**1.INTRODUCTION**

Accurate demand forecasting combined with resource planning is critical to a company’s performance and profitability. This paper describes ARMS (automated resource management system), an integrated system developed for the customer service operations to help with the operational/tactical planning and deployment of the company’s 20,000-strong field engineer workforce. OR techniques are used throughout the system, including ARIMA for forecasting, constraint satisfaction for problem modelling, heuristic search for problem solving thus demonstrating the value and relevance of OR in solving today’s business problems.

The need for automating resource management, herein referred to as RM, is well recognised and has been the subject of considerable research and development. The case for automating RM is motivated by the drive to maximise profits, improve quality of service (QoS) and reduce costs.

Successful automation of RM requires that the aforementioned tenets be fully automated. Indeed operations research and artificial intelligence methods  have been employed to automate some aspects of RM. For example the autoregressive integrated moving-average i.e. ARIMA model has been successfully employed in forecasting jobs. Constraint satisfaction [15and heuristic search methods  have been applied to resource allocation problems. In this paper we describe work we have done in developing an automated resource management system called ARMS, to automate the planning and deployment of field engineers within the customer service division.

* Engineers are multi-skilled and they can perform several tasks requiring different competencies and capabilities. There are currently nine different ‘‘skills’’ defined for the purposes of RM with each engineer having one or more of these skills depending on past training and/or experience. The resource manager can focus specific engineers to work on activities of a particular skill or type.

• Engineers have the flexibility to move around between relatively small geographical areas called ‘‘patches’’. A customer service team (CST) usually is responsible for several patches and the resource manager has the responsibility to resource each patch adequately by moving engineers from neighbouring patches or by altering their working pattern to be explained next.

• Engineers have different working patterns. They can work full day or half a day. They may also be available to come and work on overtime on voluntary basis. Furthermore, they have several business related absences or meetings scheduled which, if necessary, could be cancelled to increase availability in certain geographical areas or on specific dates.

**1.1 OBJECTIVE**

ARMS integrates a forecasting tool with a resource planning tool and a resource balancing tool providing an end-to-end automated resource management solution for the organisation.

**Overview**

Systems, methods, and devices to provide simple adaptive automated resource management of a resource system (such as but not limited to electricity, natural gas, water, data, bandwidth allocation, access to information, etc.) on a local basis, based on automatically detecting, measuring and combining time-varying resource provider preferences, resource market conditions, resource supply source conditions, environmental conditions and resource system impact on the environment, together with resource user locations, user priorities and preferences, and information about other conditions that may be relevant to the operation of the resource management system in order to optimize performance of the resource system to better meet or approach defined goals, and to measure and display the results achieved by the resource management system compared against those goals.

A system consisting of a human operator or group of operators and a machine, by means of which the operator performs a task involving, for example, the production of material goods, the management of some type of operation, or the processing of information. Human labor in a man-machine system is based on interaction according to received information with both the object of labor or control and the machine through the mediation of control elements.

Interest in man-machine systems arose in the mid-20th century, when systems of various kinds became with increasing frequency the objects of technical planning and design. The effectiveness of these systems, which included those for the control of production, transportation, communications, and space flights, was largely determined by the activity of the human operators. The combination of human abilities and capabilities of a machine or complex of technological devices significantly increases the effectiveness of control. Although there is a joint performance of control functions by the human operator and machine, each of the two components of the system is governed in its work by its own unique rules. The effectiveness of the system as a whole is determined by the extent to which characteristic features of the operator and machine, both limitations and potentials, are identified and taken into account when building the system. These features are most fully identified in the process of coordinating the external, that is, technological, means of action and the internal means of action, that is, means inherent to the operator. Coordination includes the construction of information and conceptual models.

Libraries can use Electronic Resource Management as a standalone solution

or as an integrated part of Sierra: Innovative’s trusted integrated library

management platform. As a standalone system, libraries can enjoy the

benefits of ERM’s ability to maintain resources, track licenses, and manage

coverage data. Whether fully integrated or standalone, ERM is tailor-made to take advantage of a “Quick Start” implementation program that includes over 400 ready-to-use resource records.

**2.SYSTEM STUDY**

**2.1EXISTING SYSTEM**

The manual process of managing the resources and its software and hardware

Disadvantages

**2.2 LIMITATIONS OF EXISTING SYSTEM**

* More manual process
* No robustness
* Cannot utilizes the dependency resources properly.
* More human required in maintaining
* Leads to unknown operations
* Less in productivity
* More cost
  1. **PROPOSED SYSTEM**

ARMS integrates a forecasting tool with a resource planning tool and a resource balancing tool providing an end-to-end automated resource management solution for the organisation.

Advantages

* Overcomes all the disadvantages
* Speed in processing
* Resource can be dynamically utilized
* Less human required
* Gives loose coupling of the resources
* Distributed system

**3. HARDWARE & SOFTWARE REQUIREMENTS**

**3.1 HARDWARE CONFIGURATION**

* Hardware: Dual Core
* Hard Disk: 50 GB
* Speed: 1.4 GHz
* RAM: 1GB
* Key Board: Standard Keyboard
* Touch Pad: Button Mouse
* Monitor: LED

**3.2 SOFTWARE CONFIGURATION**

* Operating System : Windows
* IDE : Net Beans 7.3.1
* Technology : Java and J2EE
* Web Server : Tomcat
* Web Technologies : Html, JavaScript, CSS
* Java Version : JDK1.7
* Database : My SQL

**4.SOFTWARE REQUIREMENT SPECIFICATIONS**

**4.1 REQUIREMENT ANALYSIS**

A software requirements definition is an abstract descriptioservices which the system should provide and the constraints under which the system must operate.

· System requirements may be either functional or non-functional requirements

**4.2 FUNCTIONAL REQUIREMENTS**

In software engineering, a functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behavior, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing.

· Administrator login

· Uploading data

· User registration

· User login

· Searching

· Ranking results based on likes and comments

**4.3 NON-FUNCTIONAL REQUIREMENTS**

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates.

Three key considerations involved in the feasibility analysis are

* **Economic Feasibility**: This study is carried out to check the economic impact that the system will have on the organization
* **Technical Feasibility**: This study is carried out to check the technical feasibility, that is, the technical requirements of the system.
* **Social Feasibility**: The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently.

**4.4 APPROACH ADAPTED**

**Agile Model**

The meaning of Agile is swift or versatile. "**Agile process model**" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.



**Fig 4.4:** Agile Model

## **Phases of Agile Model:**

Following are the phases in the agile model are as follows:

1. Requirements gathering
2. Design the requirements
3. Construction/ iteration
4. Testing/ Quality assurance
5. Deployment
6. Feedback
7. **Requirements gathering:** In this phase, you must define the requirements. You should explain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.
8. **Design the requirements:** When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high-level UML diagram to show the work of new features and show how it will apply to your existing system.
9. **Construction/ iteration:** When the team defines the requirements, the work begins. Designers and developers start working on their project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionality.
10. **Testing:** In this phase, the Quality Assurance team examines the product's performance and looks for the bug.
11. **Deployment:** In this phase, the team issues a product for the user's work environment.
12. **Feedback:** After releasing the product, the last step is feedback. In this, the team receives feedback about the product and works through the feedback.

**4.5 INTRODUCTION TO MYSQL**

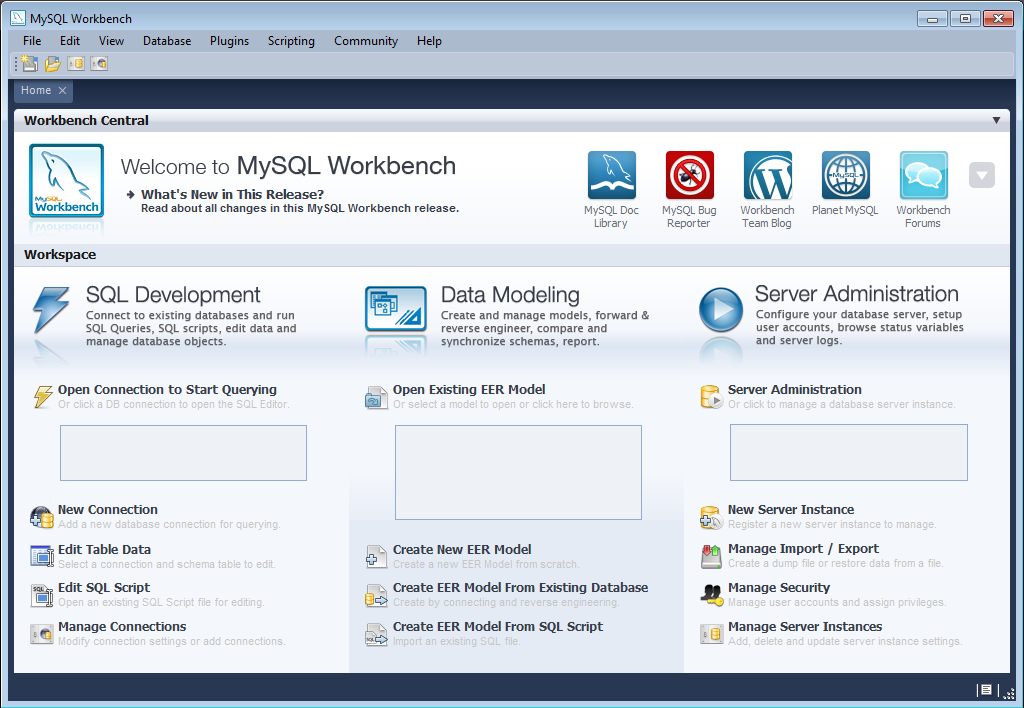
MySQL is an open source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by **Oracle Company**.

MySQL is [Relational Database Management System](https://www.javatpoint.com/what-is-rdbms) (RDBMS) software that

provides many things, which are as follows:

* It allows us to implement database operations on tables, rows, columns, and indexes.
* It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
* It provides the Referential Integrity between rows or columns of various tables.
* It allows us to updates the table indexes automatically.
* It uses many SQL queries and combines useful information from multiple tables for the end-users.



**4.6 TECHNOLOGY AND TOOL DESCRIPTION**

**NetBeans IDE Frame Work for JAVA**

In computer programming, a software framework is an abstraction in which software providing generic functionality can be selectively changed by additional user – written code, thus providing application specific software. A software framework is a universal, reusable software platform used to develop applications, products and solutions.

A software framework includes support programs, compilers, code libraries, tool sets and application programming interface that bring toget her all different components to enable development of a project or solutions.



NetBeans is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP,C/C++, and HTML5. It is also an application platform framework for Java desktop applications and others.

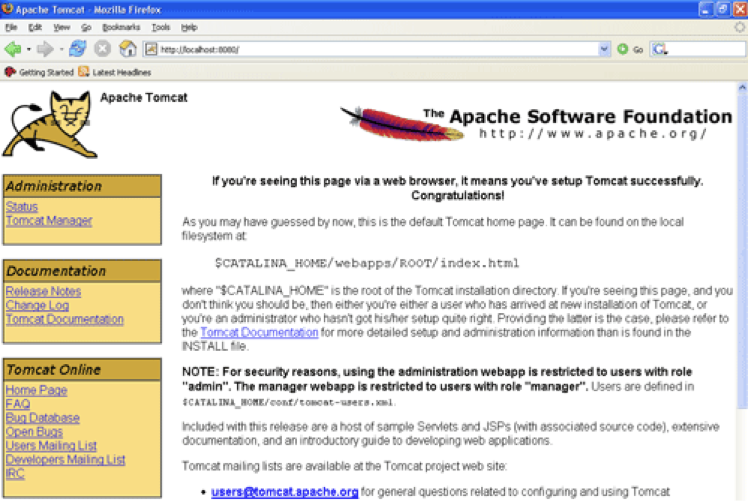
NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM. The NetBeans Platform allows applications to be developed from a set of modular software components called modules.

Applications based on the NetBeans Platform (including the NetBeans IDE itself) can be extended by third party developers. NetBeans IDE is an open-source integrated development environment. NetBeans IDE supports development of all Java application types (Java SE (including JavaFX), Java ME, web, EJB and mobile applications) out of the box. Among other features are an Ant based project system, Maven support, refactoring’s, version control.

All the functions of the IDE are provided by modules. Each module provides a well-defined function, such as support for the Java language, editing, or support for the CVS versioning system, and SVN. NetBeans contains all the modules needed for Java development in a single download, allowing the user to start working immediately.

**Tomcat 6.0 web server**

Tomcat is an open-source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs WebLogic, is one of the popular application server).To develop a web application with jsp/servlet install any web server like J Run, Tomcat etc. to run your application.

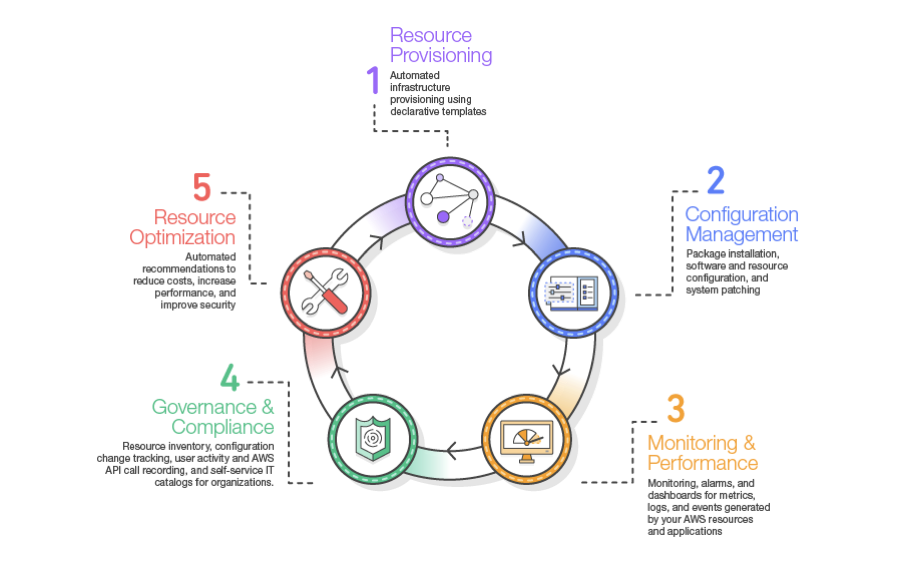


**Java**

Java is an object-oriented programming language with its runtime environment. It is a combination of features of C and C++ with some essential additional concepts. Java is well suited for both standalone and web application development and is designed to provide solutions to most of the problems faced by users of the internet era.

1. **SYSTEM DESIGN**

Systemsdesign istheprocessofdefiningthe [architecture](https://en.wikipedia.org/wiki/Systems_architecture),modules, interfaces and [data](https://en.wikipedia.org/wiki/Data) fora [system](https://en.wikipedia.org/wiki/System) tosatisfyspecified [requirements](https://en.wikipedia.org/wiki/Requirement). Systems design could be seen as the application of [systems theory](https://en.wikipedia.org/wiki/Systems_theory) to [product development](https://en.wikipedia.org/wiki/Product_development). There is some overlap with the disciplines of [systems analysis](https://en.wikipedia.org/wiki/Systems_analysis), [systems architecture](https://en.wikipedia.org/wiki/Systems_architecture) and [systems engineering](https://en.wikipedia.org/wiki/Systems_engineering)



**5.1 DATA FLOW DAIGRAM**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](http://en.wikipedia.org/wiki/Information_system). DFDs can also be used for the [visualization](http://en.wikipedia.org/wiki/Data_visualization) of [data processing](http://en.wikipedia.org/wiki/Data_processing) (structured design).

On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. 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](http://en.wikipedia.org/wiki/Flowchart), which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

**Symbols used in DFD’s:**

**Processes:**

A process transforms data values. The lowest processes are our functions without side effects.

**Data Flows:**

A data flow connects the output of an object or process to the input of another object or process. It represents the intermediate data values within the computation. It is draws as an arrow between the procedure and the consumer of the data value. The arrow is labelled with the description of the data, usually its name or type.

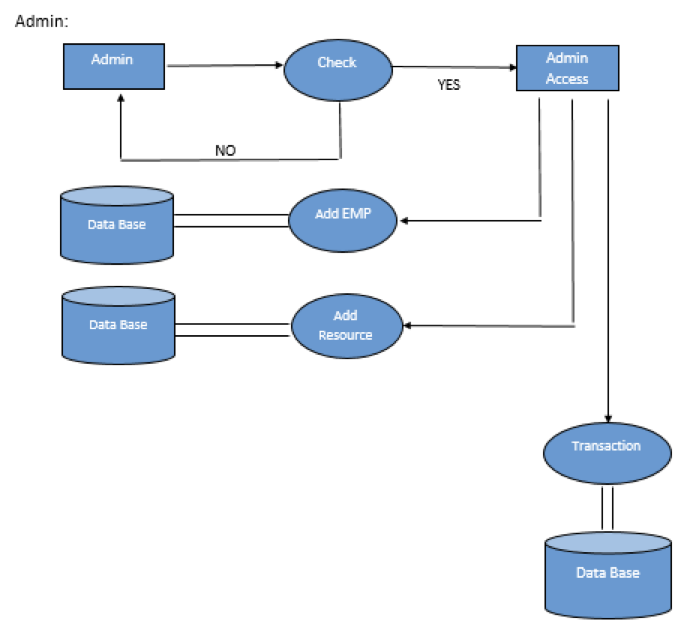
**Actors:**

An actor is an active object that drives the data flow graph by producing or consuming values. Actors are attached to the inputs and the outputs of a dataflow graph. In sense, the actors lie on the boundary of the flow graph but terminate the flow of data as sources and sinks of data, and so are sometimes called terminators.

**Data Store:**

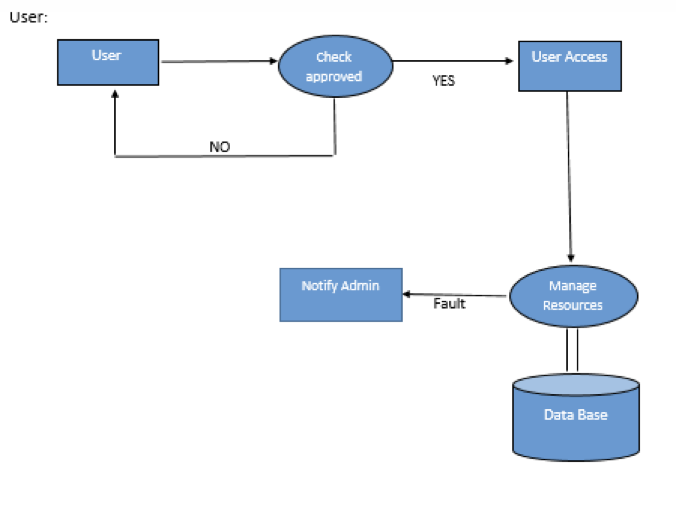
A data store is a passive object within a data flow diagram that stores data for later access. Unlike an actor, a data store does not generate any operations on its own but merely responds to requests to store and access data.

**Admin:**



**Fig 5.1.1: DFD of ADMIN**

User :



**Fig 5.1.2: DFD of user**

**5.2 USE CASE DIAGRAM**

Use case diagrams are considered for high level requirement analysis of a system. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to gather its functionalities use cases are prepared and actors are identified. Now when the initial task is complete use case diagrams are modelled to present the outside view.

**Use case:**

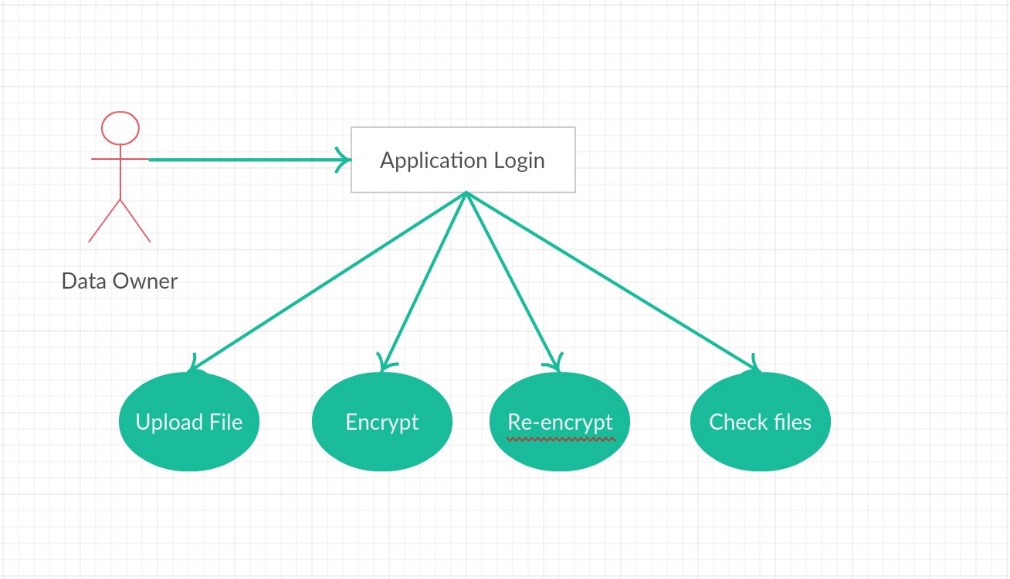
Use case diagrams are considered for high level requirement analysis of a system. So, when the requirements of a system are analysed the functionalities are captured in use cases. So, we can say that uses cases are nothing but the system functionalities written in an organized manner.

**Actor:**

Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system. The actors can be human user, some internal applications or may be some external applications.

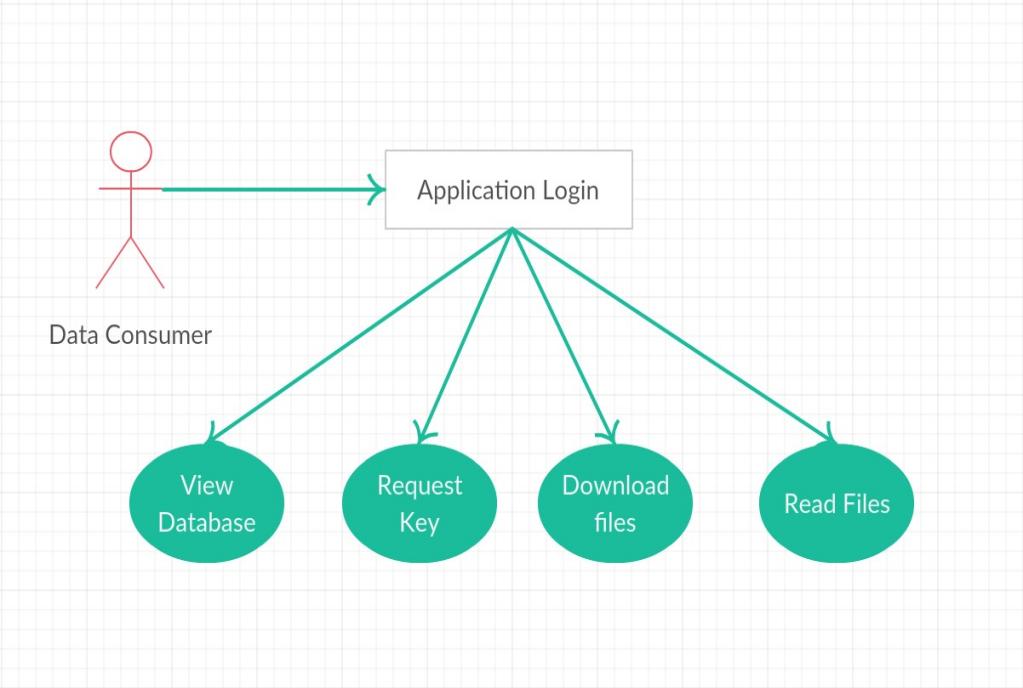
**Relationship:**

Relationships exist among the use cases and actors. Show relationships and dependencies clearly in the diagram. Do not try to include all types of relationships. Because the main purpose of the diagram is to identify requirements

**Use case diagram- Data Owner**

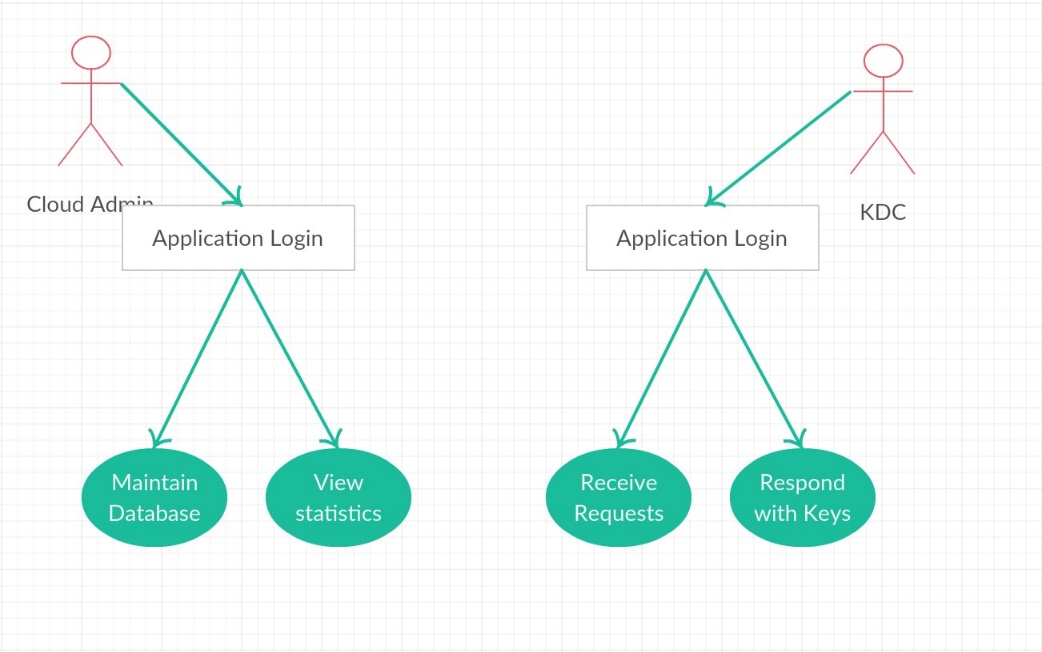
**Fig No 5.2.1: Use case diagram –Data Owner**

**Use case diagram- Data Consumer**



**Fig No 5.2.2: Use case diagram –Data Consumer**

**Use case diagram- Cloud Admin and KDC**



**Fig No 5.2.3: Use case diagram –Cloud Admin and KDC**

**5.3 SEQUENCE DIAGRAM**

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It describes interactions among classes in terms of an exchange of messages over time. Sequence diagrams are used to show how objects interact in a given situation. An important characteristic of a sequence diagram is that time passes from top to bottom: the interaction starts near the top of the diagram and ends at the bottom

**Targets/Class roles/State:**

Objects as well as classes can be targets on a sequence diagram, which means that messages can be sent to them. A target is displayed as a rectangle with some text in it. Below the target, its lifeline extends for as long as the target exists. Targets can be actor, boundary, control, entity and database.

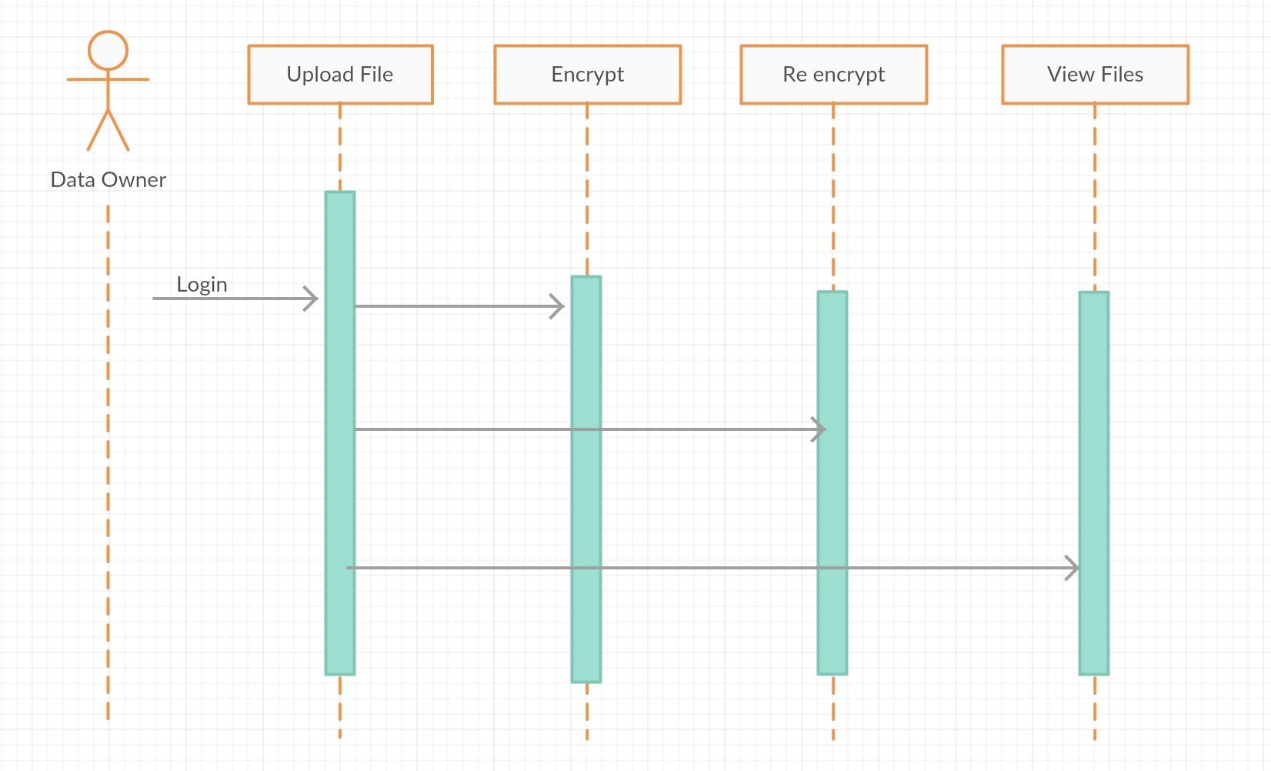
**Messages:**

Messages are arrows that represent communication between objects.

**Lifelines:**

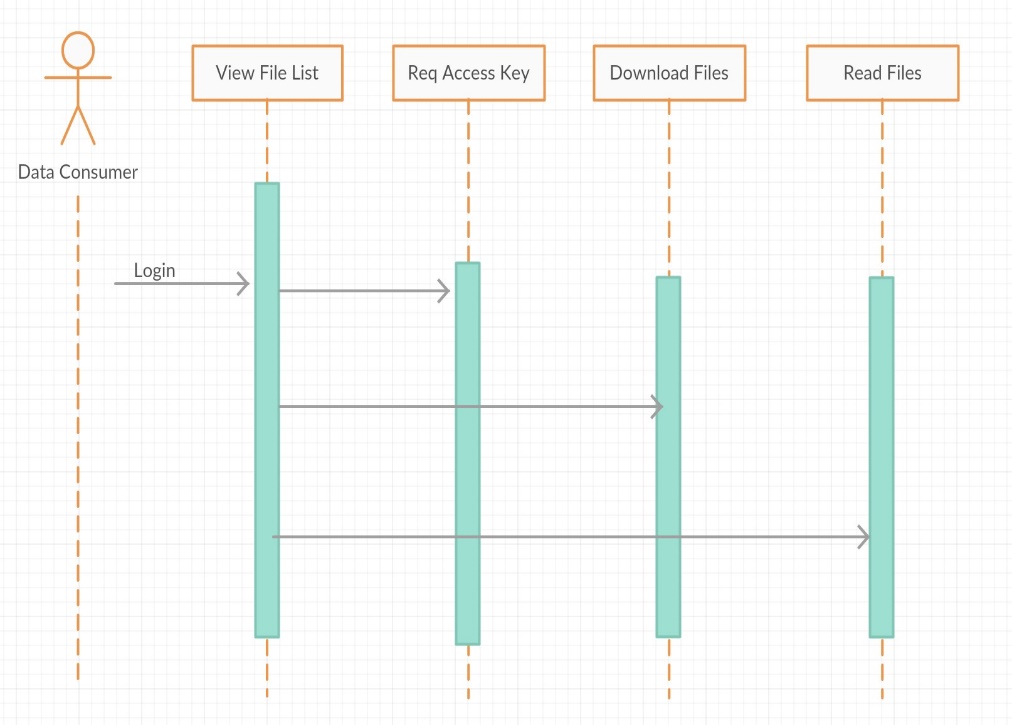
Lifelines are vertical dashed lines that indicate the object's presence over time.

**Sequence diagram for- Data Owner**



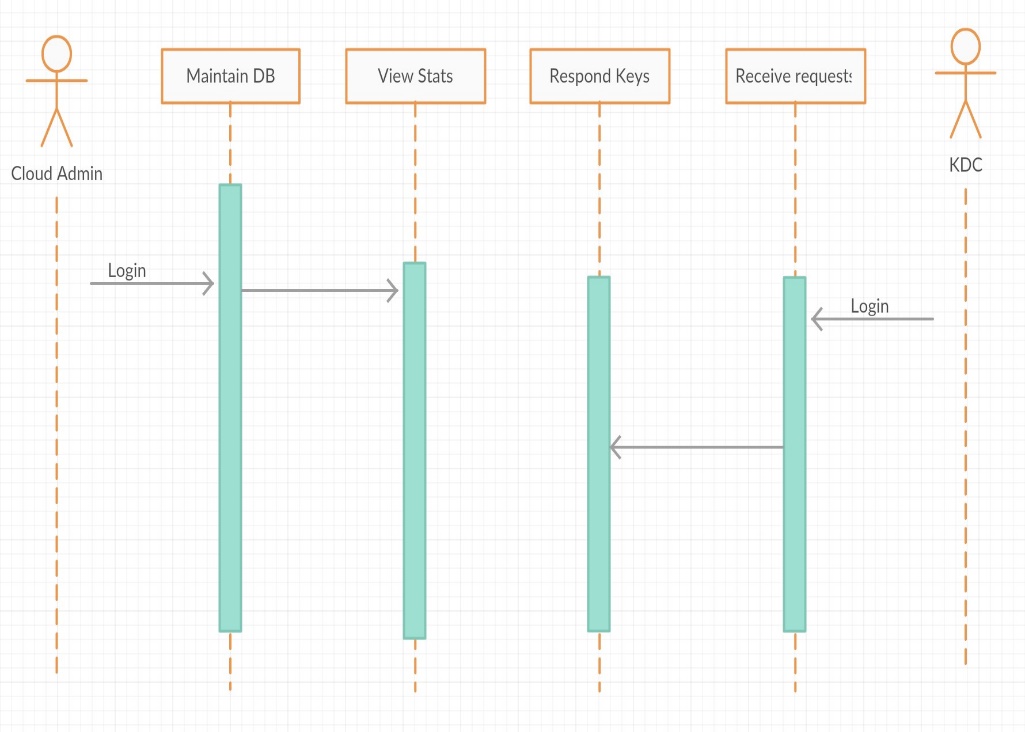
**Fig 5.3.1: Sequence diagram-Data Owner**

**Sequence diagram for- Data Consumer**



**Fig 5.3.2: Sequence diagram-Data Consumer**

**Sequence diagram for- Cloud Admin and KDC**



**Fig 5.3.2: Sequence diagram-Data Consumer**

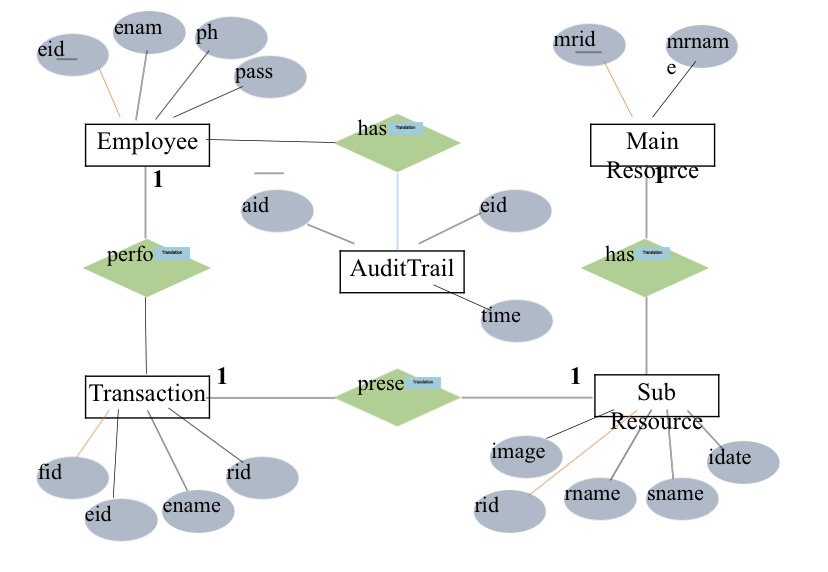
**Fig 5.3.3: Sequence diagram- Cloud Admin and KDC**

* 1. **RELATIONSHIP DIAGRAM**

Entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a [relational database](http://en.wikipedia.org/wiki/Relational_database). The main components of ER models are [entities](http://en.wikipedia.org/wiki/Entities) (things) and the relationships that can exist among them, and databases. ER Diagram is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data-an object or concept about which data is stored. A relationship is how the data is shared between entities. There are three types of relationships between entities:

* **One-to-One:** One instance of an entity is associated with one other instance of another entity.
* **One-to-Many:** One instance of an entity is associated with zero, one or many instances of another entity, but for one instance of entity B there is only one instance of entity A.
* **Many-to-Many**: One instance of an entity (A) is associated with one, zero or many instances of another entity (B), and one instance of entity B is associated with one, zero or many instances of entity A.

**ER-Diagram :**

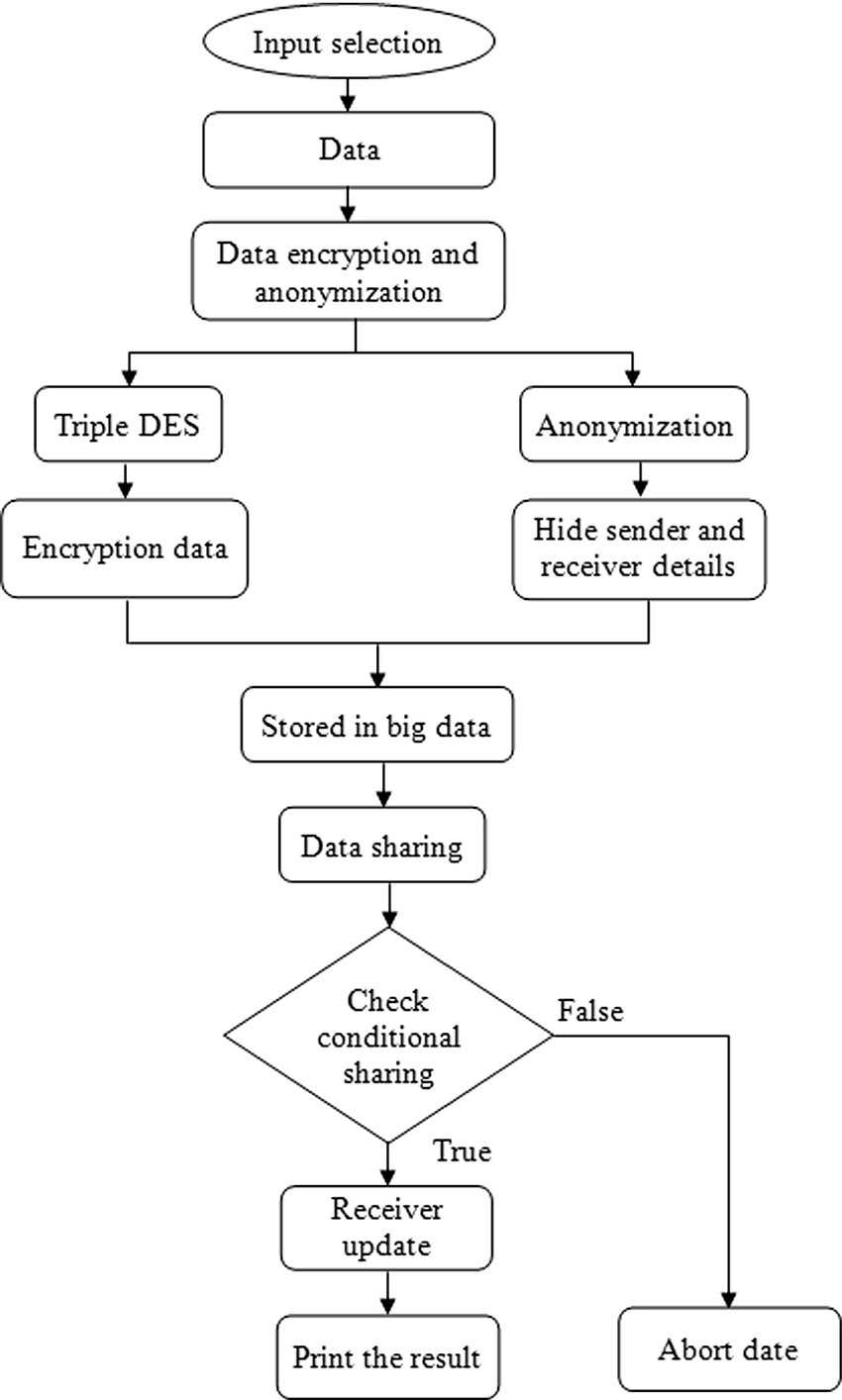
****

**Fig 5.4: E-R diagram**

**5.5 ACTIVITY DAIGRAM**

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc



**Fig 5.5: Activity diagram**

1. **IMPLEMENTATION**
   1. **MODULES**

* Forecasting: The ability to forecast the demand of work.
* Analysis: The ability to analyse resource and job profiles and identify either over or under resource utilisation. A resource profile refers to a particular collection of location (i.e. area), skill and availability (i.e. time) data. A job profile on the other hand refers to attributes of the job such as priority, start and end dates, job type and so on. Analysis comprises resource planning and scheduling. Resource planning involves profiling where resource profiles are adapted with a view to matching configuration of skills, availability and locations of the resources to the skills, timing and locations of the jobs making up a particular workload. Resource scheduling is concerned with assigning resources to actual jobs and identifying explicit execution times for those jobs. Resource planning is an essential pre-cursor to successful resource scheduling.
* Execution: The ability to execute the output of the analysis—in terms of dispatching jobs to resources, making requests for extra resources in case of over resource utilisation. Successful automation of RM requires that the aforementioned tenets be fully automated. Indeed operations research and artificial intelligence methods  have been employed to automate some aspects of RM. For example the autoregressive integrated moving-average i.e. ARIMA model has been successfully employed in forecasting job . Constraint satisfaction and heuristic search methods have been applied to resource allocation problems. In this paper we describe work we have done in developing an automated resource management system called ARMS, to automate the planning and deployment of field engineers within the customer service division of British Telecommunications plc (BT).

• Forecasting demand for several activity types (e.g. provision, repair, maintenance work).

• Planning the volumes, skills and geographical locations of engineers required in order to service that demand in the next current  up to 6 months.

• Deploying field engineers for tomorrow in the best possible manner so that incoming and existing work can be optimally scheduled by the companys automated workforce scheduling system

**User**

* Can see the 6 months update or upgrade notification.
* Can add the main resource
* Can add the sub resource
* Manage the resource to re utilizing

**Admin**

* Add the employee detail
* Add the main resource
* Add the sub resource
* View the transaction

1. **SOURCE CODE**

<%@ page  import="java.sql.\*,databaseconnection.\*"  %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

    <head>

        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

        <title>JSP Page</title>

    </head>

    <body>

       <%

String rid=request.getParameter("rid");

String rname=request.getParameter("rname");

//out.print(username1);

String udate=request.getParameter("udate");

String status=request.getParameter("status");

int i=1;

 if(i==0)

    {

Connection con1=null;

Statement st=null;

ResultSet rs=null;

int id=0;

try

{

    PreparedStatement ps1=null;

con1=databasecon.getconnection();

st=con1.createStatement();

rs=st.executeQuery("call simpleproc(?,?)");

ps1.setString(1,rname);

while(rs.next())

{

rs.getString("mid");

session.setAttribute("id",id);

}

}

 catch(Exception e)

        {

            out.println(e.getMessage());

        }

    }

Connection conn=null;

PreparedStatement ps=null;

try

{

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

conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/arms","root", "123");

//conn=databasecon.getconnection();

ps=conn.prepareStatement ("UPDATE subresource SET `rname` = ?, `udate` = ?, `status` = ?  where `rid` = ?");

ps.setString(1,rname);

ps.setString(2,udate);

ps.setString(3,status);

ps.setString(4,rid);

int x=ps.executeUpdate();

            if(x==0)

                {

                response.sendRedirect("amanageresource.jsp?message=fail");}

            else

                {

                response.sendRedirect("amanageresource.jsp?message=success");

                }

        }

        catch(Exception e)

        {

            out.println(e.getMessage());

        }

%>

    </body>

</html>

**8. TESTING**

The inspiration driving is to find defaults in the errand. The testing is the explanation behind attempting to discover every lack or deficiency in a working things. It gives the best way to check the suitability of sections, sub-assemblies and a completed thing. It is the method of working with the arrangement of confirming that the item framework meets its chucks and the customer covets and does not miss the mark in an unacceptable way. There are characteristic sorts of test. Every test category discourses a precise testing need.

**8.1 Types of Testing**

1. Unit
2. Integration
3. Functional
4. System
5. White-Box
6. Black-Box
7. Acceptance

**8.1.1 Unit Testing of System**

Testing fuses the sketch of examinations that endorse that the inner venture reason is working truly, and that program inputs extensive yields. All inside code and branches ought to be insisted. It is utilization to test the individual programming unit of an application. Unit testing is done before breaker. It is the central testing that depends heaps of its change and is intruding.

Unit testing guarantees that each exceptional strategy for a occupational system perform correctly to the recorded purposes of interest and covers obviously depicted inputs and output result.

This performs indispensable tests part by part level and test a specific occupational framework, application, structure course of action. Unit testing is for the most part decided as mutual code and unit test of the thing lifecycle.

* **Test plan and method**
  + It will be performed physically and practical and will be composed in point of interest.
* **Objectives**
* Field accesses essentially work properly.
* Folios must be triggered from the recognized link.
* The entrance display, responses must not be late.
* **Features**
* Link’s should take to the correct page.
* Confirm that the accesses are of the format.
* No same entries should be allowed.

**TEST CASES**

|  |  |
| --- | --- |
| Name of the test |  |
| Test Description |  |
| Sample Input |  |
| Expected Output |  |
| Actual result/Remarks |  |
| Passed (?) |  |

|  |  |
| --- | --- |
| Name of the test |  |
| Test Description |  |
| Sample Input |  |
| Expected Output |  |
| Actual result/Remarks |  |
| Passed (?) |  |

|  |  |
| --- | --- |
| Name of the test |  |
| Test Description |  |
| Sample Input |  |
| Expected Output |  |
| Actual result/Remarks |  |
| Passed (?) |  |

**8.1.2 Integration Testing of System**

Integration testing is needed to test urged programming sections to grasp whether they truly continue running as one structure. Exchange off particularly away to uncover the matters that climb up the mix of pieces. Event driven and it more worried with the critical aftereffect of screens or fields. This tests show that paying little personality that the pieces were solely satisfaction, as displayed by tastefully unit testing.

The mix of sections is correct and solid. Programming testing is the testing of two or more made programming parts on a single stage to go on disillusionments brought on by crossing point misshapenings. Attempt of testing is to watch that parts or programming application e.g. pieces in a thing framework or programming application at the association level to interface without lurch.

Mix testing is deliberate to test joined programming portions to make sagacity of whether they continue running as one framework. Combination testing is especially away to uncover the issues that rise up out of the mix of sections.

Testing is more stressed with the basic consequence of screens fields. Combination tests display that in spite of the way that the sections were solely gratification, as showed up by viably unit testing. Programming blend testing is the coordination testing of two or more joined programming sections on a lone stage to make frustrations realized by interface disfigurements.

The task that compromise test is to watch that parts or programming application, e.g. fragments in an item framework or programming application at the association level to interface without bungle.

**Test Results:** This test cases is passed successfully. No defects met.

|  |  |
| --- | --- |
| Name of the test |  |
| Test Description |  |
| Sample Input |  |
| Expected Output |  |
| Actual result/Remarks |  |
| Passed (?) |  |

Table 8.10 Integration Of all Unit Testing

**8.1.3 Functional Testing** **of system**

Functional tests gives exact establishes that limits make an effort showed by the occupational and specific necessities, framework certification, and customer manual. Affiliation and availability of functional test is revolved around necessities, key limits, or uncommon trials. Besides, effective allowance identifying with perceive occupational process stream, datafields, predefined structures, and techniques must be measured for testing. Additional perceived and the convincing estimation of current tests is determined.

**8.1.4 System Testing**

This testing ensures that the entire fused programming structure meets necessities. It tests a setup to ensure known obvious results. An instance of structure testing is the setup arranged structure blend test. System testing relies on upon technique focusing on pre-driven strategy associations and blend centers.

**8.1.5 White-Box Testing of System**

White box testing is a annoying in which the invention analyzer has data of internal mechanisms, structure and product. It is additionally used to test parts that can't be come to from a discovery level.

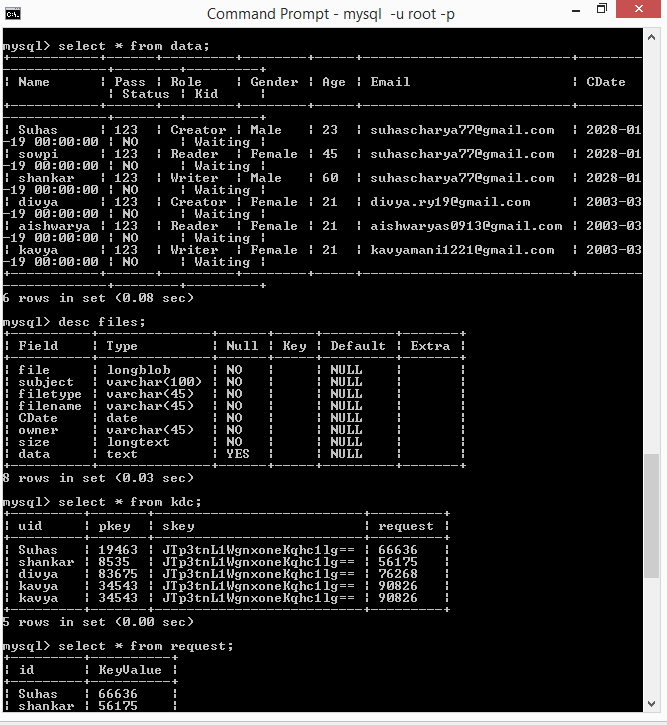
**8.1.6 Black-Box Testing of System**

Black-Box testing is the attempting the item with unknown data of the internal working, structure or lingo of the module attempted. Disclosure test, most diverse sorts of tests, must be made from an indisputable basis record.

**8.1.7 Acceptance Testing**

This testing is a risky period of interest by the end client. It additionally assurances that the framework experiences the utilitarian prerequisites.

Test Results: All the trials said above passes effectively. No deformities experienced.

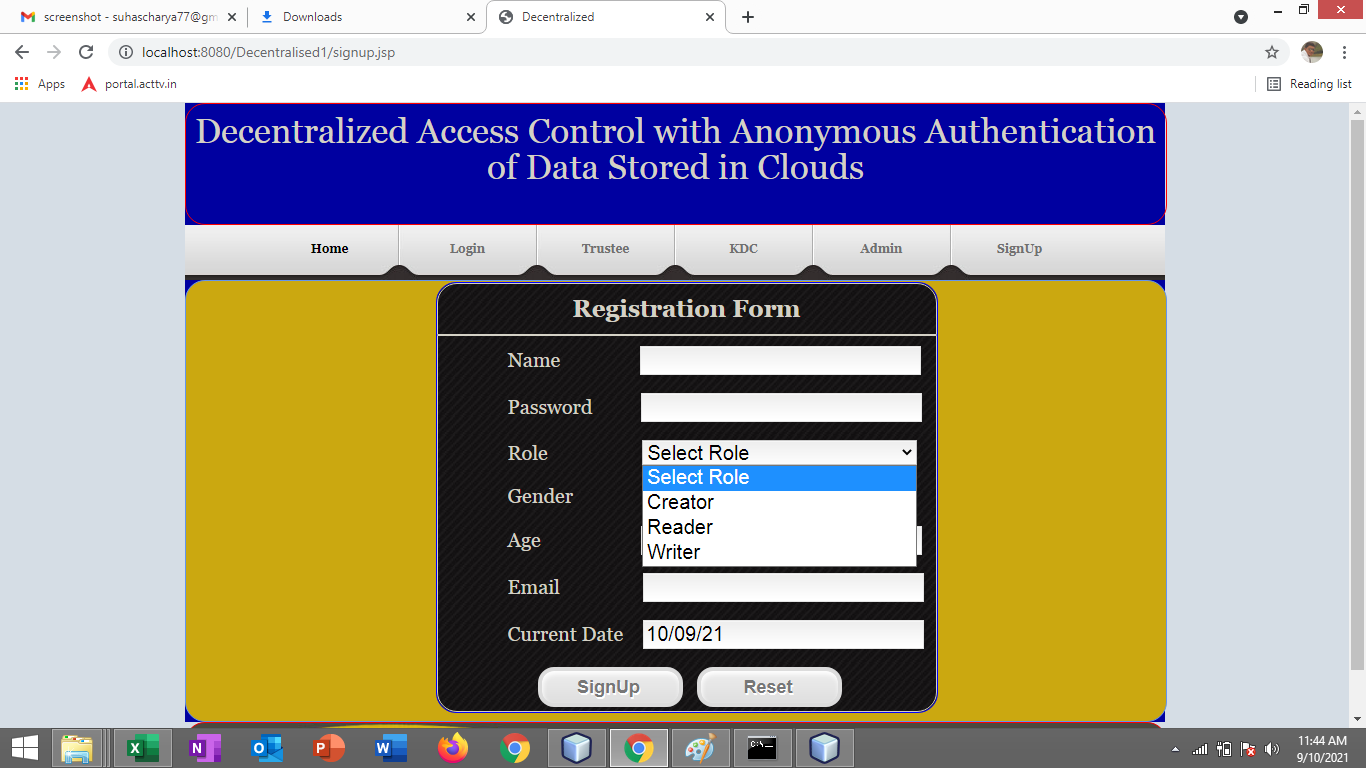


**Fig No 9.1:**

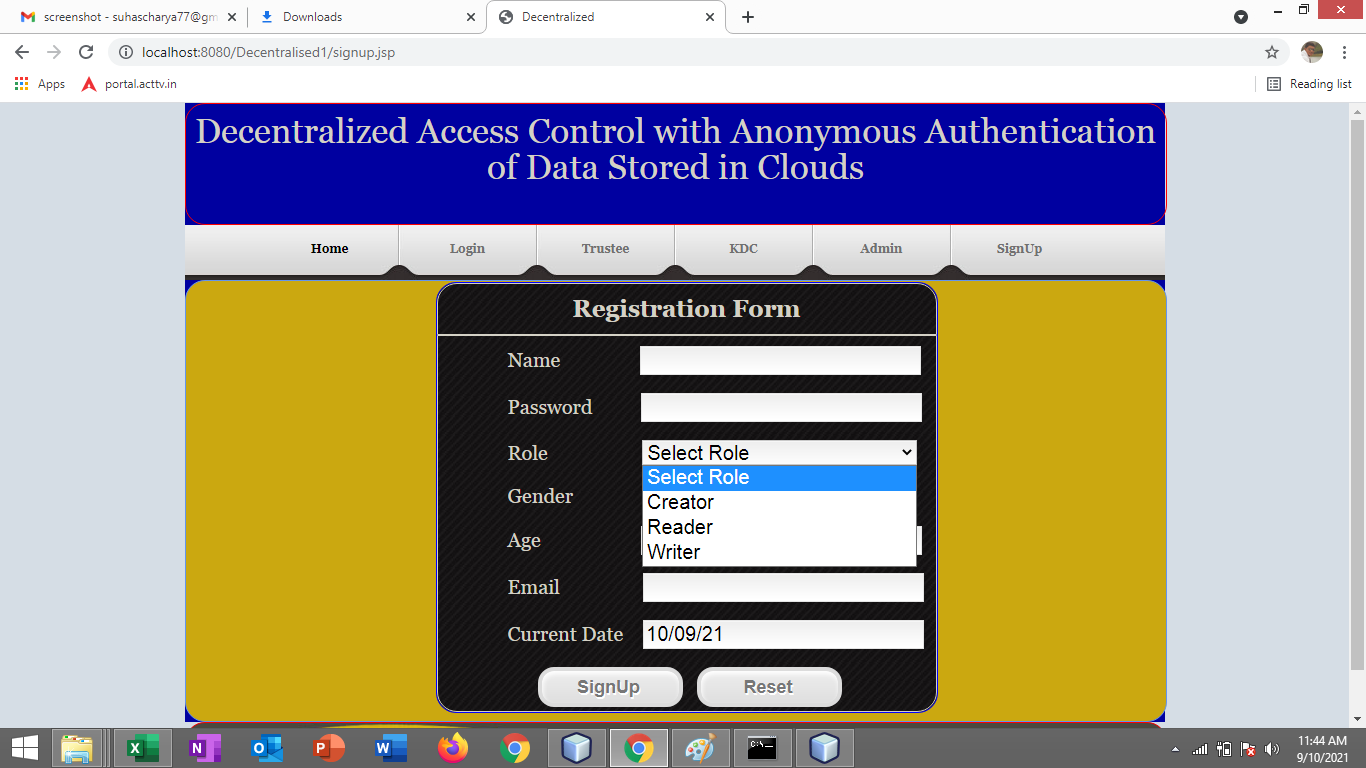


1. **Fig No 9.1: Database Tables**

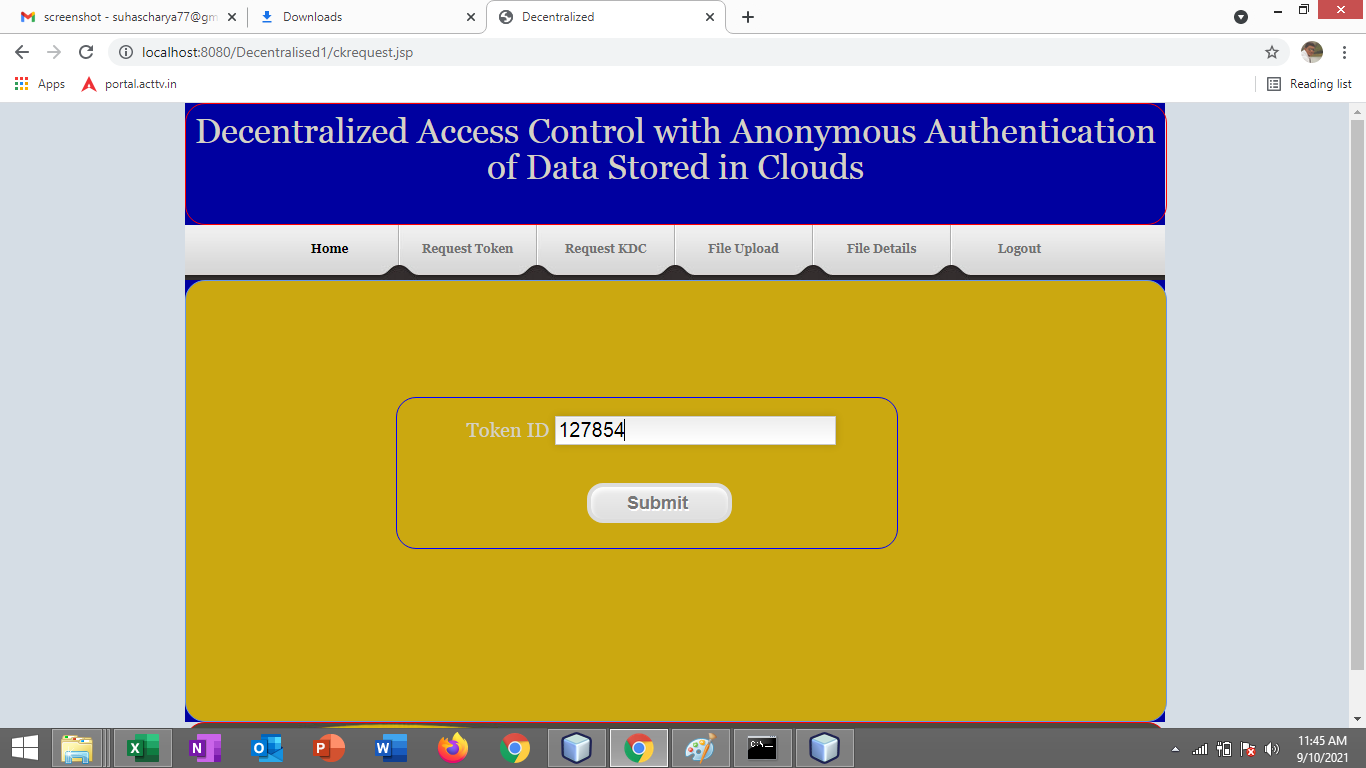
**Fig No 9.1:**



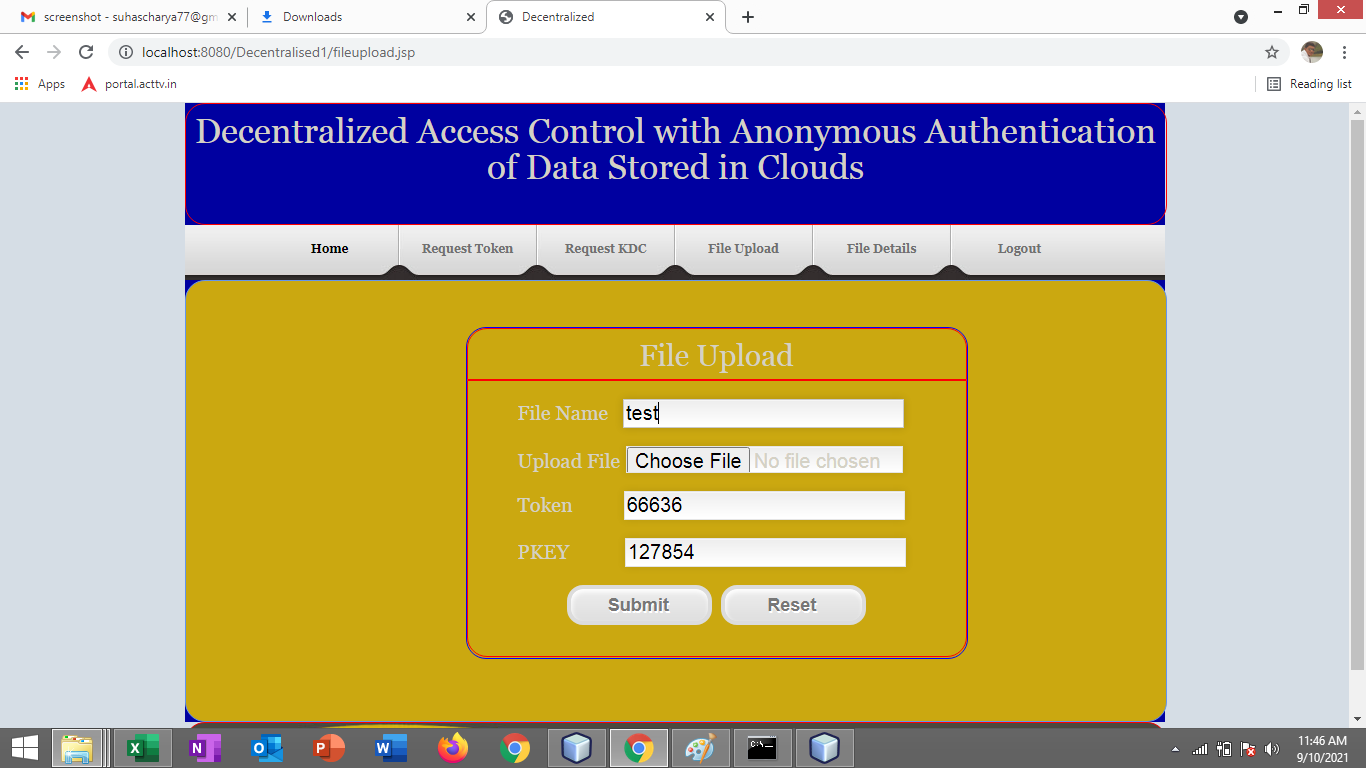
**Fig No 9.1:**



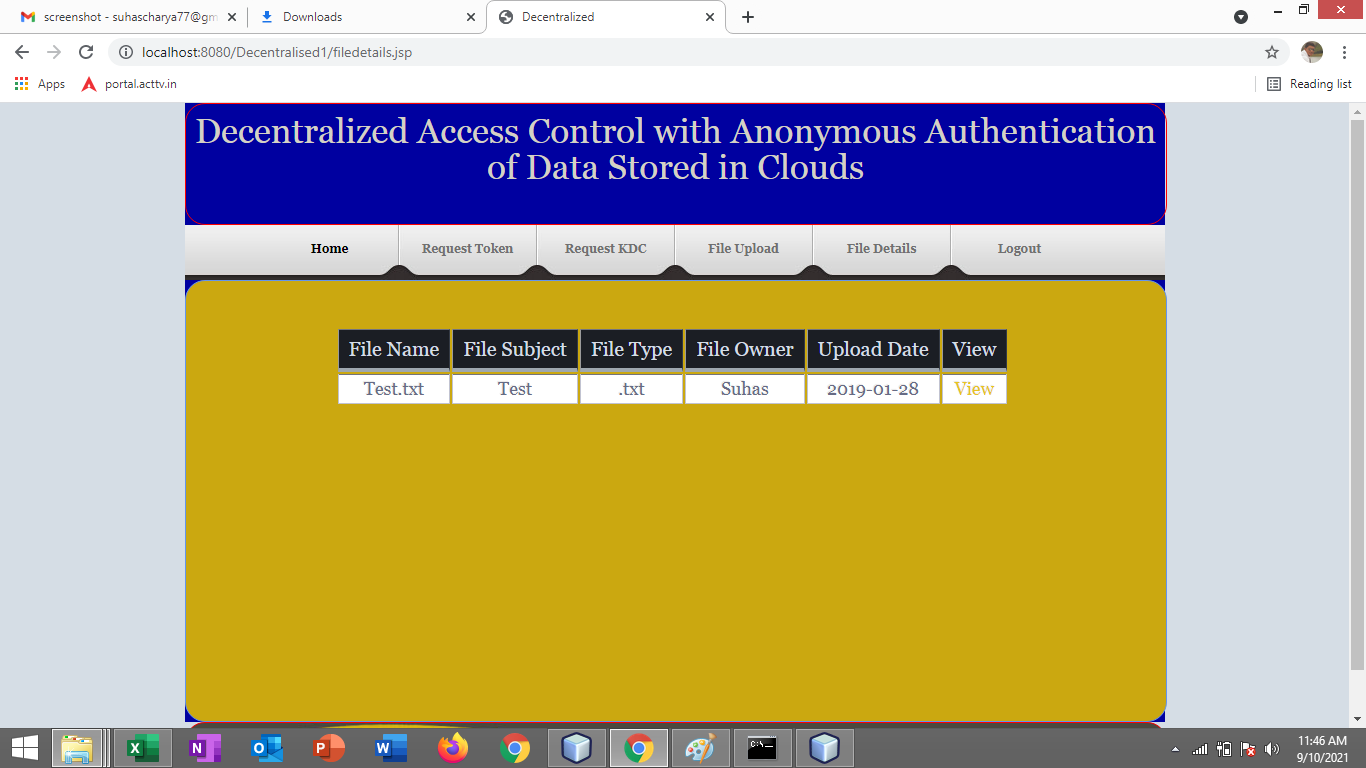
**Fig No 9.1:**



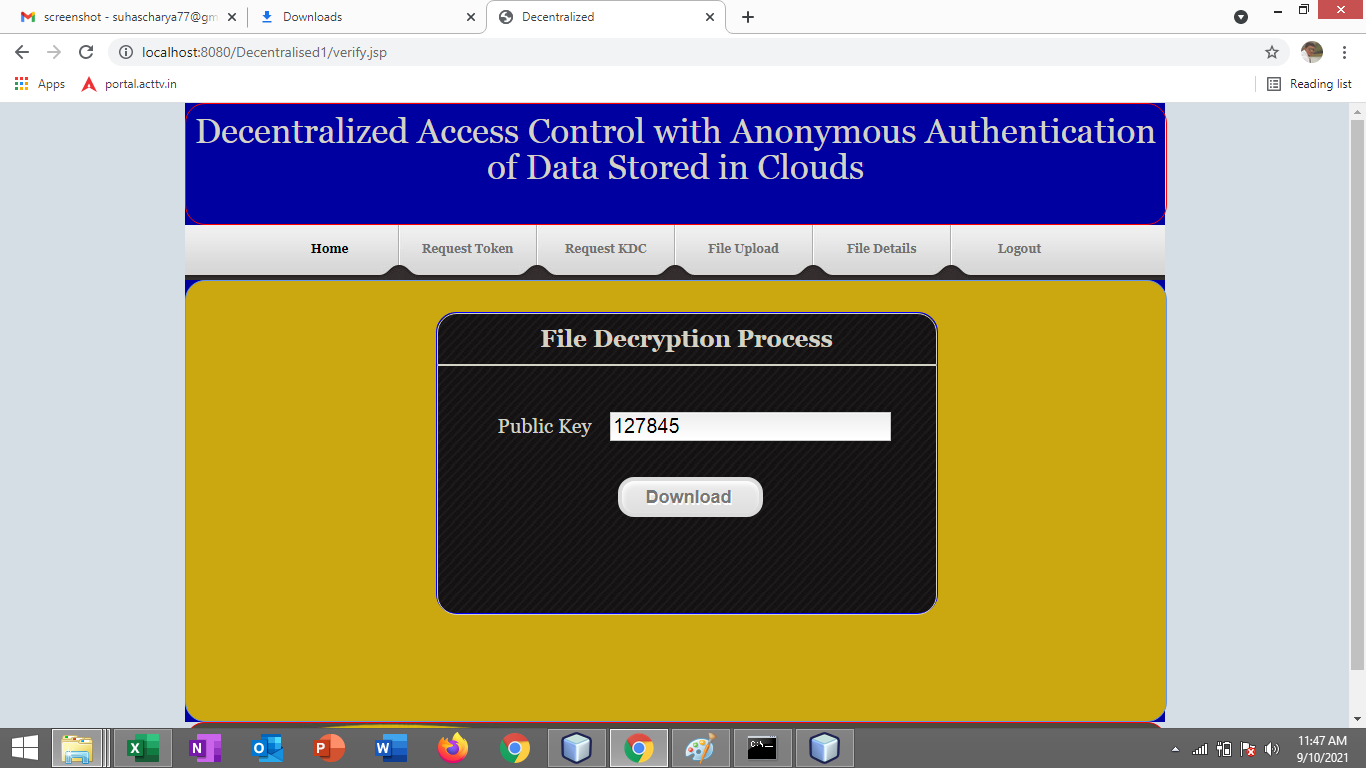
**Fig No 9.1:**



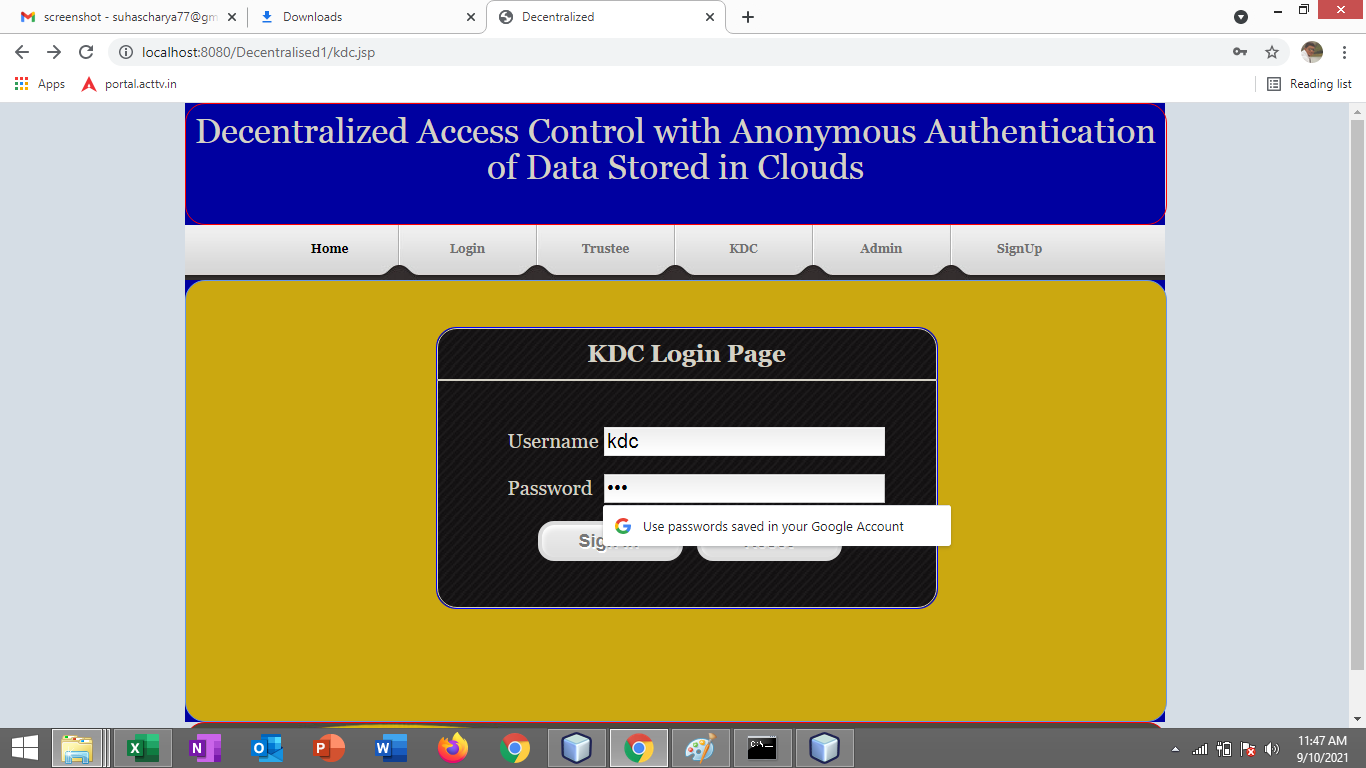
**Fig No 9.1:**



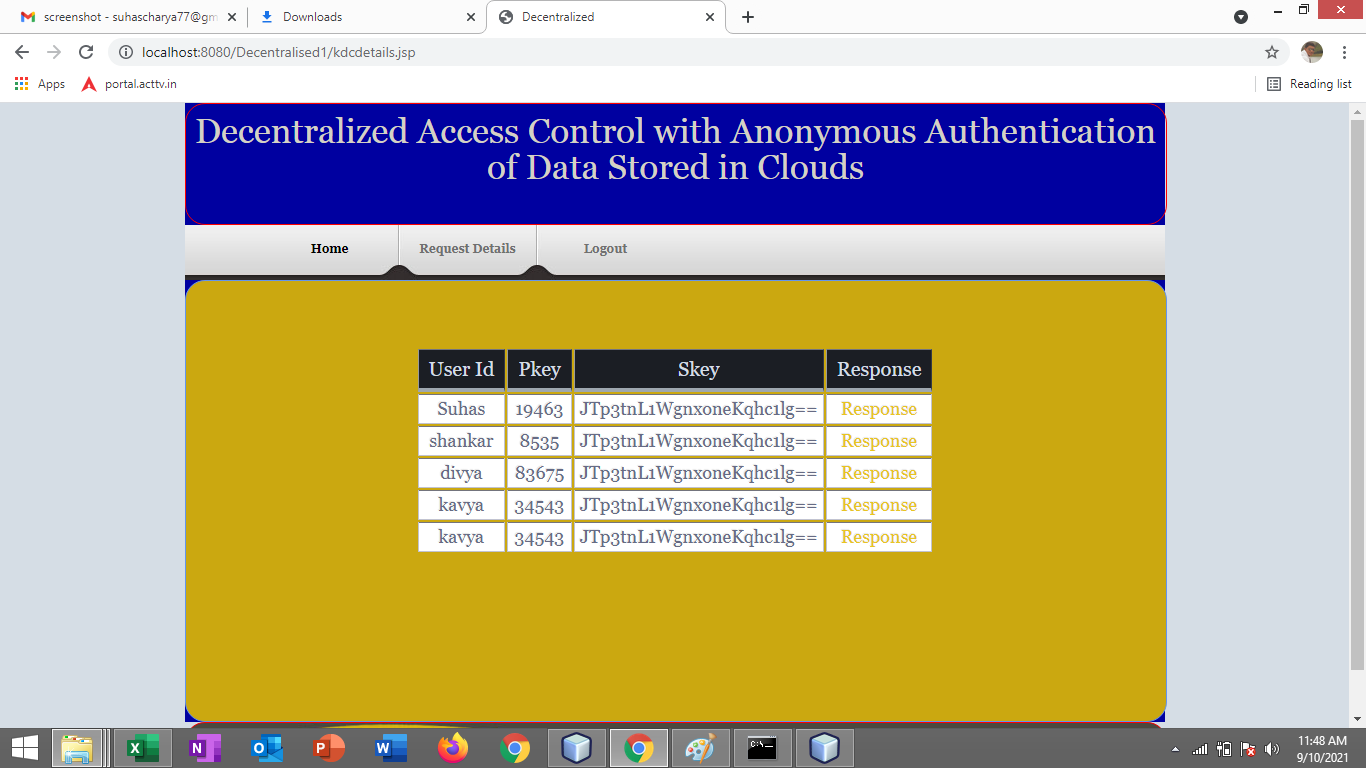
**Fig No 9.1:**



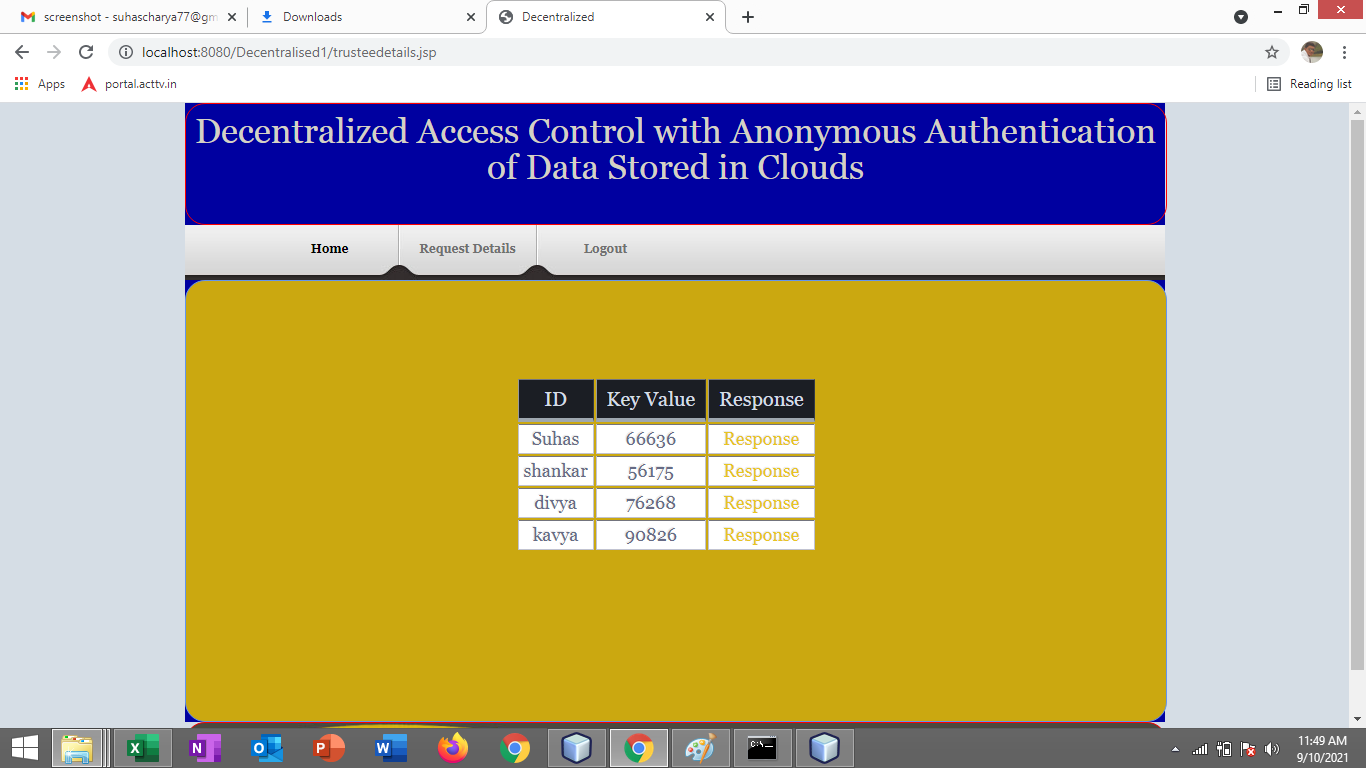
**Fig No 9.1:**



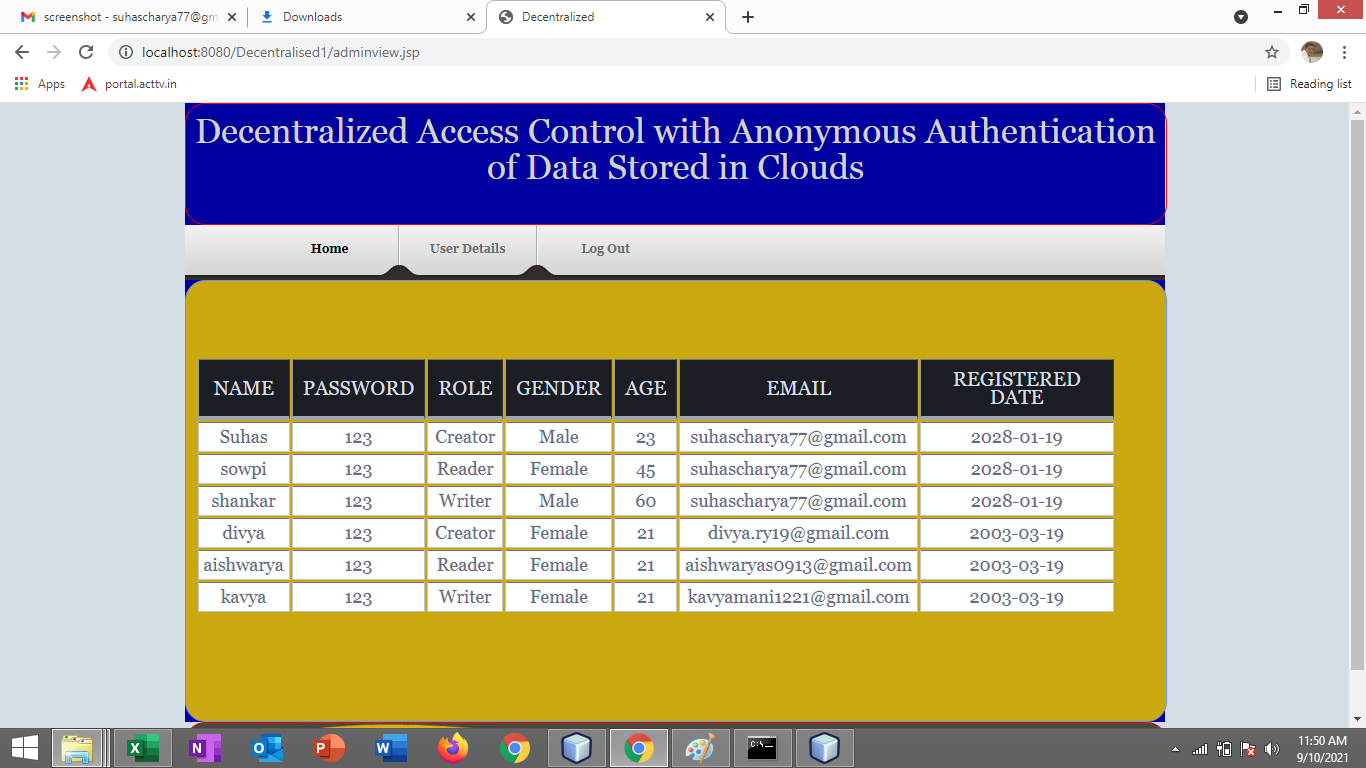
**Fig No 9.1:**



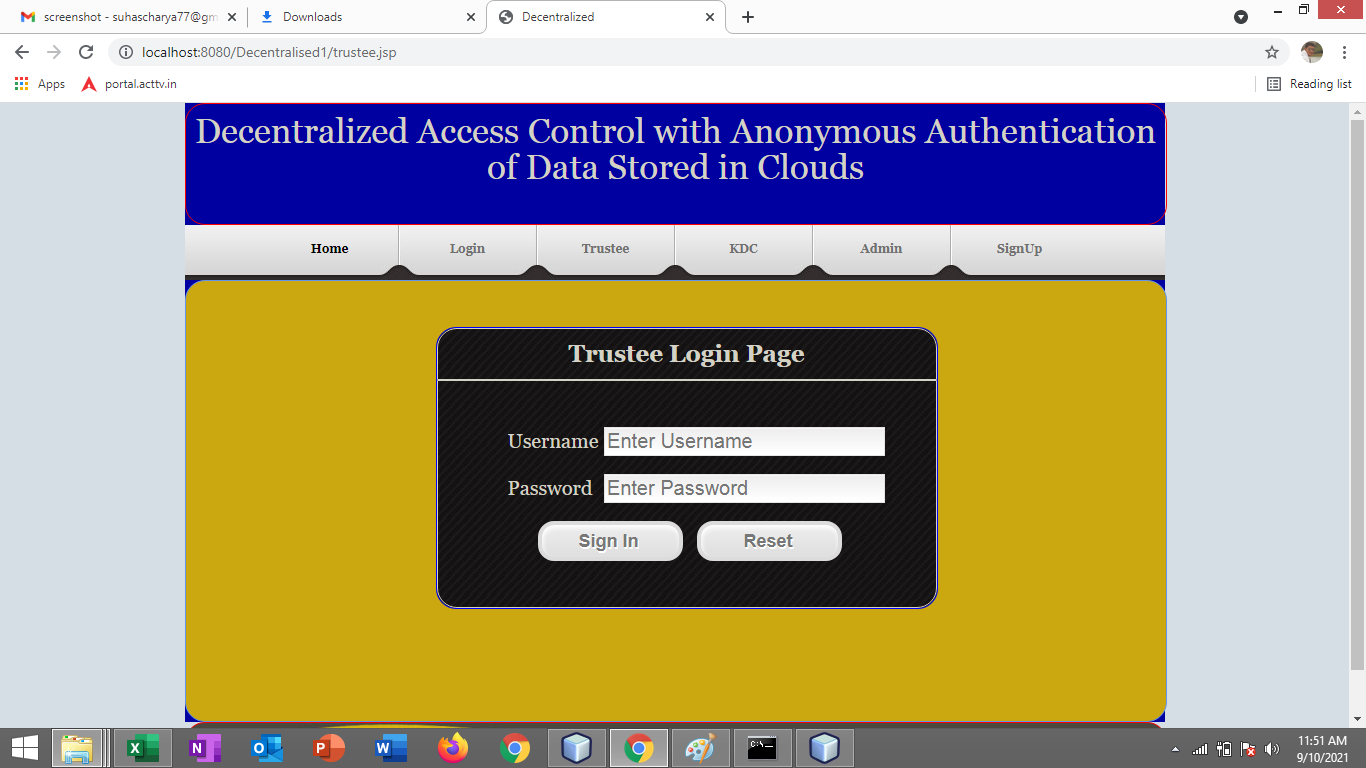
**Fig No 9.1:**



**Fig No 9.1:**



**Fig No 9.1:**



**Fig No 9.1:**

**10. CONCLUSION**

The automated resource management leads to utilize all the resource properly. The update and upgrade date and time along with the employee can be kept track. The admin keeps control of all the resource and the resources can be mapped to any main resource which leads to not wasting the resource any time hence this increases the productivity.

**11. FUTURE ENHANCEMENT**

* Currently application is using random key generation using math Random function for generating all keys the other techniques might be used to generate random key.
* One key distributor is there in this application in future many key distributor are supported.
* The key is distributing with Gmail services in future it can be sent to mobiles.
* Current application is taking text files as data input. It can take various types of data.

**BIBLIOGRAPHY**

Knox, Tim, “Digital Graphiti, Inc. Selected To Develop Web-Based E-Government System For The City of Huntsville, Alabama” February 25, 2002.

http://www.digitalgraphiti.com. 4 Kullhem, Charles A. Telephone interview with author. Montgomery Ala, April 8, 2002.

[1] http://en.wikipedia.org/wiki/Human\_resource\_ management\_system Books:

[2] James Goodwill, PURE Java Server Pages 3rd Edition

 [3] Larne Pekowsky, Java Server Pages 2nd Edition

[4] Simon Brown, Sam Dalton, Daniel Jepp, Dave Johnson, Pro JSP 3rd Edition

 [5] Thearon Willis, SQL Server 2000 2nd Edition

**APPENDIX**

**INTRODUCTION**

Cloud computing is architecture for providing, pay per use, services on demand of the user over the network. Although cryptography can be used to provide the security of data but it has a problem. This cryptography problem is that, when we convert our plain text into the cipher text then it becomes meaningless, so the intruder will interrupt the transmission on the data from sender to recipient.

On the other hand water marking technique is also widely used for the security of data. So in this paper, a merged technique, combination of cryptography and water marking, is proposed to enhance the security of data. Firstly, for hiding the original message Triple Data Encryption Standard (3DES) algorithm will be used. Secondly, the water marking technique will be used over the encrypted message. Therefore, the multiple level securities will be provided using this proposed system.

Cloud computing is an emerging computing paradigm that enables users to store their data into a cloud server to enjoy scalable and on-demand services. Nevertheless, it also brings many security issues since cloud service providers (CSPs) are not in the same trusted domain as users.

To protect data privacy against untrusted CSPs, existing solutions apply cryptographic methods (e.g., encryption mechanisms) and provide decryption keys only to authorized users. However, sharing cloud data among authorized users at a fine-grained level is still a challenging issue, especially when dealing with dynamic user groups. In this paper, we propose a secure and efficient finegrained access control and data sharing scheme.

Back in past if we had to access a file or document to another computer, we would have to copy the file to an external discand bring it to the computer on which the file was supposed to be accessed. But now we can access a file on the computers placed miles away from our personal computers without copying it or bringing it in some external drive. This process of copy and paste can be removed by using the cloud storage. We can simply share the files on the cloud and these shared files at can be accessed at any place on the world without any interruption.

**NEED OF THE PROJECT**

**EXISTING SYSTEM**

The existing is not in cloud and security is not given properly hence any one can theft the data. Recent applications used AES algorithm for the encryption and decryption process. Also centralized approach.

Disadvantages

Resource spending and maintenance issue

More space usage

Centralized approach have more performance issue

AES is less bit and easily can be hacked

**PROPOSED SYSTEM**

In our system we deploy the application in cloud with distributed approach In symmetric cryptography we only use one key for both encryption and decryption process. We use DES algorithm. It is the successor of DES algorithm which uses 168 bits key size. The key size of 3DES algorithm is 3 times bigger than the key size of DES algorithm i.e., (3\*56 bits) and the block size of 3DES algorithm is 64 bits. 48 rounds are used in 3DES algorithm for the encryption process.

Advantages

Resource and storage is not our maintenance as cloud

System is distributed and parallel concept is used for fast data fetching

3DES is used where three times encryption is used more secured

**REQUIRMENT SEPCIFICATION**

* **Hardware Requirement Specification:**
* Hardware: Dual Core
* Hard Disk: 50 GB
* Speed: 1.4 GHz
* RAM: 1GB
* Key Board: Standard Keyboard
* Touch Pad: Button Mouse
* Monitor: LED
* **Software Requirement Specification:**
* Operating System : Windows
* IDE : Net Beans 7.3.1
* Technology : Java and J2EE
* Web Server : Tomcat
* Web Technologies : Html, JavaScript, CSS
* Java Version : JDK1.7
* Database : My SQL