

# **PLEDGE STORAGE RECKON**

A REPORT

submitted by

**P.NITYASREE (17MIS1007)**

*in partial fulfilment for the award*

of

**M. Tech. Software Engineering (Integrated)**

**School of Computer Science and Engineering**



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

**JUNE 2020**



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

## **School of Computer Science and Engineering**

### **DECLARATION**

I hereby declare that the project entitled “**PLEDGE STORAGE RECKON**” submitted by me to the School of Computer Science and Engineering, Vellore Institute of Technology, Chennai Campus, Chennai 600127 in partial fulfilment of the requirements for the award of the degree of **Master of Technology - Software Engineering (Integrated)** is a record of bonafide work carried out by me. I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or university.

Signature

**P.NITYASREE**  
**(17MIS1007)**



**VIT<sup>®</sup>**

**Vellore Institute of Technology**

(Deemed to be University under section 3 of UGC Act, 1956)

## **School of Computer Science and Engineering**

### **CERTIFICATE**

The project report entitled “**PLEDGE STORAGE RECKON**” is prepared and submitted by **P.NITYASREE (Register No: 17MIS1007)**. It has been found satisfactory in terms of scope, quality and presentation as partial fulfilment of the requirements for the award of the degree of **Master of Technology – Software Engineering (Integrated)** in Vellore Institute of Technology, Chennai, India.

**Examined by:**

**Examiner     I**

**Examiner     II**

# INSERT THE CERTIFICATE OF MERIT OBTAINED FROM THE INDUSTRY WHERE YOU DID YOUR INTERNSHIP



24<sup>th</sup> September 2020

## INTERN COMPLETION CERTIFICATE

This is to certify that Ms P.Nityasree (Reg. No. 17MIS1007) a student of M.Tech (INTEGRATED SOFTWARE ENGINEERING ), Vellore Institute of Technology, Chennai has successfully completed the Internship in ANDRIOD platform from June 2020 to July 2020 in our company. During the period, she had been exposed to different processes and found to be Punctual, Hard Working and Inquisitive.

We wish her every success in life and career.

For Shia#sh Info Solutions Private Limited

Ashwini Kanniyappan

Manager – Human Resources

Shia#sh Info Solutions Private Limited  
#51, Level 4, Tower A, Rattha TEK Meadows, Old Mahabalipuram Road,  
Sholinganallur, Chennai – 600 119, Tamil Nadu, India  
+91 44 66255681 info@shia#sh.com



24<sup>th</sup> September 2020

**PROJECT COMPLETION CERTIFICATE**

This is to certify that Ms. P.Nityasree (Reg. No. 17MIS1007) a student of M.Tech (INTEGRATED SOFTWARE ENGINEERING), Vellore Institute of Technology, Chennai has successfully completed the project titled "PLEDGE STORAGE RECKON" in ANDROID Platform from June 2020 to July 2020 in our company. During the period, she had been exposed to different processes and found to be Punctual, Hard Working and Inquisitive.

We wish her every success in life and career.

For Shia#sh Info Solutions Private Limited

**Ashwini Kanniyappan**  
**Manager – Human Resources**

Shia#sh Info Solutions Private Limited  
#51, Level 4, Tower A, Rattha TEK Meadows, Old Mahabalipuram  
Road, Sholinganallur, Chennai – 600 119, Tamil Nadu, India  
+91 44 66255681 info@shia#sh.com

## **ACKNOWLEDGEMENT**

Throughout the writing of this dissertation I have received a great deal of support and assistance. I would first like to thank all the people who guided me through the work and whose expertise was invaluable in the Completion of the project.

I would like to acknowledge my colleagues from my internship at Shiash Info solutions private limited for their wonderful collaboration. You supported me greatly and were always willing to help me. I would particularly like to thank all the members for your excellent cooperation and for all of the opportunities I was given to complete the project.

I would also like to thank my tutors, faculties of the university, (Head of the Department for 5 years MTech Integrated Software Engineering Vit University Chennai Campus) Dr. Asnath Vicky Phamila Y , Dr. Jagadeesh Kannan R ( Dean of SCOPE of Vit University Chennai Campus), Dr. Geetha S (Associate Dean of SCOPE of Vit University Chennai Campus) for their valuable guidance .You provided me with the tools that I needed to choose the right direction and successfully complete my dissertation.

In addition, I would like to thank my parents for their wise counsel and sympathetic ear. You are always there for me. Finally, there are my friends, who were of great support in deliberating over our problems and findings, as well as providing happy distraction to rest my mind outside of my research.

# CONTENTS

Chapter	Title	Page
	Title Page	i
	Declaration	ii
	Certificate	iii
	Industry certificate	iv,v
	Acknowledgement	vi
	Table of contents	vii
	List of Tables	viii
	List of Figures	ix
	List of Abbreviation	x
	Abstract	xi
1	Introduction	01
2	Methodology	11
3	System Development Environment	14
4	Conclusion	15
	References	16
	Appendix – I	17

## **LIST OF TABLES**

<b>Title</b>	<b>Page</b>
Table 1 : Cloud Owner Register	4
Table 2 : Cloud File_Upload	4
Table 3 : Cloud_Owner Assignkey	4
Table 4 : Taxo About Product	5
Table 5 : Taxo File Request	5
Table 6 : Cloud _UserRegister	5



## LIST OF FIGURES

<b>Title</b>	<b>Page</b>
Fig 1 : Data Flow Diagram	06
Fig 2 : Use Case Diagram	07
Fig 3 : Class Diagram	08
Fig 4 : Sequence Diagram	09
Fig 5 : Activity Diagram	10

## LIST OF ABBREVIATIONS

Abbreviation	Expansion
DACC	distributed access control scheme in clouds
KDC	key distribution centers
M-HABE	modified hierarchical attribute-based encryption
DAC	data access control
CP-ABE	Ciphertext-Policy Attribute-Based Encryption
SRS	Software requirements specification
RDBMS	Relational Database Management System

## **ABSTRACT**

In fast forward world of technology, everyone is running behind the mobile application and they need all the information on the mobile phone itself. This application is related to providing to Mobile Cloud Computing is internet-based computing pattern through which Shared resource is provided to devices on-demand its an emerging but promising paradigm to integrating mobile devices into cloud computing and the integration performs in the cloud-based hierarchical multiuser data shared environment with integrating into the cloud computing security issues such as data confidentiality and user authority may arise in the mobile cloud computing in order to provide safe security hierarchical access control encryption and modified in to the three-layer. In specific mobile control encryption and modified into three layout In a specific mobile computing model, enormous, data which may be from all kinds of mobile devices such as smartphones. mainly implemented secure computing purpose.

Cloud computing is an internet-based computing pattern through which shared resources are provided to devise on-demand Its an emerging but promising paradigm to integrating mobile devices into cloud computing, and the integration. performs in the cloud-based a number of users sharing the data. with integrating into the cloud computing, security issues such as data confidentiality and user authority may arise in the mobile cloud computing system and it's concerning as the main constrains to the developments of mobile cloud computing this process personal information shared with the Unauthorized person.

The novel schemes mainly focus data processing storing, accessing, which designed to ensure the user with legal authority to get corresponding classified data and restrict illegal users unauthorized legal users and unauthorized legal user get access to the data

## 1. INTRODUCTION:

Data Access Control: A plurality of data access control systems based on the promising CP-ABE technique are proposed to construct the efficient, secure, fine grained and revocable access schemes. S.Ruj et al. (2011) proposed a distributed access control scheme in clouds (DACC) that supported attribute revocation.

In DACC, one or more key distribution centers (KDCs) distributed keys to data owners and users. Technically, it requires not only forward security but more indispensable backward security in context of the attribute revocation. However, DACC supported attribute revocation with vulnerable forward security. J.Hur et al. (2011) proposed an attribute-based DAC scheme with efficient revocation in cloud storage systems, whereas it was designed only for the cloud systems with single trusted authority. In addition, the above two schemes both require data owners to reencrypt the out- sourced ciphertext after revocation.

However, the data access control (DAC) issue of cloud computing systems has been escalated by the surge in attacks such as collusion, wiretapping and distort, so that DAC must be designed with sufficient resistance. DAC issues are mainly related to the security policies provided to the users accessing the uploaded data, and the techniques of DAC must specify their own defined security access policies and the further support of policy updates, based on which each valid user can have access to some particular sets of data whereas invalid users are unauthorized to access the data .

A plurality of data access control systems based on the promising CP-ABE technique is proposed to construct efficient, secure, fine-grained and revocable access schemes. S.Ruj et al. (2011) proposed a distributed access control scheme in clouds (DACC) that supported attribute revocation. In DACC, one or more key distribution centers (KDCs) distributed keys to data owners and users.

## **PURPOSE OF THE PROJECT :**

The main scope of the project is to maintain the data with more secured and reduce data loss with a high credential level. Many companies have suffered data loss from their own company due to less security. And it has to maintain the data with the security level and it has to overcome with the storage level of different types of data. That will make some important data's to be hidden for their company purpose. Because the data has to make more high credentials and most security levels for the data. In this application, users can view the companies details for their reference with complete landmarks.

The user has to search the company details on websites and the detailed information for their clearance. Instead of that, we can see the company details through in one application. It will make more useful to see the company profile in this application and the product development of each company. They can share the resource file with the most secure and efficient level data that can be transferred. And it will more helpful for developing the product with different companies. Then it will improve the stability of the user and it if check the stability for the data redundancy. It will get improved with better user interface and it will be added with the advanced technology level for the user .

## **EXISTING SYSTEM:**

Senders encrypt messages with certain attributes of the authorized receivers. The ABE based access control method uses several tags to mark the attributes that a specific authorized user needs to possess. The users with certain tag sets can get access to the specific encrypted data and decrypt it. Lots of paper introduced the scheme about the attribute-based encryption access control method in cloud computing.

In the mobile cloud computing environment, there are tremendous data which needs to be processed and marked with attributions for the convenient attributing access before storing. At the same time, the hierarchical structure of the application users needs an authentication center entity to control their attributes.

**DisAdvantages:**

- User with legal access authorities to get corresponding sensing data
- It is not secure personal data
- Developing a False Sense of Security.
- It will make requiring for all the cooperation.

**PROPOSED SYSTEM:**

In the proposed scenario, users with different privilege levels have different rights to access the part of sensing data coming from mobile devices. Therefore, one same data has to be encrypted into ciphertext once, which ought to be able to be decrypted multiple times by different authorized users. In this paper, a hierarchical access control method using a modified hierarchical attribute-based encryption (M-HABE) and a modified three-layer structure is proposed. Differing from the existing paradigms such as the HABE algorithm and the original three-layer structure, the novel scheme mainly focuses on the data processing, storing and accessing, which is designed to ensure the application users with legal access authorities to get corresponding sensing data and to restrict illegal users and unauthorized legal users get access to the data, the proposed promising paradigm makes it extremely suitable for the mobile cloud computing based paradigm. What should be emphasized is that the most important highlight of all in the proposed paper can be described as that the modified three-layer structure is designed for solving the security issues illustrated above.

**Advantages :**

- One ciphertext can be decrypted by several keys.
- Both precise level descriptions and user attributes should be supported in the access structure of the method.
- The keys in the authentication center ought to have the same hierarchical structure just as the structure of the user's privilege levels.

TABLE1:

Server: localhost:3306 » Database: id9722108\_roja » Table: CloudownerRegister

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>Name</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>Mobile</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>Email</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>Password</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>DateOfBirth</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>Address</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 8	<b>MasterKey</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Fulltext

TABLE2:

Server: localhost:3306 » Database: id9722108\_roja » Table: Cloud\_FileUpload

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>FileName</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>AssignKey</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>OwnerName</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>OwnerMail</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>File</b>	varchar(500)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>FileTag</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Fulltext

TABLE 3:

Server: localhost:3306 » Database: id9722108\_roja » Table: Cloud\_OwnerAssignkey

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>OwnerName</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>OwnerMailId</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>UserName</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>UserMailId</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>UserMobileNo</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>SendKey</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 8	<b>Status</b>	varchar(50)	utf8_unicode_ci		Yes	NULL			Change  Drop  More

TABLE4:

Server: localhost:3306 » Database: id9722108\_roja » Table: TaxoAboutProduct

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>ProductName</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>Description</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>Purpose</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>MailId</b>	varchar(40)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>RandomKey</b>	varchar(40)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>Deadline</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 8	<b>ProductStatus</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 9	<b>Document</b>	varchar(1000)	utf8_unicode_ci		No	None			Change  Drop  More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Fulltext

TABLE 5:

Server: localhost:3306 » Database: id9722108\_roja » Table: TaxofileRequest

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>Name</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>CompanyName</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>EmailId</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>FileName</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>Date</b>	varchar(50)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>Randomkey</b>	varchar(40)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 8	<b>Status</b>	varchar(25)	utf8_unicode_ci		Yes	0			Change  Drop  More

☐ Check all
 With selected:
 [Browse](#)
 Change
 Drop
 Primary
 Unique
 Index
 Fulltext

TABLE 6:

Server: localhost:3306 » Database: id9722108\_roja » Table: Cloud\_UserRegister

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Triggers](#)

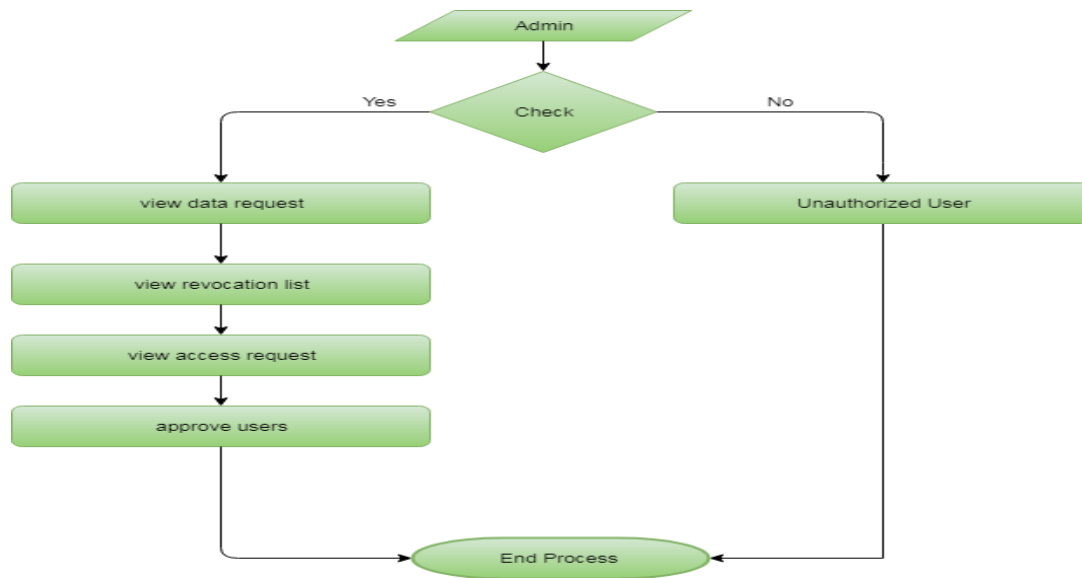
[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	<b>Id</b>	int(10)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>Name</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	<b>MobileNo</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	<b>EmailId</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	<b>Password</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	<b>Address</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/> 7	<b>DateofBirth</b>	varchar(100)	utf8_unicode_ci		No	None			Change  Drop  More

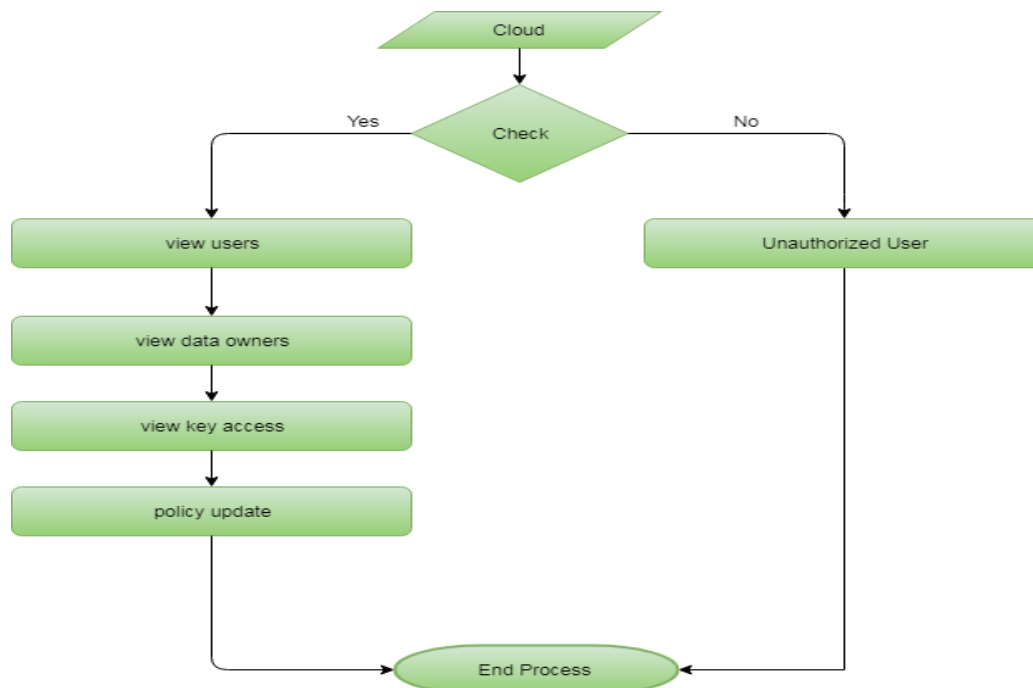


**FIG 1: Data Flow Diagram**

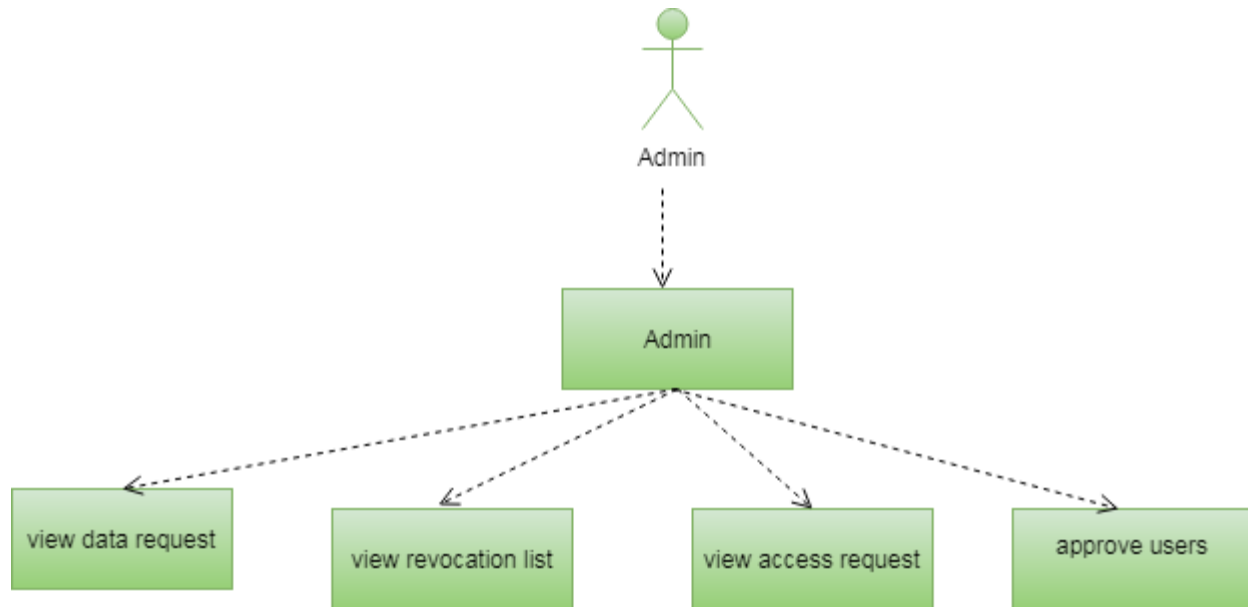
## Admin



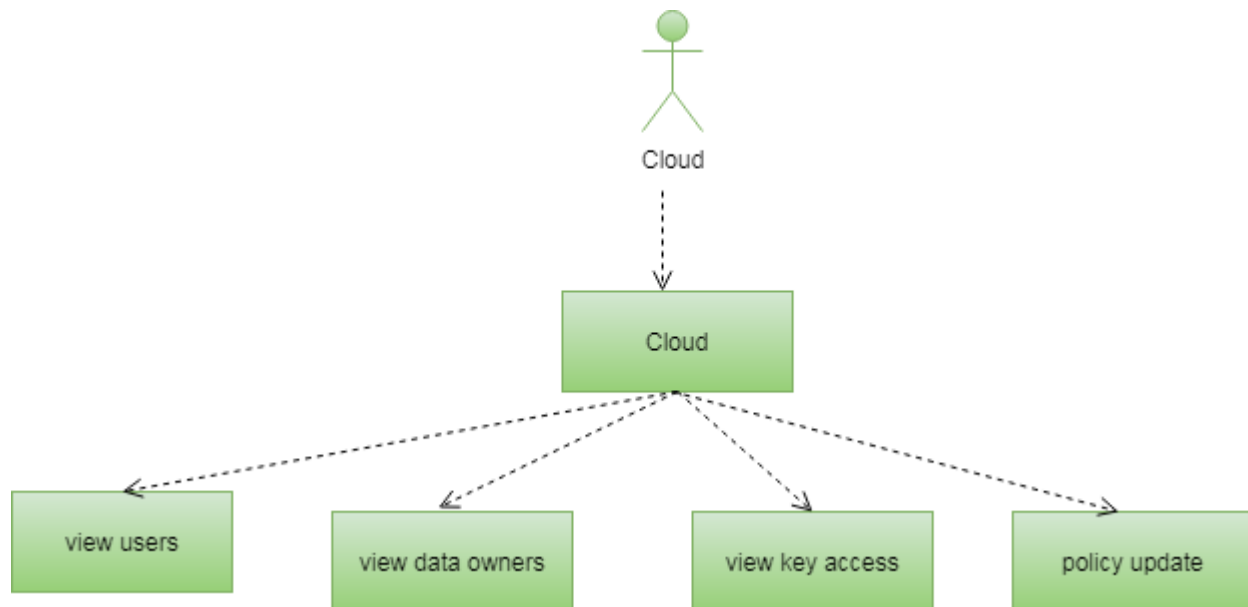
## Cloud



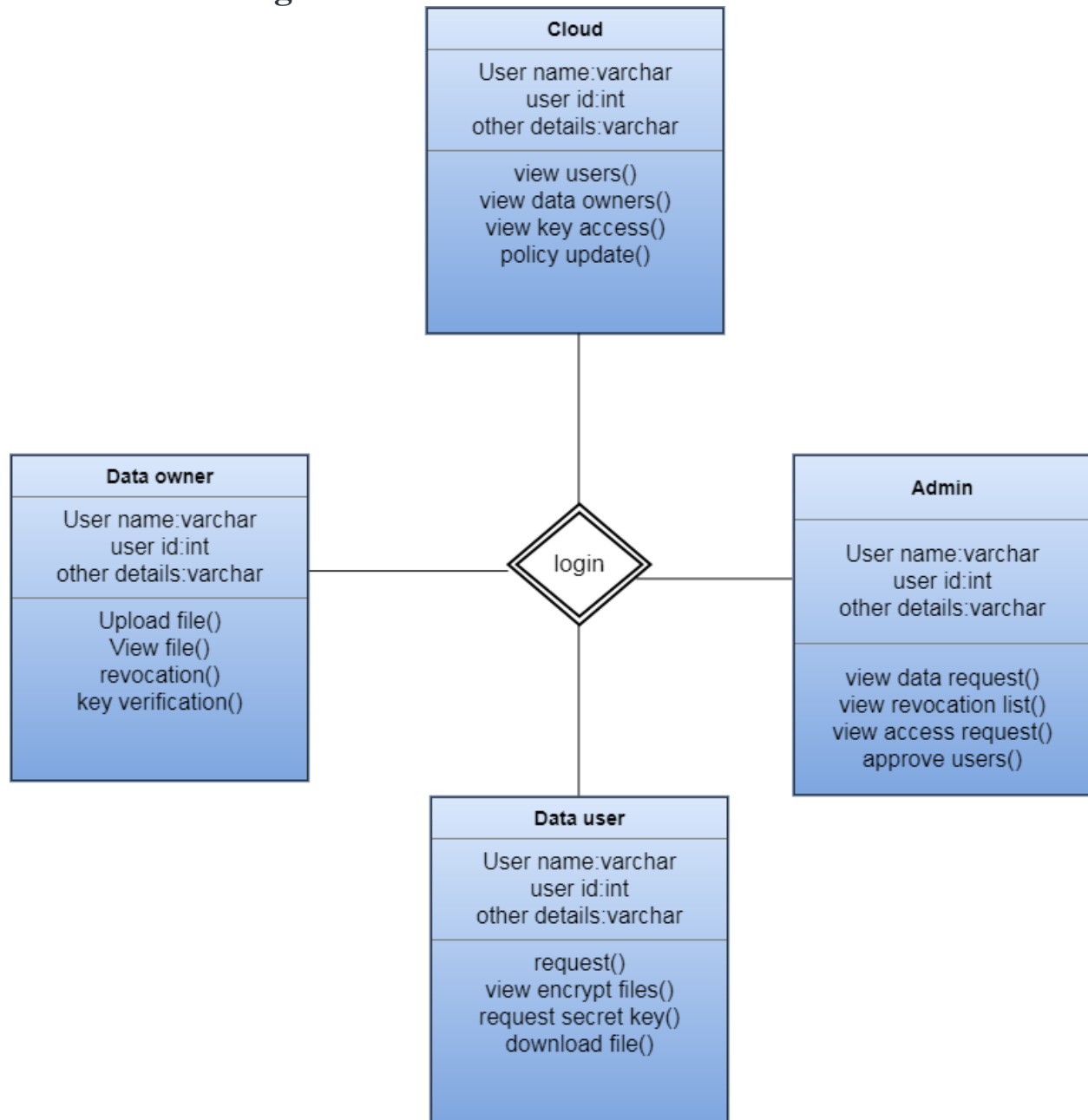
**FIG 2: Usecase Diagram**  
**Admin**



**Cloud**



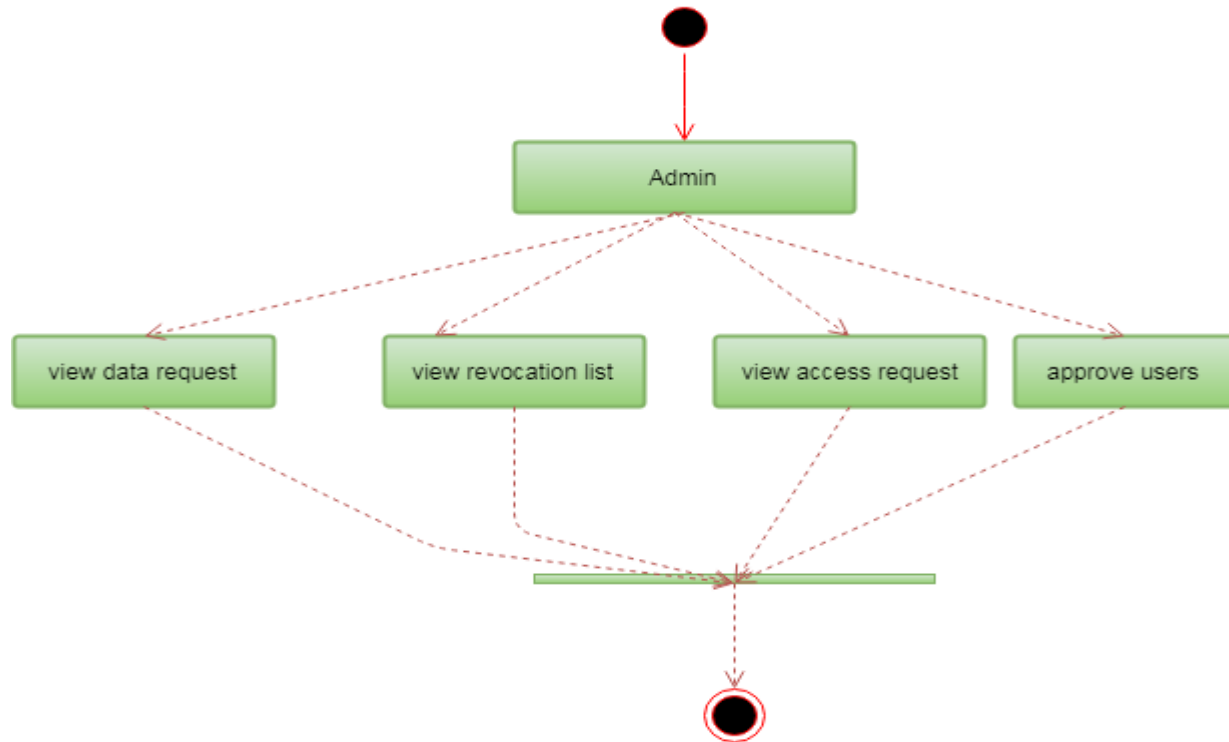
**FIG 3: Class Diagram**



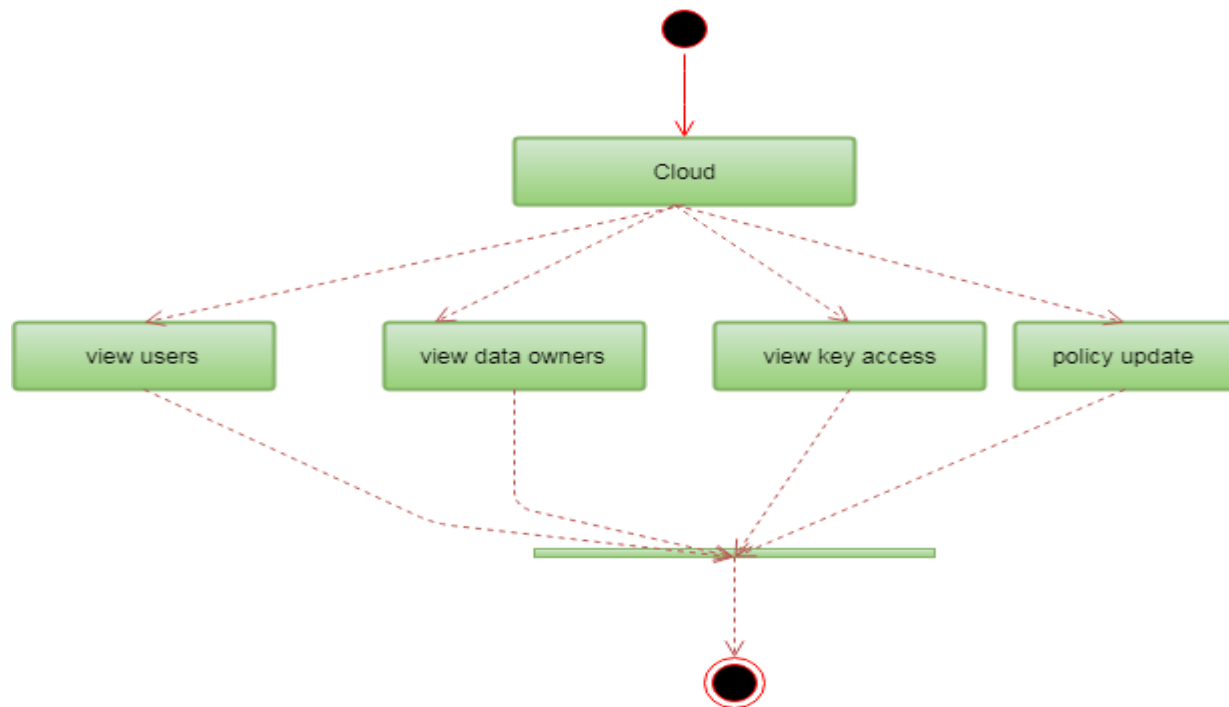
## Admin



**FIG 5: Activity Diagram**  
**Admin**



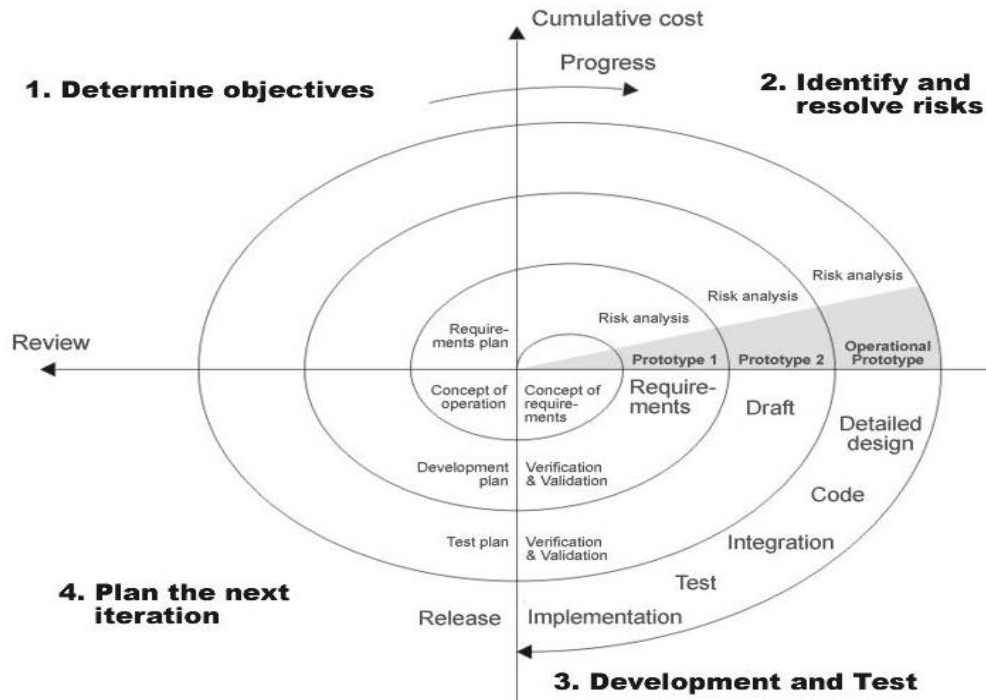
**Cloud**



## 2. METHODOLOGY

### 2.1 SYSTEM ANALYSIS :

SPIRAL MODEL:



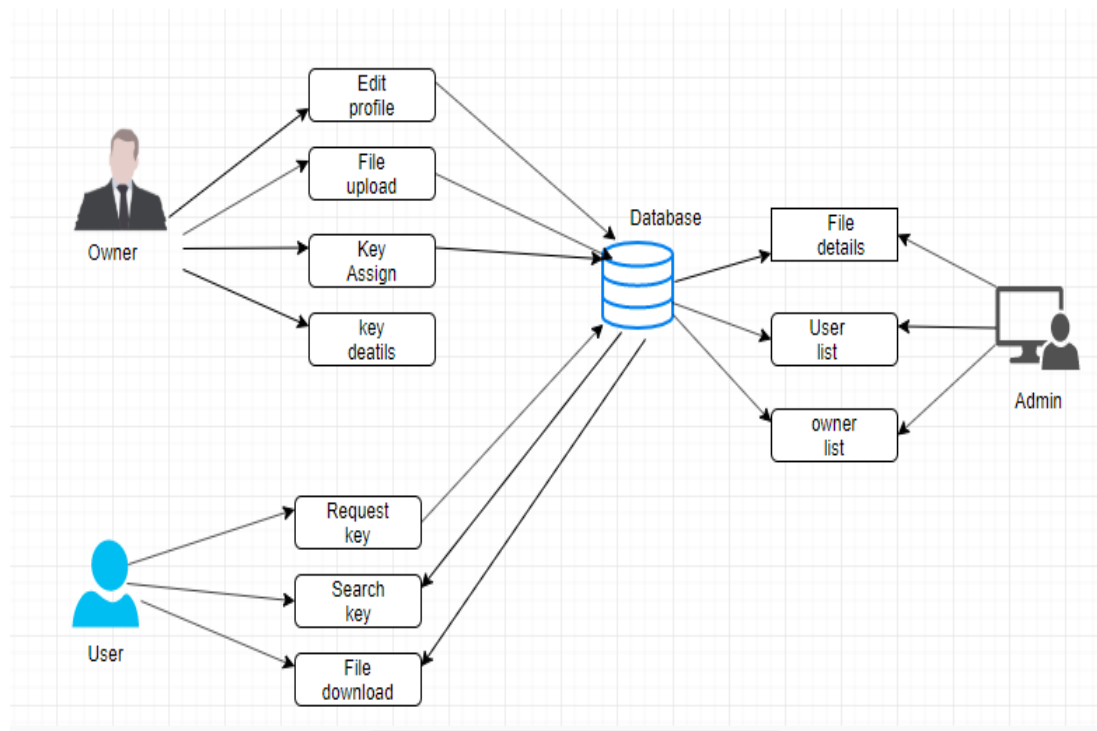
#### Software Requirements:

- FRONT END: Java
- BACK END: MySQL
- SERVER: PHP
- VERSION: JDK8
- PLATFORM: WINDOWS 10
- IDE: ANDROID STUDIO 3.5.2
- SQL TOOL: PHPMyAdmin

## MODULES:

- Authentication
- Authorization
- Checking Internet Connection
- Admin
- Register
- File Upload
- Secret Key
- User Request key

## 2.2 SYSTEM ARCHITECTURE:



## **2.3 SOFTWARE REQUIREMENT SPECIFICATION [SRS]:**

### **PURPOSE:**

The purpose of this Software Requirement Specification (SRS) is to help the project. It is provided with some requirements which are used in the Transaction Mercator System. The purpose of this document is to detail the requirements placed on the Transaction Mercator System and serves as a contract between the customer and the developers as to what is to be expected of the stock exchange, and how the components of the system are working with each other with external systems.

### **FUNCTIONAL REQUIREMENTS:**

Following is a list of functionalities of the browsing enabled system.

- An Activity with a UI that allows you to browser settings. Provide a second Activity that allows users to access the share with permission from the administrator. Handle the activity lifecycle appropriately. A precondition for any points in this part of the grade is code that compiles and runs.

### **NON-FUNCTIONAL REQUIREMENTS:**

- Each member should have a separate system. The system should ask the username and password to open the application. Secure access to confidential data (employee's details). 24 X 7 availability. Better component design to get better performance at peak time

### **PERFORMANCE REQUIREMENTS**

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the required specifications are properly given, it is possible to design a system, which will fit into the required environment. It rests largely on the part of the users of the existing system to give the required specifications because they are the people who finally use the system



## **3. SYSTEM DEVELOPMENT ENVIRONMENT**

### **3.1 JAVA:**

Java applets run in a Web browser with Java Virtual Machine (JVM), which translates Java byte code into native processor instructions and allows indirect OS or platform program execution. JVM provides the majority of components needed to run byte code, which is usually smaller than executable programs written through other programming languages. Byte code cannot run if a system lacks required JVM.

### **3.2 ANDROID:**

Android is an open-source, Linux-based software stack created for a wide array of devices and form factors. Android applications are composed of one or more application components (activities, services, content providers, and broadcast receivers). Each component performs a different role in the overall application behaviour, and each one can be activated individually (even by other applications)

### **3.3 PHP:**

PHP is open-source. It executed at the server-side. It is also called server-side scripting language. It is used to develop web applications (an application i.e. executed at the server-side and generates dynamic page).

### **3.4 MYSQL:**

**MySQL** is a freely available open-source Relational Database Management System (RDBMS) that uses Structured Query Language (**SQL**). MySQL is used for developing various web-based software applications

## **4. CONCLUSION & FUTURE ENHANCEMENT**

The paper proposed a modified HABE scheme by taking advantage of attributes based encryption (ABE) and hierarchical identity-based encryption (HIBE) access control processing. The proposed access control method using MHABE is designed to be utilized within a hierarchical multiuser data-shared environment, which is extremely suitable for a mobile cloud computing model to protect the data privacy and defend unauthorized access.

Compared with the original HABE scheme, the novel scheme can be more adaptive for mobile cloud computing environments to process, store and access the enormous data and files while the novel system can let different privilege entities access their permitted data and files.

The scheme not only accomplishes the hierarchical access control of mobile sensing data in the mobile cloud computing model but protects the data from being obtained by an untrusted third party.

With digital video transmission, encryption methodologies are needed that can protect digital video from attacks during transmission. So the developed algorithms can be applied on video images and web-based systems.

## REFERENCES

- 1) Kalmani, V. H., Goyal, D., & Singla, S. (2015). An efficient and secure solution for attribute revocation problem utilizing CP-ABE scheme in mobile cloud computing. *Int. J. Comput. Appl*, 129, 16-21.
- 2) Lewko, A., & Waters, B. (2010, February). New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In *Theory of Cryptography Conference* (pp. 455-479). Springer, Berlin, Heidelberg.
- 3) Nejad, M. S., Malakooti, M. V., & Taba, N. H. (2019, August). An Efficient Cloud-based Network Data Access Using CP-ABE Encryption Method. In *The 8th International Conference on Informatics and Applications (ICIA2019)* (p. 27).
- 4) Li, W. M., Li, X. L., Wen, Q. Y., Zhang, S., & Zhang, H. (2017). Flexible CP-ABE based access control on encrypted data for mobile users in hybrid cloud system. *Journal of Computer Science and Technology*, 32(5), 974-990.

## APPENDIX I

### SAMPLE CODE OF REGISTRATION

#### Registration:

```
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.design.widget.TextInputLayout;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import java.util.HashMap;
import java.util.Map;

public class UserRegister extends AppCompatActivity {
    TextView slogin;
    TextInputLayout
sinput_layout_name,sinput_layout_username,sinput_layout_email,sinput_layout_pass,sinp
ut_layout_repass,sinput_layout_location,sinput_layout_sdob;
    EditText sname,sphone,spass,semail,srepass,saddress,sdob;
    Button ssignup;
    ProgressDialog progressDialog;
    private int progresstatus=0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user_register);
        sname=findViewById(R.id.sname1);
        sphone=findViewById(R.id.sphone1);
        semail=findViewById(R.id.semail1);
        spass=findViewById(R.id.spass1);
        srepass=findViewById(R.id.srepass1);
        saddress=findViewById(R.id.slocation1);
        sdob=findViewById(R.id.sdob1);
        sinput_layout_email=findViewById(R.id.sinput_layout_email1);
        sinput_layout_name=findViewById(R.id.sinput_layout_name1);
        sinput_layout_username=findViewById(R.id.sinput_layout_phone1);
        sinput_layout_pass=findViewById(R.id.sinput_layout_pass1);
```

```

sininput_layout_repass=findViewById(R.id.sininput_layout_repass1);
sininput_layout_location=findViewById(R.id.sininput_layout_location1);
sininput_layout_sdob=findViewById(R.id.sininput_layout_dob1);
slogin=findViewById(R.id.slogin1);
slogin.setOnClickListener(

    new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent in=new
Intent(getApplicationContext(),OwnerSideLogin.class);
            startActivity(in);
        }
    });
progressDialog = new ProgressDialog(UserRegister.this);
ssignup=findViewById(R.id.ssign1);
final String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";

ssignup.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        String email = semail.getText().toString().
            trim();

        final String password=spass.getText().toString();
        final String repassword=srepass.getText().toString();

        if (sname.getText().toString().trim().isEmpty()) {
            sininput_layout_name.setError(getString(R.string.err_msg_name));
        }
        else if (sphone.getText().toString().trim().isEmpty()) {
            sininput_layout_name.setErrorEnabled(false);
        }
sininput_layout_username.setError(getString(R.string.err_msg_phoneno));
        }
        else if (!email.matches(emailPattern))
        {
            sininput_layout_username.setErrorEnabled(false);
            sininput_layout_email.setError(getString(R.string.err_msg_email));
        }
        else if (sdob.getText().toString().trim().isEmpty()) {
            sininput_layout_sdob.setError(getString(R.string.err_msg_dob));
        }
        else if (saddress.getText().toString().trim().isEmpty()) {
sininput_layout_location.setError(getString(R.string.err_msg_address));
        }

        else if (spass.getText().toString().trim().isEmpty()) {
            sininput_layout_email.setErrorEnabled(false);

```

```

sinput_layout_pass.setError(getString(R.string.err_msg_password));
    }
    else if (srepass.getText().toString().trim().isEmpty()) {
        sinput_layout_pass.setErrorEnabled(false);

sinput_layout_repass.setError(getString(R.string.err_msg_repass));
    }

    else if (password.equals(repassword)){
        processRequest();

    }
    else {
        sinput_layout_repass.setError("Password Doesn't Match");
    }

}

});

}

private void processRequest() {

    progressDialog.setMessage("Please Wait, We are Inserting Your Data on
Server");
    progressDialog.show();

    StringRequest sr = new
StringRequest(Request.Method.POST, "https://jashminr.000webhostapp.com/CloudSecurity/U
serRegister.php", new Response.Listener<String>() {

        @Override
        public void onResponse(String response) {
            progressDialog.dismiss();
            if(response.equalsIgnoreCase("inserted")) {

                Toast.makeText(getApplicationContext(), "success",
Toast.LENGTH_LONG).show();
                SharedPreferences pref = getSharedPreferences("LoginPrefs",
Context.MODE_PRIVATE);
                SharedPreferences.Editor edt = pref.edit();
                edt.putString("Name", sname.getText().toString());
                edt.putString("Email", semail.getText().toString());
                edt.commit();
                finish();
            }
        }
    });
}

```

```

        Intent in = new Intent(getApplicationContext(),
UserSideHome.class);
        startActivity(in);
    }
    else {

        // Showing Echo Response Message Coming From Server.
        Toast.makeText(UserRegister.this, response,
Toast.LENGTH_LONG).show();

    }
}
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        progressDialog.dismiss();

        Toast.makeText(UserRegister.this, "No Internet Connection"+error,
Toast.LENGTH_LONG).show();
    }
}) {
    @Override
    protected Map<String, String> getParams() throws AuthFailureError {
        Map<String, String> params = new HashMap<String, String>();

        // Adding All values to Params.

        String name = sname.getText().toString().trim();
        String pass = spass.getText().toString().trim();
        String email=semail.getText().toString().trim();
        String location=saddress.getText().toString().trim();
        String dob=sdob.getText().toString().trim();
        String phone = sphone.getText().toString().trim();
        params.put("Name", name);
        params.put("Mobile", phone);
        params.put("Address",location);
        params.put("DateOfBirth",dob);
        params.put("Mail",email);
        params.put("Password", pass);

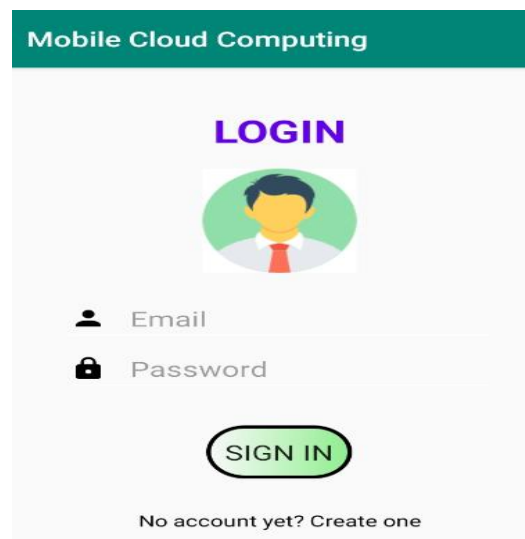
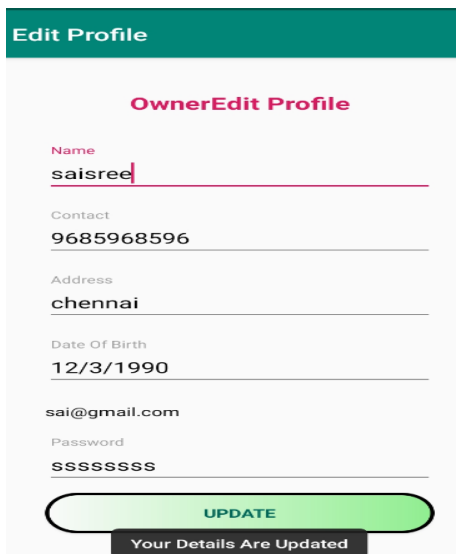
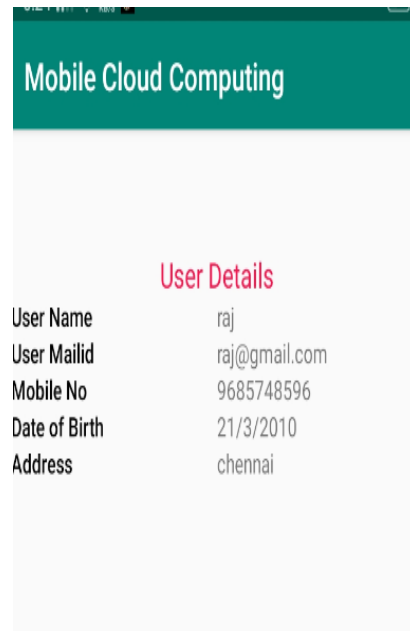
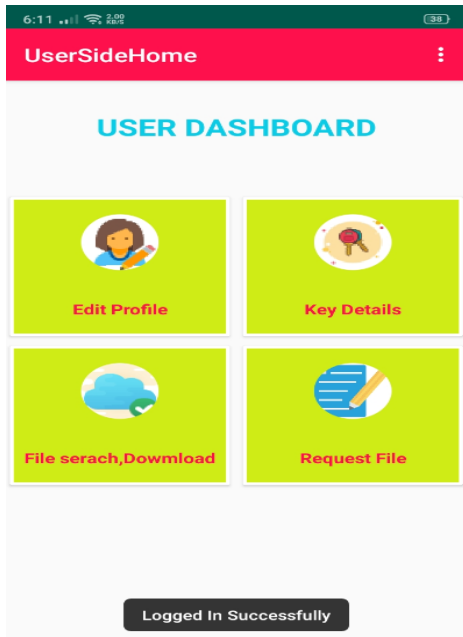
        return params;
    }
}

};

RequestQueue requestQueue= Volley.newRequestQueue(UserRegister.this);
requestQueue.add(sr);
}}

```

## OUTPUT SCREENSHOTS:





6:21 97% 37

## Upload


**Upload File**

Owner Name sai

Email Id sai@gmail.com


File Name android

File Tag text

 /storage/emulated/0/Android/data/cn.wps.moffice\_eng/.cache/KingsoftOffice/file/download/86d20cf-d6e0-41f9-b090-904c61976292/com.google.android.gm.sapi/RESUME.doc

AssignKey 123

**UPLOAD**

 Go back

6:09 2.00 38

## Mobile Cloud Computing

search for categories

gas

dfsdf

hxjxbnx

hxjxbnx


Test


TestNew


123456


## Mobile Cloud Computing


### OWNER DASHBOARD


 **Edit Profile**

 **Key Details**

 **File Upload**

 **File Details**

 **successStatus Changed Successfully !!**



**END**