Contents

[3.1 Introduction 2](#_Toc428184895)

[3.1.1 Purpose 2](#_Toc428184896)

[3.1.2 Intended audience and reading suggestion 2](#_Toc428184897)

[3.1.3 Project Scope 2](#_Toc428184898)

[3.1.4 Design and Implementation Constraints 2](#_Toc428184899)

[3.1.5 Assumptions and Dependencies 2](#_Toc428184900)

[3.2 System Features 3](#_Toc428184901)

[3.2.1 System Feature 1(Functional Requirement) 3](#_Toc428184902)

[3.2.2 System Feature2 (Functional Requirement) 3](#_Toc428184903)

[3.3 External Interface Requirements 4](#_Toc428184904)

[3.3.1 User Interfaces 4](#_Toc428184905)

[3.3.2 Hardware Interfaces 4](#_Toc428184906)

[3.3.3 Software Interfaces 4](#_Toc428184907)

[3.3.4 Communication Interfaces 4](#_Toc428184908)

[3.4 Nonfunctional Requirements 4](#_Toc428184909)

[3.4.1 Performance Requirements 4](#_Toc428184910)

[3.4.2 Safety Requirements 4](#_Toc428184911)

[3.4.3 Security Requirements 4](#_Toc428184912)

[3.4.4 Software Quality Attributes 4](#_Toc428184913)

[3.5 Other Requirements (If Applicable) 4](#_Toc428184914)

[3.5.1 Database Requirements 4](#_Toc428184915)

[3.5.2 Internalization Requirements 4](#_Toc428184916)

[3.5.3 Legal Requirements 4](#_Toc428184917)

[3.5.4 Reuse Objectives for the Project 4](#_Toc428184918)

[3.6 Analysis Models 4](#_Toc428184919)

[3.6.1 Data Flow Diagrams 4](#_Toc428184920)

[3.6.2 Class Diagrams 4](#_Toc428184921)

[3.6.3 State-transition Diagrams or Entity Relationship Diagrams 4](#_Toc428184922)

# 3. 1 Introduction

Internet has become pervasive in our daily life and cloud computing is the newest offering as service over the ubiquitous Web. Cloud computing has been considered as much-hyped phenomenon in the IT and business world promising to deliver a host of beneﬁts. Companies need to look beyond this hype and seriously consider the real value of incorporating the Cloud in their own businesses. This is aimed at helping companies analyze several characteristics of their own business as well as pre-existing IT resources to identify their favorability in the migration to the Cloud Architecture.

Here a tool is developed to monitor and analyze cost pattern on cloud accounts ( like Amazon Web Services) and capable of giving suggestions about cost optimization and cost containment.

## 3.1.1 Purpose

The main purpose of our project is to evaluate the performance of workloads on EC2 and reduce infrastructure cost from the customer’s point of view.

1. To monitor and analyze cost pattern on cloud accounts.
2. To analyze usage of the cloud services and give suggestions for future plans according to the user’s usage.
3. To give the suggestions about cost optimization and delivering cost containment.
4. To improve system performance and service quality.

## Intended audience and reading suggestion

* + - Administrators
    - Individual Node Owners
    - Cloud Controller

## Project Scope

This system analyses the problem of cost optimization in cloud computing. We also evaluate the performance of the resource monitoring and load balancing tools. This system monitors the VM node on private cloud to reduce infrastructure cost from the customer’s point of view. In our proposed model, we will be doing

**1.** **Cloud Setup** –

Creating private cloud (test bed) by using (Amazon Account)

**2. Resource Monitoring** –

Monitoring critical resources like RAM, CPU, memory, bandwidth, partition information, running process information and utilization and swap usages etc.

**3. Authentication and authorization** –

We need to connect to existing user’s Amazon account using user id and password and fetch the entire performance matrix like CPU, RAM, storage etc.

**4. Testing –**

In order to evaluate the performance of complete setup, need to deploy resource monitoring and load balancing tools on test bed and evaluate need of available resources.

**Modules:**

1. **Resource Monitoring of Cloud Nodes:**
   1. User should be able to view CPU and RAM usage utilization of Amazon ec2 nodes
   2. CPU and RAM utilization statistics should be dynamic and should refresh every second.
2. **Select Cloud Plans for popular clouds** like Amazon. Cost of service depends on region of server, memory usage, CPU etc. Cloud service providers charge for following services which need to be added in system
   1. Storage – Pricing
   2. Request Pricing
   3. Storage Management Price
   4. CPU pricing
3. **Monitor account wise VM Usage of following parameters**
4. CPUUtilization
5. DiskReadBytes
6. DiskWriteBytes
7. NetworkIn
8. NetworkOut
9. StatusCheck
10. **Propose efficient resource utilization**
11. By suggesting memory cutdown
12. By suggesting cpu cutdown
13. By suggesting storage cutdown
    * 1. **Design Constraints**

* **Error Recognition**: Error should be easily recognized and get solved out.
* **Speed**: Incoming Results speed should be good enough. So that it can propose efficient resource utilization and processing can be faster.
* Provide solutions to reduce infrastructure cost from the customer’s point of view.

**General Constraints**

* Login and password is used for identification of User and there is no facility for guest.
* This system is working for Client-Server.
* Application should not use java.io package
* Cloud node should be ON for monitoring

## 3.1.5 Assumptions and Dependencies

1. **Assumptions:**

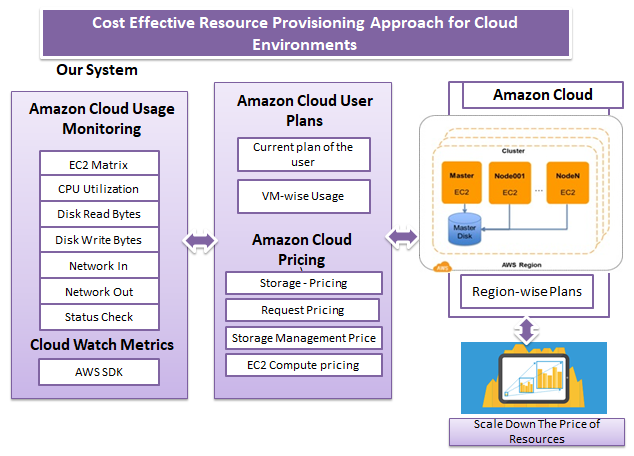
* User must have basic knowledge of computer.
* User must have basic knowledge of handling WebPages.
* Must be familiar basic with Networking and communication.
* Admin only the user who is allowing to access network.

1. **Dependencies:**

* Only Administrators will be able to edit main configurations.
* User and Administrators will communicate among themselves while executing the application.
* Escalation mechanism is limited to administrator only.

# 3.2 System Features

## 3.2.1 System Feature 1(Functional Requirement)



**Modules**:

**Resource Monitoring of Cloud Nodes:**

* 1. User should be able to view CPU and RAM usage utilization of amazon ec2 nodes
  2. CPU and RAM utilization statistics should be dynamic and should refresh every second.

. Cloud service providers charge for following services which need to be added in system

* 1. Storage – Pricing
  2. Request Pricing
  3. Storage Management Price
  4. CPU pricing

**Monitor account wise VM Usage of following parameters**

1. CPUUtilization
2. DiskReadBytes
3. DiskWriteBytes
4. NetworkIn
5. NetworkOut
6. StatusCheck

**Propose efficient resource utilization**

1. By suggesting memory cutdown
2. By suggesting cpu cutdown
3. By suggesting storage cutdown

## 3.3 External Interface Requirements

## 3.3.1 User Interfaces

* Use of AJAX atleast with all registration forms.
* Use of reports
* Then it will provide access to admin. It will allow admin to perform required task.
* User interface will provide good look and feel effect so that it will user friendly.
* And he or she can operate system very efficiently.

## 

## 3.3.2 Software Interfaces

**Software:**

**Tool to use:**

Eclipse Luna:

* Eclipse is an open source community whose projects building tools and frameworks are used for creating general purpose application. The most popular usage of Eclipse is as a Java development environment.
* Eclipse is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the Eclipse projects and helps cultivate both an open source community and an ecosystem of complementary products and service
* The independent not-for-profit corporation was created to allow a vendor neutral and open, transparent community to be established around Eclipse. Today, the Eclipse community consists of individuals and organizations from a cross section of the software industry.

**Feature Highlights:**

**JDK 1.8**

* Java is a programming language originally developed by James Gosling at Sun Microsystems and released in 1995 as a core component of Sun Microsystems Java platform.
* The language derives much of its syntax from C and C++ but has a simpler object model and fewer low level facilities. Java applications are typically compiled to byte code (class ﬁle) that can run on any Java Virtual Machine (JVM) regardless of computer architecture.
* Java is a general purpose, concurrent, class-based, object-orientedlanguagethatisspeciﬁcallydesignedtohaveasfewimplementation dependencies as possible. It is intended to let application developers write once, run anywhere.
* Java is currently one of the most popular programming languages in use, particularly for client-server web applications. One characteristic of Java is portability, which means that computer programs written in the Java language must run similarly on any hardware/operating-system platform.
* This is achieved by compiling the Java language code to an intermediate representation called Java byte code, instead of directly to platform-speciﬁc machine code. Java byte code instructions are analogous to machine code, but are intended to be interpreted by a virtual machine (VM) written speciﬁcally for the host hardware.
* End-users commonly use a Java Runtime Environment (JRE) installed on their own machine for standalone Java applications, or in a Web browser for Java applets. Standardized libraries pro- vide a generic way to access host-speciﬁc features such as graphics, threading, and networking

**MYSQL Database**

* Simplified connection wizard
* Guided installation to JDBC driver
* Editing and deployment of stored procedure.

## 3.3.4 Communication Interfaces

System shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

# 3.4 Nonfunctional Requirements

## 3.4.1 Performance Requirements

1**. Response Time:** The response time of the system should be deterministic at all times and very low, i.e it should meet every deadline. Thus, the system will work in real time.

**2. Throughput:** The throughput for all the actions should be high in-order to maintain the reliability.

**3. Accuracy:** System should correctly execute process; display the result i.e .cloud utilization accordingly. System should be able to analyze the sense of the timing as much accurate as possible.

**4. High speed:** Incoming Results speed should be good enough. So that it can Propose efficient resource utilization and processing can be faster. Provide solutions to reduce infrastructure cost from the customers point of view..

## 3.4.2 Safety Requirements

The data safety must be ensured by arranging for a secure and reliable transmission media. The source and destination information must be entered correctly to avoid any misuse or malfunctioning.

3.4.3 Security Requirements

Secure access of confidential data (user’s details).

Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction.

1. The terms information security, computer security and information assurance are frequently incorrectly used interchangeably. These fields are interrelated often and share the common goals of protecting the confidentiality, integrity and availability of information; however, there are some subtle differences between them.
2. User password must be stored in encrypted form for the security reason
3. All the user details shall be accessible to only high authority persons.
4. Access will be controlled with usernames and passwords.
5. To evaluate the performance of complete setup, need to deploy resource monitoring and load balancing tools on test bed and evaluate need of available resources.

## 3.4.4 Software Quality Attributes

Maintainable software should have

* Encourage in-code documentation (XML docs in javadoc, etc.)
* use a wiki to maintain the documentation
* Unit Tests = Good for documenting specifications
* Comments = Good for documenting design decisions.
* Unit Tests + Comments = Good for documenting specifications and design decisions. = Easily maintainable software.
* Faster feedback from any changes made to the system
* Providing better transparency into the changes happening to the system
* Propagating environmental changes and code changes more rapidly while maintaining control
* Ease integration issues by dealing with them earlier in smaller chunks

# 3.5 Other Requirements (If Applicable)

## 3.5.1 Database Requirements

The database is required to be created and maintained in MySQL Server. Stored procedures are also created to retrieve and operate on data.

## 3.5.2 Internalization Requirements

## System only support English language

## 3.5.3 Legal Requirements

Licensed copies need to be purchased for application/software that is used by client

# 3.6 Analysis Models

## 3.6.1 Data Flow Diagrams

## 3.6.2 Class Diagrams

## 3.6.3 State-transition Diagrams or Entity Relationship Diagrams