

EFFICIENT AND SECURE SERVER MIGRATION ON CLOUD STORAGE WITH VSM AND DROPBOX SERVICES

Submitted in partial fulfillment of the requirements for the award of

Bachelor Degree in
Computer Science

by

Jency Prescilla.V (Reg.No : 38290029)

Krithika.D (Reg.No : 38290037)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF COMPUTING
SATHYABAMA UNIVERSITY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU.**

APRIL 2021



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(Established under Section 3 of UGC Act, 1956)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Jency Prescilla.V** (Reg.No : 38290029) and **Krithika.D** (Reg.No : 38290037) who carried out the project entitled “**Efficient and Secure Server migration on cloud storage with VSM and Dropbox Services**” under my supervision from October 2020 to April 2021

Internal Guide

Dr. R. Aroul Canessane, M.E., Ph.D

Head of the Department

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

We, **Jency Prescilla.V** (Reg.No : **38290029**) and **Krithika.D** (Reg.No : **38290037**) hereby declare that the Project Report entitled “**Efficient and Secure Server migration on cloud storage with VSM and Dropbox Services**” done by me under the guidance of **Dr. Aroul Canessane** at Sathyabama Institute Of Science And Technology, CHENNAI is submitted in partial fulfillment of the requirements for the award of Bachelor Degree in Computer Science .

DATE:
PLACE:

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA, M.E., PH.D., Dean**, School of Computing **Dr.S. Vigneshwari M.E., Ph.D** and **Dr. L. Lakshmanan M.E., Ph.D., Heads of the Department**, for providing me necessary Support and details at the right time during the progressive reviews.

I would specially thank my internal guide **Dr. Aroul Canessane, M.E.,Ph.D.,** Department of Computer Science and Engineering, Sathyabama University, who was the instrument in providing vital encouragement without hesitation for the successful completion of project.

I would like to express my honor and gratefulness to my Project Coordinator **Dr. R. Aroul Canessane, M.E.,Ph.D,** Department of Computer Science and Engineering, who was beside me throughout the course of the project providing me all the support with keen interest.

I also thank my friends and my family members who had been providing me with the moral support. Last, but not the least, I thank God Almighty for giving me the strength and intellect to finish this project successfully.

ABSTRACT

The huge amount of sensitive data store in centralize storage space in cloud server, which is most important for providing the result based on the user's provided keywords. These service providing high efficiency with keeping user privacy. Without any configure the existing system learn plain text information. CSP (Cloud Service Provider) have outsourced data control in this condition because of some issue. There may survive unconstitutional operation on outsourced information on relation of profit or curiosity. Hence, privacy data has to encrypt for security. In existing search method over encrypted data in cloud support KNN search keyword not multi-keyword rank search based on semantics approach. The search methods aren't intelligent also neglect a few semantically

associated documents. In vision of attempt, deficiency an efficient method is suggested to solve multi rank search keyword issue over encrypted information similar search clarification sustaining synonym queries over encrypted information in cloud. To overcome these problems we proposed

DMRS search and VSM (Vector Space Model) algorithm. These two algorithms give top ranking search consequence and Security in the cloud storage system. The DMR algorithm include five components of distinct, boundary determination of empirical regions, Empirical regions characterization and filtering, DMR Statistical significance computation, DMR methylome characterization: sample clustering and descriptive statistics, gene models DMR annotation and CpG island, for optimization algorithms regional analysis convenient tools is used. To sustain consequence relevance ranking and multi-keyword search we accept VSM (Vector Space Model) to construct searchable index for accomplish accurate search results. To develop search efficiency without privacy leakage design structure of tree-based index which sustains deletion and insertion update well.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	i
	LIST OF FIGURES	iv
	LIST OF ABBREVIATIONS	v
1	INTRODUCTION	1
	1.1 DOMAIN INTRODUCTION	2
	1.2 PROJECT INTRODUCTION	3
2	LITERATURE REVIEW	5
3	SYSTEM ANALYSIS	8

3.1	SCOPE OF THE PROJECT	8
3.2	OBJECTIVE	8
3.3	PROBLEM DEFINITION	8
3.4	EXISTING TECHNIQUE	8
3.4.1	Disadvantages	9
3.5	PROPOSED TECHNIQUE	9
3.5.1	Advantages	9
4	SYSTEM REQUIREMENT SPECIFICATION	10
4.1	HARDWARE REQUIREMENTS	10
4.2	SOFTWARE REQUIREMENTS	10
4.2.1	Overview of Java	11
4.2.2	Java swing	12
5	SYSTEM DESIGN	13
5.1	FLOW DIAGRAM	13
5.2	ARCHITECTURE	15
5.3	DATA FLOW DIAGRAM	16
5.4	UML DIAGRAM	17
5.4.1	Activity Diagram	18
5.4.2	Class Diagram	19
5.4.3	Sequence Diagram	20
5.4.4	Collaboration Diagram	21
6	SYSTEM IMPLEMENTATION	
6.1	MODULES DESCRIPTION	23
6.2	ALGORITHM USED	26
7	CONCLUSION AND FUTURE ENHANCEMENT	
7.1	CONCLUSION	27
	REFERENCES	28
	APPENDIXES	29
	APPENDIX-I	
	SAMPLE CODE	29
	APPENDIX-II	
	SCREEN SHOTS	33
	APPENDIX-II	

LIST OF FIGURES

FIGURE NO.	NAME OF THE FIGURE	PAGE NO.
1.1	Cloud computing	01
1.2	Cloud Architecture	02
5.1	Architecture of server	15
5.2	Use case diagram	18
5.3	Activity diagram	19
5.4	Class diagram	20
5.5	Sequence diagram	21
5.6	Collaboration diagram	22
A.1	Server login page	33
A.2	User login page	34
A.3	User main page	34
A.4	Dataset file	35
A.5	Server file location	35
A.6	Stored file in cloud	38
A.7	Physical server location	38

LIST OF ABBREVIATIONS

CSP	- Cloud Service Provider
VSM	- Vector Space Model
BFT	- Best Fit Decreasing
IaaS	- Infrastructure as a Service
PaaS	- Platform as a Service
SaaS	- Software as a Service

CHAPTER 1

INTRODUCTION

1.1 DOMAIN INTRODUCTION

The project comes under “Cloud computing” Area

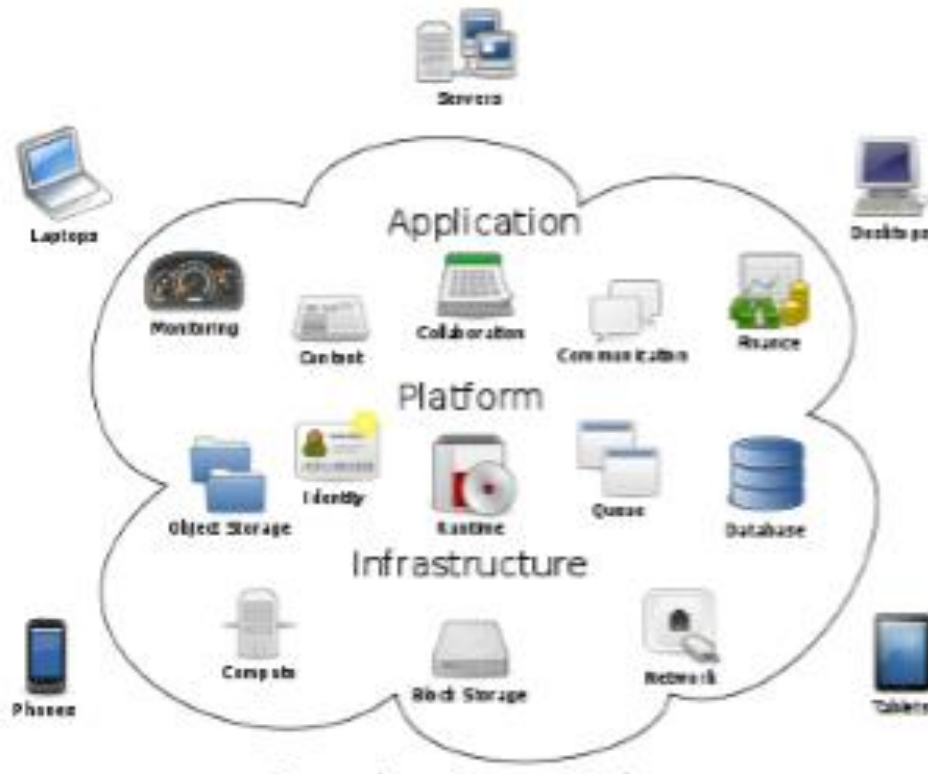


Fig 1.1 CLOUD COMPUTING

Cloud computing is the delivery of computing and storage capacity as a service to a heterogeneous community of end-recipients. The name comes from the use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts services with a user's data, software and computation over a network.

There are three types of cloud computing:

- Infrastructure as a Service (IaaS),
- Platform as a Service (PaaS), and
- Software as a Service (SaaS).

Using Infrastructure as a Service, users rent use of servers (as many as needed during the rental period) provided by one or more cloud providers. Using Platform as a Service, users rent use of servers and the system software to use in them. Using Software as a Service, users also rent application software and databases.

The cloud providers manage the infrastructure and platforms on which the applications run.

CLOUD COMPUTING ARCHITECTURE

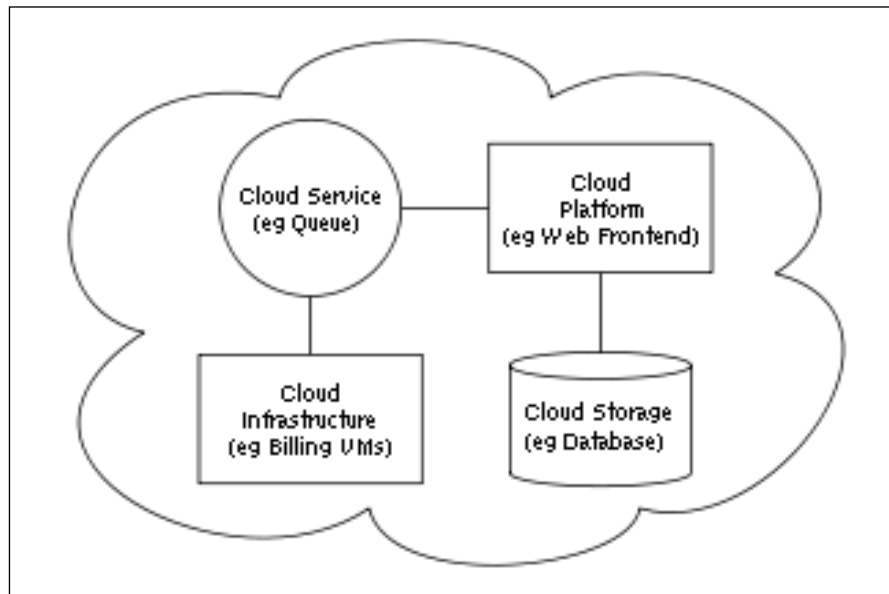


Fig 1.2 CLOUD ARCHITECTURE

Cloud architecture, the systems architecture of the software systems involved in the delivery of cloud computing, typically involves multiple cloud components communicating with each other over a loose coupling mechanism such as a messaging queue. Elastic provision implies intelligence in the use of tight or loose coupling as applied to mechanisms such as these and others.

Cloud computing types

- Public cloud
- Community cloud
- Hybrid cloud
- Private cloud

Characteristics

- Cost
- Reliability
- Flexibility
- Scalability
- Performance
- Security
- Maintenance

- Virtualization

1.2 PROJECT INTRODUCTION

VM Migration permitted by virtualization can aid in load balancing, allowing high provisioning and avoiding hot-spots in data centers which decreases energy consumption. Server consolidation helps in improving resource utilization by consolidating many VMs residing on multiple under-utilized servers onto a single server, hereafter decreasing energy consumption. Load balancing can help in decreasing energy consumption by regularly dispensing the load and decreasing the resource consumption.

Cloud computing, being an emerging technology also raises significant questions about its environmental sustainability. Through the use of large shared virtualized datacenters Cloud computing can offer large energy savings. However Cloud services can also further increase the internet traffic and its growing information database which could decrease such energy savings.

Cloud Computing is a technology that uses the internet and data centers to maintain data as well as applications. Cloud Computing allows business and consumers to access their personal files from any computer via internet. The Cloud simply denotes the default symbol of the internet in diagrams and the Computing encompasses the computation and storage. This technology allows more efficient Computing by centralized storage, processing, and band width. A simple example for this Cloud Computing is Gmail, yahoo mail etc. There is no need of software or server to use them. Just an internet connection is needed to start sending an email. There is no need to worry about the internal processing

Modern data centers operating under the Cloud computing model are hosting a variety of applications ranging from those that run for a few seconds serving requests of web applications such as e-commerce and social networks portals with transient workloads to those that run for longer periods of time simulations or large data set processing on shared hardware platforms. The need to manage multiple applications in a data center creates the challenge of on demand resource provisioning and allocation in response to time-varying workloads.

Clouds aim to drive the design of the next generation data centers by architecting them as networks of virtual services (hardware, database, user-interface, application logic) so that users can access and deploy applications from

anywhere in the world on demand at competitive costs depending on their QoS (Quality of Service) requirements . Developers with innovative ideas for new Internet services no longer require large capital outlays in hardware to deploy their service or human expense to operate it. Cloud computing offers significant benefits to IT companies by freeing them from the low-level task of setting up basic hardware and software infrastructures and thus enabling focus on innovation and creating business value for their services

Clouds aim to drive the design of the next generation data centers by architecting them as networks of virtual services. Cloud computing offers significant benefits to IT companies by freeing them from the low-level task of setting up basic hardware and software infrastructures and thus enabling focus on innovation and creating business value for their services.

CHAPTER - 2

LITERATURE SURVEY

1. IEEE Title: Energy Efficient Allocation of Virtual Machines in Cloud Data Centers

Year: 2010

Technique used:

- An efficient resource management policy for virtualized Cloud data centers.
- Heuristic Dynamic Allocation
- Best Fit Decreasing (BFD) algorithm.

Advantage:

- An efficient resource management policy for virtualized Cloud data centers.
- The objective is to continuously consolidate VMs leveraging live migration and switch off idle nodes to minimize power consumption, while providing required Quality of Service.

Drawback:

- The first part can be considered as a bin packing problem with variable bin sizes and prices.
- Higher Energy Consumption

2. IEEE Title: Dynamic Resource Management Using Virtual Machine Migrations

Year: 2012

Technique used:

- Wide-Area Virtual Machine Migration
- Holistic Approach
 - Hot Spot
 - Affinity based

Advantage:

- Outline the components required to use virtual machine migration for dynamic resource management in the virtualized cloud environment.
- Present categorization and details of migration heuristics aimed at reducing server sprawl, minimizing power consumption, balancing load across physical machines, and so on.

Drawback:

- Hot Spot is a condition of overload.
- A large discrepancy in resource utilization

3. IEEE Title: Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud Computing Environments

Year: 2011

Technique used:

- RDMA (Remote Direct Memory Access),
- sysbench and SPECjbb,
- Post-copy technology,
- Log-based migration technology, and
- Memory compression based migration technology

Advantage

- Improve the migration efficiency.

- Proposed to Multiple virtual machines with different resource reservation methods

Drawback:

- Excessive Time consuming
- Costly, and
- Excessive Energy Consumption.

4. IEEE Title: Enhancing Minimal Virtual Machine Migration in Cloud Environment

Year: 2014

Technique used:

- Two-tiered on demand resource allocation mechanism
- Min-threshold-input-length (MTIL)
- MVMM Algorithm

Advantage

- Reduces migration overhead up to 75% and the above graph plotted.
- Concentrate on minimizing VM migration and eliminate starvation.

Drawback:

- Doesn't reduce the overloading condition in virtual machines
- Doesn't minimize the migration of virtual machines

5. IEEE Title: BGM-BLA: A New Algorithm For Dynamic Migration Of Virtual Machines In Cloud Computing

Year: 2015.

Technique used:

- Virtual machine migration(VMM)
- Bucket-code learning algorithm(BLA)
- Binary graph matching(BGM)

Advantage

- Solving the DM-VM problem.
- Evaluating the candidate solutions.

Drawback

- Some open source tools like cloudsim will be employed to do the works again, for comparison.
- The generation, learning and mutation method of Bucket Code can be further improved to reduce its asymmetry property.

CHAPTER 3

SYSTEM ANALYSIS

3.1 SCOPE

The main scope of the project is to achieve efficient and secure server migration on cloud storage with vsm using drop box service.

3.2 OBJECTIVE

The main objectives of the project are

- Store data in the form of bucket code and store in the cloud space.
- Using migration tool VM migration to migrate the receiving data in active server.
- Finding the sever space using migration tool.

3.3 PROBLEM DEFINITION

The definition of the problem can be defined as a detailed and operational description of the differences between the existing situation and the desired situation. The DM-VM problem is divided into two parts: (a) forming VMs into groups (b) determining the best way to place the groups into certain physical nodes. In this the problem of dynamic migration of Cloud computing which shares the characteristics of distributed computing and virtualization technology develops at an amazing rate in recent years. Pointed out that data centers for cloud computing usually do not perform efficiently

3.4 EXISTING TECHNIQUE

The existing technique has a secure multi-keyword ranked search scheme over encrypted cloud data, which simultaneously supports dynamic update operations like deletion and insertion of documents. Specifically, widely-used TF X IDF model are combined in the index construction and query generation. We

construct a special tree-based index structure and propose a “Greedy Depth-first Search” algorithm to provide efficient multi-keyword ranked search. The secure kNN algorithm is utilized to encrypt the index and query vectors, and meanwhile ensure accurate relevance score calculation between encrypted index and query vectors

3.4.1 Disadvantage

1. These methods are not practical due to their high computational overhead for both the cloud sever and user.
2. The query unlink ability is not protected,
3. The confidentiality of the index and query are not protected that the original vectors are kept from the cloud server.

3.5 PROPOSED TECHNIQUE

In this paper, we propose a practically efficient and flexible searchable encrypted scheme supporting dynamic update. To address multi-keyword search and result ranking, we use Vector Space Model (VSM) to build document index and for enhancing the searching capability DMRS algorithm has been proposed.

3.5.1 Advantage

1. Similarity based top-ranked migration of server,
2. Our solution could return not only the exactly matched files, but also the files including the terms semantically related to the query keyword.

CHAPTER 4

SYSTEM REQUIREMENT SPECIFICATION

The purpose of the Software Requirement Specification is to produce the specification of the analysis task and also to establish complete information about the requirement, behavior and also the other constraint like functional performance and so on. The main aim of the Software Requirement Specification is to completely specify the technical requirements for the software product in a concise and in unambiguous manner.

4.1 HARDWARE REQUIREMENTS

Hardware is the physical components of the computer like microprocessor, hard disks, RAM, and motherboard. Hardware devices are the executors of the commands provided by software applications. Computer hardware as the electronic, magnetic, and electric devices that carry out the computing functions

Processor	: Pentium IV
Clock speed	: 550MHz
Hard Disk	: 20GB
RAM	: 128MB
Cache Memory	: 512KB
Operating System	: Windows XP/ Windows7

4.2 SOFTWARE REQUIREMENTS

Software is a set of instructions that are used to command any system to perform any operation. Software has the advantage to make decisions and to deliver sensible results and is useful in handling complex situations

FRONT END	: JAVA 1.7
BACK END	: My SQL

4.3 OVERVIEW OF JAVA

One of the most compelling reasons to move to java is its platform independence. Java runs on most major hardware and software platforms, including windows 98, 95, and NT Macintosh and several varieties of UNIX. Java is a general USA in 1991. Originally called oak by James Gosling, one of the inventors of the language. The java development team which included Patrick Naughton discovered that the existing language like C and C++ had limitations in terms of both reliability and

portability. However, they modeled their new language java on C and C++ but removed a number of features of C and C++ that were considered as sources of problems and thus made java a really simple, reliable, portable and powerful language. Specifically, this overview will include a bit of the history of java platform, touch of the java programming language, and the ways in which people are using java applications and applets, now and in the likely future.

After going a while down the path of consumer – electronics devices, they realized that they had something particularly cool in the java language and focused on it as a language for network computing. Sun formed the java soft group which in a little over three years has grown to over six hundred people working on java related technologies. Java quickly became a hot buzzword of the computing industry. People wanted to know Java - it was said to be great for creating dynamic interactive content for WebPages. Yet the true power of Java lies not in applets, but in its many other uses. Java is used for developing standalone applications, and for server-side programming. The face of Java has changed, but the core language remains the same. In this tutorial series, I'll teach you the basics of Java programming. You'll still need a good book as a companion to this tutorial series, but for those who are dabbling in Java; this should be enough to get your feet wet.

Java is an object-orientated language, and may not be suitable for first time programmers. Learning a new language takes some time, but learning your first object-orientated language can be exceedingly difficult. Nonetheless, if you've done some C programming before, the shift into Java shouldn't be unreachable, providing you obtain a good reference book. There are also those that believe programmers should start with an object-orientated language first, and many universities have adopted this practice. Still, you've been fairly warned :) Java is two things: a programming language and a platform.

4.4 Java Swing

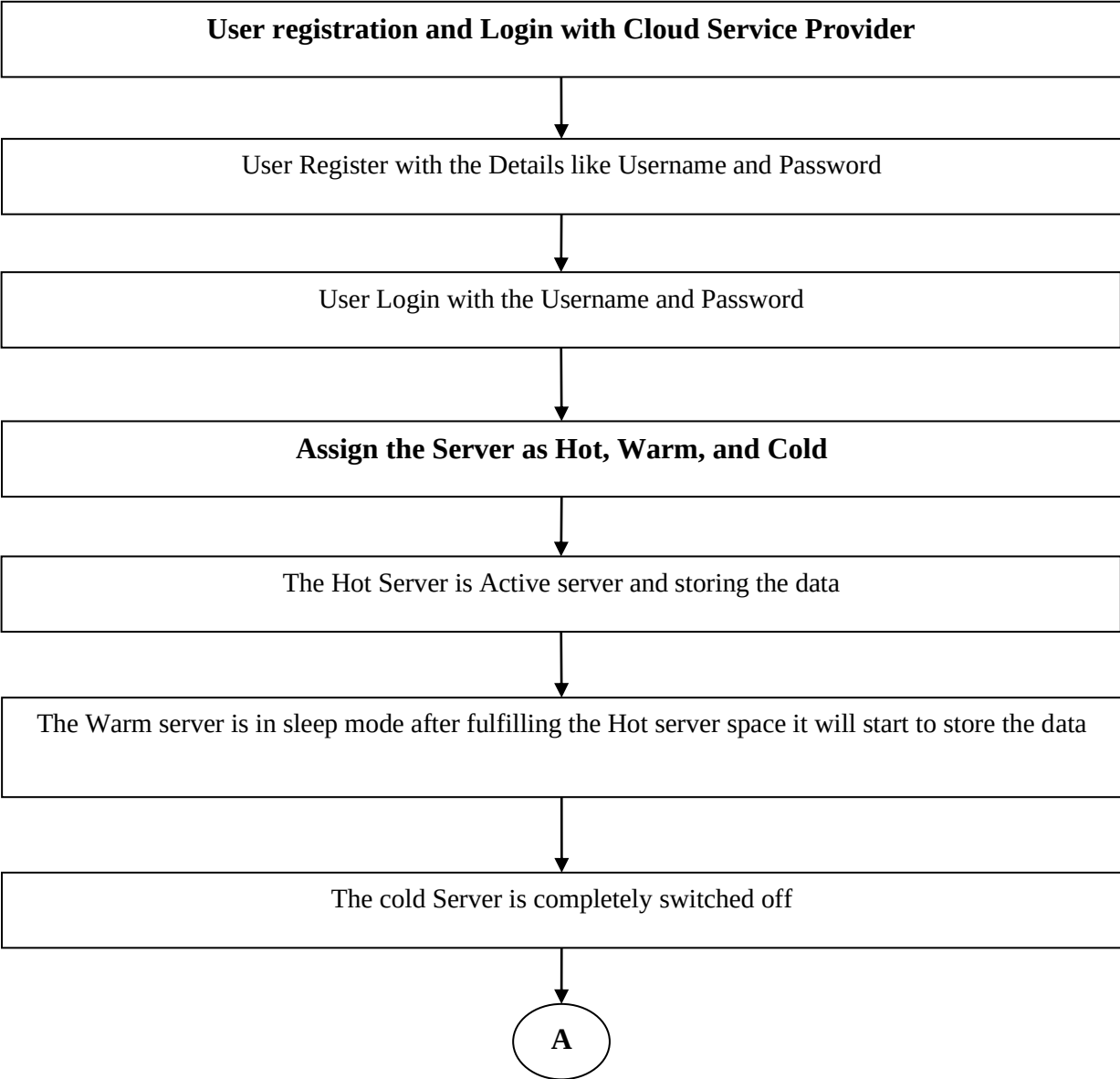
Java Swing is a Library/toolkit released by SUN Microsystems as a part of Java Language which enables Java Programmers to create GUI and rich client applications. Swing is the primary Java GUI widget toolkit. It is a part of Oracle's Java Foundation Classes (JFC) – an API for providing a graphical user interface (GUI) for Java programs. Swing was developed to provide a more sophisticated set of GUI components than the earlier Abstract Window Toolkit. Swing provides

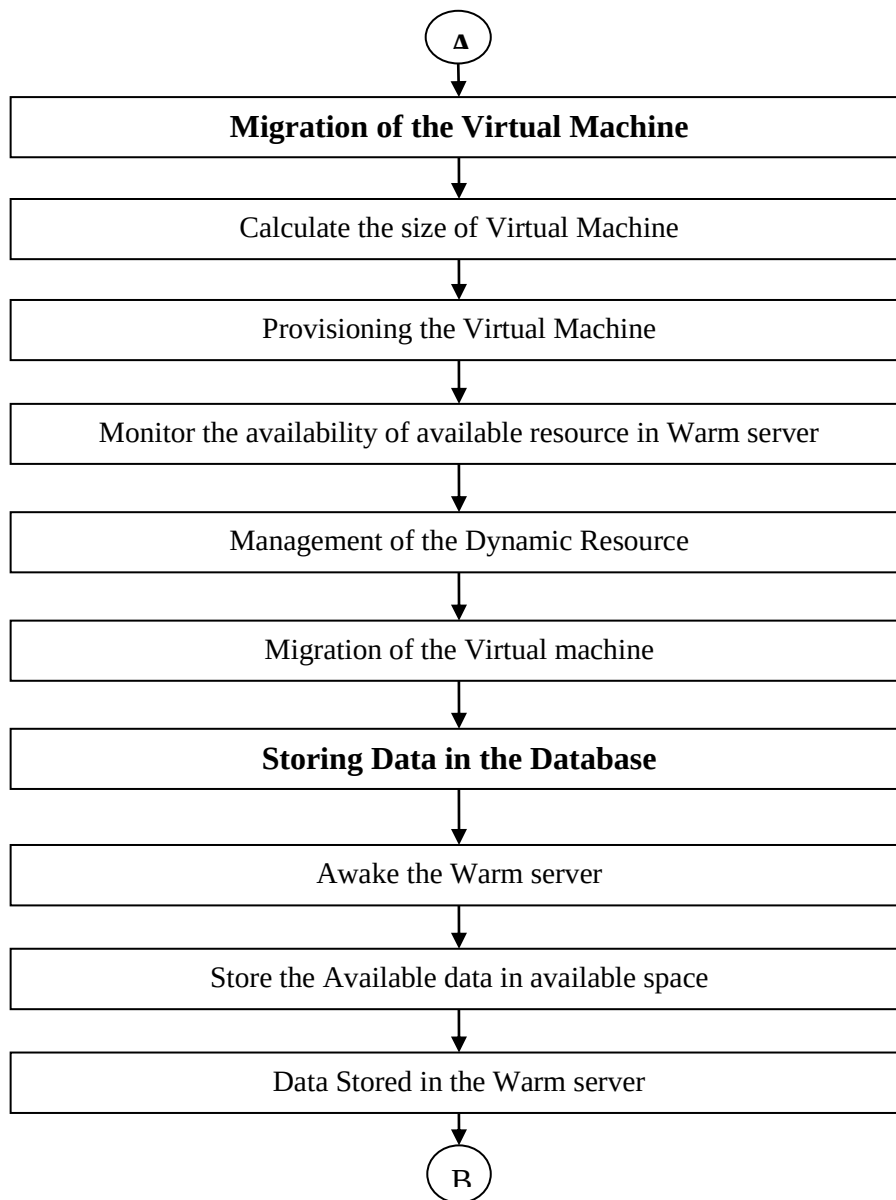
native look and feels that emulates the look and feel of several platforms and also supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components like buttons, check box and labels. Swing provides several advanced components like tabbed panel, scroll panes, trees, tables and lists. Unlike AWT components, Swing components are not implemented by the platform-specific code. Instead they are written entirely in Java and therefore are platform-independent. The term “lightweight” is used to describe such an element.

CHAPTER 5

SYSTEM DESIGN

5.1 FLOW DIAGRAM





5.2 ARCHITECTURE

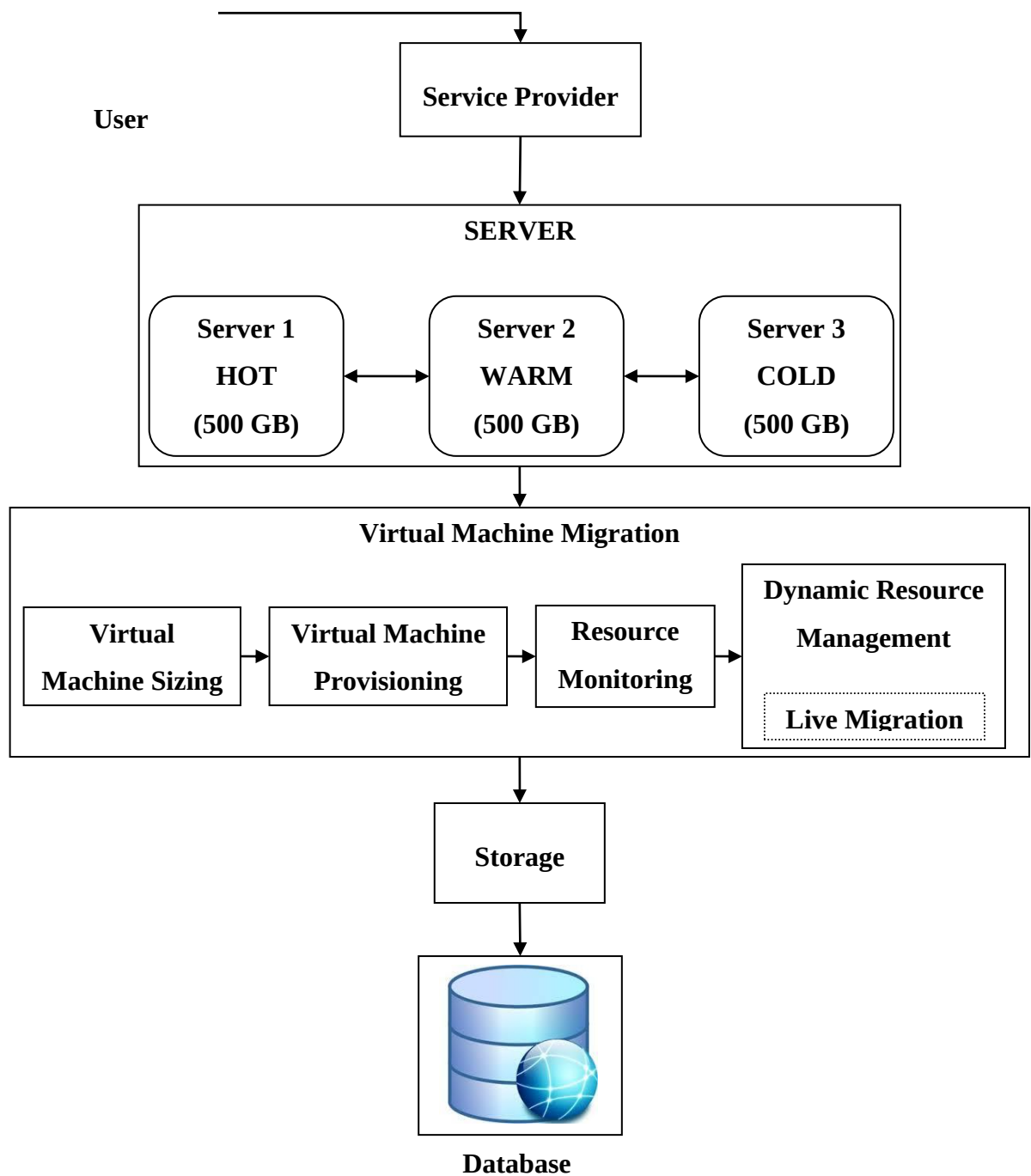


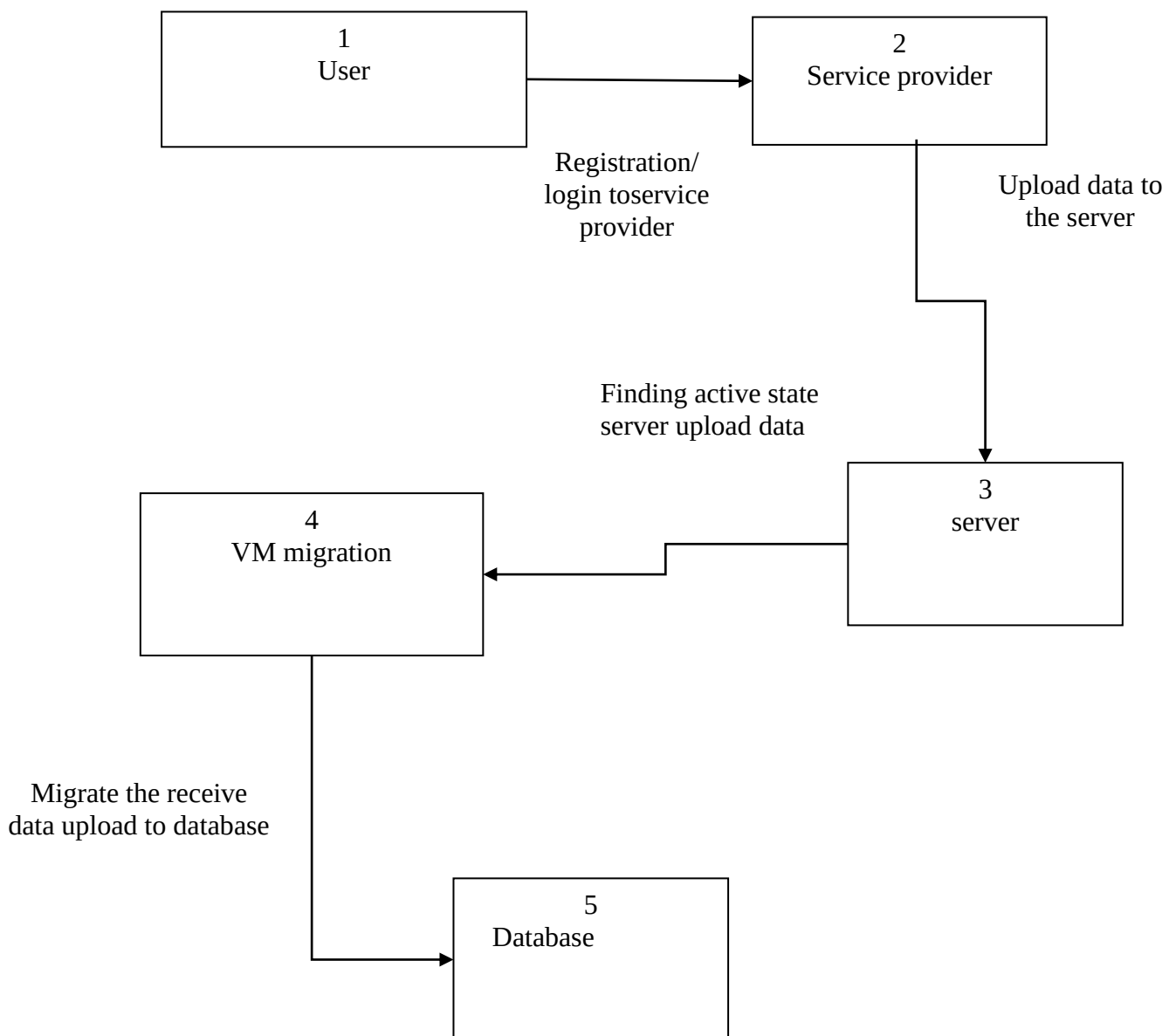
Fig 5.1 ARCITECTURE OF SERVER

5.3 DATAFLOW DIAGRAM

The Data Flow diagram is a graphic tool used for expressing system requirements in a graphical form. The DFD also known as the “bubble chart” has the purpose of clarifying system requirements and identifying major

transformations that to become program in system design. Thus DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. The DFD consist of series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system. A DFD describes what that data flow in rather than how they are processed. So it does not depend on hardware, software, data structure or file organization.

Data flow diagrams are used to describe how the system transforms information. They define how information is processed and stored and identify how the information flows through the processes. A DFD provides no information about the timing of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart.



a. UML DIAGRAMS

USE CASE DIAGRAM

Use Case Diagrams describes the relationships and the dependencies between a group of Use Cases and the Actors participating in the process. Use Case Diagrams are meant to facilitate the communication with the future users of the system and with the customer and are especially helpful to determine the required features of the system is to have. Use Case Diagrams describes what the system should do but do not and cannot specify how this is to be achieved.

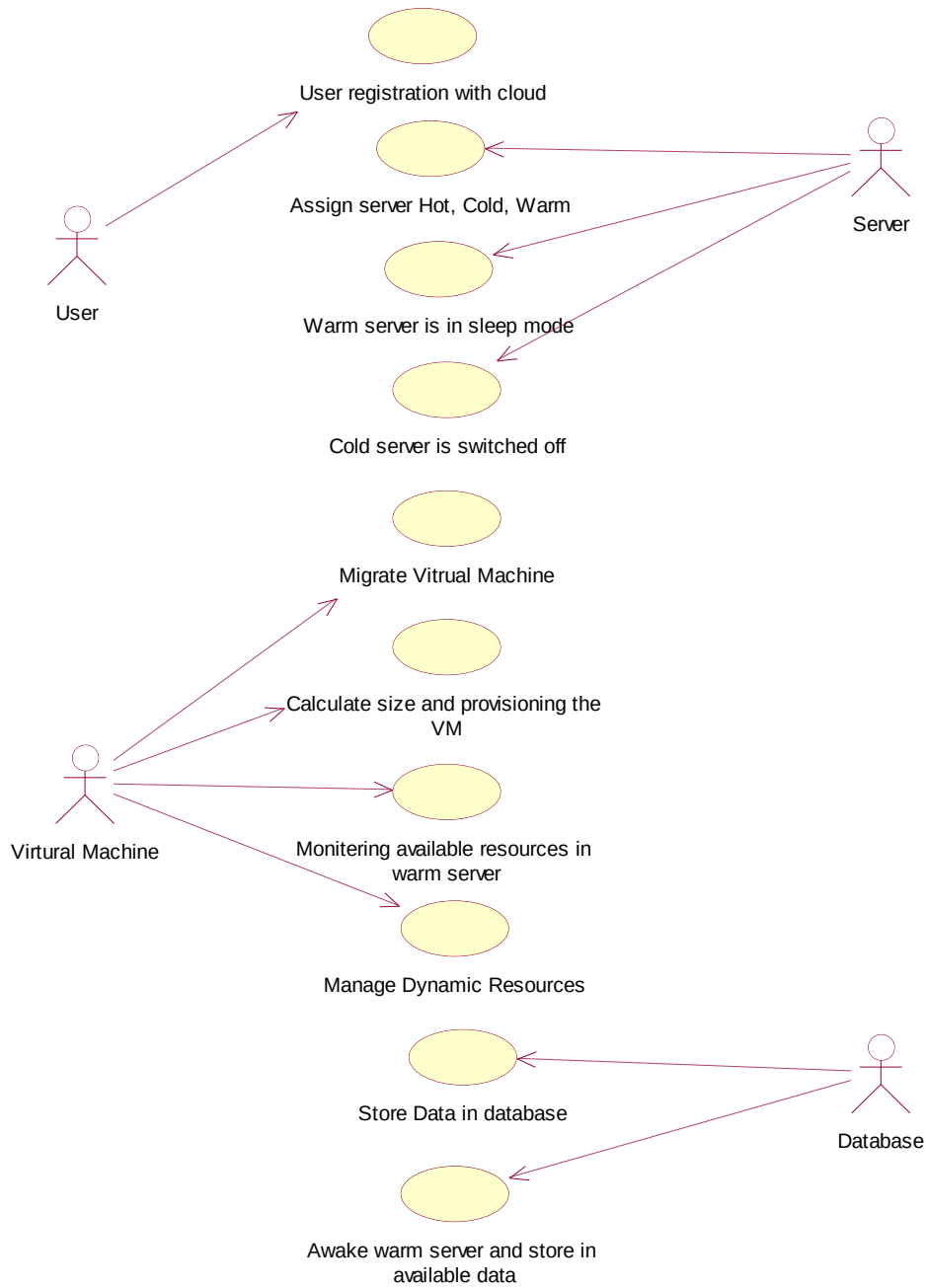


Fig 5.2 USE CASE DIAGRAM

5.4.1 ACTIVITY DIAGRAM

Activity Diagrams describes about the sequences of the activities in a system with the help of Activities. An Activity is a single step in a process. One Activity is one state in the system with internal activity and atleast one outgoing

transition. Activities can also have more than one outgoing transition if they have different conditions.

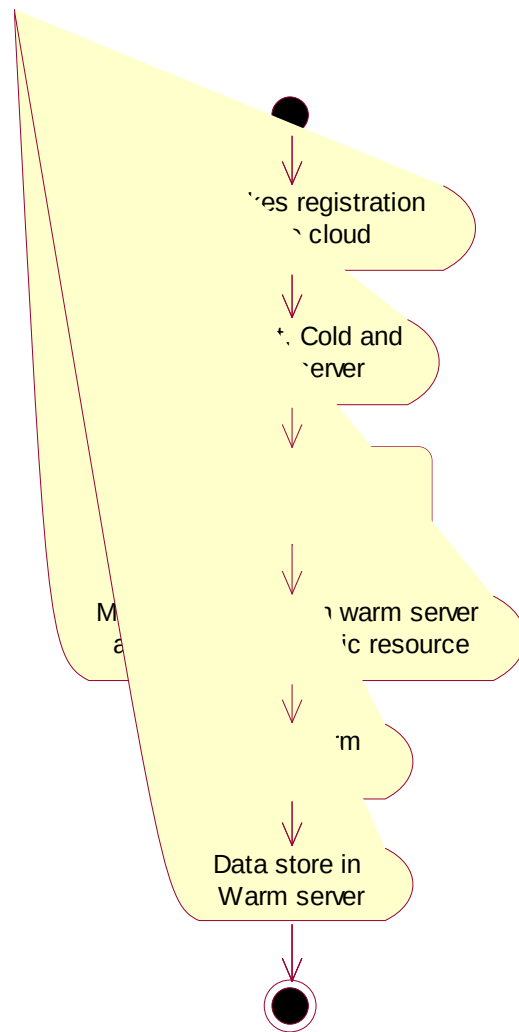


Fig 5.3 ACTIVITY DIAGRAM

5.4.2 CLASS DIAGRAM

The Class Diagram is the main building block of object oriented modeling. It is used for both general conceptual modeling of the systematic of the application and also for the detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects and or interactions in the application and the objects to be programmed. In the class diagrams these classes are represented with the boxes which contain three parts.

- The upper part holds the name of the class
- The middle part contains the attributes of the class

The bottom part gives the methods or operations of the class which can take or undertake.

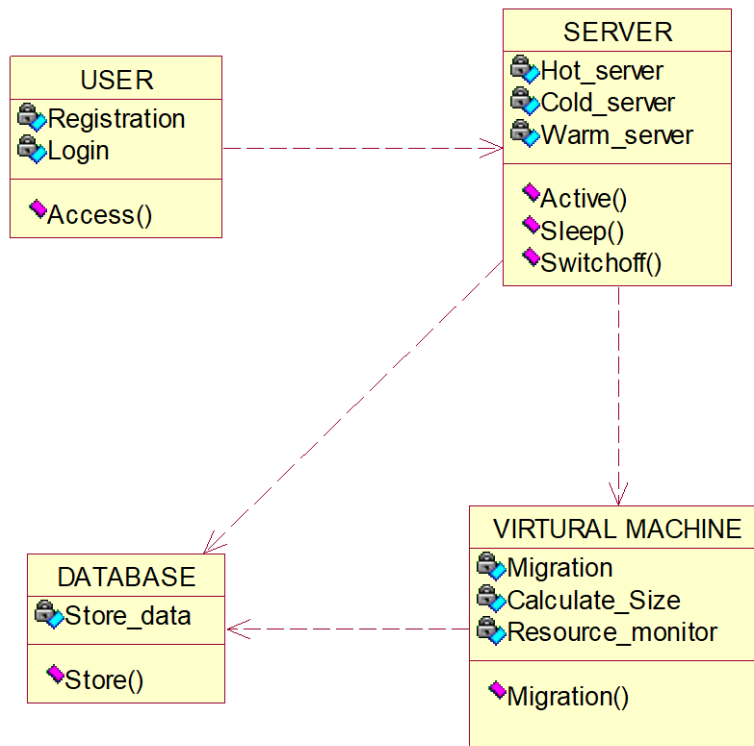


Fig 5.4 CLASS DIAGRAM

5.4.3 SEQUENCE DIAGRAM

Sequence Diagrams shows the message exchange (i.e. method call) between the several objects in a specific time-delimited situation. Sequence Diagrams put special emphasis in the order and the times in which the messages to the objects are sent. The following Diagrams explain about sequence process.

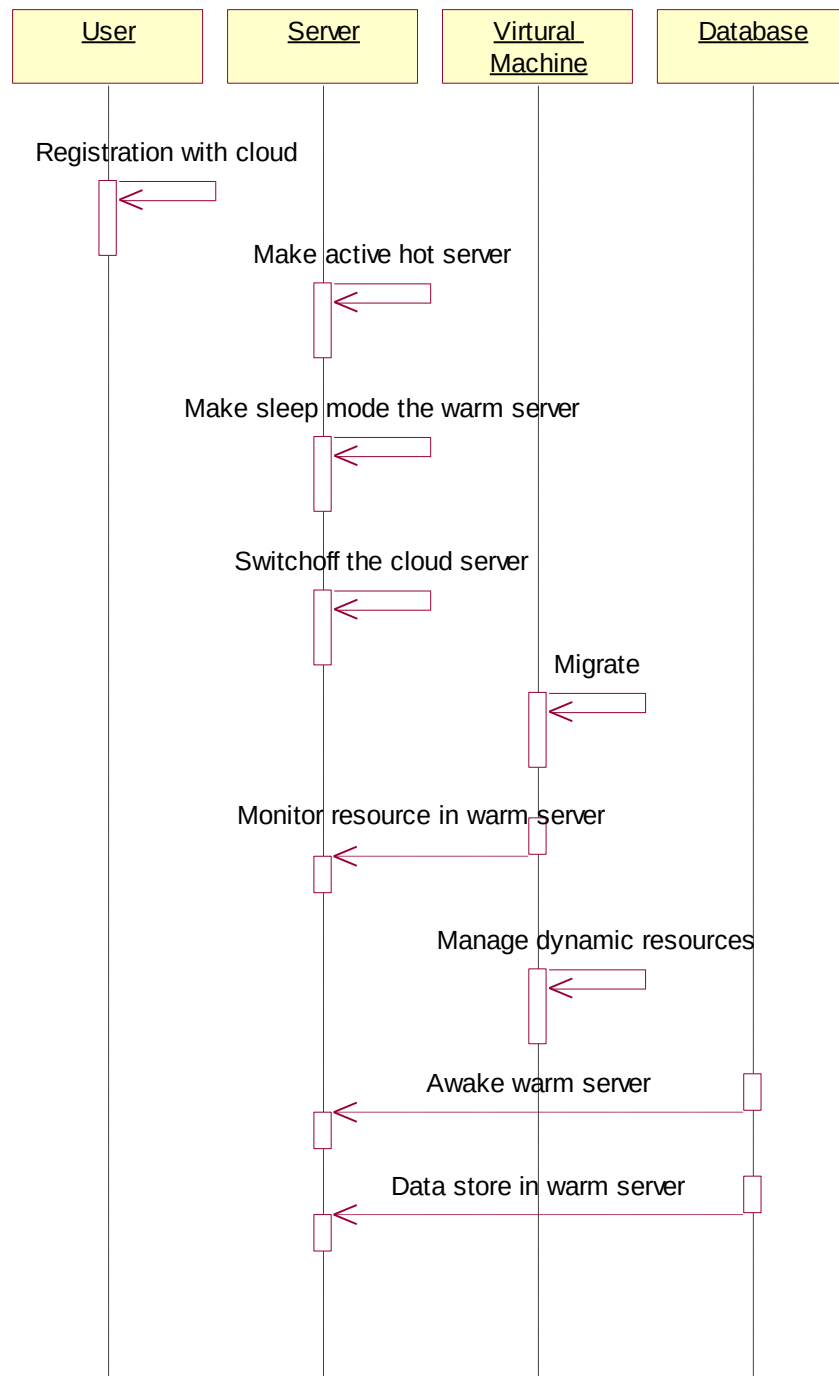


Fig 5.5 SEQUENCE DIAGRAM

5.4.4 COLLABORATION DIAGRAM

Collaboration Diagram depicts how collaborations are wired together to form larger collaborations between the entities. They are used to illustrate the

structure of arbitrarily complex systems. Collaborations are wired together by using an assembly connector with number between the connections.

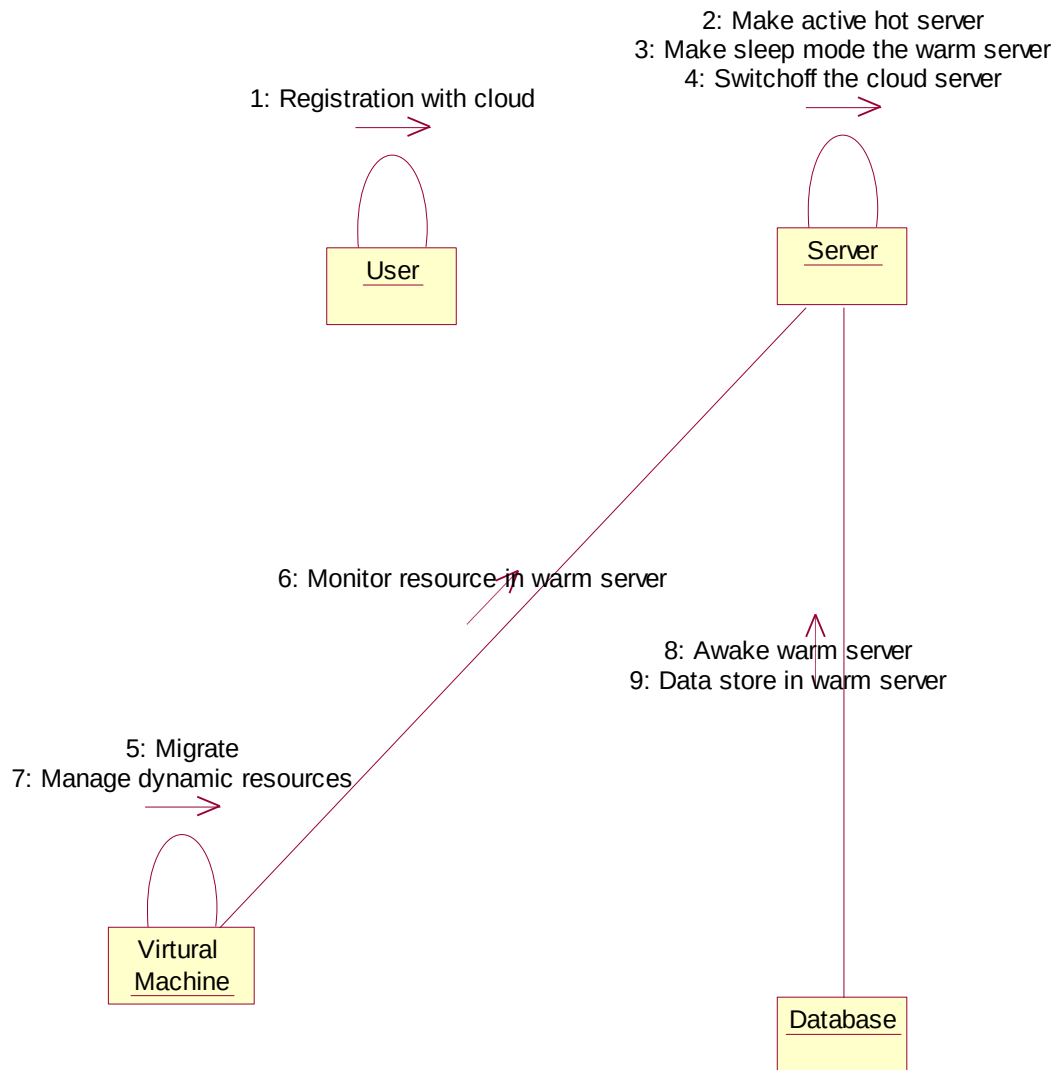


Fig 5.6 COLLABORATION DIAGRAM

CHAPTER 6

SYSTEM IMPLEMENTATION

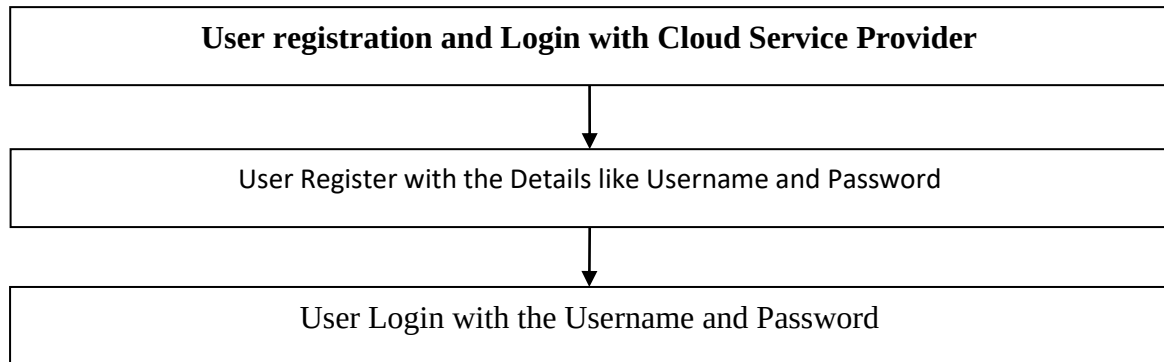
MODULES DESCRIPTION

Module description

1. User registration and Login with Cloud Service Provider

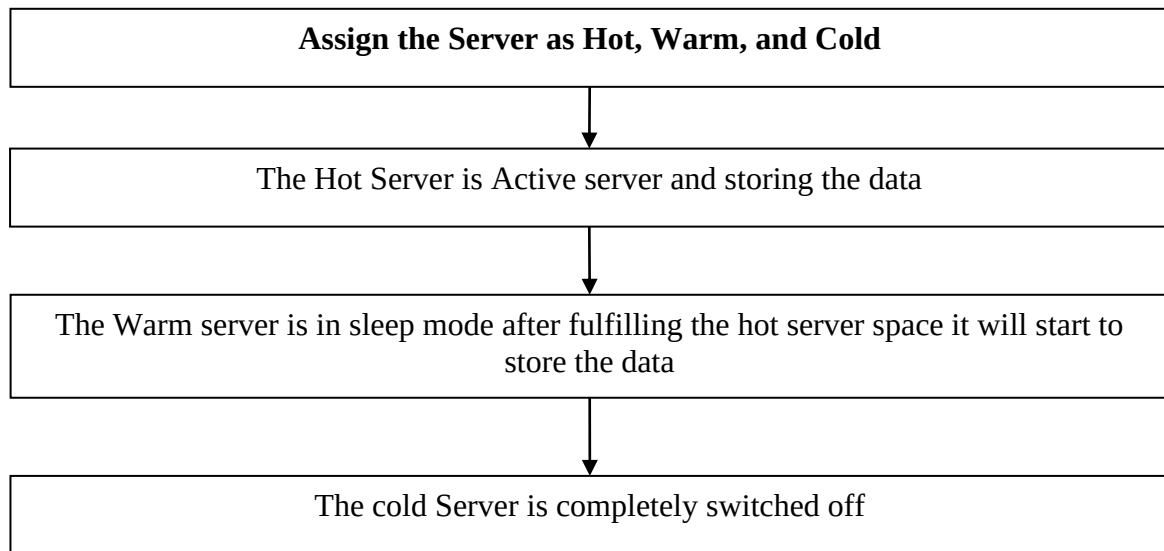
2. Assign the Server as Hot, Warm, and Cold,
3. Migration of the Virtual Machine, and
4. Storing in the Database.

1. User registration and Login with Cloud Service Provider



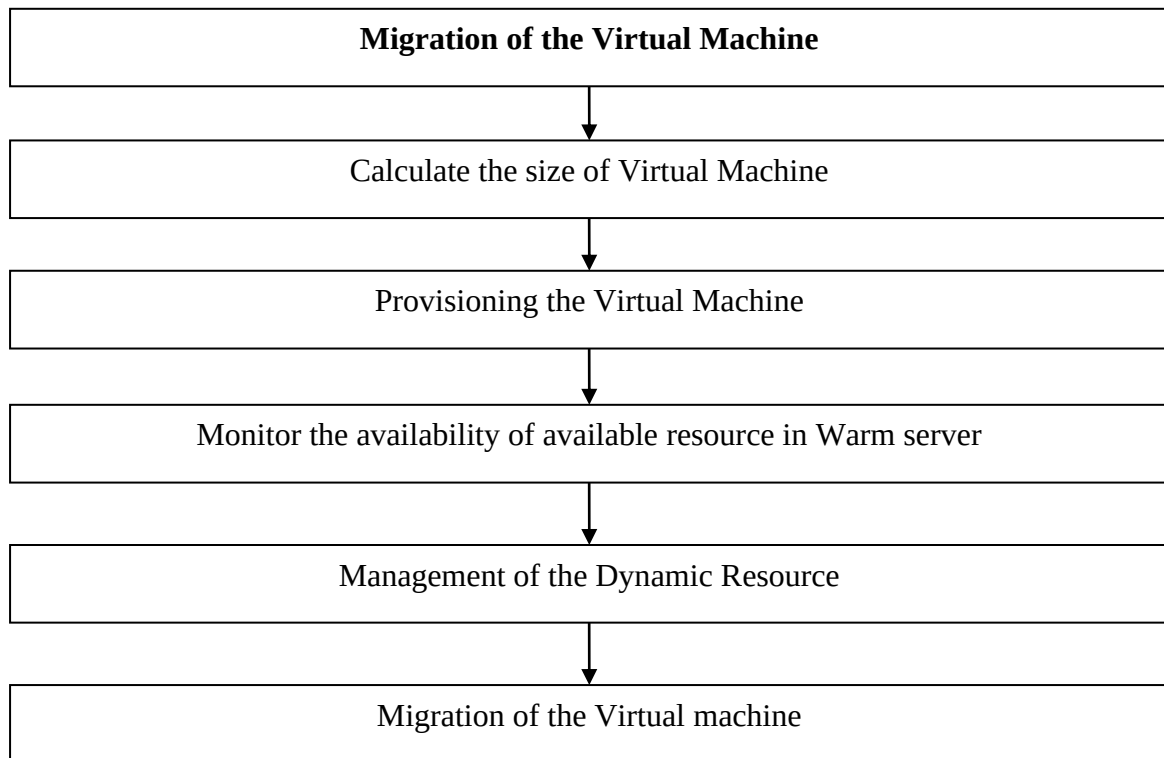
The cloud service and user are making a registration and login within the cloud services by using username and password. The information of the users are being stored in the database and validating the access right.

2. Assign the Server as Hot, Warm, and Cold,



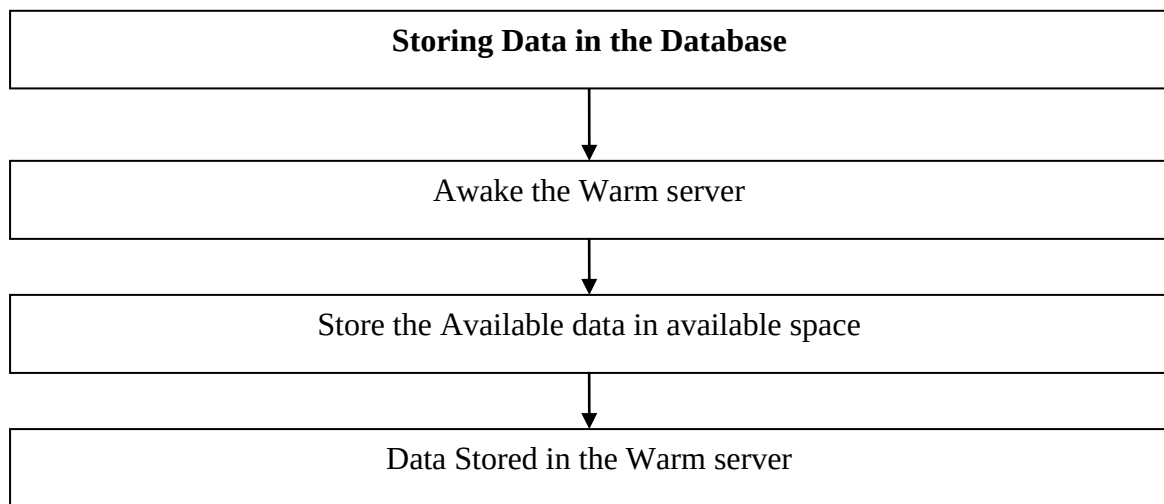
The second module is presenting the status of the cloud server by categorizing into three parts as Hot, Warm and Cold. The Hot server is standing for the active server or currently working server for storing the data in the cloud storage. The Warm serve is being ion the ready to start mode and creating an instance of active server. The third type of serve cold is being in sleep mode which not taking any part in the storage.

3. Migration of the Virtual Machine



The third module is calculating the size of server and computing the virtual machine migration instance within the available resources. The provisioning process is taking a place for communicating to previous server for making an instance of the data by migrating the server, where it's monitoring the storage space of the server and the total size of the stored data in the previous server and then locating instance into next hot server by computing the size of available space. The migration process is taking place in the dynamic management of the available resource over the instance of the server and making a live migration.

4. Storing in the Database



The above described module is presenting the database storage capability by awaking the server, the warm server is being awake once the hot server getting full memory, it also taking and storing the instance of previous hot server.

ALGORITHM USED

```
Protected void evaluateExtraParams()
{
    Boolean isEmptyList= true;
    Collection<String>actionMessage = (List) findValue("actionErrors");
    If(actionMessage: actionMessage){
    If(StringUtils.isNotBlank(message)){
    isEmptyList= false;
    break; }}}
    addParameters("isEmptyList",isEmptyList);
    addParameters("escape",escape);
}
```

CHAPTER 7

CONCLUSIONS & FUTURE ENHANCEMENT

CONCLUSIONS

This Project presents the hypothetical study of heterogeneous dynamic resource allocation techniques in cloud computing environment. We have presented the design, implementation, and evaluation of a resource management system for cloud computing services. Our system multiplexes virtual to physical resources adaptively based on the changing demand. Cloud computing can resolve obscure set of tasks in shorter time by proper resource utilization. The proposed system multiplexes virtual to physical resources adaptively based on the changing needs. To use the skewness to merge VMs with different resource characteristics judiciously so that the capacities of servers are best utilized. The algorithm attains both overload avoidance and green computing for systems with multi resource constraints.

REFERENCES

1. Sukhpal Singh and Inderveer Chana, "Energy based Efficient Resource Scheduling: A Step Towards Green Computing" 2014,

*Computer Science and Engineering Department, Thapar University,
Patiala, India*

2. RajkumarBuyya, Anton Beloglazov, and JemalAbawajy, “Energy-Efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges”.
3. Manasa H.B, AnirbanBasu, “ Energy Aware Resource Allocation in Cloud Datacenter” International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013 277
4. Ramya, “VM Assignment Algorithm Based Cost Effective Caching in Cloud Computing
5. Xi Chen, Lukas Rupperecht, Rasha Osman, Peter Pietzuch, William Knottenbelt, “CloudScope: Diagnosing and Managing Performance Interference in Multi-Tenant Clouds”

APPENDIXES

APPENDIX-1

SAMPLE CODE

```
public class ServerMainpageextendsJFrame {  
    Toolkittk;
```

```

Dimension dim;
Container con;
JPanel pan1, loginpanel;
JLabel title;

public static int server1Space1 = 2;
public static int serverSpace2 = 2;
public static int serverSpace3 = 2;
public static JTextArea jt1,jt2,jt3;
public static JScrollPane jsc1,jsc2,jsc3;
public static Vector server1status = new Vector();
public static Vector server2status = new Vector();
public static Vector server3status = new Vector();
ServerMainpage() {
    try {
this.setTitle("");

    UIManager.setLookAndFeel("com.jtattoo.plaf.aluminium.AluminiumLookAndFeel");
        dim = tk.getDefaultToolkit().getScreenSize();
        con = this.getContentPane();
con.setBackground(new Color(31, 122, 31));
con.setLayout(null);
System.out.println("server ip" + InetAddress.getLocalHost().getHostAddress());
        pan1 = new JPanel();
        pan1.setBounds(370, 40, 600, 100);
        title = new JLabel("Server MainPage ", SwingConstants.CENTER);
title.setFont(new Font("Verdana", Font.BOLD, 20));
title.setForeground(Color.black);
        pan1.setLayout(new BorderLayout());
        pan1.add(title, BorderLayout.CENTER);
con.add(pan1);
JPanel panel1 = new JPanel();
        panel1.add(new JLabel(new ImageIcon("server1.jpg")));
        panel1.setBounds(230, 250, 200, 200);
        panel1.setBorder(BorderFactory.createTitledBorder("Server1"));
con.add(panel1);

```

```

        jt1 = new JTextArea();
        jsc1 = new JScrollPane(jt1);
        jsc1.setBounds(230, 500, 200, 50);
con.add(jsc1);
JPanel panel2 = new JPanel();
        panel2.setBounds(530, 250, 200, 200);
        panel2.add(new JLabel(new ImageIcon("server2.jpg")));
        panel2.setBorder(BorderFactory.createTitledBorder("Server2"));
con.add(panel2);
        jt2 = new JTextArea();
        jsc2 = new JScrollPane(jt2);
        jsc2.setBounds(530, 500, 200, 50);
con.add(jsc2);
JPanel panel3 = new JPanel();
        panel3.setBounds(830, 250, 200, 200);
        panel3.add(new JLabel(new ImageIcon("server3.jpg")));
        panel3.setBorder(BorderFactory.createTitledBorder("Server3"));
con.add(panel3);
        jt3 = new JTextArea();
        jsc3 = new JScrollPane(jt3);
        jsc3.setBounds(830, 500, 200, 50);
con.add(jsc3);
        if( server1status.isEmpty())
            jt1.append("Size: 2mb");
        else
            jt1.append("Size: 2mb");
        if( server2status.isEmpty())
            jt2.append("Size: 2mb");
        else
            jt2.append("Size: 2mb");
        if( server3status.isEmpty())
            jt3.append("Size: 2mb");
        else
            jt3.append("Size: 2mb");
this.setSize(new Dimension(tk.getDefaultToolkit().getScreenSize().getSize()));

```

```

this.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent we) {
JOptionPane.showMessageDialog(null, "THANKS FOR USING PROGRAM");
System.exit(0);
    }
});
this.show();
    }
    catch(
    Exception Ex
    )
    {
Ex.printStackTrace();
    }
}

    public static void appendinserver1(String t){
JOptionPane.showMessageDialog(null,"current size of server1:"+t);
    }

    public static void appendinserver2(String t){
JOptionPane.showMessageDialog(null, "current size of server2:" + t);

    public static void appendinserver3(String t){
JOptionPane.showMessageDialog(null, "current size of server3:" + t);
    }
    public static void main(String ar[]) {
        new ServerMainpage();
    }
}

```

APPENDIX-2

SCREEN SHOTS

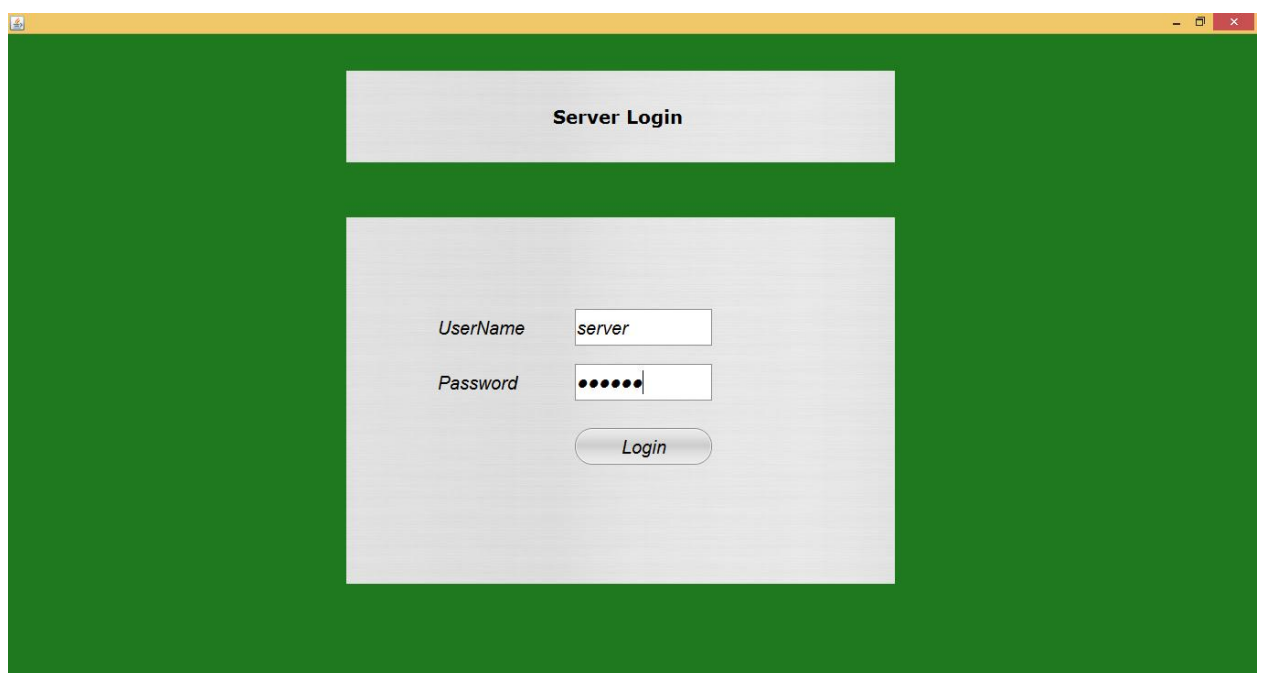


Fig A.1 SERVER LOGIN PAGE

Server main page



A screenshot of a web application window titled "User Login / Registration". The background is a solid green color. At the top center, there is a white rectangular box with the text "User Login / Registration". Below this box, there is a larger white rectangular box containing a login and registration form. The form has two input fields: "UserName" with the value "raja" and "Password" with four dots. Below the input fields, there are two buttons: "Register" and "Login".

Fig A.2 USER LOGIN

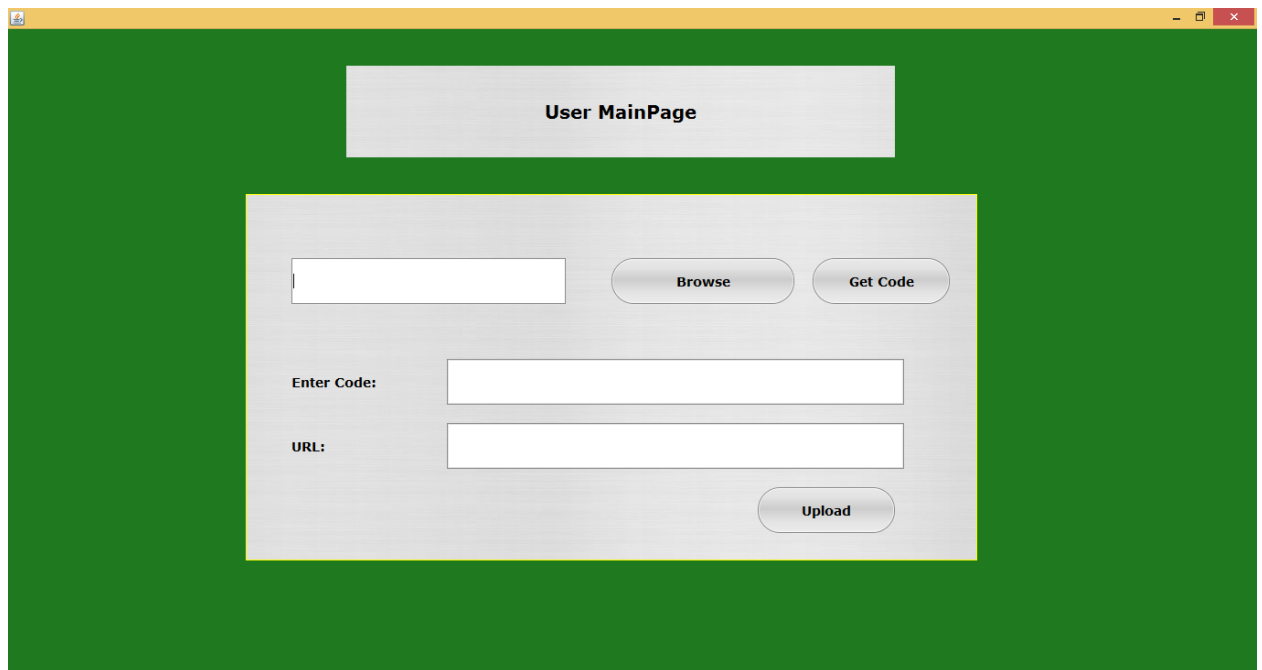


Fig A.3 USER MAIN PAGE

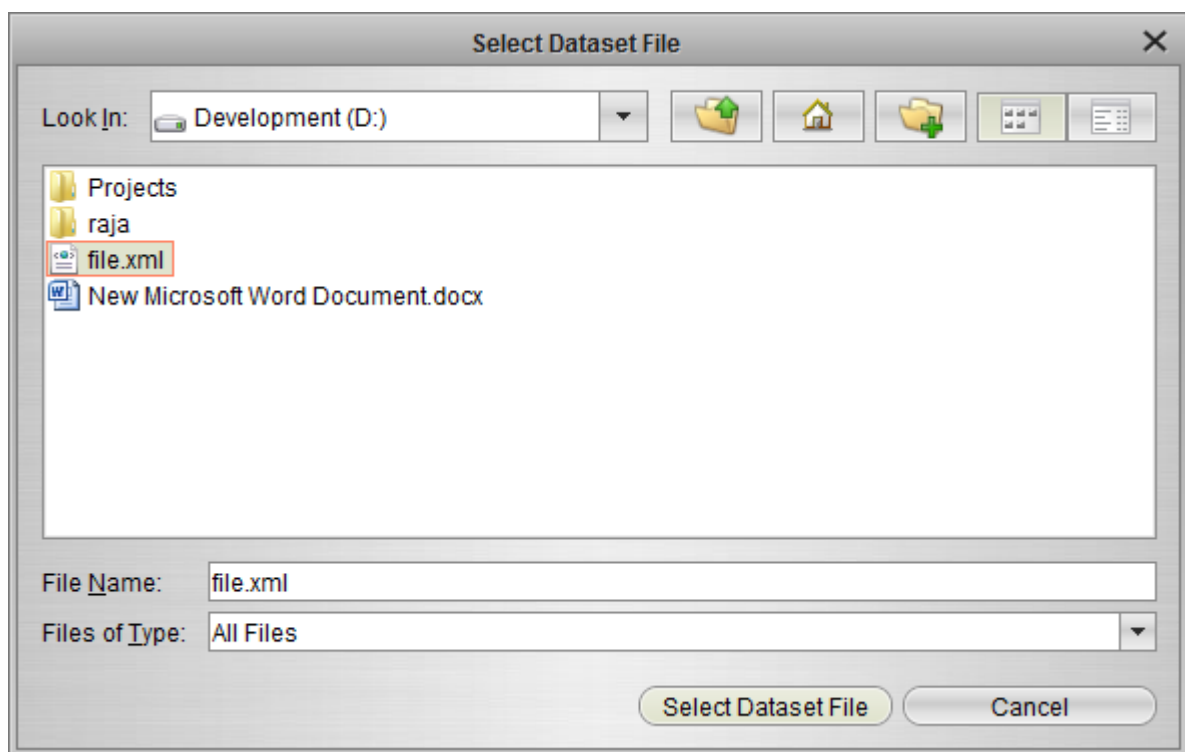


Fig A.4 DATASET FILE

Server File Access

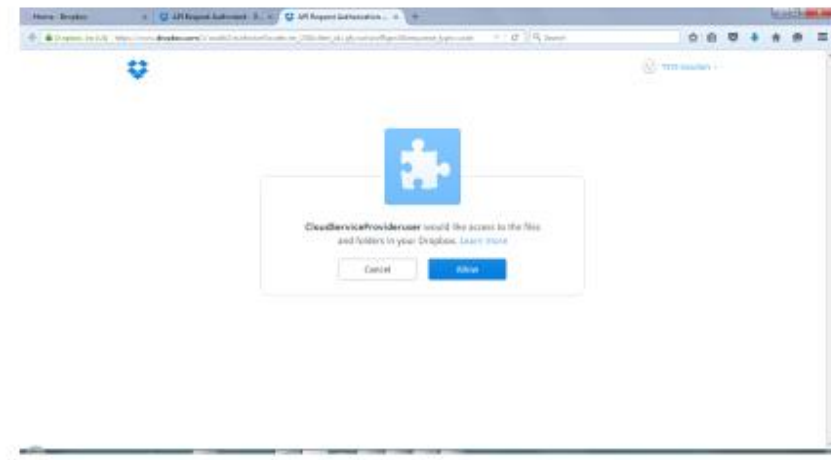


Fig A.5 SERVER FILE LOCATION

User MainPage

D:\file.xml

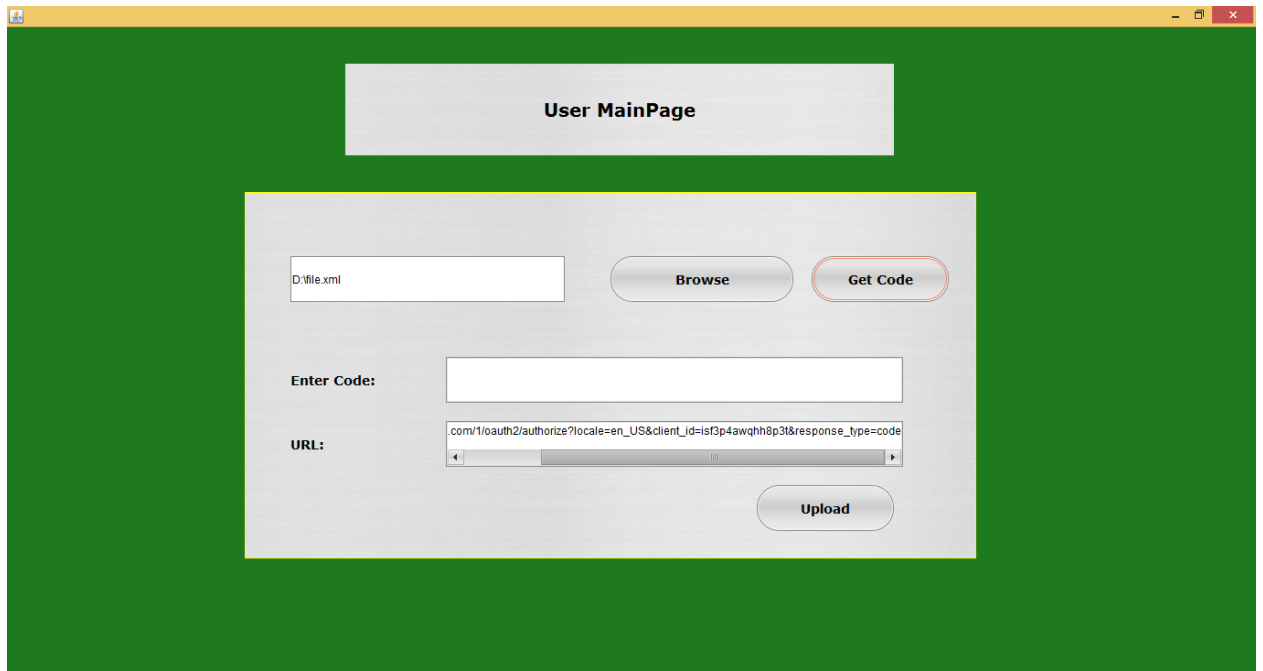
Browse

Get Code

Enter Code:

URL:

Upload



Successfully Upload of Data



File Stored in Cloud

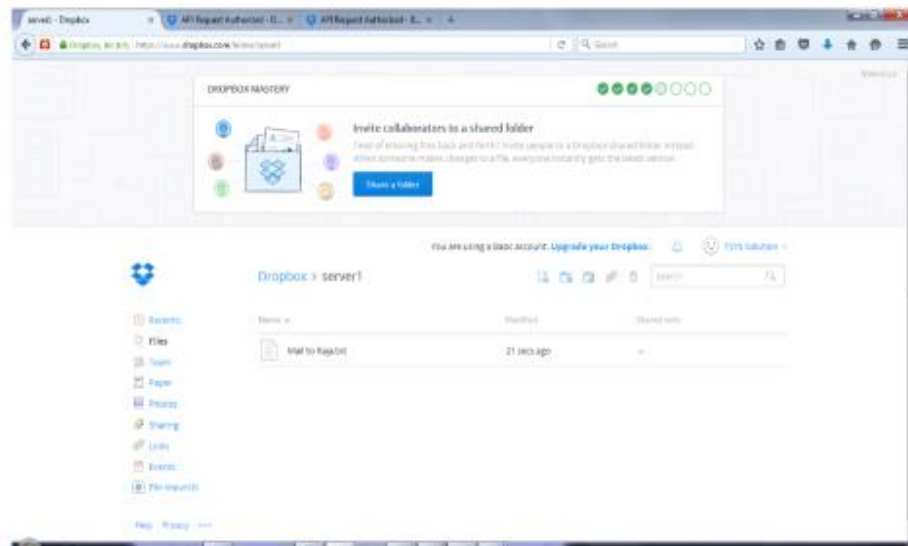


Fig A.6 STORED FILE IN CLOUD

Backup in Physical Server Location

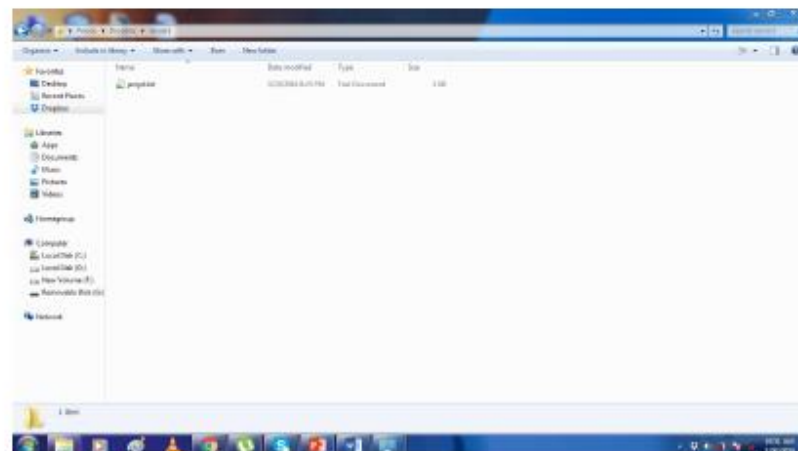


Fig A.7 PHYSICAL SERVER LOCATION