solve.

Supporting Sources

1. In the past, in order to manage the need for high user traffic **“…**timesharing servers” were utilized “due to limited computing resources”. Thus, CC is now used because it is able to “…build complex IT infrastructures” across multiples hosts (Nandgaonkar, 2016, p.734).
2. One issue that many big businesses face is how to reduce costs when there isn’t a need for demanding resources. Here, CC is suited to fix this problem by giving users the ability to create “…applications with a variable workload” i.e., scaling up or down (Nandgaonkar, 2016, p.736).
1. One of the primary goals that CC aims to achieve is to push virtualization to newer heights. Meaning that “cloud computing services can be considered as a significant step towards” giving users access to services “regardless of where they are hosted or how they are delivered” (Mesbahi, Rahmani, Mehdi, 2017, p.1)
3. In recent history, the COVID-19 pandemic highlighted the need to support secure methods of remote work and ensuring system resiliency, thus “Investing in cloud became

a convenient means to address all three of these needs". In fact, survey data shows that 70% of organizations that are currently using cloud services today have plans to increase their spending it (Gartner, 2020)

Paragraph 3

Paragraph Thesis

Here, we will analyze the core concepts of software quality principles that CC is currently being used to achieve. There are many principles mentioned in my sources that are described to be achieved by CC such as maintainability, reliability, testability, and scalability.

Supporting Sources

2. *Reliability* and *availability* in web services is achieved through CC, in one example, by offering multiple failure detection algorithms that can be used to quickly respond to and resolve faults before they turn into errors. This highlights the fact that "…fast and real-time failure detection" is "…one of the most important principles to achieving high availability and reliability in cloud systems" (Mesbahi, Rahmani, Mehdi, 2017, p.4)
3. Through CC, implementations of automated testing as a software (TaaS) have drastically increased software *testability* by giving developers the ability to automatically run tests and receive reports on them within minutes (or occasionally hours), all with the goal "…to be in control of the quality of the software they [programmers and end users] use." (Candea, Bucur, Zamfir, 2010, p.1).
4. *Scalability* is another software quality principle that can be achieved with CC. Through the use of automated tests during performance/load testing, developers are given the opportunity to run tests at higher and higher scales through various load balancing techniques, leading to an "order-of-magnitude lower bug density and higher programmer productivity" (Candea, Bucur, Zamfir, 2010, p.2).
5. CC also improves *maintainability* by offering CI/CD methods which are useful in faster delivery times, such as being able to deploy bug fixes quickly. Faster delivery to higher non-production environments also leads to the ability to verify new features quickly, ensuring that end users are constantly seeing the best version of the software. (Sorenson, 2023)

Paragraph 4

Paragraph Thesis

Although we have explored the various positive impacts that CC can have on software quality, there are still some potential pitfalls that must be assessed during any implementation of it.

Supporting Sources

1. Because of no current agreement on the standard use of CC's external interface, it is very difficult to try and migrate a service from one provider to another, essentially becoming "…locked by the provider" and ultimately putting you in "unfavorable conditions." (Luo Zhang, 2010, p. 629-639).
2. Issues of security are one of the most, if not the most conversed topics of concern when it comes to implementations of cloud services. It's not difficult to gauge that most users of a web service, and subsequently the developers of that service, worry about "their privacy

and data security [more] than traditional hosting services" (Luo Zhang, 2010, p. 629-639).

3. Data loss is another point of scrutiny when it comes to CC. While data is typically stored in multiple locations, making its loss difficult to occur, improper implementations of cloud data stores can lead to major issues if not managed correctly (Sorenson, 2023).
4. As many developers have most likely experienced, dealing with some cloud service providers can be tricky as control over the infrastructures used by other businesses is limited and inflexible. This can make it hard to manage configurations or deal with infrastructure related issues if not given the access to do so (Sorenson, 2023).

Paragraph 5

Paragraph Thesis – Concluding Statement

This paragraph will conclude the paper by summarizing the key arguments and takeaways mentioned earlier. A quick overview of the history, benefits, and possible concerns of CC will be discussed before finally coming to a conclusion on CC's impact on software quality, performance, and reliability.

Bibliography

Candea, George, Stefan Bucur, and Cristian Zamfir. "Automated Software Testing as a Service." IEEE Software, vol. 33, no. 6, Nov./Dec. -, pp. 32-38, doi: 10.1109/MS.2016.138.

Gartner. "Gartner Forecasts Worldwide Public Cloud End-User Spending to Grow 18% in 2021." 17 Nov. 2020, www.gartner.com/en/newsroom/press-releases/2020-11-17-gartner-forecasts-worldwide-public-cloud-end-user-spending-to-grow-18-percent-in-2021.

Luo, Zhiguo, and Jie Zhang. "Cloud Computing: An Overview." IEEE International Conference on Cloud Computing, 2010, pp. 626-631, doi: 10.1109/CLOUD.2010.46.

Mesbahi, Mohammad Reza, Rahmani, Amir Masoud, and Hosseinzadeh, Mehdi. "Reliability and high availability in cloud computing environments: a reference roadmap." Journal of Cloud Computing: Advances, Systems and Applications, vol. 6, no. 1, 2017, pp. 1-16.

Nandgaonkar, Suruchee V. "A Comprehensive Study on Cloud Computing." International Journal of Computer Science Issues (IJCSI), vol. 11, no. 1, Apr. 2014, pp. 733-738

Sorenson, Christopher. Engineering Manager on Suggest at Best Buy. Virtual Conference Interview. 21, Mar. 2023