

Data Security in green cloud

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree

In

Computer Science and Engineering

By

BUDDE SAIVARMA _(38110088)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

March - 2022



**SATHYABAMA INSTITUTE OF SCIENCE
AND TECHNOLOGY**

(Established under Section 3 of UGC Act, 1956)

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600119 www.sathyabama.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of Budde salvarma (Reg. No. 38110088) who carried out the project entitled "Data Security in green cloud" under our supervision from December 2021 to March 2022.

Internal Guide

Dr.sathya priya, M.E., Ph.D

..

Head of the Department

Dr. S. VIGNESHWARI M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

Budde salvarma(**Reg. No. 3811088**),

hereby declare that the Project Report entitled "**Data security in green cloud**

" done by me under the guidance of **Dr.Sathya priya , M.E., Ph.D.**, is submitted in

partial fulfillment of the requirements for the award of Bachelor of Engineering /

DATE: Technology degree in Computer Science and Engineering.

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are 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,** and **DR. S. VIGNESHWARI, M.E., Ph.D., Head of the Department, Department of Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. Sathya Priya , M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the department of **COMPUTER SCIENCE AND ENGINEERING** who were helpful in many ways for the completion of the project.

ABSTRACT

Intrusion detection systems (IDSs) are currently drawing a great amount of interest as a key part of system defense. IDSs collect network traffic information from some point on the network or computer system and then use this information to secure the network. To distinguish the activities of the network traffic that the intrusion and normal is very difficult and to need much time consuming. An analyst must review all the data that large and wide to find the sequence of intrusion on the network connection. Therefore, it needs a way that can detect network intrusion to reflect the current network traffics. In this study, a novel method to find intrusion characteristic for IDS using genetic algorithm machine learning of data mining technique was proposed. Method used to generate of rules is classification by Genetic algorithm of decision tree. These rules can determine of intrusion characteristics then to implement in the genetic algorithm as prevention. so that besides detecting the existence of intrusion also can execute by doing deny of intrusion as prevention.

TABLE OF CONTENTS

Chapter no.	Title	Page No.
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF TABLES	viii
1.	INTRODUCTION	1
	1.1 BACKGROUND	1
	1.2 MOTIVATION	
	1.3 OBJECTIVE	1
	1.4 COLLECTING DATASET	1
	1.5 EVALUATING THE RESULT	2
2.	AIM AND SCOPE OF THE PRESENT INVESTIGATION	2
	2.1 cloud computing SCOPE	2
	2.1.1 CLASSIFICATION OF cloud computing	4
	2.1.1 SUPERVISED cloud computing	5
	2.1.2 UNSUPERVISED Cloud computing	5
	2.2 DECISION TREE SCOPE	6
	2.3 GENETIC ALGORITHM SCOPE	6
	2.4 INTRUSION DETECTION SYSTEM SCOPE	6
	2.5 EXISTING SYSTEM	7
	2.6 PROBLEM STATEMENT	7

LIST OF FIGURES

Figure No.	Name of the Figure	Page No.
2.1.1	Cloud computing	04
2.1.1	Learning Phase	04
2.1.2	Inference Form Model	05
2.1.2	Machine Learning Classification	05
3.1	Decision Tree	09
3.2	Genetic Algorithm	10
3.2	Initial Population	11
3.2	Cross Over and Cross over Point	12
3.2	Mutation	13
3.2	System Architecture	15
4.1	Green cloyFlow DiagramData	16
4.1.1	Collection	17
4.2	Result	17 – 20
4.3	Code	20-37

DATA SECURITY IN GREEN CLOUD

ABSTRACT

- Green computing is defined as the study and practice of designing , manufacturing, using, and disposing of computers, servers, and associated subsystems-such as monitors, printers, storage devices, and networking and communications systems-efficiently and effectively with minimal or no impact on the environment."

- Cloud software service is a modality for providing computer facilities and deploying software via the Internet. The concept combines Cloud Computing and Software-as-a-Service (SaaS).
- Cloud computing represents a contextual shift in how computers are provisioned and accessed.
- There are many different examples of cloud software service, and this paper seeks to combine the salient elements into a composite picture of the subject matter.

OBJECTIVE

- The goal of green computing is to reduce the use of hazardous materials, maximize energy efficiency during the product's lifetime, and promote the recyclability of outdated products and factory waste.
- Green computing can be achieved by either Product Longevity Resource allocation or Virtualization or Power management.
- Power is the bottleneck of improving the system performance.

AIM

The main aim of this project, Two-Side verification is a process that involves two authentication methods performed one after the other to verify that someone or something requesting access is who or what they are declared to be.

INTRODUCTION

In recent years, cloud computing technologies have gotten rapid developments and a line of studies have been done on security issues in cloud computing, such as access control and privacy protection. As a typical service in cloud computing, cloud storage needs both data security and search functionality. In fact, user-side verifiability takes into consideration that the cloud server may be malicious, that is, the cloud server may only return part of search results or maliciously return incorrect results. The issue of user-side verifiability is firstly addressed in [1]. However, these two schemes cannot support server-side verifiability and fair payment without any trusted third party. Furthermore, server-side verifiability takes into consideration that the data owner may be malicious, that is, the data owner may maliciously outsource invalid data in the data storage phase and fraudulently claim compensation later. This concern has not been addressed and even has received little attention in the literature. Last but not least, most of the previous schemes are bank-dependent. Specifically, either the payment issue is not considered or the default traditional payment mechanism is exploited in which a trusted third party (TTP) such as a trustworthy bank has to be introduced for payment fairness. Payment fairness can promote the honest behaviors of users and cloud servers [7]. If a malicious behavior is detected based on the user-side verifiability (resp. server-side verifiability), the data owner (resp. cloud server) should get adequate compensation from the cloud server (resp. data owner) no matter what the cloud server (resp. data owner) does. Therefore, fair payment without any third party is a meaningful and challenging task and it remains in SSE.

In order to thoroughly address the aforementioned challenging issues in cloud computing, we propose TKSE, a Trustworthy Keyword Search scheme over Encrypted data without needing any third party. TKSE is proven secure and our performance evaluation shows its efficiency. In particular, TKSE is characterized by the following desirable features.

- **Keyword Search over Encrypted Data.** The encrypted data index based on the Elliptic Curve Digital Signature Algorithm (ECDSA) allows a user to search over the outsourced encrypted data.
- **User-side Verifiability.** In TKSE, a data owner can embed search requirements into the output script of a joint transaction such that the transaction can be redeemed by the cloud server if and only if the output script evaluates to true based on the returned search result. Therefore, TKSE enables the data owner to resist malicious cloud servers and user-side verifiability is realized.
- **Server-side Verifiability.** Similar to user-side verifiability, the public verification of digital signature enables the cloud server to check the validity of the outsourced encrypted data from the data owner in the data storage phase. Thus, malicious data owners can be detected by the cloud server, which realizes server-side verifiability.
- **Fair Payment and No TTP.** Based on hash functions and ECDSA, TKSE is compatible with blockchains such as the Bitcoin blockchain and the Ethereum blockchain. The global consensus and distributed nature of a blockchain enable a fair payment mechanism in TKSE without introducing any TTP.

LITERATURE REVIEW

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, then the next step is to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

The major part of the project development sector considers and fully survey all the required needs for developing the project. For every project Literature survey is the most important sector in software development process. Before developing the tools and the associated designing it is necessary to determine and survey the time factor, resource requirement, man power, economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications in the respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with the next step such as developing the tools, and the associated operations.

Identity-based encryption with outsourced revocation in cloud computing

Identity-Based Encryption (IBE) which simplifies the public key and certificate management at Public Key Infrastructure (PKI) is an important alternative to public key encryption. However, one of the main efficiency drawbacks of IBE is the overhead computation at Private Key Generator (PKG) during user revocation. Efficient revocation has been well studied in traditional PKI setting, but the cumbersome management of certificates is precisely the burden that IBE

strives to alleviate. In this paper, aiming at tackling the critical issue of identity revocation, we introduce outsourcing computation into IBE for the first time and propose a revocable IBE scheme in the server-aided setting. Our scheme offloads most of the key generation related operations during key-issuing and key-update processes to a Key Update Cloud Service Provider, leaving only a constant number of simple operations for PKG and users to perform locally. This goal is achieved by utilizing a novel collusion-resistant technique: we employ a hybrid private key for each user, in which an AND gate is involved to connect and bound the identity component and the time component. Furthermore, we propose another construction which is provable secure under the recently formulized Refereed Delegation of Computation model. Finally, we provide extensive experimental results to demonstrate the efficiency of our proposed construction.

New publicly verifiable databases with efficient updates

The notion of verifiable database (VDB) enables a resource-constrained client to securely outsource a very large database to an untrusted server so that it could later retrieve a database record and update it by assigning a new value. Also, any attempt by the server to tamper with the data will be detected by the client. Very recently, Catalano and Fiore [17] proposed an elegant framework to build efficient VDB that supports public verifiability from a new primitive named vector commitment. In this paper, we point out Catalano-Fiore's VDB

framework from vector commitment is vulnerable to the so-called forward automatic update (FAU) attack. Besides, we propose a new VDB framework from vector commitment based on the idea of commitment binding. The construction is not only public verifiable but also secure under the FAU attack. Furthermore, we prove that our construction can achieve the desired security properties.

A searchable symmetric encryption scheme using blockchain

At present, the cloud storage used in searchable symmetric encryption schemes (SSE) is provided in a private way, which cannot be seen as a true cloud. Moreover, the cloud server is thought to be credible, because it always returns the search result to the user, even they are not correct. In order to really resist this malicious adversary and accelerate the usage of the data, it is necessary to store the data on a public chain, which can be seen as a decentralized system. As the increasing amount of the data, the search problem becomes more and more intractable, because there does not exist any effective solution at present. In this paper, we begin by pointing out the importance of storing the data in a public chain. We then innovatively construct a model of SSE using blockchain(SSE-using-BC) and give its security definition to ensure the privacy of the data and improve the search efficiency. According to the size of data, we consider two different cases and propose two corresponding schemes. Lastly, the security and performance analyses show that our scheme is feasible and secure.

BlockchainBased system for secure data storage with private

Key word search

Traditional cloud storage has relied almost exclusively on large storage providers, who act as trusted third parties to transfer and store data. This model poses a number of issues including data availability, high operational cost, and data security. In this paper, we introduce a system that leverages blockchain technology to provide a secure distributed data storage with keyword search service. The system allows the client to upload their data in encrypted form, distributes the data content to cloud nodes and ensures data availability using cryptographic techniques. It also provides the data owner a capability to grant permission for others to search on her data. Finally, the system supports private keyword search over the encrypted dataset.

Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems

Electronic Health Records (EHRs) are entirely controlled by hospitals instead of patients, which complicates seeking medical advices from different hospitals. Patients face a critical need to focus on the details of their own healthcare and restore management of their own medical data. The rapid development of blockchain technology promotes population healthcare, including medical records as well as patient-related data. This technology provides patients with

comprehensive, immutable records, and access to EHRs free from service providers and treatment websites. In this paper, to guarantee the validity of EHRs encapsulated in blockchain, we present an attribute-based signature scheme with multiple authorities, in which a patient endorses a message according to the attribute while disclosing no information other than the evidence that he has attested to it. Furthermore, there are multiple authorities without a trusted single or central one to generate and distribute public/private keys of the patient, which avoids the escrow problem and conforms to the mode of distributed data storage in the blockchain. By sharing the secret pseudorandom function seeds among authorities, this protocol resists collusion attack out of N from $N - 1$ corrupted authorities. Under the assumption of the computational bilinear Diffie-Hellman, we also formally demonstrate that, in terms of the unforgeability and perfect privacy of the attribute-signer, this attribute-based signature scheme is secure in the random oracle model. The comparison shows the efficiency and properties between the proposed method and methods proposed in other studies.

Existing System

- In slight contrast, it is used and own by an organization internally, anyone within the organization can access the data, services and web application except for outside, reaffirmed the above assertions and added that, a private cloud provides the upmost degree of control over performance, trustworthiness and security.

- Many of the features that make cloud computing attractive, have not just challenged the existing security system, but have also revealed new security issues.
- This paper provides an insightful analysis of the existing status on cloud computing security issues based on a detailed survey carried by the author.
- It also makes an attempt to describe the security challenges in Software as a Service (SaaS) model of cloud computing and also endeavors to provide future security research directions.

Disadvantages for Existing System

- Data confidentiality and privacy has been achieved but identity privacy neglected.
- Less security
- It is not having any data content

Proposed System

- There are number of security issues and threats on cloud storage such as lack of proper policy to govern cloud storage, inadequate access to client

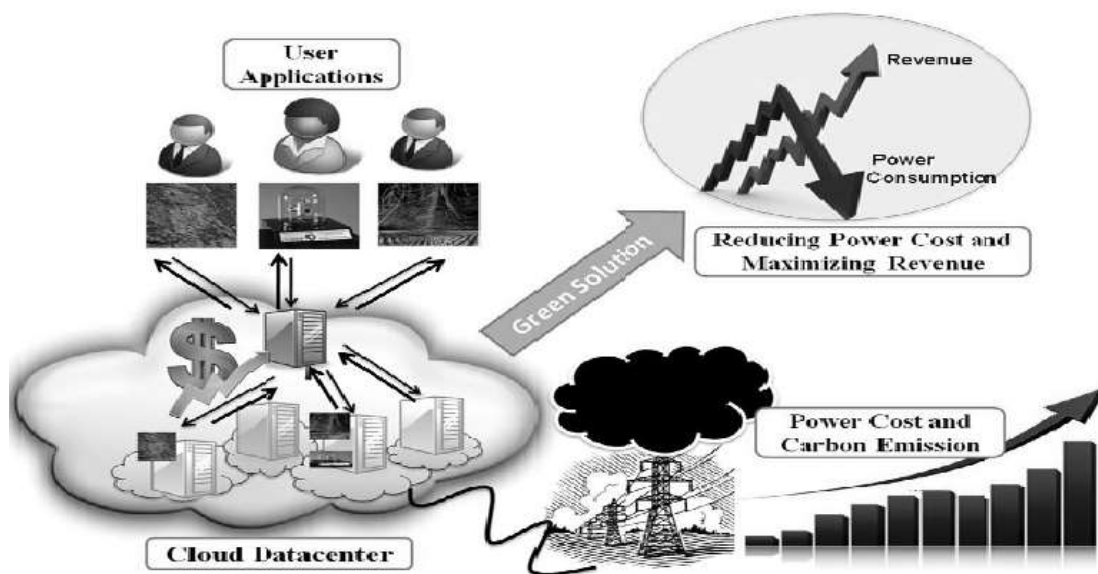
information, breach of trustworthiness, technological bottlenecks, lack ICT infrastructure, managerial problems, absence of qualified personal to manage ICT facilities/equipment, lack of data confidentiality and integrity.

- As cloud becomes the tool of choice for more data storage services, the number of service providers has also increased.
- With these choices, organizations have a wide selection of services available to move their data to the cloud. However, the responsibility to maintain the security of sensitive data stored therein remains paramount.
- This paper will discuss some of the challenges of securing a cloud storage and putting it into context by reviewing relevant literature. The challenges associated with the three important security aspects
- We aim to provide insights of security challenges and its solutions to enhance cloud storage implementation

Advantages for proposed system

- Save data management cost
- To protect user privacy and data security
- Message authentication code
- Integrity protection

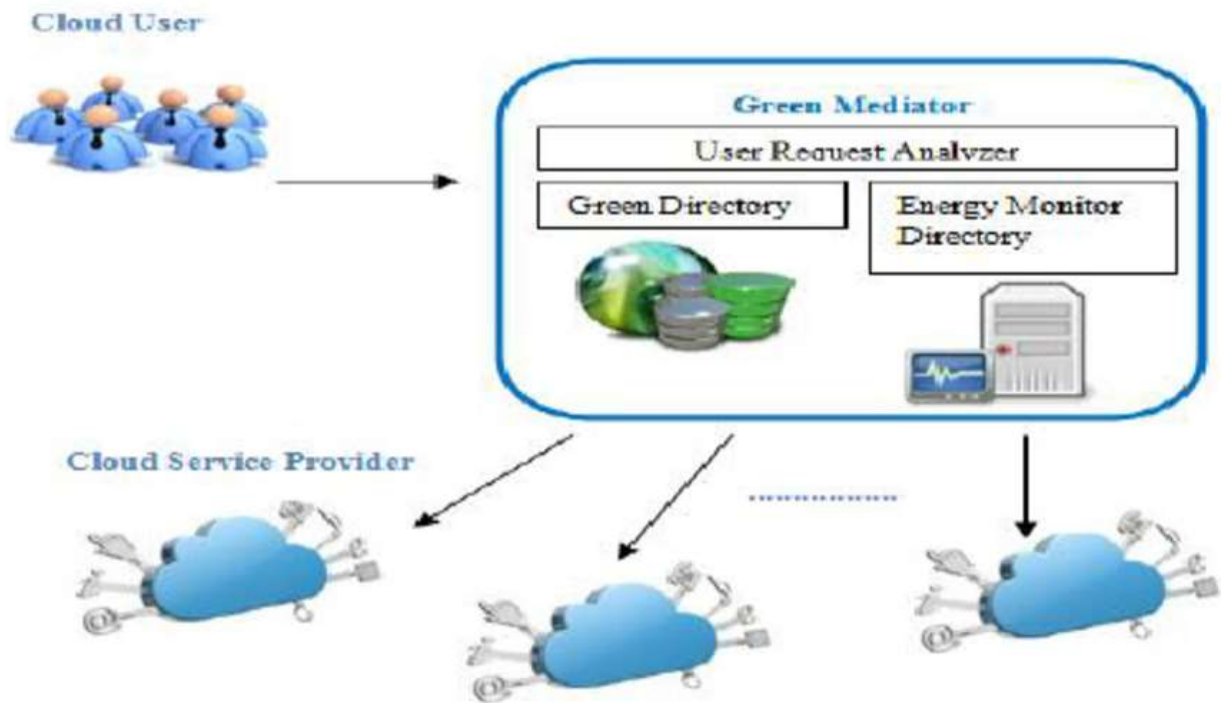
SYSTEM ARCHITECTURE:



METHODS OF IMPLEMENTATION

- This data may include PUE and cooling efficiency of Cloud datacenter which is providing the service, the network cost and carbon emission rate of electricity, Green Broker calculates the carbon emission of all the Cloud providers who are offering the requested Cloud service.
- Then, it selects the set of services that will result in least carbon emission and buy these services on behalf users.

Green Cloud service security process



SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- RAM - 512MB

SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding language - Java
- IDE - NetBeans
- Database - MYSQL

WORK FLOW:

- In the Green Cloud architecture, users submit their Cloud service requests through a new middleware Green Broker that manages the selection of the greenest Cloud provider to serve the user's request. A green request can be of three types i.e., software, platform or infrastructure.
- The Cloud providers can register their services in the form of "green offers" to a public directory which is accessed by Green Broker.
- The green offers consist of green services, pricing and time when it should be accessed for least carbon emission. Green Broker gets the current status

of energy parameters for using various Cloud services from Carbon Emission Directory.

- The Carbon Emission Directory maintains all the data related to energy efficiency of Cloud service.

NEED OF GREEN COMPUTING IN CLOUDS

- Modern data centers, operating under the Cloud computing model are hosting a variety of applications ranging from those that run for a few seconds (e.g. 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 (e.g. 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. Normally, data center resources are statically allocated to applications, based on peak load characteristics, in order to maintain isolation and provide performance guarantees.
- Until recently, high performance has been the sole concern in data center deployments and this demand has been fulfilled without paying much attention to energy consumption. The average data center consumes as much energy as 25,000 households [20]. As energy costs are increasing while availability dwindles.

Advantages Of Proposed system

- Examples mentioned will help in harnessing energy
- Lots of research to be carried out
- Maximizing Efficiency of Green Data Centers
- Developing Regions to benefit the most

Applications:

- Public cloud storage is easy to deploy through a web portal managed by the cloud provider or through a marketplace offering, such as the AWS Marketplace
- Storage capacity can be dynamically scaled up or down as needs change. And the limits on capacity are high enough for most workloads.
- Cloud storage is accessible from almost any location, because the interface to the storage environment is through a web browser over the internet.
- Additionally, most cloud providers offer multiple storage offerings, such as file services, object storage, and block storage, to meet the needs of most applications.

INTRODUCTION

Java is one of the world's most important and widely used computer languages, and it has held this distinction for many years. Unlike some other computer languages whose influence has wearied with passage of time, while Java's has grown.

APPLICATION OF JAVA

Java is widely used in every corner of world and of human life. Java is not only used in softwares but is also widely used in designing hardware controlling

software components. There are more than 930 million JRE downloads each year and 3 billion mobile phones run java.

Following are some other usage of Java:

1. Developing Desktop Applications
2. Web Applications like Linkedin.com, Snapdeal.com etc
3. Mobile Operating System like Android
4. Embedded Systems
5. Robotics and games etc.

FEATURES OF JAVA

The prime reason behind creation of Java was to bring portability and security feature into a computer language. Beside these two major features, there were many other features that played an important role in moulding out the final form of this outstanding language. Those features are;

1) Simple

Java is easy to learn and its syntax is quite simple, clean and easy to understand. The confusing and ambiguous concepts of C++ are either left out in Java or they have been re-implemented in a cleaner way.

Eg: Pointers and Operator Overloading are not there in java but were an important part of C++.

2) Object Oriented

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

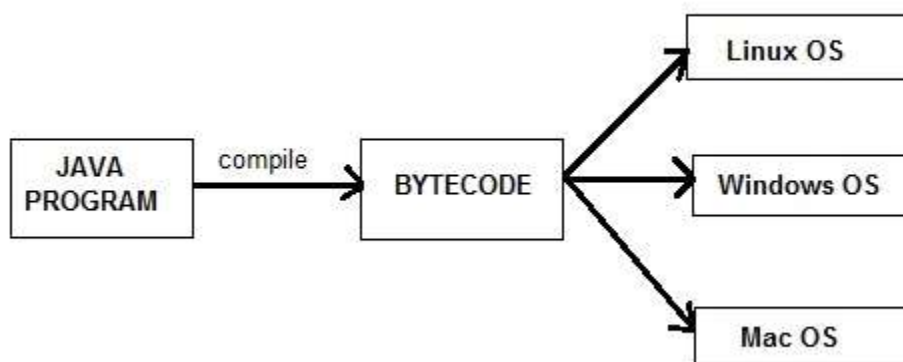
3) Robust

Java makes an effort to eliminate error prone codes by emphasizing mainly on compile time error checking and runtime checking. But the main areas which Java improved were Memory Management and mishandled Exceptions by introducing automatic Garbage Collector and Exception Handling.

4) Platform Independent

Unlike other programming languages such as C, C++ etc. which are compiled into platform specific machines. Java is guaranteed to be write-once, run-anywhere language.

On compilation Java program is compiled into byte code. This byte code is platform independent and can be run on any machine, plus this byte code format also provide security. Any machine with Java Runtime Environment can run Java Programs.



5) Secure

When it comes to security, Java is always the first choice. With java secure features it enable us to develop virus free, temper free system. Java program always runs in Java runtime environment with almost null interaction with system OS, hence it is more secure.

6) Multi-Threading

Java multithreading feature makes it possible to write program that can do many tasks simultaneously. Benefit of multithreading is that it utilizes same memory and other resources to execute multiple threads at the same time, like While typing, grammatical errors are checked along.

7) Architectural Neutral

Compiler generates byte codes, which have nothing to do with a particular computer architecture, hence a Java program is easy to interpret on any machine.

8) Portable

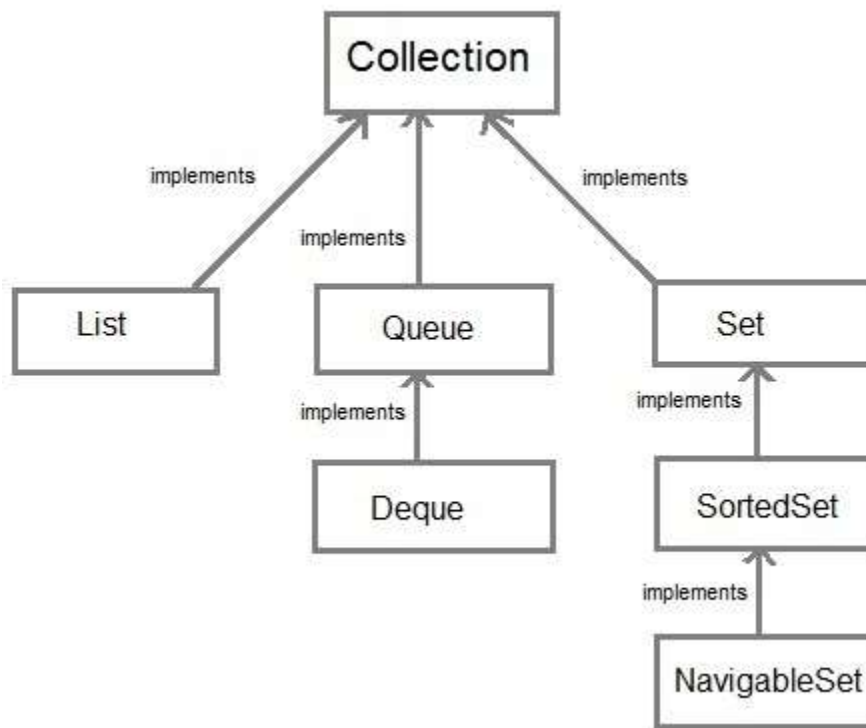
Java Byte code can be carried to any platform. No implementation dependent features. Everything related to storage is predefined, example: size of primitive data types

10) High Performance

Java is an interpreted language, so it will never be as fast as a compiled language like C or C++. But, Java enables high performance with the use of just-in-time compiler.

COLLECTION FRAMEWORK

Collection framework was not part of original Java release. Collections was added to J2SE 1.2. Prior to Java 2, Java provided adhoc classes such as Dictionary, Vector, Stack and Properties to store and manipulate groups of objects. Collection framework provides many important classes and interfaces to collect and organize group of alike objects.



MYSQL

MySQL, officially, but also called "My Sequel" is the world's most widely used open-source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases, though SQLite probably has more total embedded deployments. The SQL phrase stands 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. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks).

LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Freesoftware-open source projects that require a full-featured database management system often use MySQL.

For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr and YouTube.

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

The feasibility study investigates the problem and the information needs of the stakeholders. It seeks to determine the resources required to provide an

information systems solution, the cost and benefits of such a solution, and the feasibility of such a solution.

The goal of the feasibility study is to consider alternative information systems solutions, evaluate their feasibility, and propose the alternative most suitable to the organization. The feasibility of a proposed solution is evaluated in terms of its components.

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed

on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this 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. The user must not feel threatened by the system, instead must accept it as a necessity.

MODULES

- (i) Login
- (ii) Registration
- (iii) Create Secret Key
- (iv) Authentication Scheme

(v) Two-Side Verification

Login

Logins are used by websites, computer applications, and mobile apps. They are a security measure designed to prevent unauthorized access to confidential data. When a login fails (i.e, the username and password combination does not match a user account), the user is disallowed access. Many systems block users from even trying to log in after multiple failed login attempts.

Registration

A registered user is a user of a website, program, or other system who has previously registered. Registered users normally provide some sort of credentials (such as a username or e-mail address, and a password) to the system in order to prove their identity: this is known as logging in. Systems intended for use by the general public often allow any user to register simply by selecting a register or sign up function and providing these credentials for the first time. Registered users may be granted privileges beyond those granted to unregistered users.

Create Secrete Key

Cryptography is the practice and study of secure communication in the presence of third parties. In the past cryptography referred mostly to encryption.

Encryption is the process of converting plain text information to cipher text. Reverse is the decryption. Encryption is a mechanism to make the information confidential to anyone except the wanted recipients. Cipher is the pair of algorithm that creates encryption and decryption. Cipher operation is depends on algorithm and the key. Key is the secret that known by communicants.

Authentication scheme

- It is used to solve the problem of authenticating the keys of the person (say "person B") to whom some other person ("person A") is talking to or trying to talk to. In other words, it is the process of assuring that the key of "person A" held by "person B" does in fact belong to "person A" and vice versa.
- This is usually done after the keys have been shared among the two sides over some secure channel, although some of the algorithms share the keys at the time of authentication also.
- The simplest solution for this problem is for the two users concerned to meet face-to-face and exchange keys. However, for systems in which there are a large number of users or in which the users do not personally know each other (e.g., Internet shopping) this is not practical. There are various algorithm for both symmetric keys and asymmetric public key cryptography to solve this problem

Two-Side Verification

In this module, Two-Side verification is a process that involves two authentication methods performed one after the other to verify that someone or something requesting access is who or what they are declared to be.

SYSTEM DESIGN

INTRODUCTION

Design is a multi- step that focuses on data structure software architecture, procedural details, algorithm etc... and interface between modules. The design

process also translate the requirements into presentation of software that can be accessed for quality before coding begins. Computer software design change continuously as new methods; better analysis and border understanding evolved. Software design is at relatively early stage in its revolution.

Therefore, software design methodology lacks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines. However techniques for software designs do exist, criteria for design qualities are available and design notation can be applied.

DESIGN STRUCTURE

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- ❖ Convey information about past activities, current status or projections of the
- ❖ Future.
- ❖ Signal important events, opportunities, problems, or warnings.
- ❖ Trigger an action.
- ❖ Confirm an action.

SYSTEM TESTING

TESTING PROCESS

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product it is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produces valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

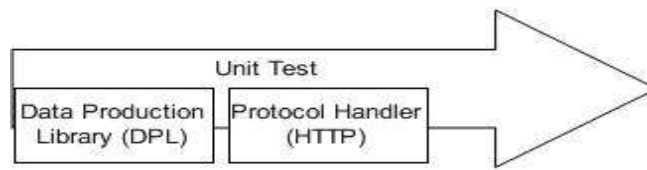


Fig. Unit Testing

Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation and user manuals.

Functional testing is centered on the following items:

Valid Input is used to identified classes of valid input must be accepted.

Invalid Input is used to identified classes of invalid input must be rejected.

Functions is used to identified functions must be exercised.

Output is used to identify classes of application outputs.

Systems/Procedures is used to interfacing systems or procedures must beinvoked. Organization and preparation of functional tests is focused on

requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successiveProcesses must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

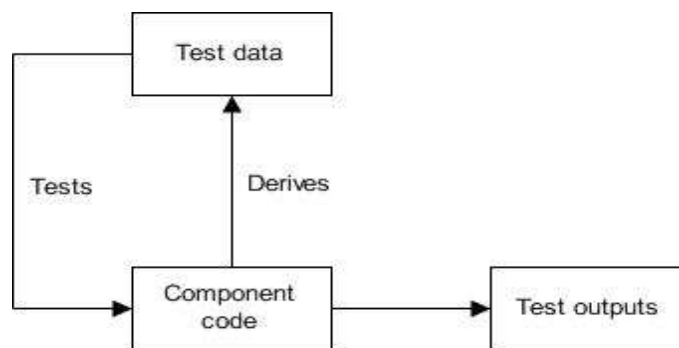


Fig. White box Testing

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source

document, such as specification or requirements document, such as specification
or requirements

document. It is a testing in which the software under test is treated, as a black box .you **cannot “see” into it. The test provides inputs and responds to outputs without**

Considering how the software works.

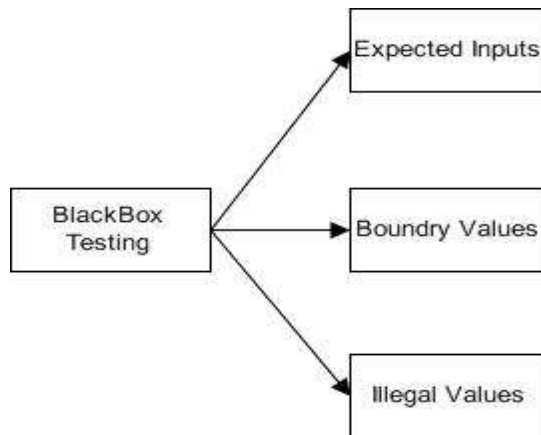


Fig. Black box Testing

TEST STRATEGY AND APPROACH

Field testing will be performed manually and functional tests will be written in detail.

Test Objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.
- Features to be tested
- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

ALPHA TESTING

In software development, alpha test will be a test among the teams to confirm that your product works. Originally, the term alpha test meant the first phase of testing in a software development process. The first phase includes unit testing, component testing, and system testing. It also enables us to test the product on the lowest common denominator machines to make sure download times are acceptable and pre loaders work.

BETA TESTING

In software development, a beta test is the second phase of software testing in which a sampling of the intended audience tries the product out. Beta testing can be considered "pre-release testing." Beta test versions of software are now distributed to curriculum specialists and teachers to give the program a "real world" test.

UML DIAGRAM

UML is a method for describing the system architecture in detail using the blueprint. UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. UML is a very important part of developing objects oriented software and the software development process. UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

UML is simply another graphical representation of a common semantic model. UML provides a comprehensive notation for the full lifecycle of object-oriented development.

ADVANTAGES

- To represent complete systems (instead of only the software portion) using object oriented concepts
- To establish an explicit coupling between concepts and executable code
- To take into account the scaling factors that are inherent to complex and critical systems
- To creating a modeling language usable by both humans and machines

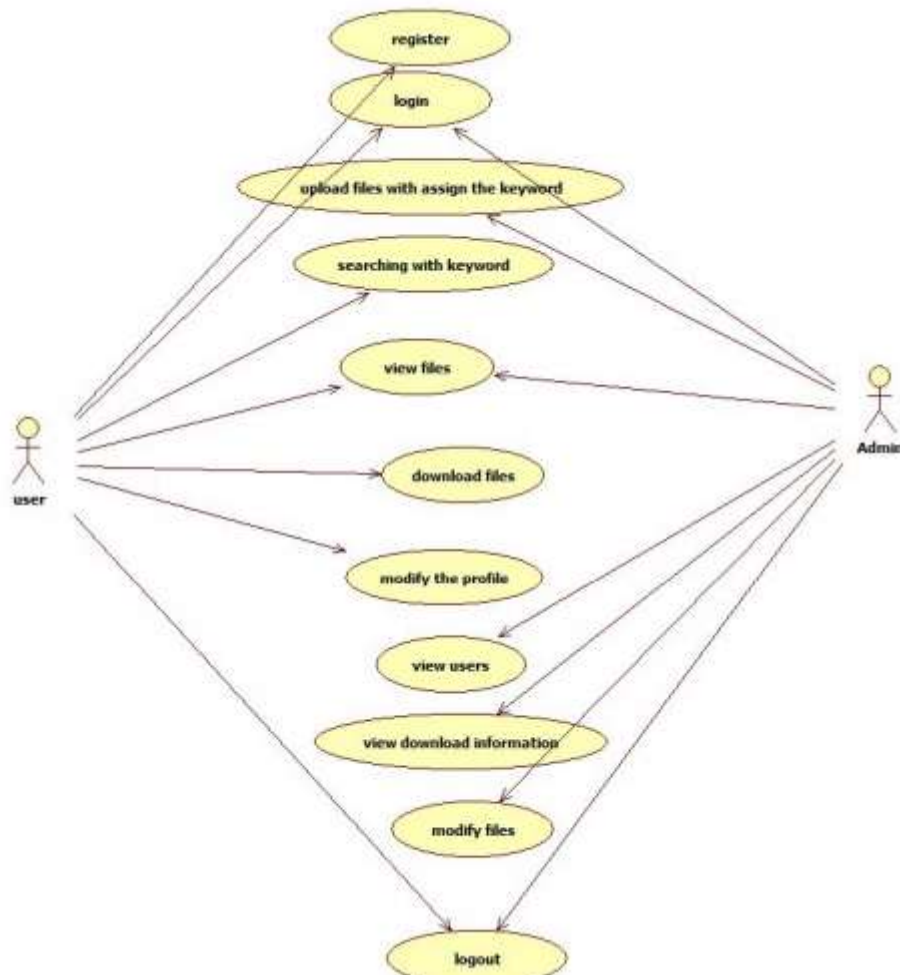
UML defines several models for representing systems

- The class model captures the static structure
- The state model expresses the dynamic behavior of objects
- The use case model describes the requirements of the user
- The interaction model represents the scenarios and messages flows
- The implementation model shows the work units

- The deployment model provides details that pertain to process allocation

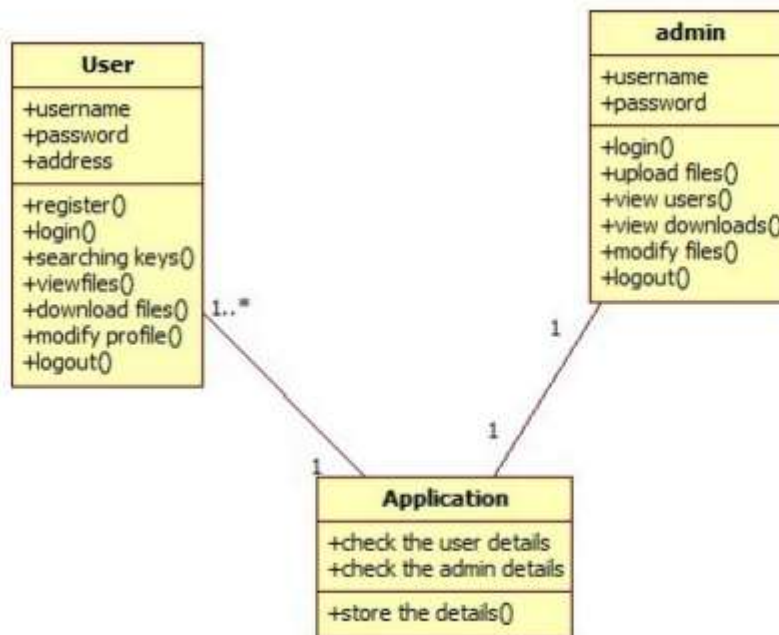
USECASE DIAGRAM

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors. An actor is represents a user or another system that will interact with the system you are modeling. A use case is an external view of the system that represents some action the user might perform in order to complete a task.



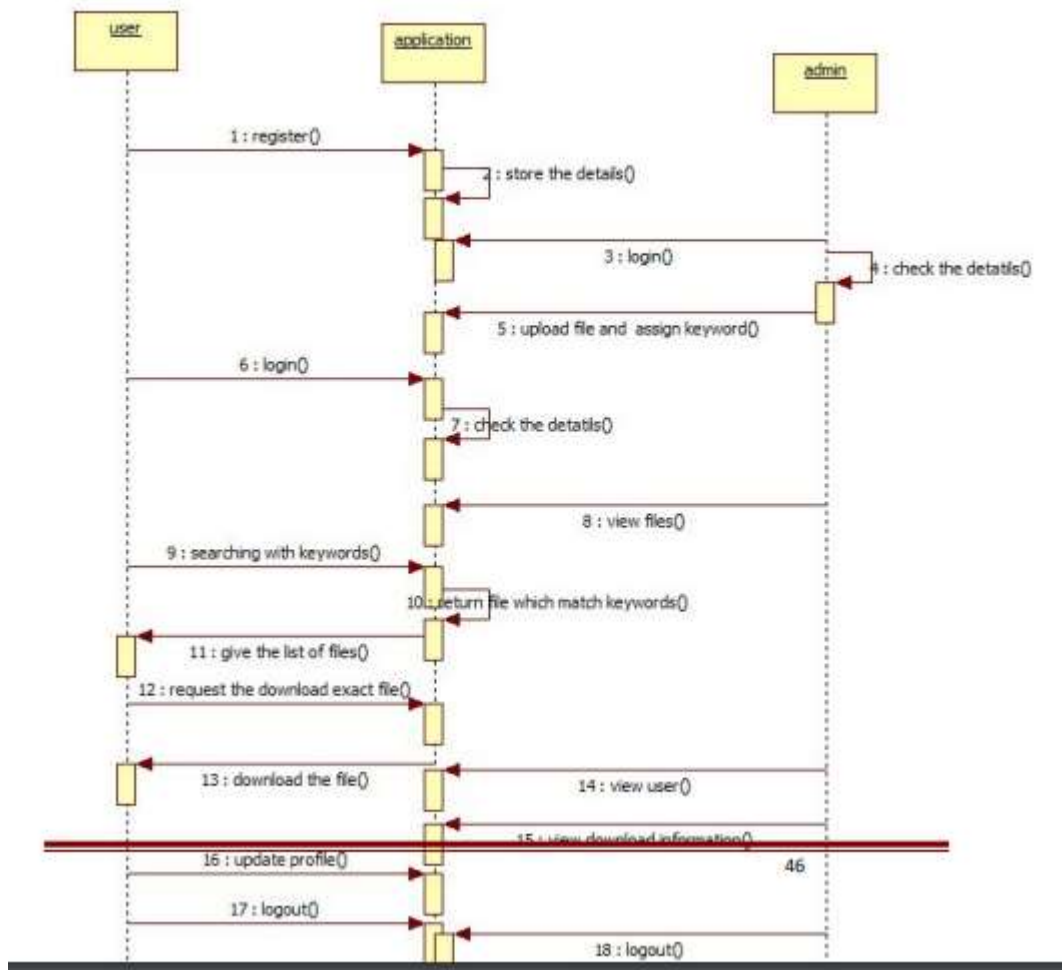
Class Diagram

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation. These perspectives become evident as the diagram is created and help solidify the design. Class diagrams are arguably the most used UML diagram type. It is the main building block of any object oriented solution. It shows the classes in a system, attributes and operations of each class and the relationship between each class. In most modeling tools a class has three parts, name at the top, attributes in the middle and operations or methods at the bottom. In large systems with many classes related classes are grouped together to create class diagrams. Different relationships between diagrams are show by different types of Arrows. Below is a image of a class diagram. Follow the link for more class diagram examples.



Sequence Diagram

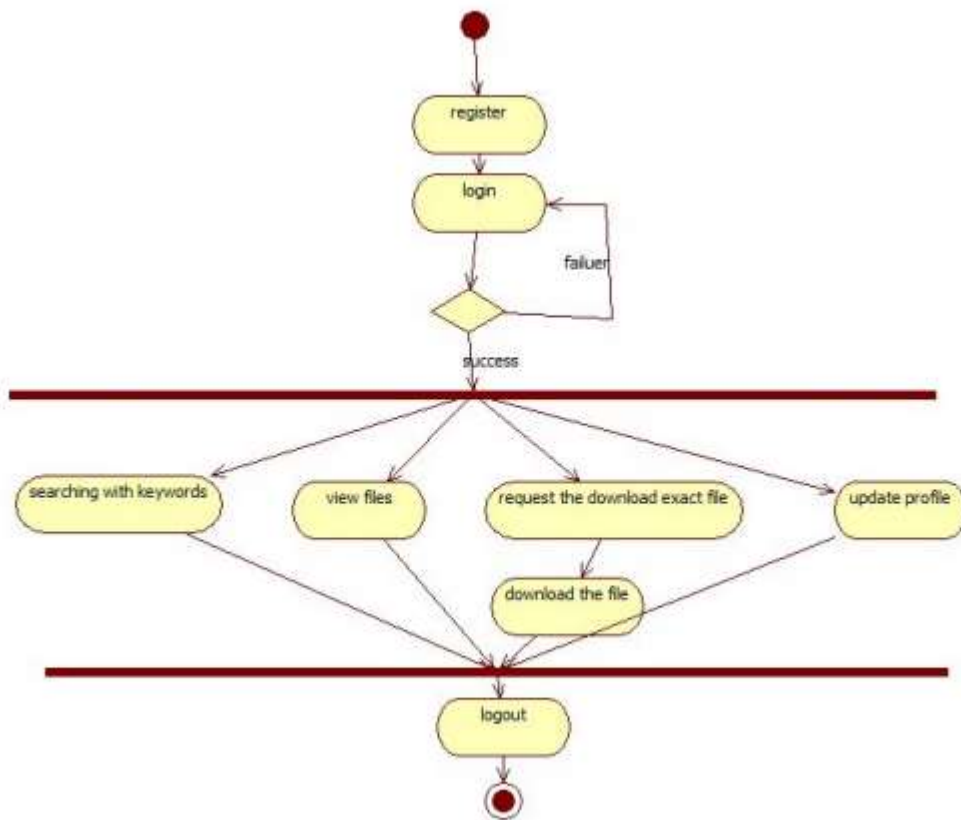
Sequence diagrams in UML show how objects interact with each other and the order those interactions occur. It's important to note that they show the interactions for a particular scenario. The processes are represented vertically and interactions are shown as arrows. This article explains the purpose and the basics of Sequence diagrams.



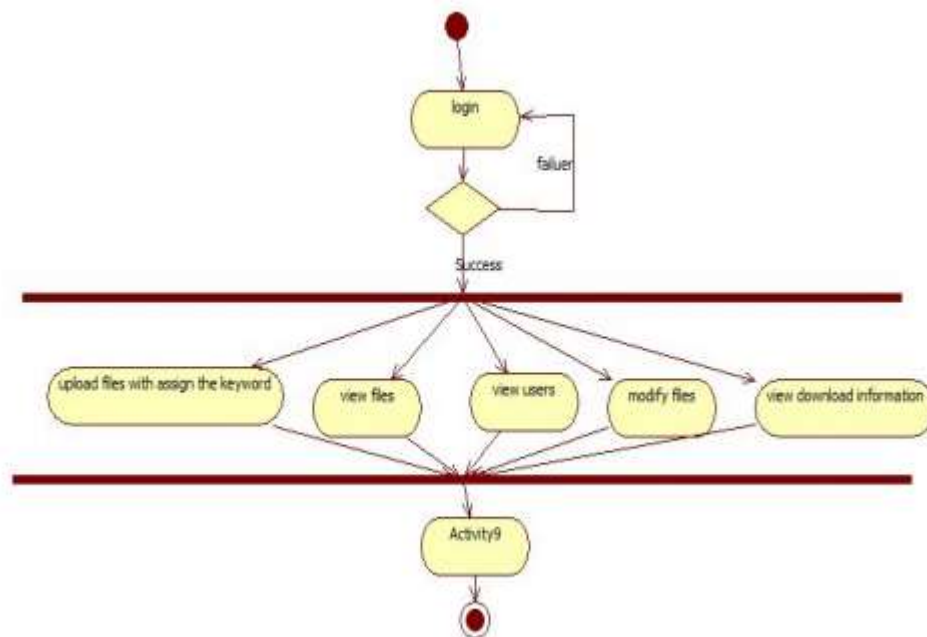
Activity diagram

Activity diagrams describe the workflow behavior of a system. Activity diagrams are similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel.

ACTIVITY DIAGRAM FOR USER



ACTIVITY DIAGRAM FOR ADMIN



REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses

the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

Robustness

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is

needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

Compatibility

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

Flexibility

The flexibility of the project is provided in such a way that it has the ability to run on different environments being executed by different users.

Safety

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

NON- FUNCTIONAL REQUIREMENTS

Portability

It is the usability of the same software in different environments. The project can be run in any operating system.

Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

Accuracy

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

CONCLUSION

- The management of power consumption in data centres has led to a number of substantial improvements in energy efficiency.

- Cloud computing infrastructure is housed in data centres and has benefited significantly from these advances.
- Techniques such as sleep scheduling and virtualization of computing resources in cloud computing data centres improve the energy efficiency of cloud computing

REFERENCE:

[1] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, "Identity-based encryption with outsourced revocation in cloud computing," *IEEE Transactions on Computers*, vol. 64, no. 2, pp. 425–437, 2015

- [2] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, "New publicly verifiable databases with efficient updates," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 5, pp. 546–556, 2015
- [3] H. Li, F. Zhang, J. He, and H. Tian, "A searchable symmetric encryption scheme using blockchain," *arXiv preprint, 2017*. [Online]. Available: <https://arxiv.org/pdf/1711.01030.pdf>
- [4] H. G. Do and W. K. Ng, "Blockchain based system for secured data storage with private keyword search," in *Services (SERVICES), 2017 IEEE World Congress on*. IEEE, 2017, pp. 90–93.
- [5] R. Guo, H. Shi, Q. Zhao, and D. Zheng, "Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems," *IEEE Access*, vol. 776, no. 99, pp. 1–12, 2018.