አዳማ ሳይንስ እና ቴክኖሎጂ ዩኒቨርሲቲ

Adama Science & Technology University

SCHOOL OF ELECTRICAL ENGINEERING & COMPUTING

DEPARTMENT OF COMPUTING

A MASTER'S THESIS

ON

AN EFFECTIVE FRAMEWORK TO MONITOR SERVICE LEVEL AGREEMENTS IN CLOUD COMPUTING ENVIRONMENT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE

OF

MASTER OF SCIENCE IN SOFTWARE ENGINEERING

BY:

ABEL **A**DANE

FEBRARY 2017

ADAMA ETHIOPIA

አዳማ ሳይንስ እና ቴክኖሎጂ ዩኒቨርሲቲ

Adama Science & Technology University

SCHOOL OF ELECTRICAL ENGINEERING & COMPUTING

DEPARTMENT OF COMPUTING

AN EFFECTIVE FRAMEWORK TO MONITOR SERVICE LEVEL AGREEMENTS IN CLOUD COMPUTING ENVIRONMENT

BY

ABEL ADANE

Name and Signature of Members of the Examination Board

MESFIN ABEBE (PhD)	_____	**Date**
Advisor Name	Signature	
_____	_____	_____
External examiner's name	Signature	
_____	_____	_____
Internal examiner's name	signature	
_____	_____	_____
Chairperson's name	signature	
_____	_____	_____
Department Head Name	Signature	
_____	_____	_____
School Dean Name	Signature	

DECLARATION

This thesis is my original work, and it has not been presented for a degree in any other university. All sources of materials used for the thesis have been acknowledged

Abel **A**dane

This thesis has been submitted for examination with my approval as university advisor.

Mesfin Abebe (PhD)

FEBRARY, 2017

Acknowledgement

First and above all, I praise God, the almighty for providing me this opportunity and granting me the capability to proceed successfully. This thesis appears in its current form due to the support and guidance of several people.

First I would like to express my sincere gratitude and heartfelt thanks to my advisor Dr. Mesfin Abebe (PhD) for his dedicated help, advice, inspiration, encouragement and continuous support throughout my thesis work. Secondly, I would like to offer my sincere thanks to all of the other people supporting and guiding me with patience.

Lastly, I would like to thank my family for giving birth to me and supporting me spiritually throughout my life. Thank you all my families and friends who support me in different ways.

Abstract

Cloud computing is a new technology that most of the big organizations are adopting today all over the world. This technology is very important for service consumers to save their time, cost minimization, save human power and they can get services being anywhere at any time and pay per use as well. However there are different kinds of problems in cloud computing environment like service level agreement is violated in most of the service providers when services are delivered. How to consider SLA in a general way for different cloud environments and SLA monitoring problems. In this research we had solved the problems we specified in our scope of the research. We have developed SLA Monitoring framework to solve the identified problems. The framework contains SLA fetcher, Running SLA, Monitoring agent, cloud provider and consumer and cloud services delivery models. We have implemented our prototype using CloudSim on java net beans, MySQL database for creating necessary databases and tables for our research, E_Draw max for destining purposes. We have used three scenarios to evaluate our prototype. The first scenario is giving the resources the same with SLA. The second scenario is we gave less resources than the SLA and in the third scenario we gave more resources than SLA. In the first case there is no violation, but there is violation in the second and third scenario.

Keywords: Cloud Computing, CloudSim, SLA (Service Level Agreement), Service Consumer, Service Provider, Monitoring SLA, SLAMA.

Table of Contents

Table of Figures

Lists of Tables

List of Abbreviation and Acronyms

AHP	Analytical Hierarchy Process
API	Application Programming Interface
ASP	Application Service Providers
AWS	Amazon Web Services
CBSLA	Cloud Bank Service Level Agreement
CMS	Course Management Systems
CRM	Customer Relationship Management
CSLA	Cloud Service Level Agreement
CSP	Cloud Service Provider
EC2	Elastic Compute Cloud
Edu Cloud	Educational Cloud
EHEI	Ethiopian Higher Education Institution
E-Mail	Electronic Mail
ERP	Enterprise Resource Planning
EthERNet	Ethiopian Education and Research Network
IAAS	Infrastructure as a Service
IBM	International Business Machine
ICT	Information and Communication Technology
ISO	International Standard Organization
ISP	Internet Service Provider
IT	Information Technology
KPI	Key Performance Indicator

LMS	Learning Management System
MIPS	Millions of Instructions per Second
NIST	National Institute of Standards and Technology
PAAS	Platform as a Service
PSLA	Personalized Service Level Agreement
QoS	Quality of Service
RAM	Random Access Memory
RIA	Rich Internet Application
RUP	Rational Unified Process
SAAS:	Software as a Service
SIS	Student Information System
SLA	Service Level Agreement
SLAAS	Service Level Agreement Aware Service
SLAMA	Service Level Agreement Monitoring Agent
SLM	Service Level Management
SLO	Service Level Objective
SOA	Service-Oriented Architecture
SQL	Structured Query Language
UP	Unified Process
VLE	Virtual Learning Environment
VM:	Virtual Machine
WSLA	Web Service Level Agreement

CHAPTER ONE

1. INTRODUCTION

1.1 Background of the study

Information and Communication Technology (ICT) is core technology for all business, such as healthcare, education, science, entertainment and defence. In the current era of information technology every sector demand higher level of IT interaction such as instant online access to information, products and services through their mobile devices.

Today almost all big organizations had been migrated to cloud computing all over the world because of different benefits from cloud computing like cost minimize for IT infrastructures. Higher Educational Institutions worldwide have become highly dependent on information Systems for their IT provision and service delivery. Thus, the necessities for educational resources like hardware, software, study materials, teaching tools, teaching stuff sets [1].

Information Technology has always been considered a major point of enterprise organizations, from the perspectives of both cost and management. However, the information technology industry has experienced a dramatic shift in the past decade factors such as hardware commoditization, open-source software, virtualization, workforce globalization, and agile IT processes have supported the development of new technology and business models. Cloud computing now offers organizations more choices regarding how to run infrastructures, save costs, and delegate liabilities to third-party providers. It has become an integral part of technology and business models, and has forced businesses to adapt to new technology strategies [2].

Cloud computing employs a service driven business model where resources are offered as a service, usually over an internet connection and for which a provider charges a fee as public and in which the cloud is deployed inside a firewall and managed by the user organization as private. The main drivers for cloud computing adoption include scalability, elasticity, virtualization, cost, and mobility, collaboration, and risk reduction. Major concerns include security, interoperability, control, and performance [3]

The National Institute of Standards and Technology's, states that Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing

resources. These are networks, servers, storage, applications, and services that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing is a model for providing computing service such as storage, servers, services and applications, without physically acquiring them via the internet. Access to such service could be free or pay as per use. So that the organizations does not have to spent time and cost in managing them [4]. Cloud computing represents another evolutionary step in the continued virtualization of IT systems. As computing power and network bandwidth increased with advances in circuit density, packaging, and design, IBM and other vendors continue to provide new ways of sharing computing resources between increasing numbers of users and their computing workloads. Where virtualization creates the appearance that each user or workload has dedicated computing resources while the actual physical resources are being shared, cloud computing is concerned with the dynamic allocation of these virtual computing resources and the delivery of IT services over the network in response to rapidly changing business needs [5].

1.2 Service level agreement

According to [6] Service Level Agreement (SLA) is defined as a legal agreement between the service provider and customer. Design of such agreement aim is to maximize the profit to provider as well as assured availability of services to customer. However the restricted choice of appropriate parameters in SLA affects the interacting of end user with cloud services and creates risks of user data. Users are concerned about their data and how it will be stored in cloud and how the data is recovered in the case of failure of disaster. However in reality, none of the SLA considers user perspective while conducting SLA [6].

The service contracts or SLA documents are legal cloud documents that define broadly the service data, delivery mode, service agent details, quality metrics and cost of the service including penalty terms, if any [7].

According to [8] CSLA includes five parts: cloud service, cloud technology, cloud quality report, and cloud security, cloud business

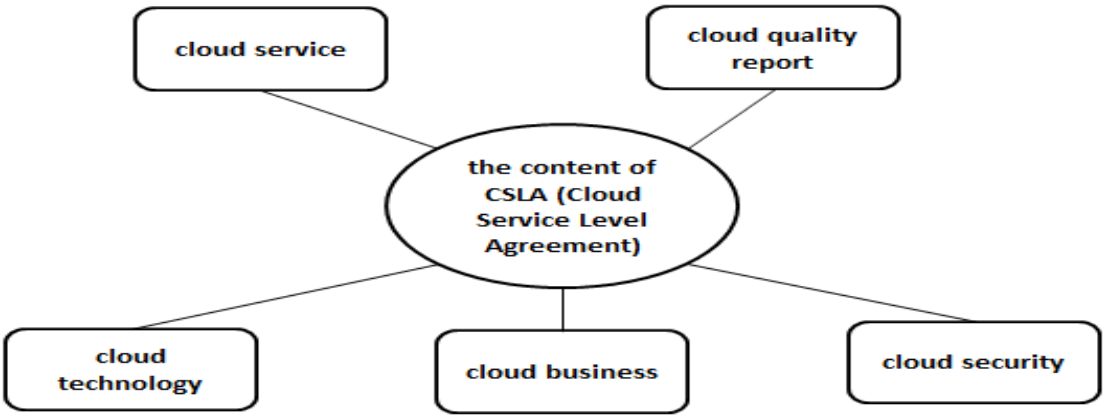

Figure 1-1 CSLA structure [8]

- **Cloud service** describes the negotiated cloud services, including cloud service information identity, cloud services, cloud service level, cloud services billing, contract modification or termination
- **Cloud component technology** represents a common business measure of cloud technology parameters, targets set and performance monitoring
- **Cloud reports section** refers to the cloud services quality monitoring report of consumers and providers, including the cloud services quality data and statistics specified in CSLA.
- **Cloud Business section** refers to the description that when the cloud service provider does not meet the promised cloud service level (i.e. Service violation), the cloud service provider should compensate according content, methods, and irresistible factors in.
- **Cloud security** refers to the major cloud security service side.

A typical Service Level Agreement (SLA) within the CSA describes levels of service using various attributes such as availability, serviceability or performance. Well-designed SLAs can significantly contribute to avoiding conflict and can facilitate the resolution of an issue before it escalates into a dispute. Service level guarantees for IaaS are better defined than for SaaS or PaaS, but that does not mean that they meet the customer's expectations. Most public cloud infrastructure services are available only through non-negotiable standard contracts which strictly limit the provider's liability. As a result, the remedies offered in case of non-compliance do not match the cost to the customer of the potential service disruptions. Furthermore, most

IaaS providers put the burden of SLA violation notification and credit request on their customers [9].

Cloud-based services have become the backbone of IT industry nowadays and the promising technology that offers a commoditized service to the software, the platform and the infrastructure where they are delivered as a service. Because several providers have started to offer a vast diversity of Cloud services, Cloud customers became unable to decide whose services they should use and what is the basis for their selection. Therefore, a legal contract is necessarily needed for negotiating these both Cloud parties. This contract is referred to Service Level Agreement (SLA). A SLA negotiation between Cloud parties assists in defining the Quality of Service (QoS) requirements of critical service-based processes. In addition, the role of third party in the negotiation process that is represented in Cloud broker will recommend customers to achieve the required service efficiently when negotiating with multiple providers. Currently, there is no framework that can allow customers to evaluate Cloud offerings and rank them based on their interests.

1.2.1 Monitoring and Measuring

Service level management, based on SLOs, is how performance information on the cloud is gathered and handled. This is how it is employed [10]

- The **cloud provider** uses service level management to make decisions about its infrastructure; for example, if throughput isn't always meeting a customer's requirements, the provider can reallocate bandwidth or add more hardware. Or decide to make one customer happy at the expense of another one. For providers, SLM is designed to help make the best decisions based on business objectives and technical realities.
- The **cloud consumer** uses SLM to decide how he wants to use cloud services; like whether or not to add in more virtual machines and at what price point that option becomes too expensive to justify the return. For consumers, SLM helps them make decisions on the way they use the cloud. And sometimes on how to automate those decisions.

1.2.2 SLA requirements

According to [11] a list of 14 responsibilities to consider when considering an SLA:

1. **Security**: A consumer must understand his security requirements and what controls and federation patterns are necessary to meet those requirements. A provider must understand what they must deliver to the consumer to enable the appropriate controls and federation patterns.
2. **Data encryption**: Data must be encrypted while it is in motion and while it is at rest. The details of the encryption algorithms and access control policies should be specified.
3. **Privacy**: Basic privacy concerns are addressed by requirements such as data encryption, retention, and deletion. An SLA should make it clear how the cloud provider isolates data and applications in a multi-tenant environment.
4. **Data retention, deletion**: How does your provider prove they comply with retention laws and deletion policies?
5. **Hardware erasure, destruction**: Same as #4.
6. **Regulatory compliance:** If regulations must be enforced because of the type of data, the cloud provider must be able to prove his compliance.
7. **Transparency:** For critical data and applications, providers must be proactive in notifying consumers when the terms of the SLA are breached. This includes infrastructure issues like outages and performance problems, as well as security incidents.
8. **Certification:** The provider should be responsible for proving required certification and keeping it current.
9. **Performance definitions:** What does *uptime* mean? All the servers on every continent are available? Or just one is available? It pays to define those definitions. (The authors of this paper suggest standardizing performance terminology to make it easier.)
10. **Monitoring:** For issues of potential breaches, you might want to specify a neutral third-party organization to monitor the performance of the provider.
11. **Auditability:** Because the consumer is liable for any breaches that occur with loss of data or availability, it is vital that the consumer be able to audit the provider's systems and procedures. The SLA should make it clear how and when those audits take place. They can be disruptive and costly to the provider.

12. **Metrics:** These are the tangible something's that can be monitored as they happen and audited after the fact. The metrics of an SLA must be objectively and unambiguously defined. Following this list is a list of common metrics.
13. **Providing a machine-readable SLA:** This can allow for an automated, dynamic selection of cloud broker. In other words, if your SLA requires that the broker use the cheapest possible provider for some tasks but the most secure provider for others, this type of automation makes it possible. (This type of service is not readily available yet, but is something to keep in mind when contributing to the cloud SLA standardization discussion.)
14. **Human interaction:** On-demand self-service is one of the basic characteristics of cloud computing but your SLA should take into account that when you need a human being, one is made available to you.

1.3 Motivation

Today almost all big organizations are adopting cloud computing to their organization for simplifying communication services and cost minimization in order to increase their benefits. However there are different kinds of problems in cloud computing environment. When we talk about cloud computing we have to know how and what mechanism are used to use cloud computing. The user cloud computing are accessing cloud services through internet connection. In order to get service in a well manner there is a contract between cloud service providers and cloud service users which is known as service level agreement (SLA). Different reasons motivated me to do this research, the following are some of the reasons 1. SLA violation, service level agreement is violated in most of the service providers when services are delivered. The SLA signed between two parties and services the consumers consumed sometimes violated and the violation is reported to the service providers but not to the consumers, so there was discrimination here.2. Service level agreement is monitored from provider side and customers have no option to monitor service level agreement and this situation makes costumers unsatisfied. As I explained above cloud computing is very important technology for the organizations in order to save money, time and human power and to serve the customers of their organization in a well manner. So the reason that motivated me is the second list of the reasons I studied implemented in this research.

1.4 Statements of the problem

Information Technology is very essential in Academic areas for different purposes i.e. for student references, for Instructor guides, for research and Administration purposes and others

issues. Currently cloud computing is a new technology that reduces the challenges that the old IT services faced. Cloud computing is very essential for an organizations or institutions for cost minimizing to build IT infrastructures as well as simplify the way of obtaining ICT services at any time and places through internet for the above listed stakeholders namely organizations and institutions. Cloud computing solves a lot of problems by cost minimizing, saving man power, getting services anywhere, pay-per-use and so on . This can be done if and only if the service level agreement is monitored in a well manner.

However there are different kinds of problems to conduct service level agreement between service providers and service consumers [6]. Service provider is the only part that can monitor service level agreement and SLA is violated, as a result users are not satisfied. This problem is a problem of conducting service level agreement between cloud service providers and service customers. There are other problems like resource management (how to manage resources) and accessibility (techniques how to us of services). Without monitored service level agreement cloud computing may not have advantages for customers. The service level agreement has been monitored by cloud service providers only and the service customers have no option to monitor the service level agreement signed by the two parties, cloud provider and cloud customers. The common problems are listed as: 1. SLA violation, service level agreement is violated in most of the service providers when services are delivered. 2. Service level agreement is monitored from provider side and customers have no option to monitor service level agreement and this situation makes costumer's unsatisfied 3. Reliability and Trust 4. Cloud integration and interoperability 5. How to consider SLA in a general way for different cloud environments 6. How to describe the SLA terms between a cloud provider and a cloud customer, such as service levels objectives, or penalties in case of SLA violations?

Therefore, the research tries to get answers of the following research questions.

Q1. Are there any problems in the current cloud based service level agreement?

Q2. What solutions are taken to solve the problems?

Q3. How to implement the solution?

1.5 Objectives of the study

1.5.1 General objectives

The general objective of this research is designing a Framework to Monitoring Service Level Agreements in cloud computing environment.

1.5.2 Specific objectives

- Understanding about cloud computing and cloud based service level agreement
- Analyses of the current framework on cloud based service level agreement(SLA)
- Studying about services and service level agreement
- Developing a framework that show how to monitor service level agreement over cloud in both side (provider and consumer side).
- Implementation of the framework using cloud simulation tool i.e. CloudSim, and programming language (java NetBeans) and database MySQL.

1.6 Significance of the study

The General Uses of service level agreement are to understand roles and responsibilities, Understand services and deployment models, Identify critical performance objectives, Evaluate security and privacy requirements, Identify service management requirements, Prepare for service failure management, understand the disaster recovery plan and Define an effective management process. Service level agreement is the core important component is cloud computing. In our research have developed an Effective to Monitoring service level agreement over cloud for both parties. So our research have the following significance.

1) Significance to the service consumers
   - Since our developed framework flexible, the consumers will have high confidence to use cloud computing.
   - The consumers will be satisfied to the cloud providers because they know what they agreed and what they used according to our framework.
   - The members of the organization can access services according to their needs.
   - Researchers and Administrative staff can access services for research purposes and management purposes respectively.
   - The organization and institutions can save their money, time and human power in ICT installation and for buying infrastructure and needed software.
   - Consumers know whether the service is based on SLA or not.
   - The computation between the providers will increases and the cost and QoS increases for the customers.
   - Avoid conflict with provider

- Increases the adoption of cloud computing in the society of the organization.

2) Significance for service providers

   Our research is significance to the providers as well. The following are the significance to the providers

- If the service consumers are satisfied on the framework, the need of using cloud computing in increases. So if number of consumers increases the benefit or income of the providers increases.
- When cloud service level agreement is monitored traditional users of ICT will move to cloud computing.
- If service level agreement is monitored in a well manner the QoS will be increased and customer's satisfaction increases and the provider will get more users as a result provider is benefited.

1.7 Scope of the Study

There are different kinds of solution for the above listed problems. So we focused on service level agreement between service providers and Consumers. In his research we have:-

- Studied the current situation in deployment and using ICTs services in different organization and institutions.
- Designed a framework that Monitor Service Level Agreement (SLA) over cloud.
- Implemented the prototype using CloudSim simulation.
- Included Bandwidth, MIPS, RAM, Storage and Core that are the parameters in our implementation.

1.8 Limitations of the study

Our research has some limitations. These are:-

- Real implementation was difficult because we did not have enough resources to install real environment.
- Our prototype did not calculate the difference between resources required and allocated because it needs another algorithm.
- We did not get the latest IEEE paper because of websites blocked.

1.9 Research Document Organization

Chapter One: Presents the background of the study, the statement of problem, objectives, Scope and significance of the study

Chapter Two: Presents Literature Review and Related work.

Chapter Three: Presents the methodology used in the study

Chapter Four Presents selection of cloud providers as well as cloud tools in details.

Chapter Five Presents the Proposed Solution and the Designed framework.

Chapter Six Describes the Evaluation of the proposed system

Chapter Seven Presented Conclusion, Recommendation and future work.

CHAPTER TWO

2. LITERATURE REVIEW AND RELATED WORK

2.1 Cloud computing

With the huge increasing of the speed of the network over the last decades, there is a significant rise in its usage involved millions of Web queries a day. This increasing demand is handled through huge datacenters. Datacenters consist of hundreds or thousands of servers with other infrastructure such as storage and network systems. These huge datacenters are operated by many IT global companies such as IBM, Google, Amazon, and eBay. The commercialization of these developments is defined currently as cloud computing, where computing is delivered as utility on a pay-as-you-go model [12].

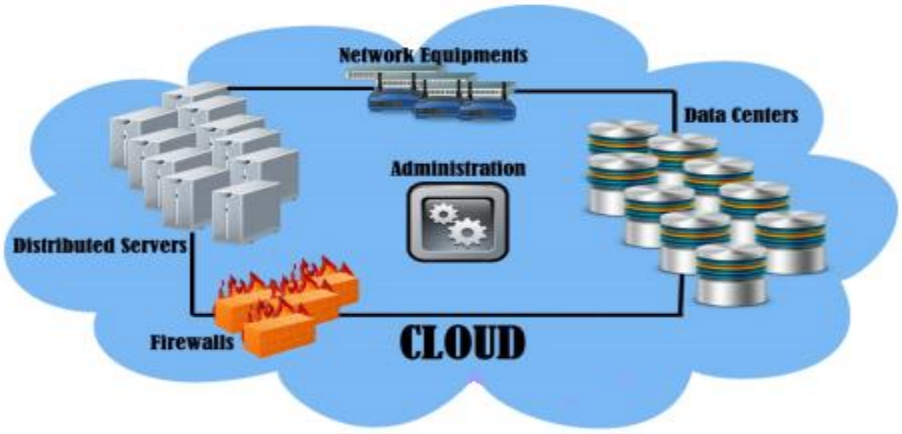

Figure 2-1 Overview of cloud computing From IT Infrastructure View

Cloud computing is a distributed computing paradigm allows to outsource all IT needs(hardware and software) such as processors, storage, platforms, software and services. These resources are accessible and delivered as service over a network (typically the Internet). A clear example of the cloud computing is document-sharing service such as Google Docs.

The literary meaning of cloud computing comes from the terms cloud and computing. "Cloud" in the reality has no shapes and fixed positions, and the cloud server can have any Setting and configuration. The location of cloud server can be also anywhere in the world with Internet as fundamental medium. Computing term achieved using set of networked resources which are offered on subscription.

The following definitions of cloud computing have been taken from well-known scientists and organizations:

I. Defines the cloud computing in terms of its utility to the end user [13]: "A cloud is a type of parallel and distributed computing system consisting of a collection of

interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreement established through negotiation between the service provider and consumers"

II. National Institute of Standards and Technology (NIST) defines the cloud computing as follows: "Cloud computing is a model for enabling convenient, on demand network access to a shared pool of configurable computing resources such as networks, servers storage, applications and services that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and for deployment models."

III. According to Mell, P., and Grance, T. Cloud Computing is a model for enabling 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 [14].

According to Staten cloud computing is a pool of abstracted, highly scalable, and managed computing infrastructure capable of hosting end-customer applications and billed by consumption [15].

A Cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers [16].

National Institute of Standards and Technology, 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, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [17].

2.1.1 Cloud Computing Characteristics

The different definitions of cloud computing shown in the previous section and all other definitions existing in many resources and through the internet, lead back to some main characteristics of cloud computing. These characteristics are the main demands of the industry for future IT.

Figure 2.2 [12] shows the main characteristics: virtualization, service oriented, elasticity, dynamicity, economy of scale, Market-Oriented (Pay as you go) and autonomic.

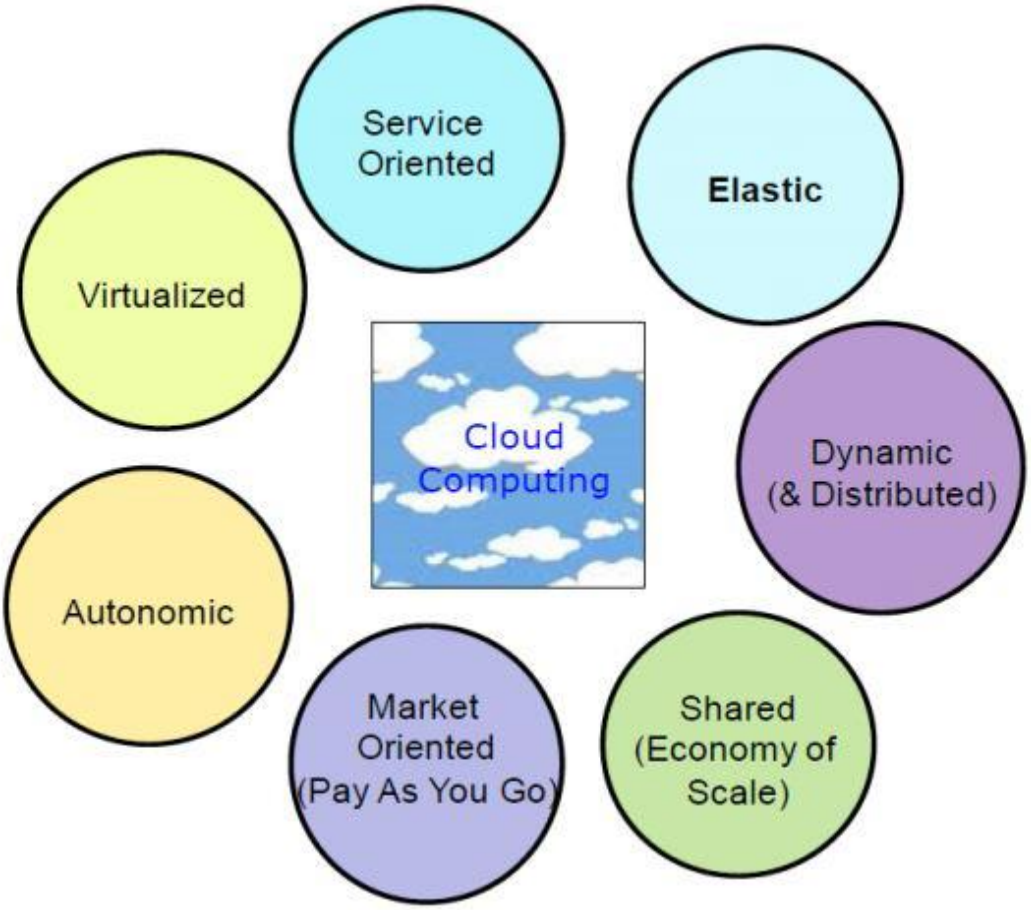

Figure 2-2 Characteristics of cloud computing

Virtualized: Resources in clouds i.e. computing, storage, and networking are virtualized. The virtualization is achieved at virtual machine and platform levels. Applications can run within their operating systems running on the same physical machine. Application can easily be migrated from one physical server to another. In the platform level, every application is mapped to one or more resources offered by different Cloud infrastructure providers.

Service-Oriented: everything in the cloud is a service. Software, platform and infrastructures are offered as services made accessible over the network. To achieve this facility, Cloud is implemented using Service-Oriented Architecture model. SOA is an architectural style for building application, loosely coupled, allowing composition. Cloud can benefit from this paradigm in building any of its services such as infrastructure, so it's called service-oriented infrastructure.

Elastic: Cloud provides the flexibility dynamically provision and vary the resources allocated for its application. The resources (computing, storage, network capacity) increases or decreases at runtime depending on the user QoS requirements.

Dynamic and distributed: to add more performance and reliability to the cloud services delivery; cloud resources are made distributed. The resources are also dynamic and can be changed (increase or decrease) according to the requirements of the user.

Economy of scale: cloud computing is often defined as a technology. However, it is actually a significant shift in the business and economic models for consuming IT that can lead to a significant cost saving. The cost saving can be achieved by the resource pooling; this capability makes the resource shared among the users. Thus, resources can be dynamically allocated according to the applications demand. That is also known as multi-tenant sharing model. Generally, the users neither have any control over the physical resources nor they are aware of the resource location.

Market-Oriented (Pay as you go): this characteristic addresses the utility dimension of cloud computing. Cloud computing is billed much like a utility, so it can be affordable for even the smallest business. The user only pays for the services that are used.

Autonomic: Cloud services are highly reliable. Therefore, clouds show autonomic behavior by managing themselves in case of failures or the performance deprivation.

2.1.2 Evolution of cloud computing

The growth of cloud computing area started alongside with the internet. In the beginnings of the internet, ISPs offered some services and those services were not cheap and limited. The area of interests of the internet was very small and focused on the scientist and organization not on the normal customers. With the time, the internet became more popular, and then the ISPs started to make more attractive offers for the customers. Some new services have been added such as emails or access to servers in their data centers [18].

At this time, the services provided were more related to computing infrastructure services.

Thus, the next evolution step added more to the existing ones. The Internet service providers started the transformation from pure computing infrastructure to application service providers (ASPs). The result of this process is hosting some applications consumed by the customers.

These applications including the infrastructure are owned and managed by the application service providers. The main difference to what known today as SAAS (Software as a service) is that the ASPs provide the service to customer as a single tenant and that includes the infrastructure, while SAAS provides a shared multi-tenant environment [18]

Following this sequence of events, cloud computing appeared and as mentioned before in this chapter there is no clear definition for cloud computing and still in the transformation process. Cloud computing integrate some old concepts mentioned above in this section and their steps in evolution of ISPs and service providers with support of new technologies such as

virtualization, Service Oriented Architecture (SOA) and other characteristics to achieve the main demands of the industry. So, cloud computing is not exactly new technology, but it is a new architectural concept merges old and new concepts, and it has been supported by newly evolved strong technologies. Figure 2.3 [18] shows the different evolution stages which lead to cloud computing.

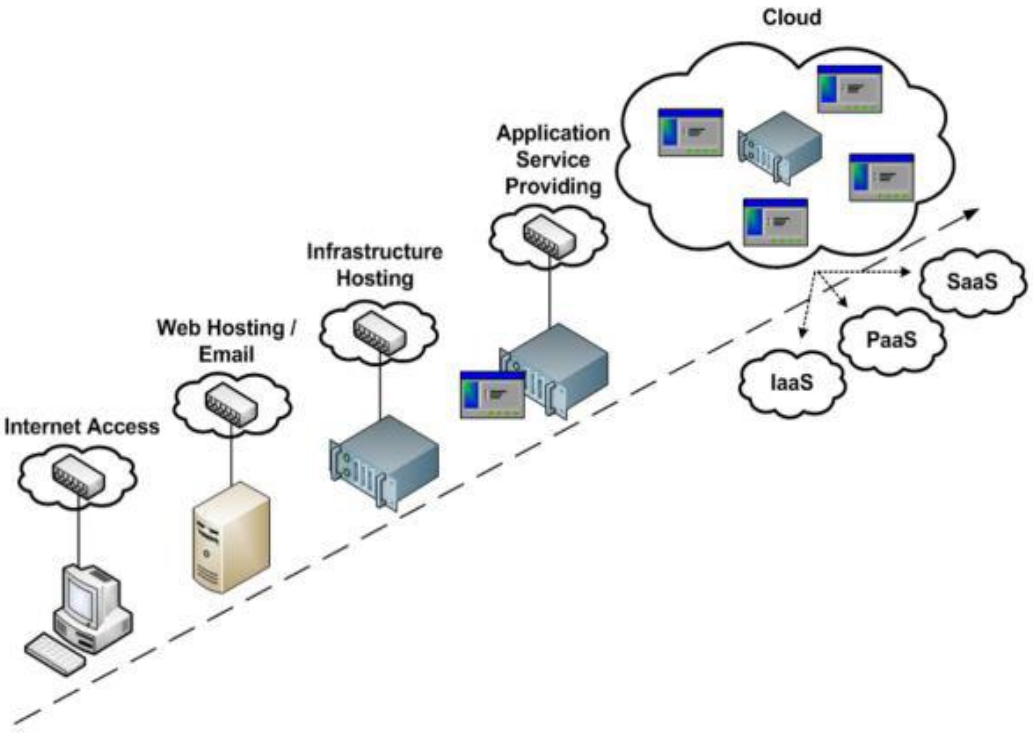

Figure 2-3 Cloud computing evolution

2.1.2.1 Drivers to cloud adoption

Economies of scale: Determining the new IT infrastructure is a complicated process and it is always based on a project which evaluates in its sizing phase the requirements of the hardware or software to accomplish business requirements. Estimating the actual percentage of the performance is difficult as well. Thus, Under-estimations and overestimations are a daily occurrence. Cloud computing enables on demand and accurately allocated computing resources as needed by businesses. The results are shorter evaluation projects and a better risk management, because lots of unknown factors are eliminated by default [18].

Small initial investment and low ongoing cost: Cloud computing eliminates the expenses cost companies need to build the IT infrastructures. As everything is hosted to the cloud, the only expenditure required is for accessing cloud services such as networks and client systems. In addition, operating costs are reduced since the billing made for the actual level and usage of the resources on a pay for a use basis. Another important point for expenditure saving is the

reduction of maintenance cost. Since cloud computing uses less physical resources, there is less hardware to maintain and power. Through cloud provider, companies can decrease the number or the working hours of server, storage, and virtualization experts; cloud providers are the responsible [18].

Open Standards: Most cloud providers expose well-documented Application Programming Interfaces, but every provider has its own unique implementation and thus it lacks to interoperability. Unless a critical mass of cloud consumers and the market's demand for modularity and standardized cloud interfaces is reached, vendor lock-in is a threat to cloud computing. Nevertheless, these standards will come, e.g. VMware as the biggest virtualization software vendor, announced "Project Redwood" as an interface for interoperability between clouds [18].

Sustainability: This driver is an immense driver for companies of all sizes that benefits from cloud. Moving the IT infrastructure, applications, and services to the cloud allows the enterprises to focus more on their core business and change it quickly in response to changing market demands. Cloud providers are responsible of security, risks and disasters management. In addition, the cloud service provider removes any technological bottlenecks. Sustainability is realized, because both cloud providers and enterprises concentrate on their own core business, and they are not aware about the other task [18].

2.1.3 Cloud Computing Service Models

Cloud computing providers offer their services according to three fundamental models. It is mainly composed of these three models (layers) which form the computing stack of the cloud system. Each of these layers offers different services to the end user.

Figure 2-4 Cloud Service layers

2.1.4 Infrastructure as a service (IAAS)

The lowest layer in the computing stack, it consists of virtual or physical machines, storage and clusters. IAAS layer can also contain database management systems and other storage services. The infrastructure is managed by an upper management layer that ensures application isolation, runtime environment customization, accounting and quality of services. A virtualization tool, such as hypervisors sit in this layer to manage the resource pooling, and to partition the physical infrastructure into virtual machines [19] [20].

Amazon Web Services (AWS) is an example of IAAS. The infrastructure is billed on a pay per use model. The customers can get servers, storage without caring about co-location, rental or datacenters.

According to [21] Infrastructure of cloud has multiple components which are used in infrastructure service model. They are the shared resources among the users, so they have multiple challenges for service providers. Some components are:

- Cloud Software
- Computer Hardware
- Network and Internet Connectivity
- Platform Virtualization

- Utility Computing
- Service Level Agreements

Before deploying the service providing architecture of cloud computing, one of the desired concerns of the customers is the reliability and quality of the offered services [21]. The expectations of these services changes from customer to customer so a consensus is needed between service provider and the customer. This agreement about the quality and reliability of service is called Service level Agreement.

It aims at defining, describing and negotiating on terms and conditions of contract between service provider and customers to reach a final verdict. QoS attribute is also part of SLA and it needs to be defined properly. QoS (Quality of Services) may include feature like response time, memory usage and throughput etc. And they need to be monitored closely [21]. It is very critical part because if any term or condition is skipped or not discussed, it will create problems for the users and service providers. SLA includes other features like services measurement, service condition evaluation and management services [21].

Cloud-based services have become the backbone of IT industry nowadays and the promising technology that offers a commoditized service to the software, the platform and the infrastructure where they are delivered as a service. Because several providers have started to offer a vast diversity of Cloud services, Cloud customers became unable to decide whose services they should use and what is the basis for their selection. Therefore, a legal contract is necessarily needed for negotiating these both Cloud parties. This contract is referred to Service Level Agreement (SLA). A SLA negotiation between Cloud parties assists in defining the Quality of Service (QoS) requirements of critical service-based processes. In addition, the role of third party in the negotiation process that is represented in Cloud broker will recommend customers to achieve the required service efficiently when negotiating with multiple providers. Currently, there is no framework that can allow customers to evaluate Cloud offerings and rank them based on their interests. Thus it is important to have a methodology that might map the customers' requirements referred as Service Level Objectives (SLO) with the most reliable Cloud service [22].

2.1.5 Platform as a service (PAAS)

Platform as a service provides users with complete development environment to support the development lifecycle from design, implementation, debugging, testing, deployment, operation and support of rich internet application (RIA) and online services. With PAAS an entire software environment can run at the service provider while not worrying about the technology underneath it. Google AppEngine, Aneka and Microsoft Azure are some of most prominent examples [13] [20].

The main advantage of PAAS is the cost saving in all the development cycle steps. PAAS providers allow cloud users to focus on the application development by reduction the cost resulted from the upgrading of the underlying platform.

2.1.6 Software as a service (SAAS)

Software-as-a-service is the top most layer of the cloud computing architecture. SAAS is defined as a software delivery model in which the software (application) is hosted on the cloud with its associated data. It provides on-demand access to applications typically using a thin client or web browser. SAAS became a common delivery model for many business applications in different enterprises from small to large business, including customer relationship management (CRM), enterprise resource planning ERP, and human resource management (HRM) and several other business applications. SAAS providers constitute other layers (services) of cloud computing, and thus, it provides the services required to maintain and store the customer data in addition to configure the applications according to the customer's requirements. SAAS results in a considerable reduction in the cost of buying new software and infrastructure. The customers do not need to keep or maintain any infrastructure or install the applications on every machine. They just need high network speed that enables them to access their applications normally by web browsers [12].

The key feature of SAAS is the multi-tenancy architecture. This architecture distinguishes SAAS from other packaged software solutions. With this model, a single version of the application, with a single configuration for tenants (customers) is used. That allows the companies to outsource the effort of maintaining and upgrading applications, managing large hardware infrastructure, and optimizing resources by sharing the cost among the large user base. Hence, SAAS is interesting for these companies who access software shared between multiple users and configured to the companies' specific needs [12].

Cloud computing is an emerging technology which introduced itself as a service oriented technology. It is working on the principle of on demand service and scalability. It is providing services in many ways including software, platform and infrastructure and making the users

free of installing and administering these services. In spite the fact that cloud computing provide high performance, high available, fault tolerant services; the issues it comes with are also very serious in nature. Of the worth mentioning are data and network security, data authenticity and audit ability, lack of user control over data and security polices and virtualization problem. In order to attract the organizations and build the confidence of customer, these issues need to be well researched and resolved [21].

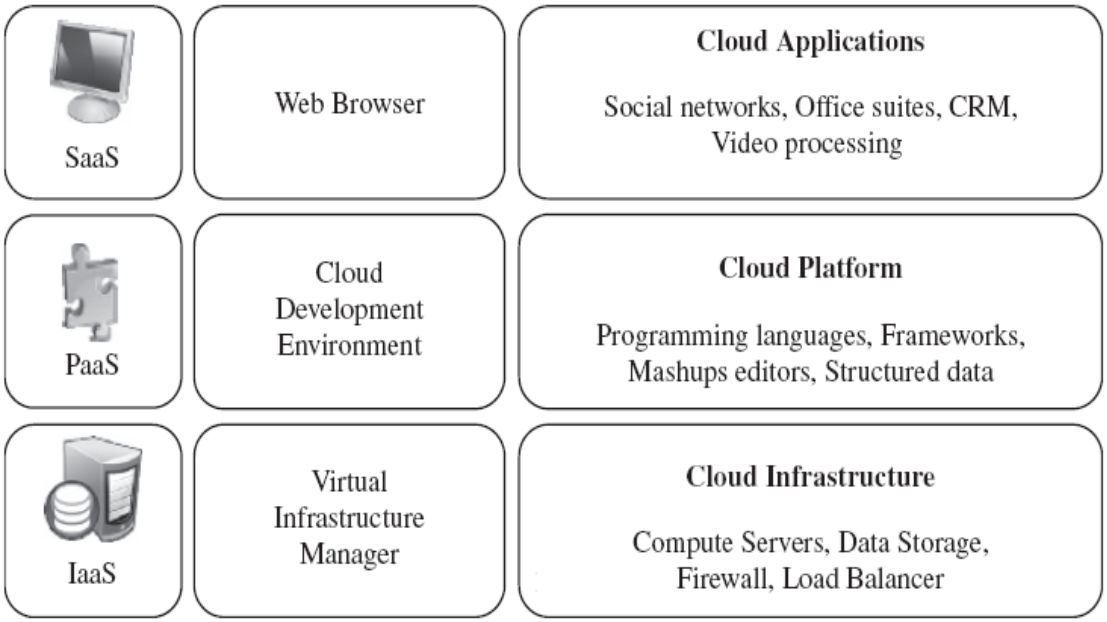

Figure 2-5 Cloud computing service model [21]

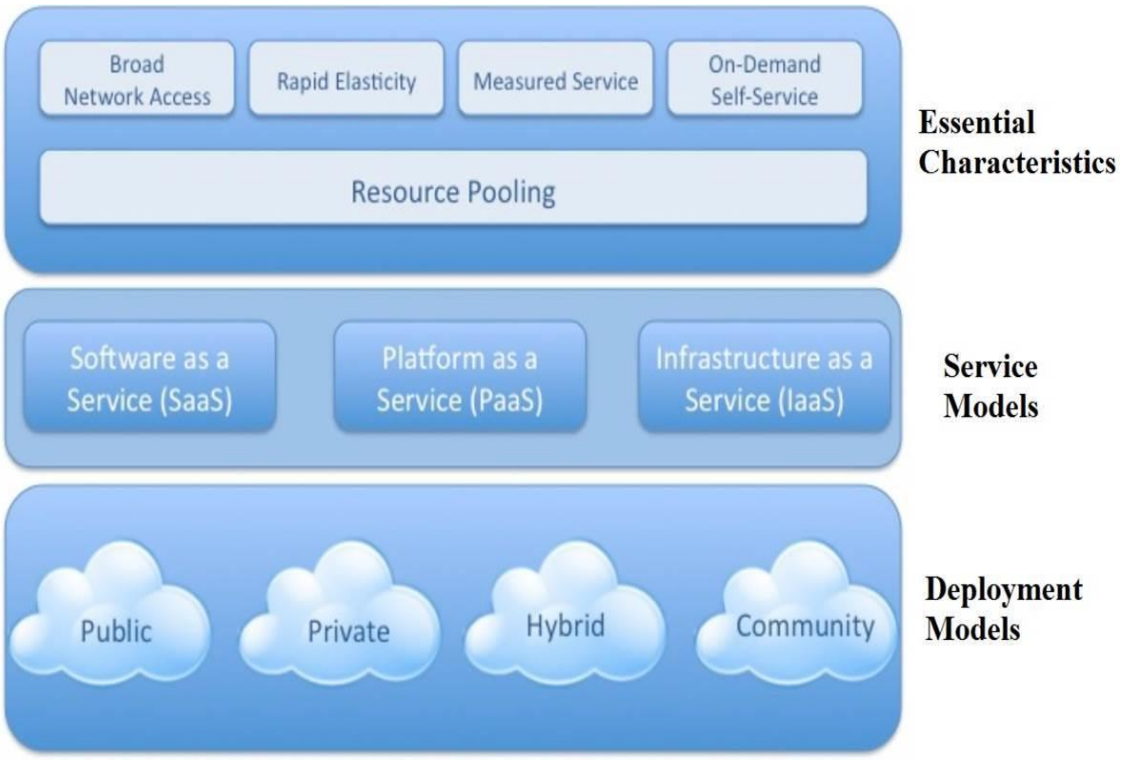

Figure 2-6 Cloud computing service model NIST [23]

2.2 Cloud Computing Deployment Models

Cloud computing is a paradigm of offering different kind of services in on-demand delivery model. According to the previously mentioned services and cloud computing characteristics, cloud computing deployment models are classified to work in harmony with other elements into four types: Public cloud, Private Cloud, Community Cloud, and Hybrid Cloud.

2.2.1 Public Cloud

Public cloud is one of the deployment models, in which a service provider makes the services (application, storage, Infrastructure) available to anyone (General Public) on Internet. Public cloud services can be free or purchased on a pay-per-usage model.

The main benefits of using public cloud services are:

- The overall cost of the system installation is covered by the provider, and so it provides an easy and inexpensive set-up.
- Scalability to meet the needs. To support thousands of public domain users, public cloud providers build quite large datacenters consisting of thousands of servers with high speed network.
- Pay for what you use, no resource wasting.

Examples of public clouds: Amazon Elastic Compute Cloud (EC2) provides infrastructure as a service, Google AppEngine provides an application development platform as a service, and Salesforce.com offers software as a service. Public clouds provide a very good option to handle peak loads on the local infrastructure and for effective capacity planning. Thus, it offers very good solutions to the customers having small enterprises or with infrequent infrastructure usage [19].

Quality of service and security are two main issues have to be ensured in the management process in public clouds. Consequently, a considerable part of the software infrastructure is dedicated to monitor and manage cloud resources, log the history of the cloud usage for each customer, and bill the used services according to the users' contracts [19].

2.2.2 Private Cloud

Although public clouds provide very good solutions that reduce the cost of installation and resources management, in addition to providing a scalable system that can handle the peak loads on the local infrastructure, there are still some cases where company may want to maintain their own specialized clouds. For instance, the health care industry maintains confidential medical data which cannot be stored in public clouds. Thus, private clouds are deployed and operated solely for a single organization to provide IT services to its internal users.

According to [23] the main benefits of private clouds are:

- Offering greater control over the infrastructure. Due to the fact that the infrastructure is on-site, the organization is involved in the process of maintaining and monitoring its data.
- Improving security by restriction the access to one organization.
- Reducing the time needed to deliver what users demand. The private cloud is deployed inside the firewall on an organization's intranet that means the transfer rates are significantly increased.

2.2.3 Hybrid Cloud

Private clouds provide suitable solutions for organizations seek more control on their data and greater security. Moreover, private clouds increase the performance of accessing the applications data. However, the cost and maintenance are high disadvantages for private clouds. The higher cost resulted from that in the private clouds the organization need hardware and maintenance personnel, in addition to investment in new hardware or using the existing

infrastructure in the company. Being some parts of the infrastructure deployed on-site of the organization, that raises up the maintenance problems and risk management [23].

According to the discussion above, hybrid clouds come to compromise a solution between the public and private cloud. In hybrid clouds, organizations host the non-critical and infrequent used data to public clouds, while keep the important information and services on their sites (Private Clouds) [23].

2.2.4 Community Cloud

Community cloud is a deployment model in which infrastructure is shared between several organizations they have the same computing concerns such as security, compliance, performance requirement and jurisdiction. The organizations with the community clouds can take the advantages of a public cloud such as multi-tenancy and pay-as- you-use billing model. Moreover, some benefits related the privacy and security associated to a private cloud can be gained. The cloud may be managed by organizations or a third party and may be existing on premise or off-premise [23].

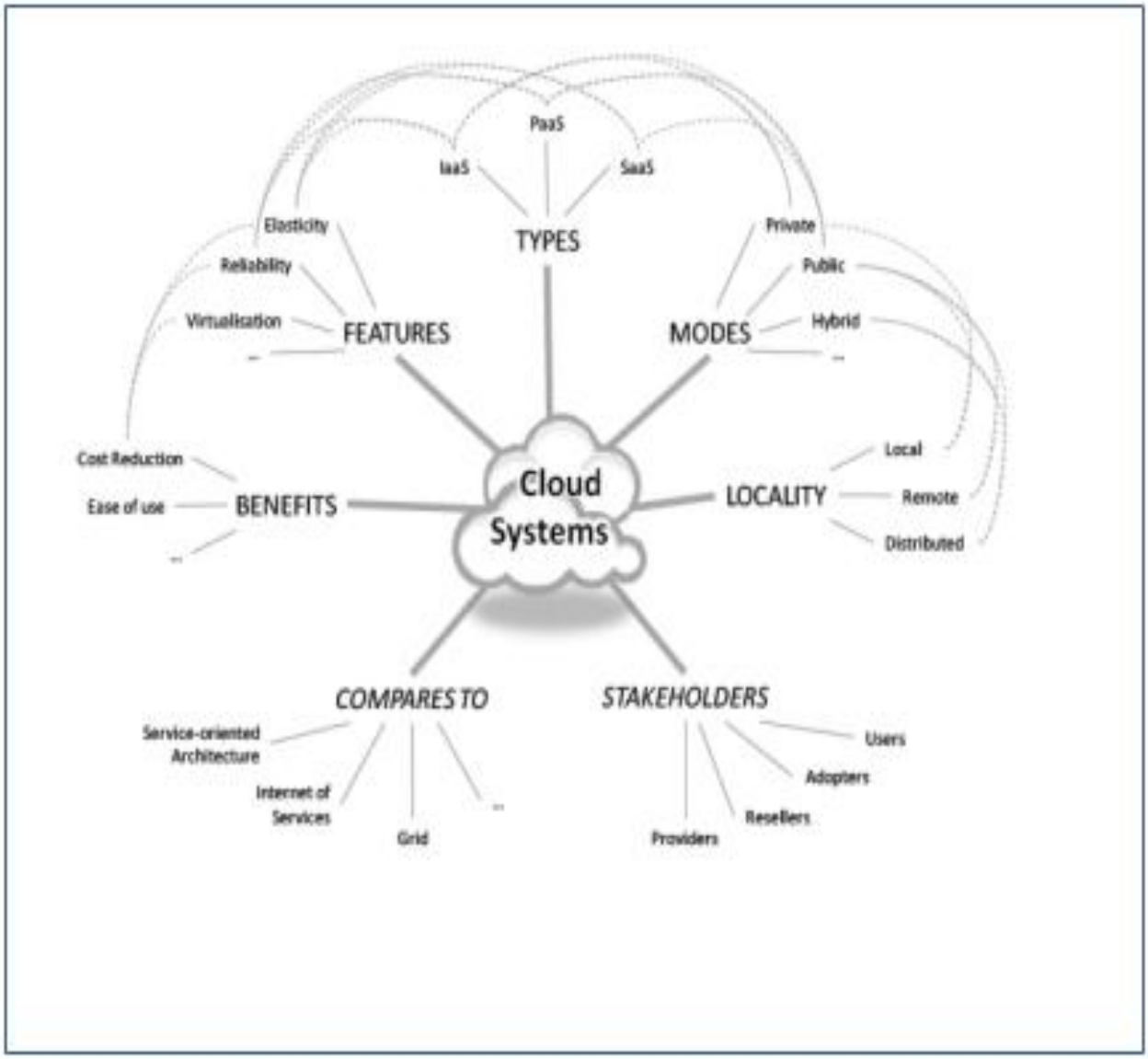

Figure 2-7 Main Aspects Forming a Cloud System

2.3 Components of Cloud Computing

According to [24], there are three main elements in Cloud solution; these are Clients, Data Centres, and Distributed servers. Each element has got its own specific purpose and plays significant role in delivering cloud based application. Now, let's look these elements closer.

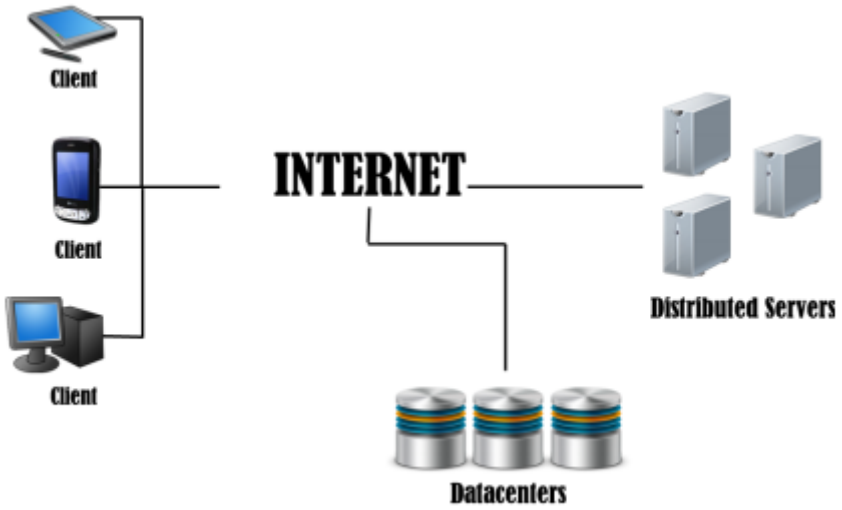

Figure 2-8 Three main elements of Cloud

- **Clients:** is simply a service requester from servers for any service through networks. So, while we think physically, Clients are the devices such as Mobile Phones, Laptops, Tablet Computers, PDAs, or typical desktop computers that the end users interact with to use their resources and access any data on the Cloud.
- **Data Centers:** Data Center is a collection of physical servers where the applications which are subscribed by client are stored. A data center usually includes backup equipment, power supplies, data communication connections, environmental controls such as air cooling, fire suppressions, etc. The main purpose of Data Center is to store and provide the circulation of data that will be used by clients and distributed servers on the Cloud.
- **Distributed Servers:** Distributed servers are a collection of physical servers which are working together as one system through network. By distribution, Servers do not have to be in the same location. They can be geographically in different locations but can work in one common system. This gives more flexibility in options and security according to Antony T. Velte because, if something happens to part of servers at one location, causing a failure, the service would still be accessed through another location. In addition, if Cloud needs more physical hardware, Cloud Provider does not have to add more servers into same, or significant location, Provider can add new hardware at another location and make it part of the same.

2.4 Understanding Roles & Responsibilities for cloud computing

According to the ISO/IEC 17789 cloud computing has 3 main roles for cloud computing:

- Cloud service customer
- Cloud service provider
- Cloud service partner

The cloud service provider and the cloud service customer are the most significant roles in the provision and use of cloud services while the cloud service partner is a party engaged in support of the activities of the cloud service customer and/or the cloud service provider.

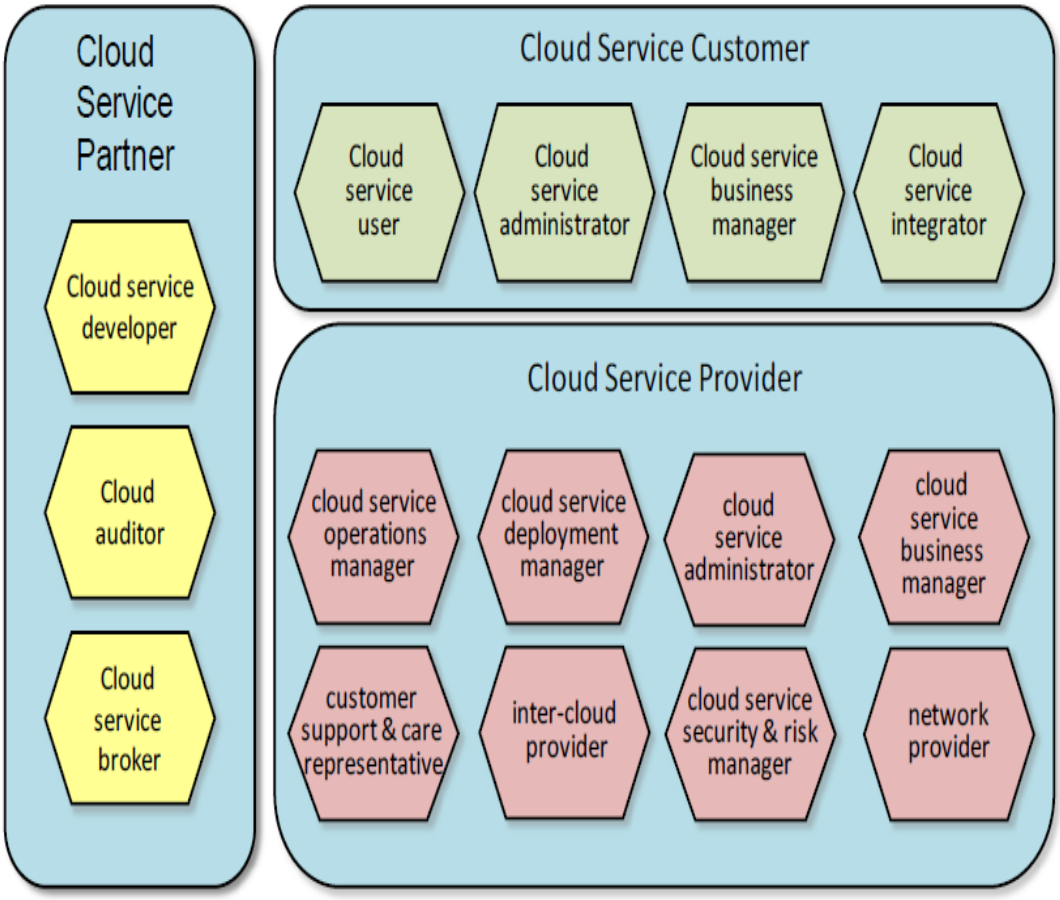

Figure 2-9 Cloud Computing Roles and Sub roles

2.5 Overview of Cloud computing in Different Sector

Cloud computing have a great role in every sectors all over the world for those user of cloud computing. In the field of health sector, educational sector, agricultural sector, market, transport, all organizations and institutions are using cloud computing. We have detailed cloud computing in education sector in 2.5.1

2.5.1 Cloud Computing in the Education Sector

Cloud Computing is based on a collection of many old and few new concepts in several research fields like Service-Oriented Architectures (SOA), distributed and grid computing as well as virtualization, it has created much interest in the last few years. This was a result of its huge potential for substantiating other technological advances while presenting a superior useful advantage over the currently under- utilized resources deployed at data centres .

The education sector has traditionally been cautious to embrace new technology due to factors such as cost and risk. But the 21stcentury presents new financial, demographic and industrial pressures to the education sector that are making it imperative for schools and universities to align themselves to the latest technology. In such a scenario, the Cloud could offer a highly acceptable solution to the education sector, helping it adopt evolving technologies without the burden of excessive cost and complexity.

The Cloud has already started making its presence felt in the education sector globally. Schools have started leveraging the Cloud for student e-Mails, collaboration tools and virtual desktops.

The applications and benefits of cloud computing are practically numerous. It is mostly of low-cost or even free in some cases (e.g., Google Apps and Microsoft's Education Software Solutions). The huge initial cost of educational technology infrastructure is no more a major barrier to technology adoption by educators. Cloud computing does not require expensive computers with large hard disk space and software or their upgrades, thus minimizing the usual resource costs. Consumers get operational benefits, automated update of software, more storage, and a high level of flexibility, reliability and security. Thus, they can save on hardware, physical space, IT support, and overheads. An immediate significant consequence of all these benefits is the opportunity for large scale adoption of technology by all educational and training institutions, and it can dramatically increase opportunity for collaboration amongst leaners and interaction with the tutor resulting in an improved pedagogy [25].It can increase student engagement, motivation and their enthusiastic response to the learning tasks at hand, and thus it can increase the time students are learning without increasing the cost. Increased collaboration and engagement can enhance the quality of learning and also help students to develop the 21st century skills that they need to be able to survive in today's world. Thus cloud computing provides not only cost effectiveness but also pedagogical benefits.

Cloud computing offers many benefits to e-learning solutions by providing the infrastructure, platform and educational services directly through cloud providers and by using virtualization, centralized data storage and facilities for data access monitoring [26].

Cloud computing has been presents significant opportunities for the education sector from different angels. Following are some of the opportunities that can be gained because of implementing cloud computing in the sector.

2.5.1.1 The Cloud offers an opportunity to reduce IT costs:

Cloud computing, and cloud services in particular, can offer the Research and Education sector huge opportunities to both maximize effectiveness and reduce the capital investment. They offer the Research and Education community a number of key benefits like reduced capital costs and reduced support costs.

2.5.1.2 The cloud reduces management burden for schools

In the Cloud, the deployment and maintenance of software, and IT infrastructure would be the responsibility of the Cloud services provider. Hence, schools that use Learning Management Systems (LMSs) or Student Information System (SISs) as SaaS need not worry about annual upgrades of software. Schools offering online distance education programs need to be concerned only about the content of the programs. Technical issues related to the hosting and maintenance of online portals would be the responsibility of the Cloud services provider. And also schools that encourage research and development activities need dedicated IT teams for maintaining data centres and server farms. With The Cloud offering PaaS and IaaS, maintenance of IT infrastructure is taken up by the Cloud service providers.

2.5.1.3 The cloud offers collaboration in education

Sharing documents leads directly to collaborating on documents. To many users, this is one of the most important advantages of cloud computing.

For example, a colleague in Addis Ababa office, and a consultant in Europe all need to work together on an important project. Before cloud computing, you had to email the relevant documents from one user to another, and work on them sequentially. Not so with cloud computing. Now each of you can access the project's documents simultaneously; the editing done by one user makes automatically reflected in what the other users see onscreen. It's all possible, of course, because the documents are hosted in the cloud, not on any of your individual computers. All you need is a computer with an Internet connection, and you're collaborating.

Of course, easier group collaboration means faster completion of most group projects, with full participation from all involved. It also enables group projects across different geographic locations. No longer does the group have to reside in a single office for best effect. With cloud computing, anyone anywhere can collaborate in real time.

2.5.1.4 E-books replacing textbooks thus reducing Students' burden

With textbooks getting dearer with time, education has become fairly expensive for students with respect to the costs associated with buying textbooks. The contents of textbooks also require periodic revisions. With technology all the books can be put online and accessed through personal computers, tablets, laptops, cell phones, etc. Instead of buying a number of books over the entire course duration, students would be required to only invest in just one of these devices to access the Cloud.

2.5.1.5 Opening up research options

One of the prominent features of cloud computing is 'resource pooling', as it allows users to share common resources. This feature opens up new opportunities for research. The Cloud can support schools encouraging research and development by reducing the costs involved with computation, as most of the research requires capacity computing, with systems that share their computing power with several users.

2.5.2 Cloud computing challenges in the education sector

Regardless of various advantages that the Cloud has to offer to the education sector, there are still challenges that need to be overcome to leverage its complete potential in education sector.

2.5.2.1 Security and Privacy Challenges

The security of cloud, and associated privacy concerns, give many organizations pause as they think through their particular cloud computing concerns. Security and privacy concerns include physical security and simple access to facilities and equipment, as well as logical security, industry compliance requirements, auditability, and more. Security and Privacy issues are raised by legal officers and senior administrators more than the end users students, as they are not aware of issue of privacy. Educators and research communities are concerned about sensitive research data being stored beyond the bounds of their university. Further, high-end usage of the Cloud such as admission process and online assessment have issues related to validation, information, and identity theft.

2.5.2.2 Governance, Service Level Agreements and Quality of Service

A critical set of potential cloud obstacles include governance, service level agreements (SLA), and overall quality of service (QoS) assurance. Much as governance dominates the SOA discussion in ensuring appropriate end-to-end governance across the IT and services lifecycle in support of SOA initiatives, governance for cloud must also include this.

2.5.2.3 Reliability and Trust:

Cloud outages are well documented and highly publicized, especially when the primary proponents of cloud computing, such as Amazon, Google, Rack space, and others experience such challenges. If the cloud dials tone cannot be assured, such that cloud consumers know that their cloud-enabled resources will always be there for them, and then cloud will be relegated to niche needs where network availability can be ensured. Cloud, like SOA, is a trust-based model where lack of trust will severely cripple cloud adoption by the masses.

2.5.2.4 Cloud integration and interoperability:

The integration and interoperability of private to private clouds, public to private clouds, public to public, and hybrids poses a great challenge in the absence of industry standards for APIs and cloud interfaces, interoperability standards, and related technical standards.

2.5.2.5 Cross-cloud composition, collaboration, and orchestration of applications:

The concept of composing distributed business applications across clouds, orchestrating business processes from services hosted in different clouds, both private and public, and of integrating multiple hybrid clouds together into a seamless business application fabric is new. There will be progress on these fronts as industry standards emerge to address these potential needs.

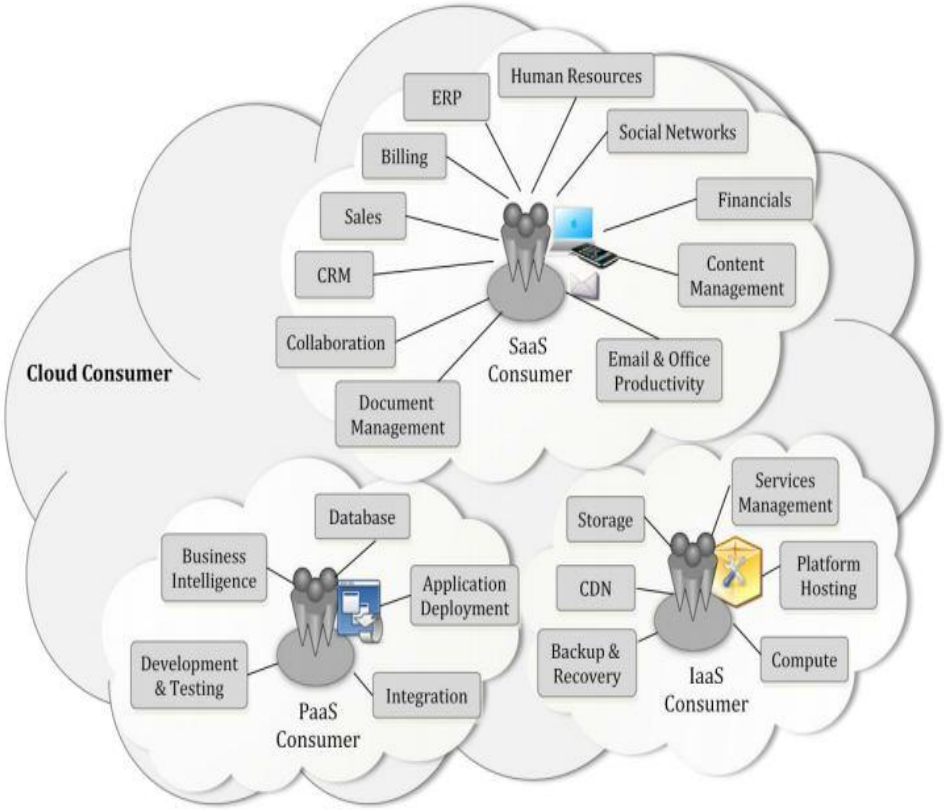

Figure 2-10 Example of Services Available to a Cloud User

2.6 Cloud computing Architecture

When we are talking about cloud computing Architecture we are talking about the components and sub components that build or create cloud computing. It consist of a front end platform (client, mobile device), back end platforms (servers, storage), a cloud based delivery, and a network (Internet, Intranet, Inter cloud). The Combination of these components make up cloud computing architecture .Cloud computing Architecture has three main layers such as software as a service (saas), platform as a service (paas), and infrastructure as a service (Iaas) which are commonly known as cloud services Model. When users use software as a service they do not think about the platform and infrastructure they used and do not think about platform and software when using the infrastructure as well. When users use the platform as a service they do not think about software a service and infrastructure as a service .These have their own functionality in differently and they are indicated in the following Fig with examples listed respectively.

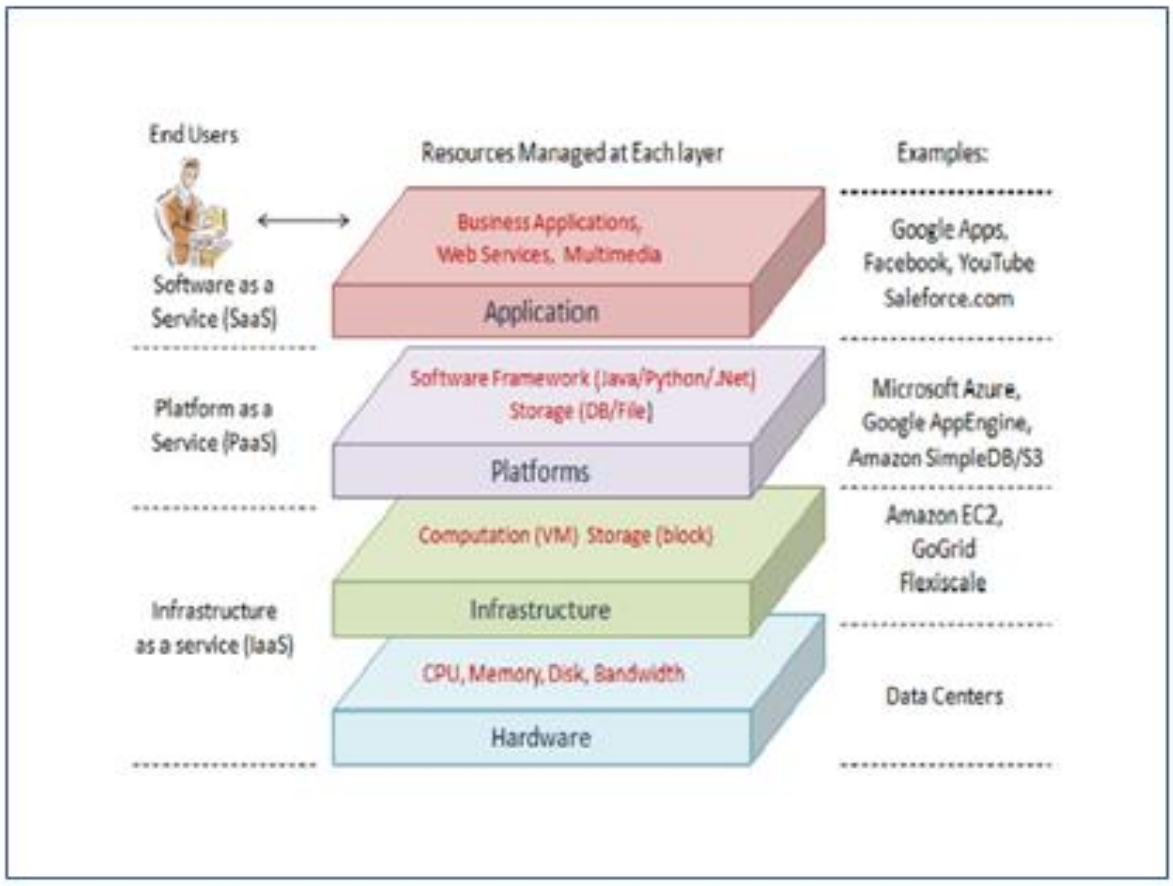

Figure 2-11 Cloud Computing Architecture [28]

2.7 Cloud Monitoring

The operation must ensure the SLA compliance by achieving the quality of Service (QoS) targets. The continuous resource monitoring is desirable in order to ensure that the developed software and hard ware resources run at the required level to satisfy the SLA. This process involving about detecting the running resources and detecting abnormal system behavior according to the service level agreement.

2.8 Related Works

The aim of the researcher in [27] was monitoring of cloud services including Infrastructure layer which covers monitoring of both physical and virtual resources energy and security, Network layer which covers IP service, throughput, performance and reliability, Application layer which covers availability, efficiency, reliability and safety of a service and Customer/provider layer in which service level agreement is included.

Regarding the management of services, an SLA acts as a valuable auditing instrument both for clients and service providers. The verification of SLA compliance is a cross-layer task spanning

(covering) all the layers described above The general term cloud monitoring covers monitoring those layers, however the target foundation of the researcher was SLA auditing and accounting (usage and cost) of a specific service.

The researcher in [28], tried to use average of CPU usage as a metric for monitoring the SLA violations that was not allocated to an application when requested. Monitoring cloud services usage is also relevant due to the elastic nature of cloud environments, associated with the business model "pay-as-you-go", therefore, measuring use and cost become vital aspects [29]. The researcher [30] proposed framework SLA assessment that presents measurement of the quality of services KPIs systematically which included resources utilization efficiency and the response time. Since there was no recommendation for the customers by the decision support system (Cloud broker), they unnecessarily over requested some of the resources and hence they were not fully utilized. This problems makes the researchers to do the research. Analytical Hierarchy Process (AHP) has been applied to evaluate the Cloud services and ranking the providers by comparing each value result with efficient KPIs. The customers' requirements have been met by the provider despite of the fact that some of resources were not fully utilized. As a result the researcher suggested that cloud broker and Auditor are needed to match requirements and resource so that there will be no waste of resources.

According to [8] the researchers tried to study cloud security model based on SLA (Service Level Agreement) . From the security challenges of cloud computing , cloud computing security threats was analyzed comprehensively, the service level agreements was expanded, the CSLA (Cloud Service Level Agreement) structure was proposed, the cloud security reference framework was designed, the cloud security level architecture and cloud services pricing and charging models was proposed.

Alexander Keller & Heiko Ludwig presented web service level agreement for define and monitor web services having three components which are parties, services definition and service objection. Supporting party and signatory party are discussed in WSLA. Service definition component define the service objectives and its parameters. The obligation component defines the service guarantees. WSLA is a good framework for cloud computing but it does not discuss the performance of services and how to convert the low level metrics to high level parameter to measure them [31].

Christpher Redl, Ivan Breskovic, Ivona and Schahram worked on Automatic service level agreement. It uses past knowledge, user requirement and characteristics evaluation to automatically match any SLA. The developed framework communicates two service level agreement which is defined by market participant. Before creating SLA, consumers and providers put their templates of SLA in cloud market. SLA mapping are used to match two SLA templates. SLA mapping convert the same meaning word to one word and different units to one unit. It automatically searches similar SLA from the market. ASLA reduce the market cost. Automatic SLA save too much time and cost.

Another researchers done their research on personalized service level agreement. It acts as being broker between service provider and service consumers. PSLA collects different kinds of services from different service providers and provides these services to further customers according to their needs. Without PSLA customers must have to translate their resources needs according to service provider's needs. PLSA solved this problem. Users only upload their requirement to PSLA broker and then it provides services according to customer's needs. Personalized SLA solved the major problem of service provision but Personalized SLA is combination of services and each service has different QoS and SLA [32].

According to [33] they have done their research that gives a platform to convert low level metrics to high level parameters for measurement. The platform contains the repository of mapped metrics and agreed SLA. When any new customer request comes to the system this framework mapped it with mapped metrics repository. This framework saves the customers from costly SLAs and futures failures. LoM2HiS is automatic SLA management and enforcement framework.

Feng chuan Zhu, Hao Li and Jan Lu worked on Bank cloud service level agreement. In Cloud Bank Service Level Agreement services are taken by service provider and this services are then stored in services pool. According to the researchers one SLA is signed by service provider and cloud bank, and the second SLA is signed between service consumer and cloud bank. CBSLA provides supporting services. Cloud Bank is used for the supervision of SLA. It provides the quality of service aspects. Cloud Bank works as broker of Cloud Services.

Table 1 Conclusion of Related Work

Author(s)	***Proposed Solution***	***Contribution***	***Gap***
Alexander Keller & Heiko Ludwig	Framework to Service level agreement for define and monitor web services.	WSLA enable service provider and consumer to define wide variety of parameters of SLA and also enable them how to relate and measure them	The framework does not discuss the performance of services and how to convert the low level metrics to high level parameter to measure them and SLA is not monitored consumer side
Christpher Redl, Ivan Breskovic, Ivona and Schahram	Automating service level agreement	ASLA reduce the Market cost. Automatic SLA save too much our time and cost	The framework does not give any guarantee to ensure Quality of Service and performance and SLA is not monitored
Teixeira, Jennifer Ortizy and Victor	personalized service level agreement	PSLA hire different type of services from Different service provider and then provide these services to further customers according to their needs.	Personalized SLA is combination of services and each service has different QoS and SLA. SLA is not monitored consumer side

VincentC.Emeakaroha, IvonaBrandic, Michael Maurer and Schahram DAustadar	SLA framework gives a platform to convert low level metrics to high level parameters for measurement	This infrastructure contains the repository of Mapped metrics and agreed SLA. When any new customer request comes to the system this framework mapped it with mapped metrics repository	The framework does not discuss how to measure these metrics. How to analyze and how to integrate it with any SLA for implementation. SLA is not monitored consumer side.
Zhu, Tengchaun, Li, Hao and Lu	Framework for Cloud Banking Service level agreement	CBSLA provides supporting services	Pooling services is also a difficult task so it is very complex to use and implement

We have tried to identify number of related works to our research. Those related works are important to understand and identify problems and solution before our research is done. It also important to ignore redundancy in our researcher with past worked research. Even though those related works give us clue and understanding our research is different ways. There are different gaps in those related work and we discussed in **table 2**. In Our research we developed a monitoring framework which monitor the SLA (Service Level Agreement) between the two parties cloud provider and consumers. It is not biased, in a sense the cloud provider and consumer will be benefited from the framework equally. We have done a prototype that will be installed in the database of cloud provider side and this prototype works according to the need of providers and consumers, if any violation of SLA occurred d alert is generated to the consumers.

CHAPTER THREE

3. RESEARCH METHODOLOGY AND PROPOSED SOLUTION

3.1 Research Methodology

In order to achieve the specific and general objectives of the study and answer the research questions, the following research methodology with selected methods were used.

3.1.1 Literature review

In order to justify the identified research question and gape in currently available solutions, a number of previously related research literatures such as journal papers, research thesis and dissertation, research project reports were reviewed and analysed in the context of the target research problem. In addition to understand the domain with available technologies concepts from books and sites were studied and referenced. Using review of literature, it was also analysed that what type of research strategies, methods and tools were used by the salient researchers contributed in the same or allied area of studies. Based on the detailed survey, different techniques and tools which were relevant for the current research study were critically analysed and evaluated with detailed analysis of parameters considered for selecting the appropriate tools and methods in service level agreement monitoring. Review of literature covered concepts and related research works have been covered in Chapter 2.

3.1.2 Selection of Tools & Techniques

Different software and designing tools and techniques were used to accomplish the research tasks. Open source Cloud Computing tool i.e. CloudSim was selected for simulation of monitoring service level agreement from cloud data centre. Java software platform Net beans has been selected. MySQL data base was selected and we used E-DrawMax modelling tool. Detailed of tools and techniques has been covered result Analysis and discussion.

CloudSim cloud sim is the most important simulator in cloud computing Environment. It used java programming language. This simulator is the base for another cloud simulators like cloud Analysist, iCan cloud and for others. Cloud Sim was suitable for my research.

Java NetBeans cloud sim is running on Java platform, net beans or eclipse, but in our case we selected java NetBeans.

MySQL we used MySQL server to create database and tables to connect different components of our framework.

E-DrawMax we used E-Draw for modelling and designing our framework. It is flexible and standard

We have tried to show our methodology from starting to the end of our research using the following figure.

3.1.3 An Overview of the Research Methodology

The research needed flow of ideas and organization of each document in a well manner. So we have followed the sequence of activities to complete our research. These include understanding of existing cloud service level agreement, identification of their problems in detail, literature review, designing SLA Monitoring framework, Implementation of SLA Monitoring, evaluation of SLA Monitoring and SLA Monitoring Product. We have used the following figure to illustrate the research methodology overview in figure 3-1.

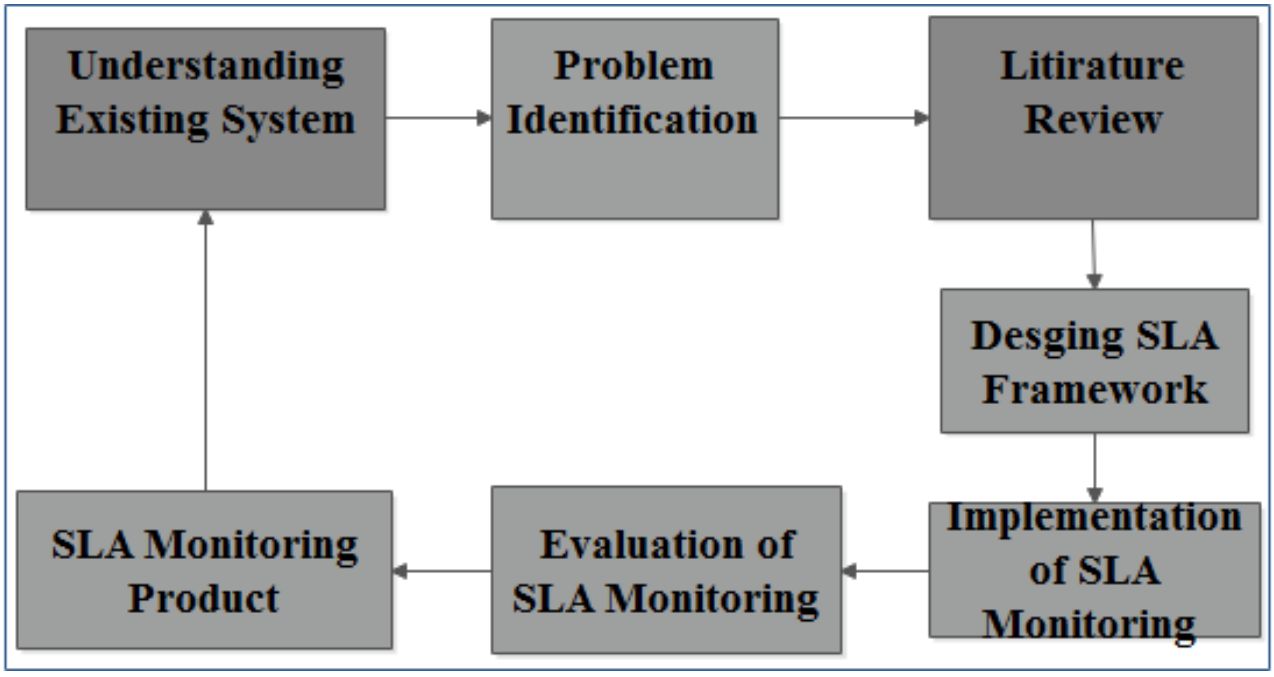

Figure 3-1 Research Methodology

3.1.3.1 Understanding existing system

Before starting to do my research I have been understood the current amazon web service level agreement in details. When I read in details there are an obligation for both service consumers and service provider differently. The services are offered according to the service level agreement which is signed between the two parties (service consumers and service providers). So by reading the Amazon web service level agreement I understood what is service level agreement.

3.1.3.2 Problem Identification

After I understand the service level agreement ,I have identify what were the main problems in the current service level agreement as I mentioned them in the statement of the problems in chapter one.

3.1.3.3 Literature Review

To understand more about the service level agreement and to know what is done already about the service level agreement, I had revised different literature, document and web sites.

3.1.3.4 Designing SLA Monitoring framework

While reviewing literature review we knew what are done around our research area and Cleary understand the objectives of our research. This taken us to the designing of the proposed solution. The proposed solution was indicated using the framework and we used E-Draw max design tool. Our framework is the best solution for the problem we identified and decided to do in the scope in this research.

3.1.3.5 Implementation of SLA Monitoring Framework

The framework was implemented using different languages like java net beans 8.1, MySQL database, and CloudSim simulator. This is the phase in which we changed the design into the coding.

3.1.3.6 Evaluation of the Framework

The frameworks have been evaluated in order to check whether the proposed framework is the correct one to solve the problem or not. We had used different evaluation mechanism in our research and we had discussed it in chapter 6 in detail.

3.1.3.7 SLA Monitoring Product

The last but not least phase of our methodology is the end product of the research. This deal with the result of the proposed solution which was solved the problem listed in the scope.

3.1.3.8 Conclusion &Recommendation

This part was the end of our research: it concluded what are done in this research as well as recommends. At the last of our research we recommended to use our framework in order to monitor service level agreement in a well manner.

3.2 Overview of Proposed solution

As we had mentioned the current problems face in cloud computing in chapter on of our research there are a lot of problems. The selected problems are taken to solve in the scope of the research in the same chapter. We had developed framework and implemented the prototype to solve the problem identified in this research. We detailed about proposed solution in chapter five of our research.

3.3 Research Design

For any research studies, there are four commonly used approaches which are quantitative, qualitative, design oriented and mixed method. The research design refers to the overall strategy that you choose to integrate the different components of the study in a coherent and logical way, thereby, ensuring you will effectively address the research problem; it constitutes the blueprint for the collection, measurement, and analysis of data. Qualitative Research is primarily exploratory research. It is used to gain an understanding of underlying reasons, opinions, and motivations. It provides insights into the problem or helps to develop ideas or hypotheses for potential quantitative research. Qualitative Research is also used to uncover trends in thought and opinions, and dive deeper into the problem. Qualitative data collection methods vary using unstructured or semi-structured techniques.

My research is a scientific research that focused on the science of cloud computing environment, cloud providers and cloud consumers. The research is qualitative because qualitative research explain about the behaviour of the research and shows the quality of services.

So our research was qualitative research because we have been collected data from different sources that are listed in qualitative type of the research.

3.4 Process Model

In software Engineering, process model is a description of the sequence of activities carried out in the software project and the relative order of these activities. There are several process models which include prototyping, unified process, incremental and agile. We can rarely combine process models or use together in software development process [34]. In this study, prototyping and Unified process models were used together to accomplish the overall objectives. Prototyping is about creating some parts of the system being developed. It is not a

standalone and complete development methodology rather it is a model to test specific features in the situation of complete methodology like unified process.

According [35] to **Prototyping** is a process model that offers a solution to the problem and it is the practice of building an early version of a system which shows particular features of the actual system. There are two types of prototyping which are throwaway and evolutionary prototyping. In **throwaway prototyping** the different parts of the system are developed and then will be ignored after demonstrating the basic functions that the system will carry out. Thus the final deliverable system will differ from the prototype. But **evolutionary prototyping** technique implements the prototype of the system that demonstrates its basic features and finally it will be part of the final deliverable system.

The **unified process** (UP) or rational unified process (RUP) is the most recent software development process model which provides overall plan and strategy to be used as part of it. It has two purposes, the first one is meeting user's need by the use of techniques like use cases, iteration and emphasis on the architecture of the software [35]. The second purpose is accommodating risks by making use of iterative technique. UP consists of four phases: inception, elaboration, construction and transition. **Inception phase** consists of a feasibility study to establish whether the proposed system is worthwhile. In **elaboration phase**, the creation of general architecture of the system and requirements lists is completed. In **construction phase**, the development of the actual system, in this case the prototype of the actual system is performed [35]. The **transition phase** involves making the system in to use by installing and running this system. Generally, the unified process model importantly uses use cases, iteration and software architecture techniques; and the prototyping model is concerned with the development of incomplete version of the system. Therefore these are the reasons to select these models for this study.

CHAPTER FOUR

4. SELECTION OF CLOUD SIMULATORS AND CLOUD PROVIDERS

4.1 Introduction

This chapter provided an overview of the selected cloud simulations on this research and their comparison base on different criteria. We were conducted an open source and commercials for comparison and only selected the open source one because only freely available tools were needed for this study as an academic purpose. On the other hand we studied Amazon Web Service as a cloud providers to detailed about service level agreement over cloud computing. As a result, we have identified around 13 cloud simulators like cloud sim, cloud Analysist, iCan cloud, open stack, open nebula and so on. We have selected only the 5 tools to study and 3 for comparison. The tools selection process is shown in Fig 4.1.

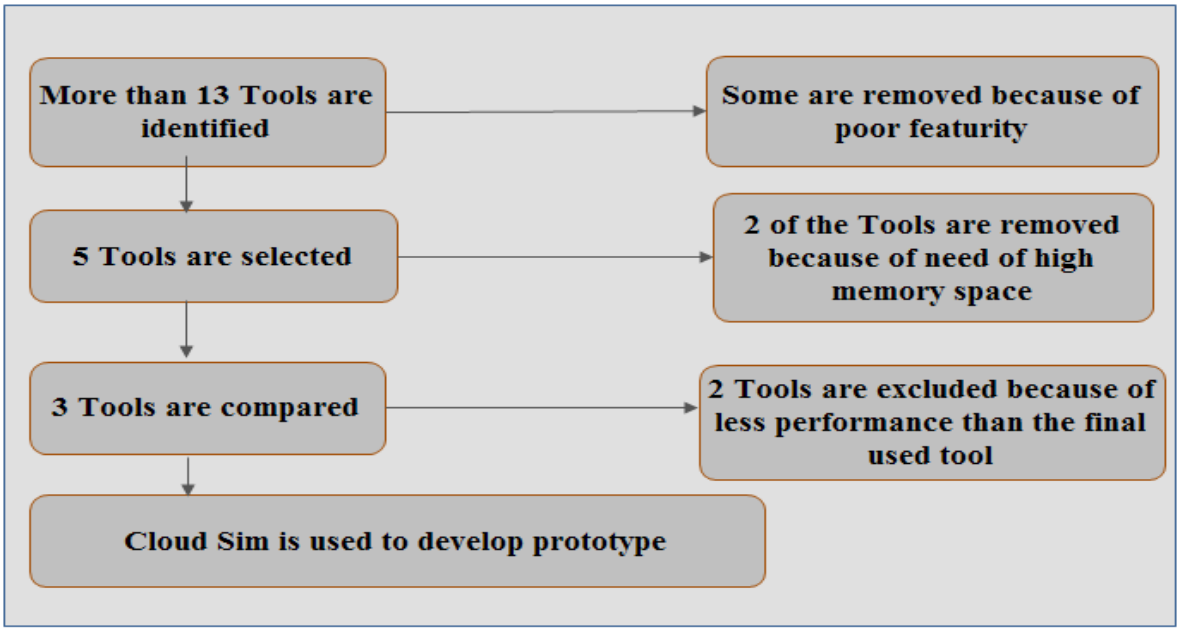

Figure 4-1 Selection of tools flowchart

4.2 Analysis and Comparison of cloud Tools to monitor Service Level Agreement

We have searched a lot of tools available on Google in order to identify which tool is suitable for our research solution in service level agreement monitoring over cloud. At first we have identified around thirteen tools, after rejecting some tools due to complexity and poor feature, we have selected the 5 tools such as Open nebula, open stack, Cloud Analysist, iCan cloud and Cloud Sim. Out of these tools, we have taken only the three tools for comparison and analysis. Namely OpenStack, Open Nebula and CloudSim. Testing all the simulator is time taken and these three are used currently by different researchers all over the world.

We have used five criteria such as programing language they support, compatibility of them with different operating system, cost, memory space they need to install, complexity of the components of the simulator. The list of cloud simulation that we have compared are: Open nebula, open stack and CloudSim. A brief comparison among these cloud tools is shown in table 4.1.

Feature	***Open Nebula***	***OpenStack***	***Cloud Sim***
programing language they support	Open-source cloud computing framework for building private, public and hybrid cloud environments. Its goal is to provide an open, flexible and extensible management layer to automate and orchestrate the operations of existing (on-premises) or remote hardware infrastructure including networking, storage, virtualization, and monitoring and user management. It used Java	It is written in Python and uses many external libraries such as Eventlet (for concurrent programming), Kombu (for AMQP communication), and SQLAlchemy (for database access). [36]	Java and C languages
compatibility of them with different operating system	It is working on Linux operating system like Ubuntu	It is working on Linux operating system like Ubuntu	Windows as well as Linux operating system
cost	It need high cost to install the infrastructures to prepare the laboratory to do the simulation. It is used for real simulation. it consume time, memory space	It need high cost to install the infrastructures to prepare the laboratory to do the simulation. It consume time and memory space.	It is simple to import the simulator easily
memory space they need to install	It needs a large memory space and have different components.	It needs a large memory space.	It does not need large space
complexity of the components of the simulator	It is complex than cloud sim	It is complex than cloudSim	It is not complex to do with

Table 2 comparison of cloud Tools

Based on the above comparison and analysis, we have used CloudSim for our study to implement the prototype of the proposed framework. CloudSim allows a wide variety of choice of environments, flexibility and accessibility of different programming languages including java will benefit us greatly. As a result, this is the best tool for us to spend more of our time on code and studied classes and packages on the simulator. We have used CloudSim version 3.0.3.

4.3 Overview of cloud service providers and their service level agreement

There are number of cloud providers all over the world. Namely Amazon web service, Google ApplEngine, Microsoft Azure Compute, Microsoft Azure Storage, Amazon EC2, Amazon S3 and others. However different cloud providers have different service level agreement and monitoring service level agreement mechanisms are different too. We have studied most of the providers and their service level agreement. Most of the providers monitor service level agreement by third parties and service consumers have no chance to participate on monitoring service level agreement. They signed an agreement to obtain services, hover the service may or may not delivered according to the service level agreement they signed. For example Amazon web service monitor service level agreement using cloud watch application. This application is functional only for the provider. If violation occurred they give credit to the consumers as a penalty. But service level agreement does not have guarantees and quality of services. We have tried to show some of the cloud providers, their service level agreement and mechanism in which they monitor service level agreement using table

Table 3 Illustrated cloud providers and how to monitor service level [37]

Cloud Service Provider	*Service Type*	*Step 1: Discover Service Provider*	*Step 2: Define SLA*	*Step 3: Establish Agreement*	*Step 4: Monitor SLA Violation*	*Step 5: Terminate SLA*	*Step 6: Enforce Penalties for SLA Violation*
Amazon EC2	IaaS Computing	Discover manually (e.g. via web site)	Pre-defined SLA terms and QoS parameters	Pre-defined SLA document by provider	Can use third party monitor systems (e.g. Cloud Watch)	By user, or provider program-matically or manually	Service Credit given by provider
Amazon S3	IaaS (Storage)	Discover manually	Pre-defined SLA terms and QoS parameters	Pre-defined SLA document by provider	Can use third party monitor systems (e.g. Cloud Status)	By user, or provider program-matically or manual-ly	Service Credit given by provider
Microsoft Azure Compute	PaaS	Discover manually (e.g. via web site)	Pre-defined SLA terms and QoS parameters	Pre-defined SLA document by provider	Can use third party monitor systems (e.g. Moni-tis)	By user, or provider program-matically or manual-ly	Service Credit given by provider
Microsoft Azure Storage	PaaS	Discover manually	Pre-defined SLA terms and QoS parameters	Pre-defined SLA document by provider	Can use third party monitor systems (e.g. Moni-tis)	By user, or provider program-matically or manual-ly	Service Credit given by provider

4.4 CloudSim Overview

According to [38] CloudSim is toolkit for simulation of cloud computing written in Java languages. CloudSim is built in CLOUDS (Cloud Computing and Distributed System) Laboratory by the University of Melbourne, Australia.

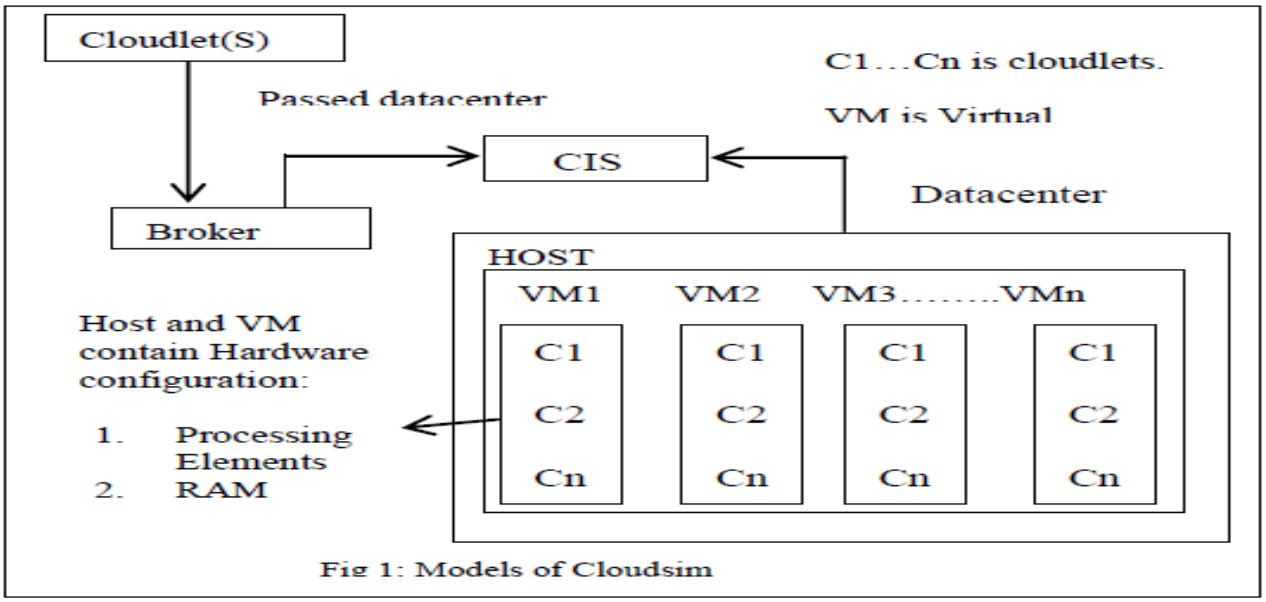

Figure 4-2 an overview of CloudSim

It is used to model data centers, service brokers, scheduling and allocation policies of large-scaled cloud computing platforms. CloudSim uses Java language and is built on top of SimGrid framework. The main components of CloudSim are: **Datacenters, Hosts, Virtual Machines (VM) and Cloudlets**. The data center entity manages a number of host entities. The hosts are assigned to one or more VMs based on a VM allocation policy that should be defined by the Cloud service provider. Here, the VM policy stands for the operations control policies related to VM life cycle such as: provisioning of a host to a VM, VM creation, VM destruction, and VM migration. Similarly, one or more application services can be provisioned within a single VM instance, referred to as application provisioning in the context of Cloud computing. In the context of CloudSim, an entity is an instance of a component. A CloudSim component can be a class (abstract or complete) or set of classes that represent one CloudSim model (data center, host).

Figure 4.3 shows the multi-layered design of the CloudSim software framework and its architectural components. The CloudSim simulation layer provides support for modeling and simulation of virtualized Cloud-based data center environments including dedicated management interfaces for VMs, memory, storage, and bandwidth. The fundamental issues, such as provisioning of hosts to VMs, managing application execution, and monitoring dynamic system state, are handled by this layer [38].

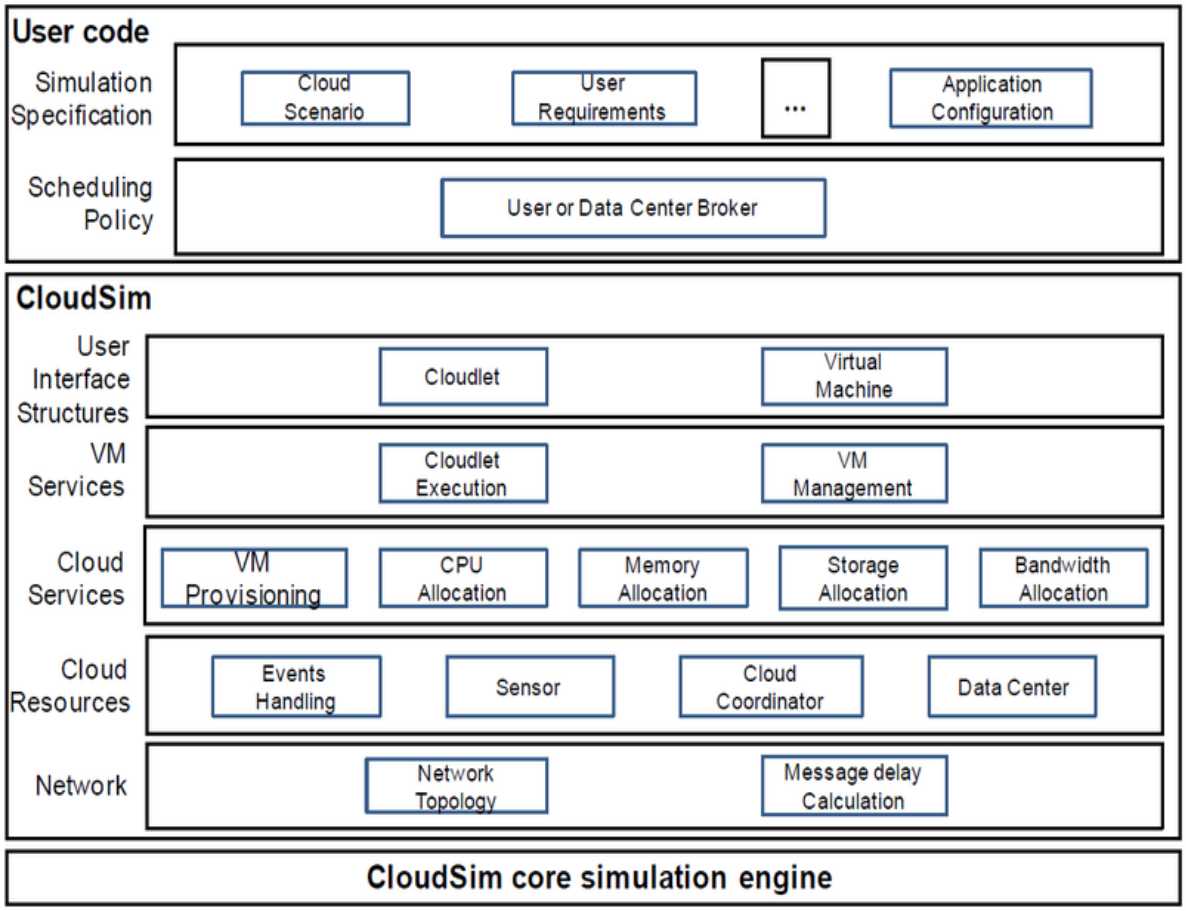

Figure 4-3 Layered CloudSim architecture [38]

The main features of Clouds:

- Backing for modelling and simulation of large scale Cloud computing data centers, virtualized server hosts, energy-aware computational assets, data center network topologies and message-passing applications, federated clouds
- Backing for dynamic insertion of simulation components, end and begin of simulation

- Backing for user-defined policies for allocation of hosts and resources to virtual machines
- Supplies modelling and simulation of large scale Cloud Computing surroundings, including Data centers on a single physical computing node
- An autonomous platform for modelling Clouds, Service Brokers, provisioning, and allotment approaches
- Supplies simulation of network connections among the simulated arrangement components
- Facility for simulation of league cloud surroundings that inter-networks assets from both private and public domains, a feature critical for research studies related to Cloud-

CHAPTER FIVE

5. PROPOSED SOLUTION

In this chapter we detailed with framework how to monitor service level agreement over cloud computing environment from both side (cloud service providers and service consumers).

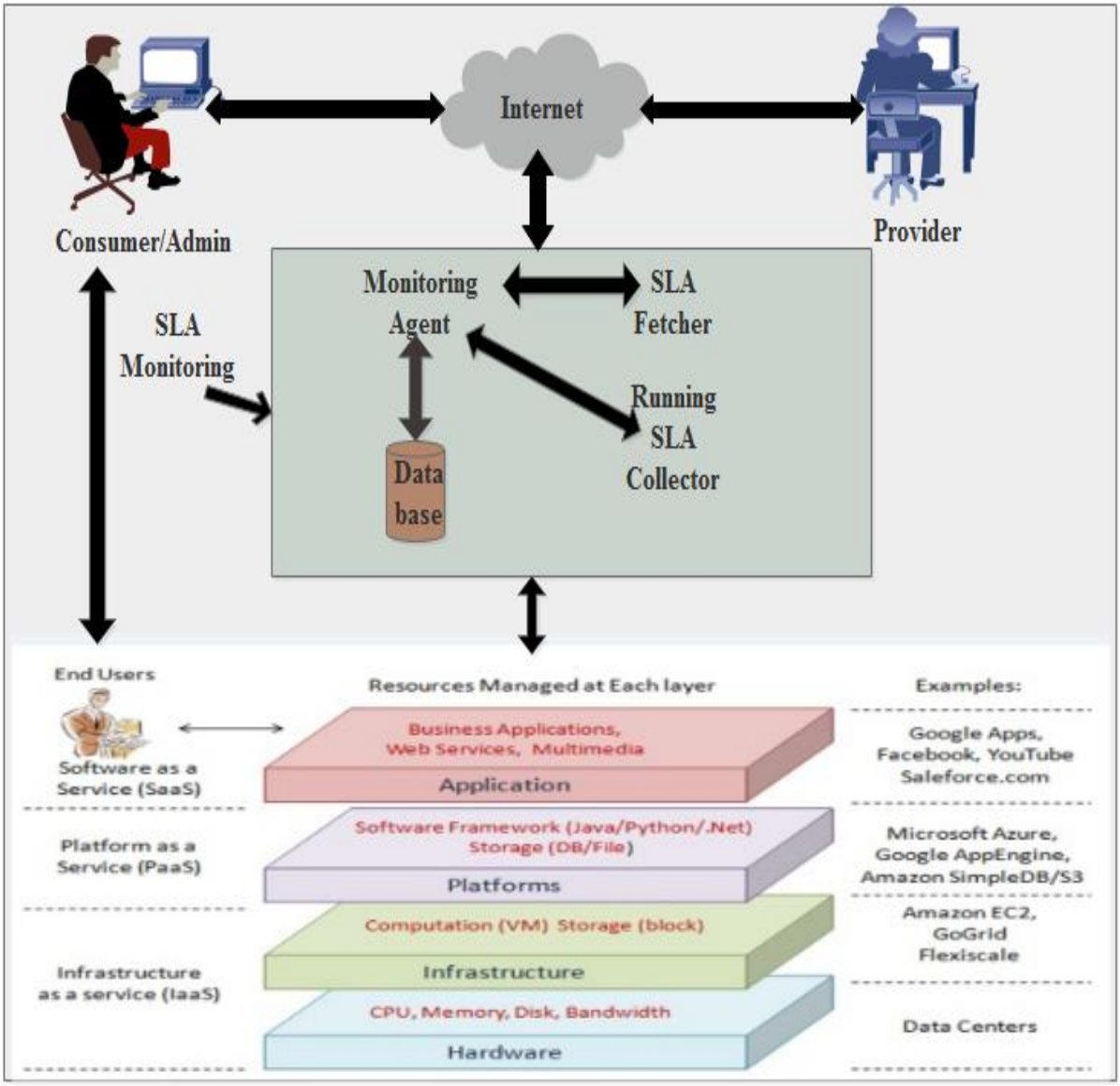

Figure 5-1 Proposed Monitoring Service Level Agreement Framework

5.1 Description of the framework

This framework is used to monitor service level agreement (SLA) over cloud. Before the delivery of the services from cloud provider and consuming services as a user's, there is a contact between the two parties which is known as service level agreement. After service level

agreement is signed between the parties (provider and consumer) they can provide and receive services respectively.

Service level agreement used to guide provider as well as consumers. So it is a core important in cloud environment. However, before this SLA was violated because of different reasons in cloud environment. If SLA is violated the quality of service decreases and as a result consumers are unsatisfied. If consumers are not satisfied, business from provider side will be fail.

In order to overcome the above problem this framework is developed. In this framework there are four components or Agents, the first component is SLA Fetcher component. This component gathers signed SLA from cloud providers to the monitoring agent to compare signed SLA and running SLA at run time.

The second component is Running SLA collector. This component used to gathers information about the running SLA parameters to the monitoring agent. At this point monitoring agent get information from both SLA Fetcher and running SLA collector, the information about the signed SLA and running SLA at run time respectively.

The third component is monitoring agent/ component. This component used to compare the signed SLA and running SLA and stores in to the database. If any violation occurs the alert sends to the cloud provider and service consumer concurrently. As a result this is very important mechanism /technique because the monitoring application is separated from cloud provider as well as cloud service consumer. So the application was worked without interferences of the two parties in a sense it is equal not biased one party from the other party.

This framework is implemented using CloudSim simulator on java NetBeans for running cloud sim, MySQL as a database, and JADE for GUI to show alert if any violation occurs. So our framework is the best solution which increases the benefits of the two parties cloud providers and service consumers in different ways i.e. the confidence of the consumers increases because the guarantee increases since the monitoring agent is work separately not for provider or consumer separately but for both equally.

Based on the above best solution the service provider provide their service in a well manner and can monitor their services as well. In another word service consumers also get more advantage than provider because the alert or notification will come from the monitoring agent not from cloud provider, so this increases the confidence they will be satisfied.

Cloud provider and cloud consumer communicated through internet. So the agreement takes place online. The application installed cloud provider side because the storage is from cloud provider. This application was connected to the internet and all services are connected to the internet through the application.

These service are software as a service, platform as a service and infrastructure as a service. All services must be provided according to the service level agreement signed between the two parties cloud provider and cloud consumer.

We have used cloud sim simulator to run our research. Our application detected whether the service level agreement is violated or not while the simulator started to run.

5.2 Design of our Implementation by Layer

The following figure indicated the implementation of our research layer by layer. As the figure showed below there are three layers.

1) Presentation layer

   Presentation layer is the layer in which an information displayed for the users as well as provider. While cloud sim started to run and if user wanted to check every information about what they access and how much resource they used. In another word if any violation occurred the application of the system displayed notification for the users and well as provider.

2) The second layer is Business Logic Layer

   In this layer there are three components and their communication. These components are SLA fetcher, Monitoring Agent and Running SLA parameters collector.

   SLA fetcher gathered service level agreement from cloud provider.

   Running SLA parameters collector: - This component used to collect running SLA information at run time. It gathers information of SLA parameters at run time to the Monitoring Agent.

   The other and main component of Business Logic Layer is Monitoring Agent, this component used to compare the two components SLA fetcher and Running SLA parameter and displays or sends the violation messages to the presentation layer if any violation occurred.

3) The third layer is data base Layer

   This layer used to store the SLA that signed between cloud provider and cloud consumer before the services are started to offer. This data base contain different tables the defined

in the implementation part of this research. This tables includes resources like data centre, Host, Virtual Machine, cloudlets.

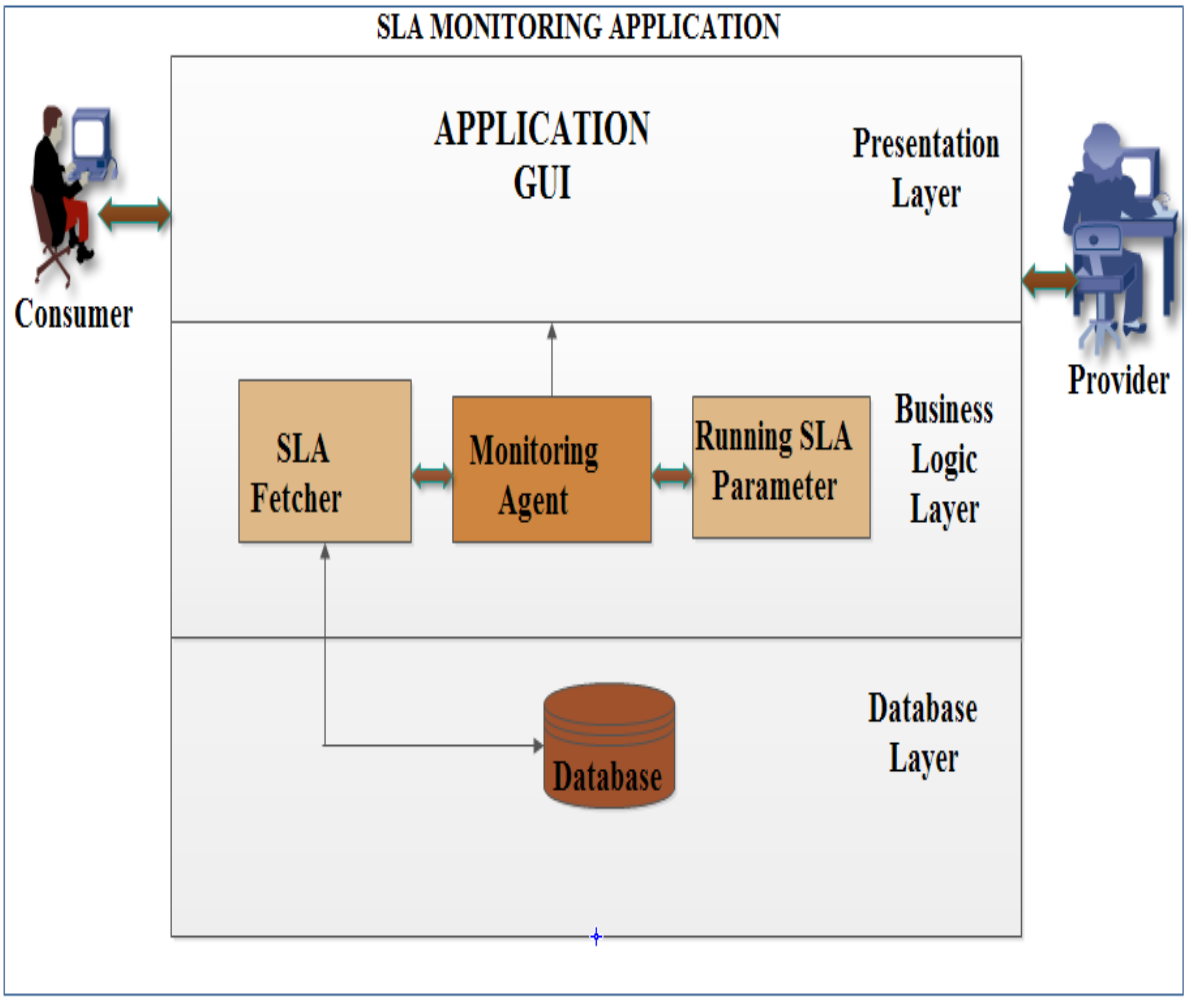

Figure 5-2 Design of SLA Monitoring

The following figure showed that the activity of the SLA fetcher in our prototype. As soon as the simulator started to simulate, the SLA fetcher fetches the service level agreement from cloud provider. Knowing the service level agreement is used to compare the SLA at run time and to check the violation. Figure 5.3 showed this idea.

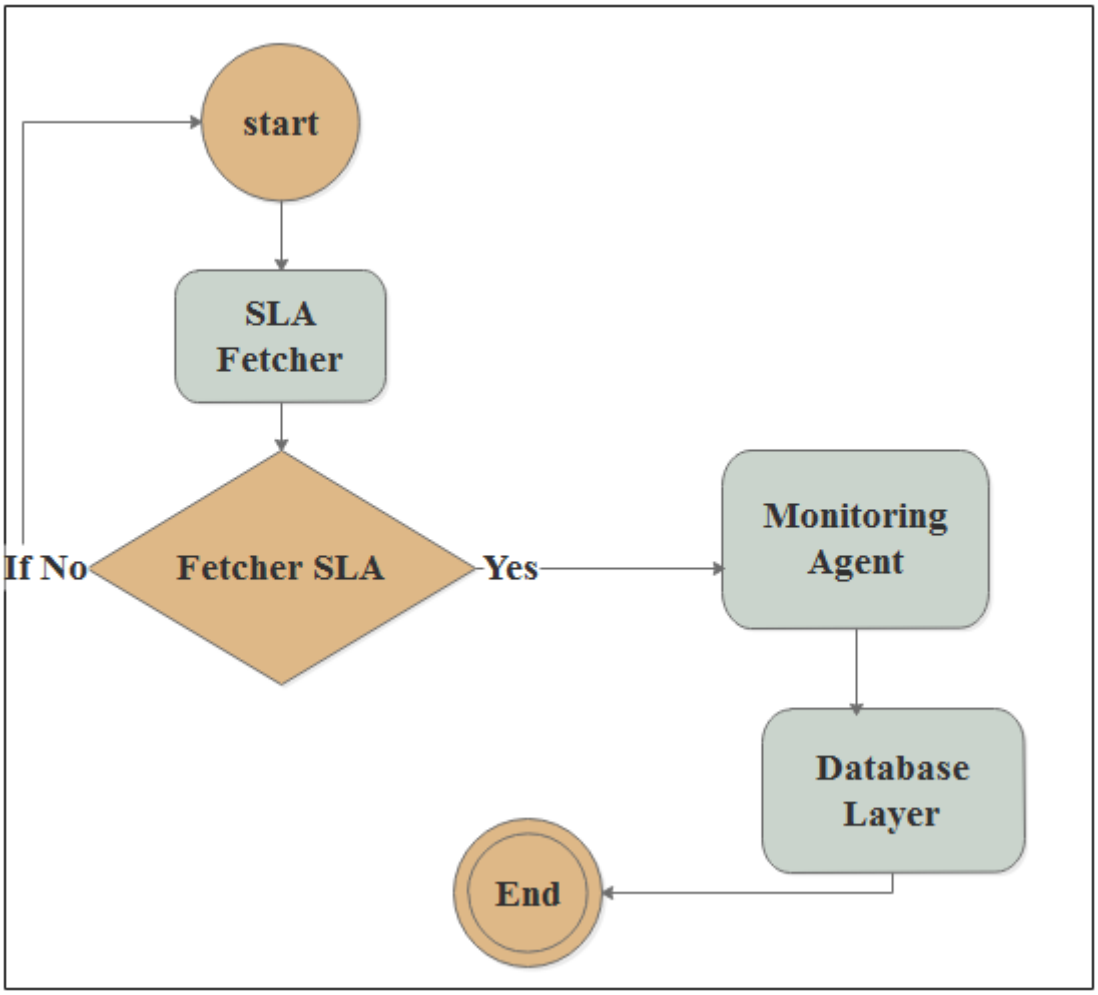

Figure 5-3 Activity Diagram to show the activity of SLA fetcher from database

The following figure showed the activity of Running SLA parameter in the implementation of our prototype. Running SLA parameter collects information at run, time; this is the general information of the allocated resources. So the monitoring agent depends on the two information from SLA fetcher and Running SLA parameter. Figure 5.4 showed this idea.

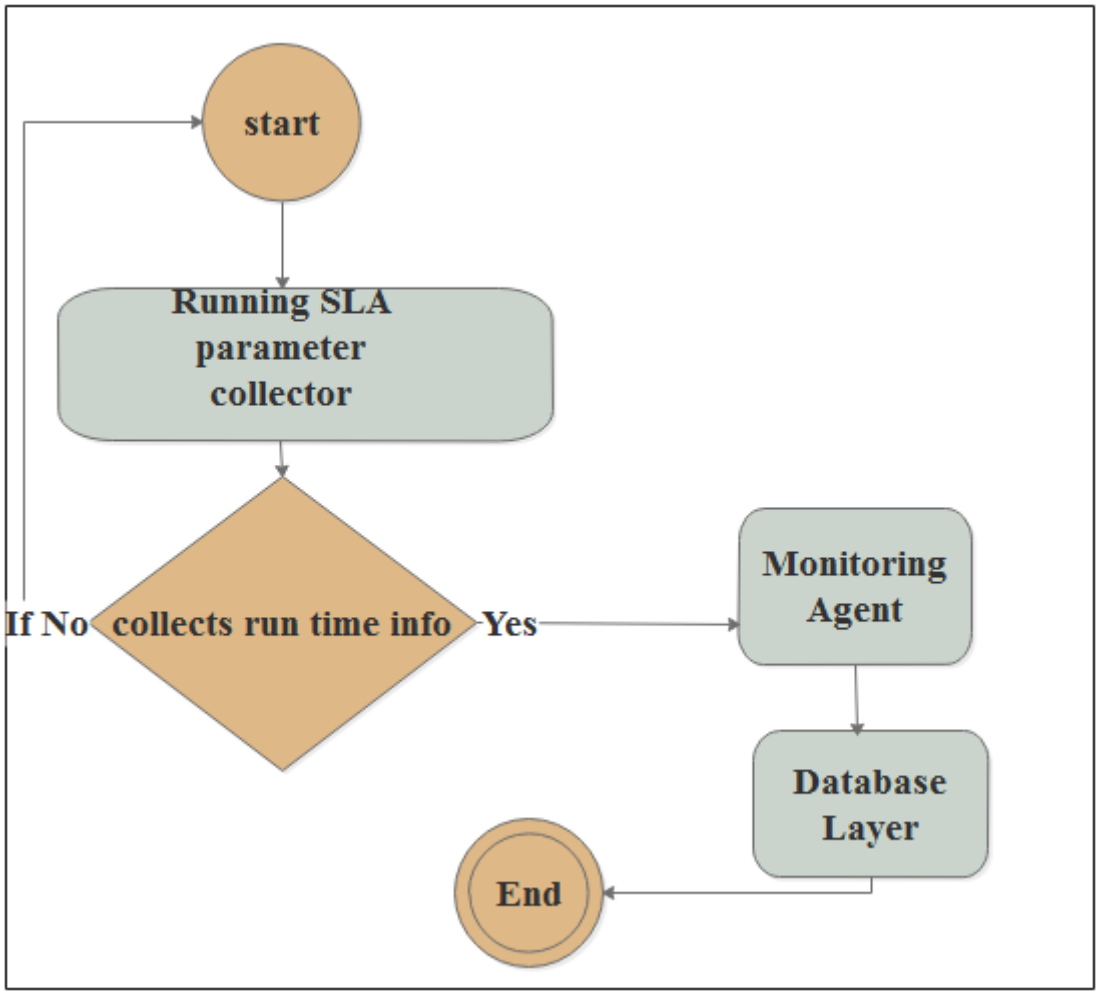

Figure 5-4 Activity Diagram show the activity of running SLA collector

The following showed the activity of the layers in our prototype. There are three layers database layer, Business Logic layer and presentation layer; the information started from database layer and ends with presentation layer. Database layer holds preexisting information, Business logic layer is where the main function of the prototype id done and presentation layer is the layer used to present the full information of work done to the consumer as well as provider in order to keep the guarantee of SLA in cloud environment Figure 5.5 showed this idea.

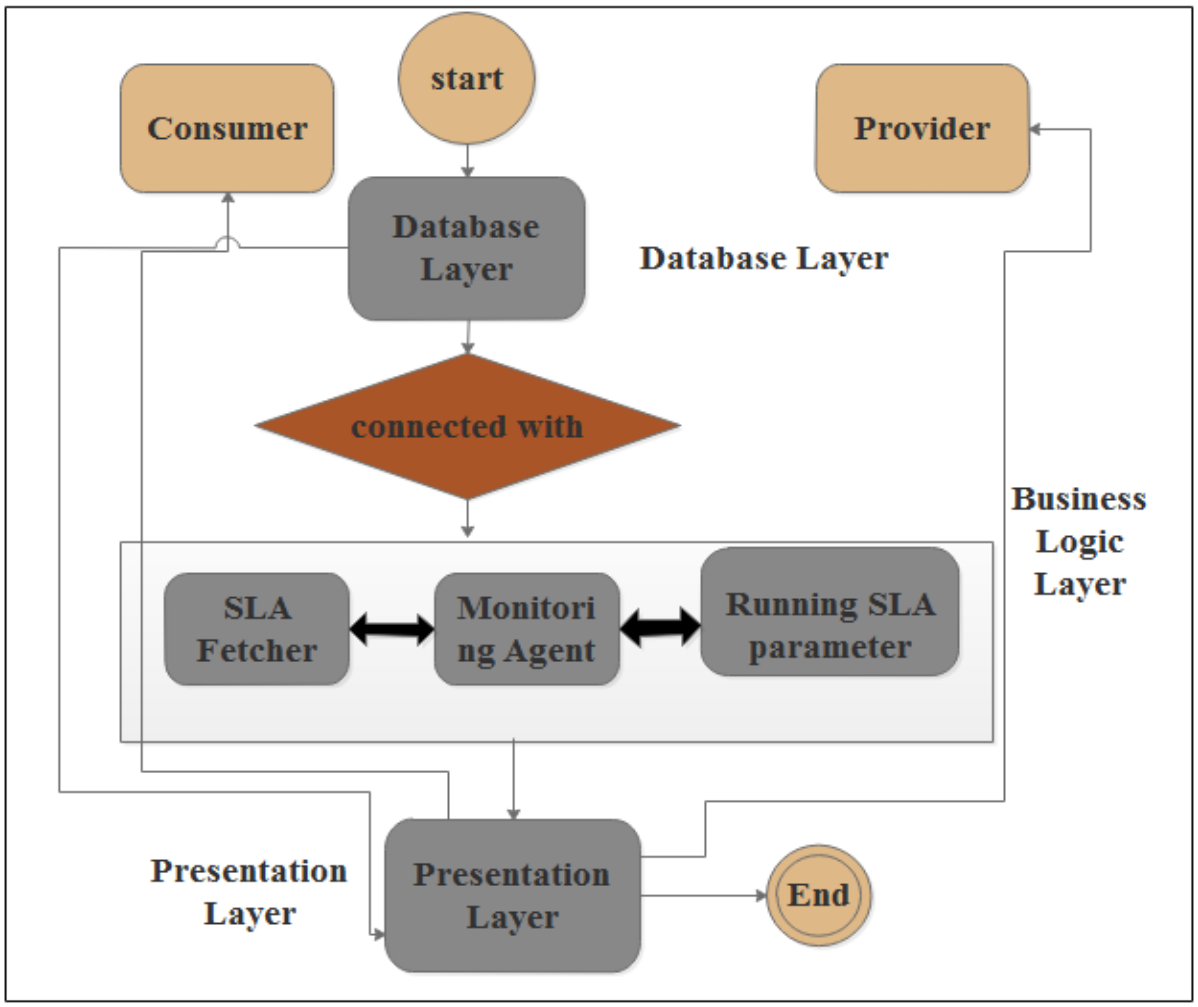

Figure 5-5 Activity Diagram that show the whole activity of the prototype

5.3 Implementation and Discussion

We have developed a framework that monitor service level agreement on cloud computing. This framework contain five important components. SLA monitoring component, this component contains SLA fetcher which used to gather agreed service level agreement from cloud service provider. Cloud provider have SLA that signed with service consumer which is signed before service is started to offer. So SLA fetcher read or gather signed SLA to the monitoring agent.

Running SLA parameters gathers running SLA parameters to the monitoring agent in order to compare the two SLA before service started and after service started to allocate for the service consumer respectively. The Monitoring agent is the main important component in our research, it compared the SLA collected by SLA fetcher and SLA collected from SLA running parameters. Parameters includes bandwidth, MIPS, Core and storage. Database used to store information of signed service level agreement or SLA fetcher, allocated service level agreement Running SLA parameters, information of data centre, information of host and information of cloud users or cloudlets. If the allocated service is less or greater than an agreed service level agreement violation will occur. If violation occur the SLA Monitoring agent generates alert to the cloud service consumer and cloud provider.

The other component of our research are cloud service provider and cloud consumer. Cloud provider is the owner of the services to be offered to the service consumers. Consumer can control the services after he/she received from cloud providers and shared services for end users in an organization or institutions.

The other component are cloud services. These service are divided into three main categories namely software as a service (Saas), platform as a service (Paas) and Infrastructure as a service (Iaas). Services are presented to the service consumer based on the service level agreement. But in some times the service level agreements are violated because of weak monitoring mechanisms. SLA Monitoring framework we developed is responsible to monitor service level agreement both for cloud service provider as well as service consumers. Our framework is suitable for all organization and institutions. Different cloud providers may have different service level agreement, but our framework can monitor for all providers and consumers.

We have created five tables in one database. The first tables contains datacentre and its characteristics. The second table is host table which contains host's characteristics. The third table is required services table which contains different kinds of service parameters. Resource

allocated and users' tables are table fourth and fifth respectively. Figure 5-6 represents the SLA Monitoring graphical user interface.

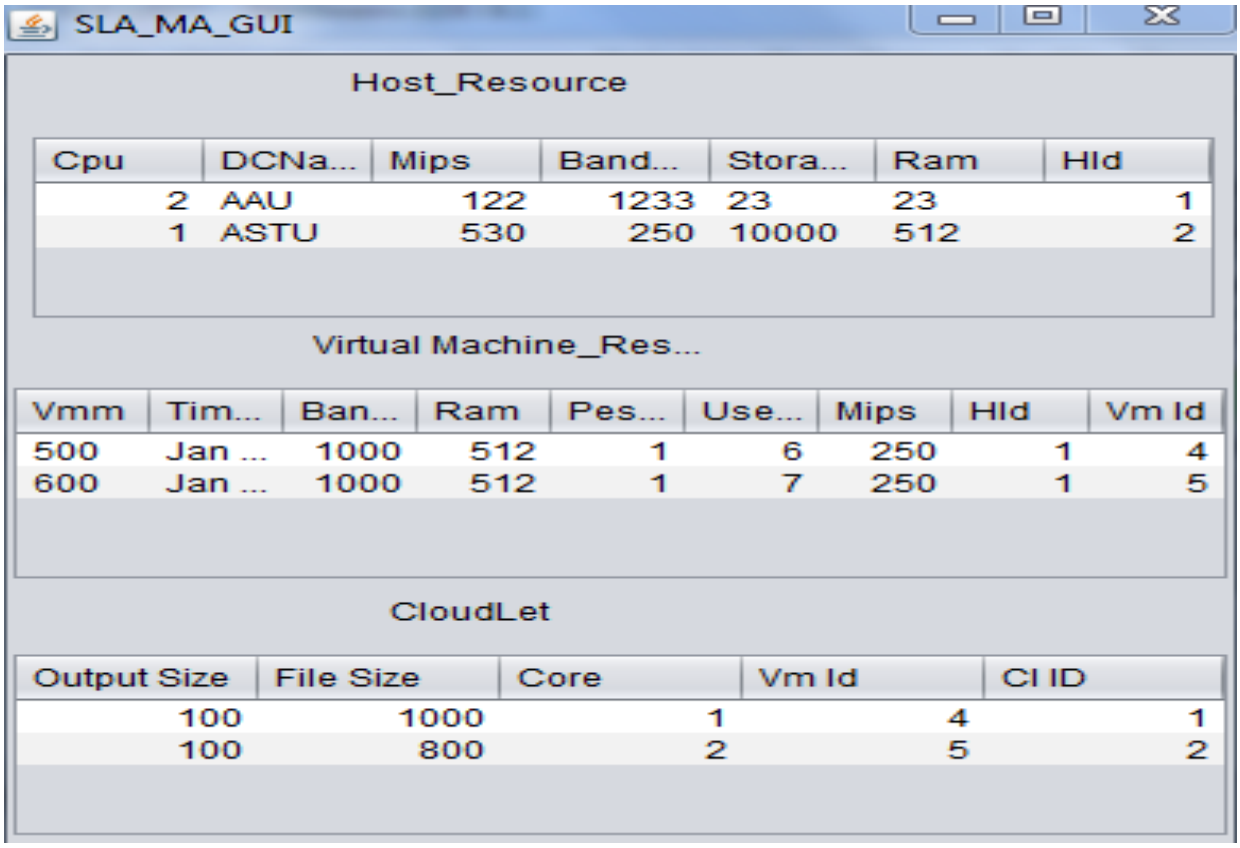

Figure 5-6 Represent the SLA Monitoring GUI

Figure 5-6 indicated resources and their parameters. These resources are specified during an agreement is signed between cloud service provider and service consumers. Resource is inherited from the higher class. First we have created data centre, host inherited from Data centre. We have connected data centre with host by foreign key primary key of the data centre. Next to creating the host we have created virtual machine and cloudlets form virtual machine. So graphical user interface displays the general service level agreement for consumers.

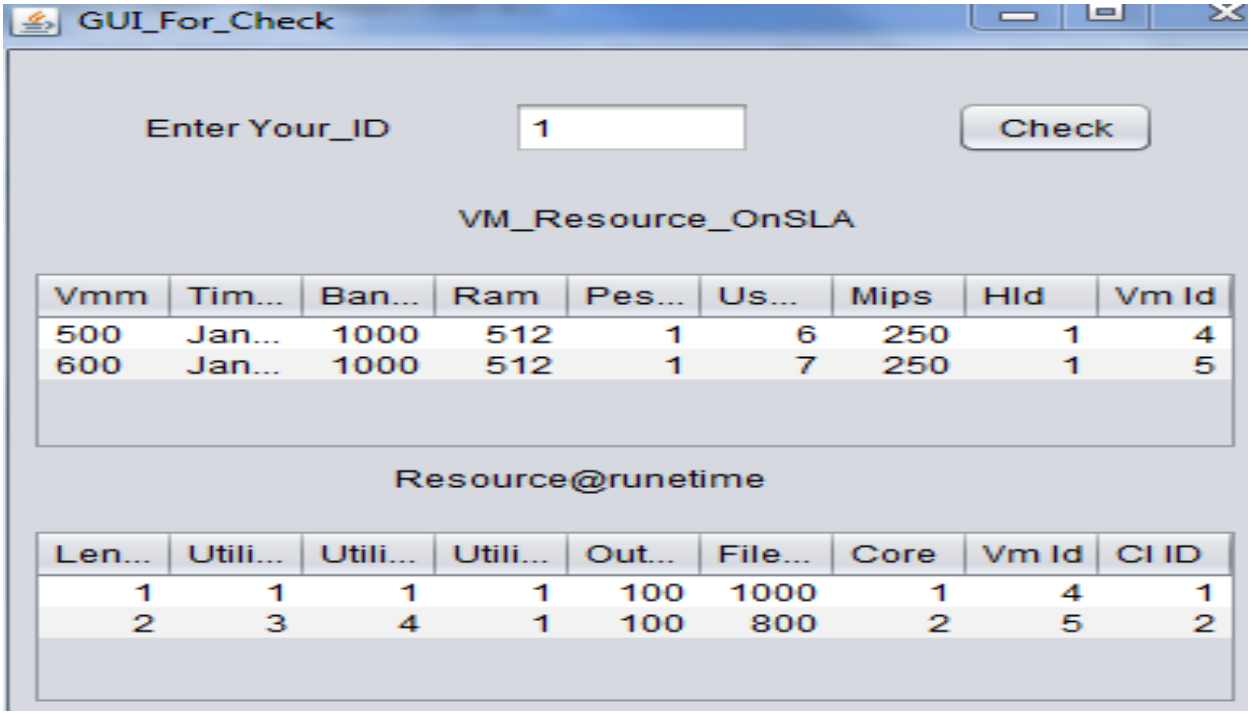

Figure 5-7 Represents Violation checking graphical user interface

Figure 5-7 indicated the way consumer check whether the service level agreement is violated or not. Using this graphical user interface the consumer can check how much services and their parameters are used by them at run time.in this GUI the first table indicated the agreed service level agreement virtual machine and its parameters. The second table indicated the obtained services with their parameters. And the last table indicated the difference of agreed service level agreement and obtained service and we know that if any difference occurred there is violation of service level agreement.

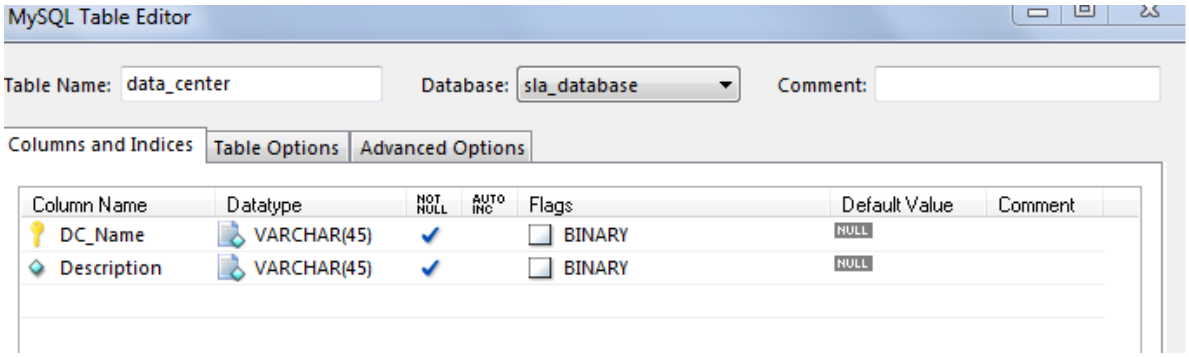

Figure 5-8 Data center creation

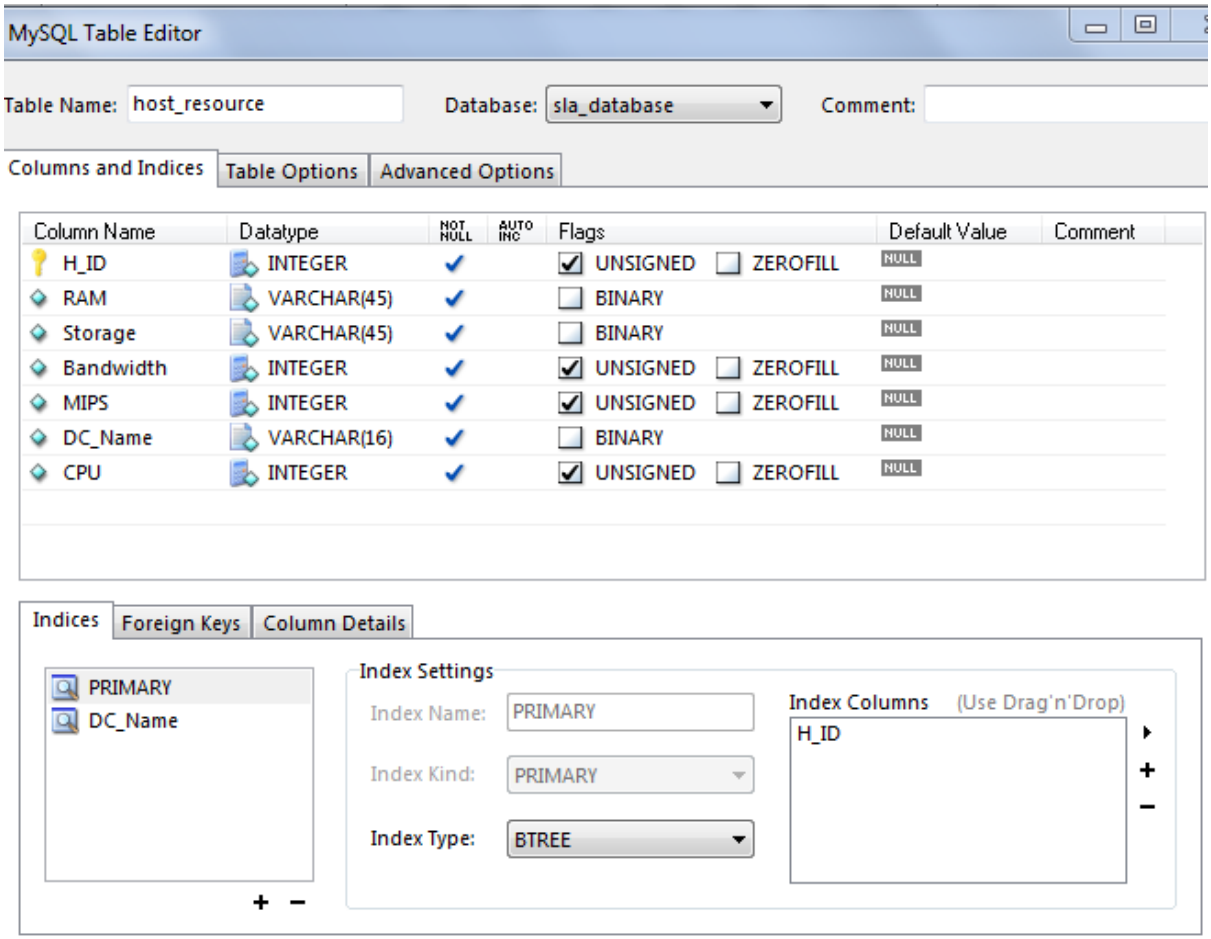

Figure 5-9 Host table

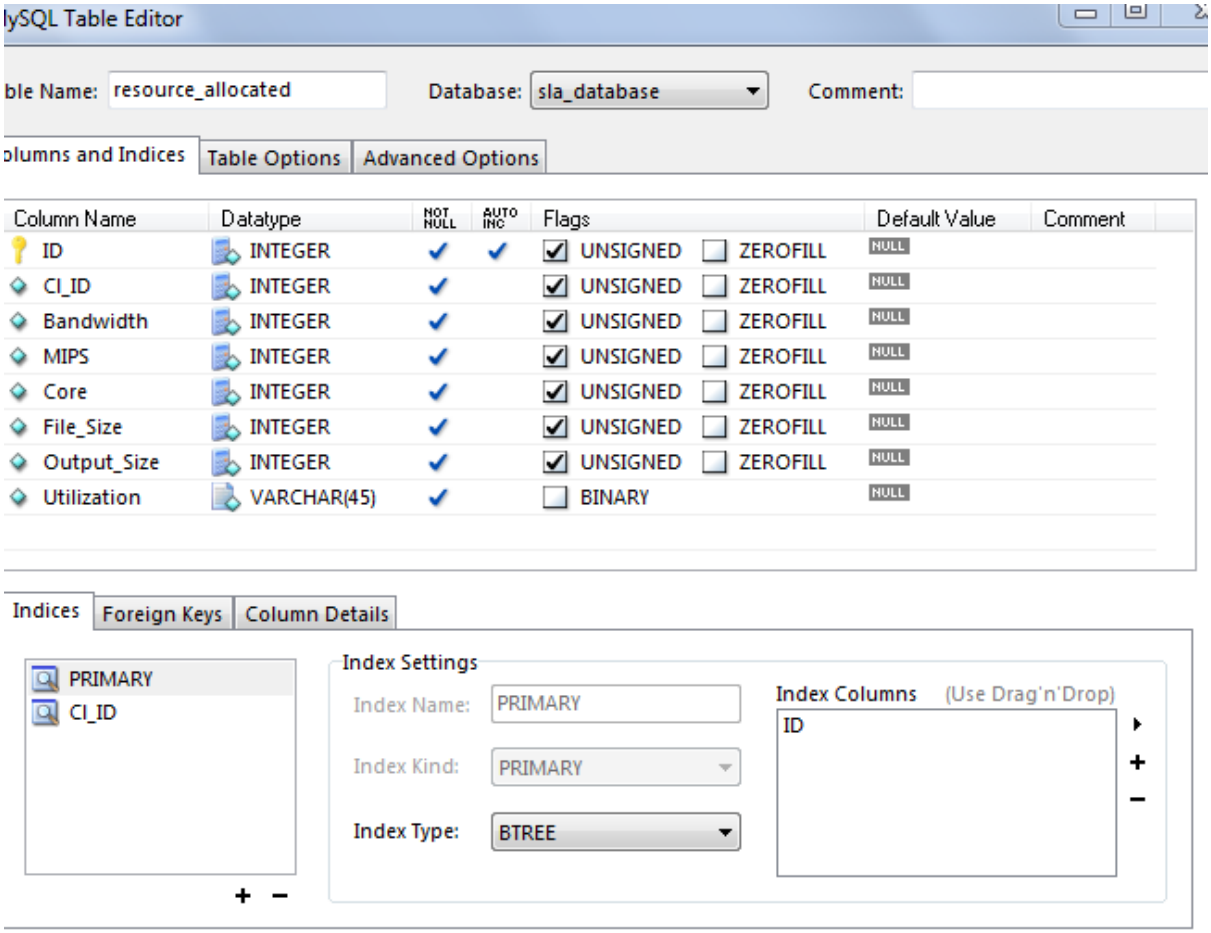

Figure 5-10 Resource allocation table

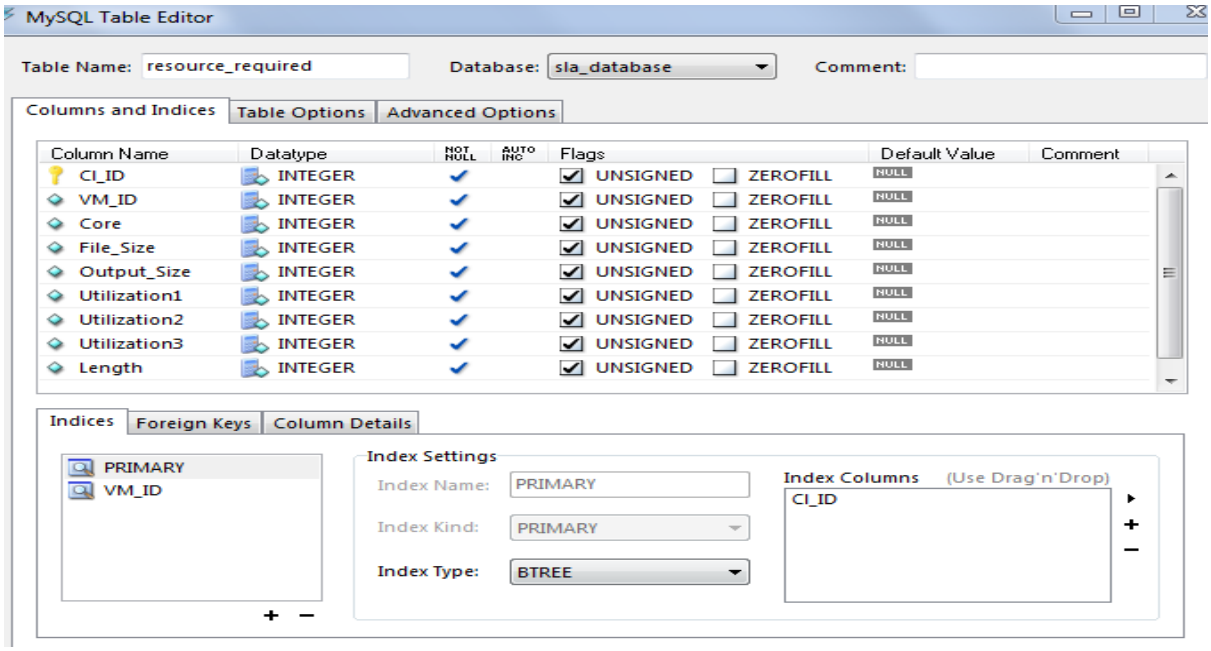

Figure 5-11 Signed SLA table

CHAPTER SIX

6. EVALUATION OF THE PROTOTYPE

The implemented SLAMA is required to be evaluated for its functionality. The evaluation includes identification of resource used, test cases scenarios, generation of test data and evaluating the functionality of agent implemented. The implemented agent is evaluated by integrating in simulator, CloudSim. The evaluation steps are detailed below:

For the evaluation purpose we used one datacenter and three Virtual Machines (VM) and we deployed those three VM on two Hosts. The configurations of VMs and Hots are described in Table 4 and 5 respectively.

Table 4 Resource details of Virtual Machines

RESOURCE	VM#1	VM#2	VM#3
Image Size	10,000	10,000	10,000
RAM(MB)	512	512	512
Core	1	1	1
MIPS	250	250	250
Bandwidth	1000	1000	1000

Table 5 Resource details of Virtual Machine Hosts

RESOURCE	HOST#1	HOST#2
Core	4	2
RAM(MB)	16384	16384
Storage	100,0000	100,0000
Bandwidth	10,000	10,000
MIPS	1000	1000

We indicated in the above tables 4 and 5 that we have as resource to meet the research goal which is monitoring the service level agreements between cloud user (broker) and cloud

provider. To show how service level agreement face challenge; we used cloudlet as user of the cloud provider resource (host and VM).

For evaluation we have created 15 cloudlets and we assigned them to VM of Hosts. The configurations of cloudlets is described in Table 6.

Table 6 Requirement Details of Cloudlets

CLOUDLETS	PARAMETER	SIZE
All Cloudlets	Length	40000
	File Size	300
	Output Size	300
	CPU	1

In the world of cloud computing it's known that the cloud providers and users make an agreement for the service usage. If cloud providers (i.e. data centers) have sufficient resources for their service user's will no violation of SLA unless they increase number of cloud users without increase the resource for service. In our case as we discussed cloudlets through broker are users of data center resource as show in Figure 6-1.

Based on VMs and Hosts, three scenarios are identified to evaluate SLAMA. The outcome of each scenario would be either no violation or violation. The load is varied in each test case to know the outcome. The three different scenario and their outcome described in Table 7:

Table 7 Test cases for three Load types

Test Case	Cloudlet Load	Remark
1	Normal Load	No SLA Violation
2	Below Load	SLA Violation
3	Over Load	SLA Violation

The following sample code show how we created each test case for evaluation in cloud sim environment.

```
num_of_cloudlet=15;
cloudletList = createCloudlet(num_of_cloudlet); // crea

setCloudlettoVM(cloudletList, 0, 1, 7);
setCloudlettoVM(cloudletList, 1, 8, 12);
setCloudlettoVM(cloudletList, 2, 13, 15);
```

Figure 6-2 Normal Load of Cloud user to Cloud provider

```
num_of_cloudlet=3;
cloudletList = createCloudlet(num_of_cloudlet); // creat

setCloudlettoVM(cloudletList, 0, 1);
setCloudlettoVM(cloudletList, 1, 2);
setCloudlettoVM(cloudletList, 2, 3);
```

Figure 6-3 Below Load of Cloud user to Cloud provider

```
num_of_cloudlet=30;
cloudletList = createCloudlet(num_of_cloudlet);

setCloudlettoVM(cloudletList, 0, 1, 10);
setCloudlettoVM(cloudletList, 1, 11, 19);
setCloudlettoVM(cloudletList, 2, 20, 30);
```

Figure 6-4 Over Load of Cloud user to Cloud provider

Table 8 Numerical representation of SLA and their Status

Test Case	Cloudlet Parameters	Required /agreed Value	Allocated Value	Difference Value	Remark
1	Length	40000	40000	No	No violation
	File size	300	300	No	No violation
	Output size	300	300	No	No violation
	Core	1	1	No	No violation
2	Length	40000	< 40000	yes	Violated
	File size	300	<300	yes	Violated
	Output size	300	<300	yes	Violated
	Core	1	<1	yes	Violated
3					
	Length	40000	>40000	yes	Violated
	File size	300	>300	yes	Violated
	Output size	300	>300	yes	Violated
	Core	1	>1	yes	Violated

Table 8 indicated the evaluation of our framework in terms of numerical (number). The given parameters are taken from amazon web service. In amazon web service the length of cloudlet is 40000, file size is 300, output size is 300 and core is 1. We have taken this standard for implementation and evaluation of our research.

CHAPTER SEVEN

7. CONCLUSION AND FUTURE WORK

Cloud computing is a core important in today's world by many means like cost minimize, to minimize human power, to save time, to access service being anywhere at any time through internet connection. In cloud computing environment there are many problems as we discussed in statement of the problems in chapter one. There are three common service delivery models in cloud computing, SaaS (software as a service), PaaS (platform as a service) and IaaS (Infrastructure as a service). Those services are offered based on the service level agreement signed between the two party's service providers and service consumers. However sometimes service level agreement on cloud computing is violated. As number of cloudlets or users increases the resources provider side become challenged, in sense the quality of services decreases and as result cloud users are become unsatisfied. Service level agreement is controlled and monitored cloud provider side and consumers have no chances to do so.

We have conducted this research to overcome this problem. In our research an Agent is developed to monitors and controls service level agreement between cloud provider and cloud consumers. The prototype worked for both parties equally. So in our implementation of the prototype we used CloudSim as a simulator, net beans to write code and MySQL database to create necessary data base and tables for our implementation. We had three layers in our Architecture namely database layer, Business Logic Layer and Presentation Layer. Database layer is the SLA storage area. Business Logic Layer is the layer in which SLA fetcher and Running SLA parameters communicated with SLAMA and the Presentation Layer is where the full information about SLA is displayed in.

Our research is evaluated in using three cases. Namely Case 1: Normal Load of Cloud user to Cloud provider case 2: Below Load of Cloud user to Cloud provider case 3: Over Load of Cloud user to Cloud provider.

We have developed a framework and a prototype to monitor service level agreement in our research. The future work for us and for others researchers is the full implementation of this framework to monitor cloud based service level agreement in general for both service provider and service consumers in the world of cloud computing.

References

[1] 2. H. 3. A. 1Zeyana Saif Alkindi, "A Conceptual Architectural Framework of Cloud Computing for," *Journal of Emerging Trends in Computing and Information Sciences,* 2015.

[2] E. Gorelik, "Comparison of Cloud Computing Service and Deployment Models," *Composite Information Systems Laboratory (CISL),* 2013.

[3] D. B. s. Smith, "The Impact of Emerging Software Paradigms on Software Quality and User Expectations.," *springer,* vol. Vol. 6759, 2011.

[4] S. Kelkar, "Challenges and Opportunities with Cloud," *International Journal of Innovative Research in Computer,* 2015.

[5] IBM, "Performance and," *RedBook,* 2013.

[6] 2. S. S. K. 1Satinder Kaur, "A Comprehensive Study on the Security Framework in," *ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print),* 2015.

[7] K. P. J. a. C. Pearce, "Automating Cloud Service Level Agreements using Semantic Technologies," *In proceedings of CLaw Workshop, IEEE International Conference on Cloud Engineering,* 2015.

[8] G.-C. TANG, "Research on the Model for Cloud Security Based on SLA," *International Workshop on Cloud Computing and Information Security,* 2013.

[9] c. s. c. clouncil, "Practical Guide to Cloud Service Agreements," *Cloud Standards Customer Council.,* 2015.

[10] d. c. c. editors, "Review and summary of cloud service levelagreements," 2010.

[11] I. d. work, "Review and summary of cloud service levelagreements," *developerWorks cloud computing editors(dwcloud,* 2010.

[12] s. B. Garg, "Green Cloud Computing and Environmental Susustainability.," 2011.

[13] R. Buyya, "Cloud computing and emerging IT platforms," *Vision,hype,and reality for,* 2009.

[14] P. a. G. T. Mell, "The NIST definition of cloud computing.," *National Institute,* 2009.

[15] Staten, "Is cloud computing ready for the enterprise?," *Forrester Research,* 2008.

[16] C. F. &. S. L. C. Fang, "Collaborative Learning Using Service-Oriented," *A Framework Design', Knowledge-Based Systems,* 2009.

[17] R. A. A. H. Muhammad Mannir Ahmad Getso, "Applications' of Cloud Computing in Academic Institutions," *International Journal of Information Systems and Engineering (online),* 2014.

[18] D. Aumueller, "IT Complience Analysis for Cloud Computing," Germany, 2010.

[19] s. B. Garg, "Green Cloud Computing and Environmental Susustainability," 2011.

[20] M. F. Armbrust, "Above the Clouds. California," 2009.

[21] I. Ashraf, "An Overview of Service Models of Cloud Computing," *International Journal of Multidisciplinary and Current Research,* 2015.

[22] 1. El-Awadi, "A Framework for Negotiating Service Level Agreement of," in *International Conference on Communication, Management and Information Technology (ICCMIT*, 2015.

[23] V. Buyya, "Mastering Cloud Computing," *Tata McGraw Hill Education,* 2013.

[24] Velte A.T. et al., "Cloud Computing: A practical Approach," McGraw-Hill , United States, 2010.

[25] Thomas P.Y., "Harnessing the potential of cloud computing to transform higher education," in *Cloud Computing for Teaching and Learning: Strategies for Design and Implementation*, 2012.

[26] Thorsteinsson G. et al., "Using Virtual Reality for Developing Design Communication," *Studies in Informatics and Control,* pp. 93-106, 2010.

[27] S. R. L. a. P. C. Nuno Palhares1, "A Multidimensional Model for Monitoring Cloud Services," Braga, Portugal., 2015.

[28] S. G. A. C. C. T. G. M. L. V. L. N. K. a. R. B. Clayman, "Monitoring Service Clouds in the Future Internet," *IOS Press, pp. 115-126,* 2010.

[29] T. K. N. L. T. K. D. a. L. J. Choi, "Autonomic Management Framework for Cloud-based Virtual Networks," *In 13th APNOMS, pp. 1-7,,* 2011.

[30] El-Awadi, "A Framework for Negotiating Service Level Agreement ," *International Conference on Communication, Management and Information Technology (ICCMIT,* 2015.

[31] A. K. a. H. Ludwig, "The Web Service Level Agreement: Specifying and," *Journal of Network and Systems,* March 2003.

[32] J. O. a. V. Teixeira, "A Vision of Personalized Service Level Agreement," DanaC 2013, New York, NY, USA, 2013.

[33] I. B. M. M. a. S. D. Vincent C. Emeakaroha, "Metric to Highs Level SLAs- LoM2HiS Framework:," *HPCS,* pp. 48-54, 2010.

[34] D. Bell, "Software Engineering for students A Programming Approach fourth edition," Henry Ling Ltd., Great Britain:, 2005.

[35] "https://en.wikipedia.org/wiki/Software_prototyping," [Online].

[36] [Online]. Available: http://docs.openstack.org/icehouse/training-guides/content/operator-getting-started.html.

[37] L. W. a. R. Buyya, "Service Level Agreement (SLA) in Utility Computing Systems," *Cloud Computing and Distributed Systems (CLOUDS),* 2010.

[38] t. F. F. (. n. p. l. S. Numérique), "Cloud Computing Simulators: State of the Art," *McKinsey Global Survey results,,* 2010.

[39] C. C. U. C. D. Group, "Cloud Computing Use Cases, white paper," 2010.

[40] K. M. J. G. B. a. A. J. Stamou, "Service Level Agreements as a Service - Towards Security Risks Aware SLA Management," 2012.

[41] P. R. A. a. S. A. Patel, "Service Level Agreement in Cloud Computing," *Technical Report,,* 2009.

[42] S. Saidhbi, "A Cloud Computing Framework for Ethiopian Higher Education," *IOSR Journal of Computer Engineering (IOSRJCE),* 2012.

[43] Nabil S., "Cloud computing for education: A new dawn?," *International Journal of Information Management , vol. 30,* p. 109–116, 2010.

[44] Yang Z. and Zhu Z., "Construction of OSS-Based E-Learning Cloud in China," in *2nd International Conference on Education Technology & Computer (ICETC)*, 2010.

[45] Vaquero L.M., "EduCloud: PaaS Versus IaaS Cloud Usage for an Advanced Computer Science Course, Education," *IEEE Transactions on Education,* 2011.

[46] I. B. I. a. S. Christpher Redl, "Automatic SLA Matching and Provider," 2012.

[47] T. L. H. a. L. Zhu, "A Service Level Agreement Framework of Cloud computing," *Computer Science and Automation Engineering (CSAE), 2012,* pp. 255-259, 2012.

[48] Y. S. C. Makhan Kumbhkar, "Analysis of Cloud Computing in Higher Education," *International Journal of Advanced Research in Computer Science and Software Engineering,* 2015.

[49] E. R. Babbie, "The Practice of Social Research. California, United States," *Wadsworth,* 2012.

[51] K. D. Trivedi, "Cloud based Conceptual Framework of Service Level Agreement for University," *International Journal of Computer Applications,* 2013.

[52] R. El-Awadi, "A Framework for Negotiating Service Level Agreement of," in *International Conference on Communication, Management and Information Technology (ICCMIT*, Cairo, Egypt, 2015.

[53] E. R. Babbie, " A Cloud Computing Framework for Ethiopian Higher Education," *The Practice of Social Research,* 2012.

[54] M. Y., "Create highly efficient design patterns for scalability, redundancy, and high availability in the AWS Cloud," *Cloud computing service model,* 2015.

[55] D.Daniel, "A Novel Approach for Scheduling Service Request in," in *International Conference on Signal Processing, Communication, Computing and Networking Technologies*, Tamil Nadu, 2011.

[56] S. B. Y. K. b. F. A. d. O. J. b. Damián Serrano a, "SLA guarantees for cloud services," *Future Generation Computer Systems,* 2016.

[57] D. U. Singh1, "CloudSim Simulator Used for Load balancing in Cloud Computing," *International Journal of Emerging Technology and Advanced Engineering,* 2016.

[58] M. K. A. O. Mohmed Sirelkhtem Adrees, "CLOUD COMPUTING ARCHITECTURE FOR," *International Journal of Database Management Systems,* 2015.

[59] D. Sullivan, "PaaS Providers List," *Comparison And Guide,* 2014.

[60] [Online]. Available: http://www.snapsurveys.com/blog/what-is-the-difference-between-qualitative-research-and-quantitative-research/.

[61] [Online]. Available: http://www.soa.si/2011/09/19/using-opennebula-hooks-and-java-rpc-api-for-cloud-process-automation/.

[62] "https://en.wikipedia.org/wiki/Software_prototyping," [Online].

Appendences

Appendix: A

SLA Monitoring agent main sample code

```
Public class SLAMAC_Agent {

Private static List<Cloudlet> cloudletList;

Private static List<Vm> vmlist;

Private static List<? Extends Vm> vmList;

Private static Integer Gv_host_ID;

List<Host> hostList = new Array List<> ();

Public static void main (String [] args)

{

int num_of_cloudlet;

try {

int num_user = 2;   // number of grid users

Calendar calendar = Calendar.getInstance();

boolean trace_flag = false;  // mean trace events

// Initialize the CloudSim library

CloudSim.init(num_user, calendar, trace_flag);

GlobalBroker globalBroker = new GlobalBroker("GlobalBroker");

// Second step: Create Datacenters

//Datacenters are the resource providers in CloudSim. We need at list one of them to run a CloudSim simulation

Datacenter datacenter = createDatacenter("Datacenter_0","cloud provider");

//Third step: Create Broker

DatacenterBroker broker = createBroker("Broker_0");

int brokerId = broker.getId();


//Fourth step: Create VMs and Cloudlets and send them to broker
```

```
vmList = createVM(brokerId, 3,0); //creating 3 vms
num_of_cloudlet=15;
cloudletList = createCloudlet(num_of_cloudlet); // creating 15 cloudlets
setCloudlettoVM(cloudletList, 0, 1, 7);
setCloudlettoVM(cloudletList, 1, 8, 12);
setCloudlettoVM(cloudletList, 2, 13, 15);
broker.submitVmList(vmList);
broker.submitCloudletList(cloudletList);
// Fifth step: Starts the simulation
CloudSim.startSimulation();
// Final step: Print results when simulation is over
List<Cloudlet> newList = broker.getCloudletReceivedList();
newList.addAll (globalBroker.getBroker ().getCloudletReceivedList());
CloudSim.stopSimulation();
}
catch (Exception ex)
{
JOptionPane.showMessageDialog(null, ex);
}
}
private static Datacenter createDatacenter(String name,String des){
try
{
MyDatabase db=new MyDatabase();
db.insertto_DC_table(name,des);// to insert data into data center table

}catch(Exception ex)
{
JOptionPane.showMessageDialog(null, ex+"data center sending");
```

```
}
List<Host> hostList = new ArrayList<>();
List<Pe> peList1 = new ArrayList<>();
int mips = 1000;
peList1.add(new Pe(0, new PeProvisionerSimple(mips))); // need to store Pe id and MIPS Rating
peList1.add(new Pe(1, new PeProvisionerSimple(mips)));
peList1.add(new Pe(2, new PeProvisionerSimple(mips)));
peList1.add(new Pe(3, new PeProvisionerSimple(mips)));
int cpu=4;
List<Pe> peList2 = new ArrayList<>();
peList2.add(new Pe(0, new PeProvisionerSimple(mips)));
peList2.add(new Pe(1, new PeProvisionerSimple(mips)));
int cpu2=2;
int hostId=0;
int ram = 16384; //host memory (MB)
long storage = 1000000; //host storage
int bw = 10000;
hostList.add(
new Host(
hostId,
new RamProvisionerSimple(ram),
new BwProvisionerSimple(bw),
storage,
peList1,
new VmSchedulerTimeShared(peList1)
)
);
try
```

{

MyDatabase db=new MyDatabase();

db.insertto_host_table(name,hostId,ram,storage,bw,cpu,mips);// to insert data in host table

// Gv_host_ID=hostId;

// createVM(,Gv_host_ID);

}catch(Exception ex)

{

JOptionPane.showMessageDialog(null, ex+"host 1 sending");

}

hostId++;

hostList.add(

new Host(

hostId,

new RamProvisionerSimple(ram),

new BwProvisionerSimple(bw),

storage,

peList2,

new VmSchedulerTimeShared(peList2)

)

);

try

{

MyDatabase db=new MyDatabase();

db.insertto_host_table(name,hostId,ram,storage,bw,cpu2,mips);

}

```
}
}
```

Appendix: B

Sample Code to Creates New Form SLA Monitoring _GUI

```
Package Cloud_Sim_SLA;
Import java.sql.Connection;
Import java.sql.DriverManager;
Import java.sql.PreparedStatement;
Import java.sql.ResultSet;
Import java.sql.SQLException;
Import javax.swing.JOptionPane;
Import net.proteanit.sql.DbUtils;
Public class SLAMAC_GUI extends javax.swing.JFrame {
/**
* Creates new form SLAMAC_GUI
*/
Connection conn = null;
ResultSet rs;
PreparedStatement pst;
public static Connection connectionDB()
{
try{
try {
Class.forName("com.mysql.jdbc.Driver");
}
catch (ClassNotFoundException ex) {
JOptionPane.showMessageDialog(null, ex);
}
```

```
Connection
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/sla_database","root","abel
");
JOptionPane.showMessageDialog(null,"Connected");
System.out.println("abel");
return  conn;
}   catch(SQLException e)
{
JOptionPane.showMessageDialog(null, e);
return null;
}
}
public SLAMAC_GUI() {
initComponents();
conn=connectionDB();
Load();
Load2();
Load3();
}
private void Load()
{
String sql="Select * from host_resource";
try {
pst=conn.prepareStatement(sql);
rs=pst.executeQuery();
Host_Info.setModel(DbUtils.resultSetToTableModel(rs));
} catch (SQLException ex) {
JOptionPane.showMessageDialog(null, ex);
}
}
```

```
private void Load2()
{
String sql="Select * from vm_master";
try {
pst=conn.prepareStatement(sql);
rs=pst.executeQuery();
VM_info.setModel(DbUtils.resultSetToTableModel(rs));
} catch (SQLException ex) {
JOptionPane.showMessageDialog(null, ex);
}
}
private void Load3()
{
String sql="Select * from resource_required";
try {
pst=conn.prepareStatement(sql);
rs=pst.executeQuery();
Cloudlet_info.setModel(DbUtils.resultSetToTableModel(rs));
} catch (SQLException ex) {
JOptionPane.showMessageDialog(null, ex);
}
}
/**
* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
jScrollPane1 = new javax.swing.JScrollPane();
Host_Info = new javax.swing.JTable();
jScrollPane2 = new javax.swing.JScrollPane();
VM_info = new javax.swing.JTable();
jScrollPane4 = new javax.swing.JScrollPane();
Cloudlet_info = new javax.swing.JTable();
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("SLA_MCA_GUI");
setSize(new java.awt.Dimension(400, 300));
Host_Info.setModel(new javax.swing.table.DefaultTableModel(
new Object [][] {
{null, null, null, null},
{null, null, null, null},
{null, null, null, null},
{null, null, null, null}
},
new String [] {
"Resource", "Title 2", "Title 3", "Title 4"
}
));
jScrollPane1.setViewportView(Host_Info);
VM_info.setModel(new javax.swing.table.DefaultTableModel(
new Object [][] {
{null, null, null, null},
{null, null, null, null},
{null, null, null, null},
{null, null, null, null}
},
```

```
new String [] {
"Title 1", "Title 2", "Title 3", "Title 4"
}
));
jScrollPane2.setViewportView(VM_info);
Cloudlet_info.setModel(new javax.swing.table.DefaultTableModel(
new Object [][] {
{null, null, null, null},
{null, null, null, null},
{null, null, null, null},
{null, null, null, null}
},
new String [] {
"Title 1", "Title 2", "Title 3", "Title 4"
}
));
}
```

Appendix: C

Sample code for SLA Monitoring _Client GUI

```
Public SLAMAC_GUI_Client() {

initComponents();

conn=connectionDB();

}

Connection conn=null;

ResultSet rs;

PreparedStatement pst;

ResultSet rs2;

PreparedStatement pst2;

Public static Connection connectionDB()

{

try{


try {

Class.forName("com.mysql.jdbc.Driver");

}

catch (ClassNotFoundException ex) {

JOptionPane.showMessageDialog(null, ex);

}

Connection
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/sla_database","root","abel
");

JOptionPane.showMessageDialog(null,"Connected");

return  conn;

}   catch(SQLException e)

{
```

```
JOptionPane.showMessageDialog(null, e);
return null;
}}
```