

1. *Introduction*

The main aim of the project “**Secure Email Service in Cloud by using Modified Homomorphic Encryption**” is to provide security in email service through the Cloud . Cloud computing is one in all the foremost emerged net based Technology that garnered a good attention of researchers from tutorial and trade . Cloud computing provides on demand Services over Network(SON) i.e., "Access services anytime from anyplace in pay-per-use fashion." a bent to any or all apprehend that the cloud or on-demand computing brings heaps of advantage to the computer science of those days and tomorrow. The adoption of cloud usage depends on security and protection that the cloud service produce ensure and also the manner a consumer can keep their personal information confidential Our basic construct was to encode the data before effort to the Cloud provider. But there is a haul still faced by the consumer. as a results of the Cloud provider should perform the calculations on data to retort the request from the consumer so he ought to provide the key to the server to rewrite the data before execute the calculations required, that might have an impression on the confidentiality of knowledge hold on inside the Cloud. how modify to perform the operations on encrypted data whereas not decrypted them is that the Homomorphic writing.

The Data transferred to the Cloud tend to use customary cryptography ways to secure this knowledge, when try to do the calculations on information placed on a far off server, it is necessary that the Cloud provider has access to the knowledge, so it will decipher them. throughout this paper tend to propose the applying of how to perform the operation on encrypted information whereas not decrypted and provide constant result likewise that the calculations were administrated on info whether or not or not you are running applications that share photos to a lot of mobile users or you're supporting the crucial operations of your business, a cloud services platform provides speedy access to versatile and low value IT resources. With cloud

computing, you don't ought to build large direct investments in hardware and pay an excellent deal of it slow on the work of managing that hardware. Instead, you may provision exactly the proper kind and size of computing resources you would like to power your newest bright arrange or operate your IT department.

You may access as many resources as you would like, nearly instantly, and only line up of what you utilize. Theory of Evolution.at the moment his student Rube Goldberg increased genetic rule within the year 1989. Genetic rule it's a tool to unravel numerous optimization issues like whole number non linear issues. it's oft utilized for locating higher best answer for all combination of issues.

Homomorphic Encryption

Homomorphic encryption is the conversion of data into ciphertext that can be analyzed and worked with as if it were still in its original form.

Homomorphic encryptions allow complex mathematical operations to be performed on encrypted data without compromising the encryption. In mathematics, homomorphic describes the transformation of one data set into another while preserving relationships between elements in both sets. The term is derived from the Greek words for "same structure." Because the data in a homomorphic encryption scheme retains the same structure, identical mathematical operations -- whether they are performed on encrypted or decrypted data -- will yield equivalent results.

Homomorphic encryption is expected to play an important part in cloud computing, allowing companies to store encrypted data in a public cloud and take advantage of the cloud provider's analytic services.

The development of the System rests on:

- A strategy where we architect, integrate and manage technology services and solutions.
- A robust development methodology which used for security in cloud

Speed:

They understand the importance of timing, getting details of necessary information as well as storing the information. A rich portfolio of reusable, modular frameworks helps jump start projects.

A full service portfolio:

Administrator to maintain the users, architect, integrate and manage through the system.

Services:

This system provides its services to users, which have common services like data privacy in cloud and message communications

KMMIPS is an educational institute established in 2001 by a group of retired civil servant. This educational institute which serves for the betterment of the students. Initially this college offers MCA course.

1.1 Organization Profile

KMMIPS is an educational institute established in 2001 by a group of retired civil servant. This educational institute which serves for the betterment of the students. Initially this college offers MCA course.

KMM Institute of Post Graduate Studies:

- Provides good higher-level educational services.
- Develop good communication skills to students.
- Provide good managerial skills MCA students.

“Sathyameva Jayathe” -Truth always Triumph is the motto of our KMM Social and Educational Development Society, Tirupati.

Nothing is permanent except change. The ongoing advances in computer communications technology continue to have profound effect on the way people work and play. Both the technology itself and the expectations of the people who use it are altering the features of the information system that analysis, design and the widespread deployment of information systems in

changing the very nature of the society in which the systems are used. The development of the information systems has played a dominant role in evolution of information economy.

KMM Institute of postgraduate studies very popularly known as KMMIPS has emerged as a major Technological Institute managed by KMM SOCIAL AND EDUCATIONAL DEVELOPMENT SOCIETY, Tirupati. The KMM society has taken the lead role to establish the institute in academic year 2001-2002. Sprawled over an area of 25 acres, permanent infrastructural facilities are being developed near Tirupati-Madanapalli state Highway at Ramireddipalle. KMMIPS has already secured the approval from the All India Council of Technical Education (AICTE), New Delhi, and Government of Andhra Pradesh and is affiliated to Sri Venkateswara University, Tirupati.

KMMIPS offers admission into the professional course MCA with annual intake of 180 seats. The institution is governed by the chairman Sri S.Srinivasulu garu, retired IRS (Indian Revenue Service) officer with the support of an advisory body consisting of the eminent personalities form different fields.

1.2 Project Report Layout

This project consists of eight chapters with an overview of A Secure Email Service in Cloud By Using Modified Homomorphic Encryption.

Chapter 1 Introduction deals with the overview of the organization and its administrative hierarchy.

Chapter 2 Genesis of the Study outlines the existing problem, the solution proposed, methodology used and looks at how the preliminary investigation is carried out, what are all its scope and objectives and limitations of existing manual system.

Chapter 3 Feasibility Study is used to test the feasibility of the project i.e., Operational, Technical and Economical feasibilities of the system.

Chapter 4 System Analysis deals with software requirement analysis. The various entities and their relationships are discussed using E-R Diagrams.

Chapter 5 System Requirements explains about the various hardware and software requirements and their features.

Chapter 6 System Design contains the description of all the tables and its attributes, design principles, HIPO charts and user interface design.

Chapter 7 System Testing presents software-testing strategies and techniques like white box testing, black box testing.

Chapter 8 Implementation gives the software and hardware implementation details of the system.

Chapter 9 Conclusion.

Appendix contains User Manual, Test Screens and Reports, Base Paper, Conference Paper, Journal Paper.

Bibliography gives detailed information of references i.e., books, sites, guides etc.

2. Genesis of Study

This chapter deals with the study of the existing system and describes the need for the proposed system to overcome the drawbacks in the existing system. It also specifies the objectives, scope of the system and also the methodology for the system development. Hence, the genesis of the study clearly depicts the factors regarding the beginning of the existing system and its extension to the proposed system and it includes the following.

2.1 Aim

The main aim of the project is to develop Security based email service in Cloud Computing. A good problem definition identifies the problem clearly, not an imposed solutions to the problem. The first step in planning a software project is to prepare a concise statement of the problem, which is to be solved, and the constraints that exists for its solution. homomorphic secret writing technique which is in security on cloud. Homomorphic secret writing may be a new thought of security that allows providing results of calculations on encrypted knowledge while not knowing the data on that the calculation was meted out, with respect of the information confidentiality. during this paper I actually have projected Paillier algorithmic program for homomorphic secret writing victimization proxy Re-encryption algorithmic program that forestalls cipher knowledge from Chosen Cipher text Attack (CCA). So this method is safer than existing system. In future will work efficiently of the system by reducing size of the key. Security of cloud computing supported Homomorphic secret writing may be a new thought of security that is change to produce the results of calculations on encrypted knowledge while not knowing the raw entries on that the calculation was meted out respecting the confidentiality of knowledge. Our work relies on the applying of Homomorphic secret writing to the protection of Cloud Computing

2.2 Problem Description

This Problem begins with analyzing the existing system practice and problems with regard to present situations. Further it discovers the potentials and significance in introducing computer system for effective maintenance. It's not uncommon for a unsuspecting employee to click on a link or download an attachment that they believe is harmless and loss data during transfer -- only to discover they've been infected with a nasty virus, or worse. As such, never click on a link that you weren't expecting or you don't know the origination of in an e-mail.

2.2.1 Description of Existing System

If emails aren't encrypted, sending confidential data is like telling hackers "please, come steal our information." We don't want that to happen to any business, but it happens all too often. Have a clear policy about what should and shouldn't be sent over email and ensure any confidential data is encrypted.

2.2.2 Drawbacks of Existing System

Some of the frequent occurring problems in the present manual system are as follows:

1. It is time consuming to do it manually and leads to errors while manual interventions.
2. Redundancy problem.
3. Less security.
4. Maintaining various books is difficult.
5. Difficulty in identifying mistakes or accounts tallying is difficulty.

In order to row out these challenges, a computerized system is needed to be developed.

2.2.3 LITERATURE SURVEY

MahaTebba et al. inspected the core application eventualities of various Homomorphic coding cryptosystems eg: (RSA, Paillier, El Gamal, Gentry etc.) on a Cloud Computing environment[1]. Further, comparison is being performed supported main four specialities "Homomorphic coding type", "Privacy of data", "Security applied to" and "the keys used". Reem Alattas et al[2]. introduced the applying of pure mathematics Homomorphic coding mechanism, supported Fermat's very little Theorem on cloud computing for higher security[3]. To fix the difficult drawback of knowledge privacy at the side of confidentiality within the cloud, totally Homomorphic Encryption (FHE) mechanism is Associate in Nursing explication, wherever the encrypted data is handled[4], and it returns the leads to encrypted manner. In spite of, totally homomorphic coding mechanism runs in relatively slower mode thence, the quicker totally homomorphic coding mechanisms square measure considerably required[5]. Gentry's projected coding mechanism is totally homomorphic however having impediment of slower performance. Lot of varied mechanisms are recommended in recent years to remarkably speed up the performance action of totally homomorphic coding schemes. Symmetric key Cryptography is shown[6]. Here for encryption, plain text is converted into cipher text, with use of secret key[7]. And at decryption time it using again same secret key to convert cipher text into plain text.

Bell laboratories discovered that all tamperproof devices of cryptosystems, which use public key cryptography for user authentication without special countermeasure[8], are at the risk of the occurrence of hardware faults [9]. For example, smart cards that are used for data storage, cards that personalize cellular phones, cards that generate digital signatures or authenticate users for remote login to corporate networks are all vulnerable to this attack[10]. The hardware fault attack is that the adversary induces some type of fault into the devices so that the system will have erroneous responses or produce faulty results[12]. Then the adversary is able to obtain the secret information of the system using the erroneous responses or results from the system[13]. The hardware fault attack of the cryptosystem is

composed of two steps[14]. The first step is to inject some fault into the system at appropriate time[15]. The second step is to exploit the erroneous responses or results to obtain the secret information of the cryptosystem[17]. The process of the fault-based attack[18]. The success of the hardware fault attack depends on whether the following three conditions are met or not [19], [20]: (i). The message to be signed is known to the attacker. (ii). A random fault occurs during the system calculations. (iii). The faulty results or erroneous responses are sent out of the system[21].

Guaranteeing that one or more of the above three conditions is not met is one way to protect the RSA devices against such attack[22]. Concerning the first condition, some countermeasures have been proposed to make sure the attacker has no access to the message to be signed[23].

2.4 Methodology

The Paillier theme, was fictional by Pascal Paillier in 1999 it's a probabilistic theme that's homomorphic with relevance addition (the add of 2 ciphertext is capable the ciphertext of the add of the 2 plaintext equivalents) and to multiplication by a relentless. Paillier could be a form of keypair-based cryptography. this suggests every user gets a public and a personal key, and messages encrypted with their public key will solely be decrypted with their personal key. Suppose E is that the paillier secret writing perform then we've the subsequent 2 properties :

$$E(a)+E(b) = E(a+b)$$

$$E(a)^b = E(a * b)$$

Key-Gen Algorithm

- 1) Choose two prime numbers p & q and calculate $n=p*q$ and $\lambda = \text{lcm}(p-1, q-1)$ such that $\text{gcd}(p*q, (p-1)*(q-1)) = 1$
- 2) Select $g \in \mathbb{Z}^{*n^2}$ and calculate $\mu = (L(g^\lambda \bmod n^2))^{(-1)} \bmod n$ where $L(x) = x-1/n$
- 3) n, g acts as a public key
- 4) λ, μ acts as a private key

Encryption Algorithm:

- 1) Let $m \in \mathbb{Z}_n$ be the message
- 2) Choose $r \in \mathbb{Z}^*_n$
- 3) Required Cipher text is $c = g^m * r^n \bmod n^2$

Decryption Algorithm:

- 1) Compute $m = L(c^\lambda \bmod n^2) * \mu \bmod n$

Example:

```
p= 17  q= 19
g= 45  r= 59
=====
Mu:           66      gLambda:      144
=====
Public key (n,g):           323 45
Private key (lambda,mu): 144 66
=====
Message:      10
Cipher:              336
Decrypted:     10
```

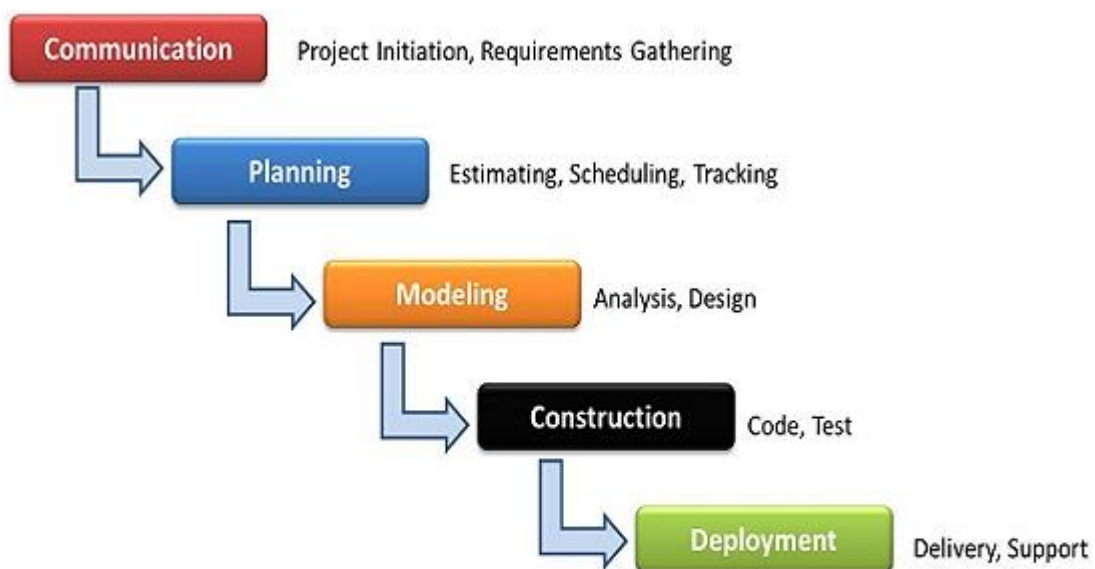
Paillier consists of 3 algorithms actually 3 algorithms are unit necessary to form associate secret writing theme. First you would like a Key generation algorithmic rule, second associate secret writing algorithmic rule and last a decipherment algorithmic rule let's see however Paillier implement those based on Paillier homomorphic encryption algorithm. I am developed this modified algorithm.

The source of information for developing the proposed system is gathered directly from clients of end user who is going to use the package becomes the primary source to give information.

A detailed study and understanding of the existing system is done either by questionnaires or by conducting interviews before developing the proposed one. Different inputs, process and output are well understood before designing the system.

Application software like PHP and MySQL Server is used for storing database information and it is used to construct the required code, so as to give the results as per the requirements of the system. The concepts of software engineering have been implemented successfully and uniformly throughout the system. The performance of the integrated system will be uniform.

For building this project, we followed Waterfall Model as the requirements of this project are completed analyzed at the beginning of the project itself.



The Waterfall Model: A Traditional Approach of SDLC

“The **Waterfall Model** was first Process Model to be introduced. It is very simple to understand and use. In a Waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases” The **waterfall model** is relatively linear sequential design approach.

2.5 Proposed System

The primary purpose of encryption is to protect the confidentiality of digital data stored on computer systems or transmitted via the internet or any other computer network. Developed this project I am using a modified encryption algorithm. By using this algorithm we can provide security in email. The Cloud email service I am using the public key for encrypting data meanwhile using private key for decrypting your data. Modern encryption algorithms also play a vital role in the security assurance of IT systems and communications as they can provide not only confidentiality

2.5.1 Objectives of Proposed system

- Using a Modified encryption algorithm we can do secure email service.
- Using the asymmetric method for developing this project
- We cant read original data until the key enter

2.5.2 Advantages of Proposed System

Encryption is now an important part of many communication services.

The proposed system successfully overcomes the drawbacks of the existing manual system. The system is proposed by taking the following aspects into consideration.

- Efficiency
- Extended security
- Maintainability
- Easy usability

Data security is another key area in which the new system takes special care and reserves valuable data from being corrupted.

Several reports can be generated in desirable format and the required information can be retrieved quickly.

Thus, the proposed system has provisions for faster and efficient computation. This saves enormous time and completely reduces the errors that are resulted in manual system.

Greater Processing Speed:

Using computers inherently able to calculate, sort, retrieve data with greater speed than that of the manual doing, we can get results in less time.

Better Accuracy and Improved Consistency:

The computer carries out computing steps including arithmetic's accurately and consistently from which really human is escaped which yields more fatigue and boredom.

Cost Reduction:

Using computerization, we can do the required operations with lower cost than any other methods. Hence by computerization, we can reduce the cost drastically.

2.5.3 Limitations of Proposed System

Encryption can protect the contents of an email message, but it can't hide who sent the message and who received it. That can be valuable information. Say that law enforcement officials are interested in a particular encrypted email that a suspect sent. If it can learn from the suspect's carrier who the recipient was, it might be able to seize that person's phone and read the message free of encryption. No muss and no fuss.

3. Feasibility Study

Generally the feasibility study is used for determining the resources, cost, benefits and whether the proposed system is feasible with respect to the organization or not. The feasibility of proposed **Secure Email Service Using by Modified Homomorphic Encryption** for email could be evaluated as follows. There are three types of feasibility which are equally important are:

- Operational feasibility
- Technical feasibility
- Economical feasibility

3.1. Operational Feasibility:

The operational feasibility is the willingness & ability of the users to run the developed system with out any difficulty. The present system has been developed in such a way in Windows and PHP the users can use the system with minimum knowledge to run the system and the users does require little training. The data entry can be done with out any difficulty because the screens has been developed as user friendly and by seeing the screens itself, he can get an overview of the process to be done. Any errors if arises at the time of the usage by the user can be debugged with ease by the user itself. Therefore the proposed system is Operationally Feasible.

3.2 Technical Feasibility:

Technical feasibility deals with the existing technology, software & hardware requirements for the proposed system. The proposed system “**Secure email service using by modified homomorphic encryption**” is planned to run on any Windows Platform with higher versions than Windows XP. From PHP can be used to deploy and run the system. The backend database is MySQL which is an open source and easily be managed with minimum knowledge of SQL. The work for the project can be done with current equipment, existing

software technology and with available personnel itself. Hence the proposed system is technically feasible.

3.3 Economical Feasibility:

This method is most frequently used for evaluating the effectiveness of a system. In this project “**Secure Email Service Using by Modified Homomorphic Encryption**”, the development of the system required PHP and above, MYSQL which are freely available and minimum memory. So the system can be developed and can run with the current equipment, with existing software technology. Since the required Hardware and software for developing the system is already available in the organization, it does not cost much for developing the proposed system. Thus, this project is economically feasible.

4. *System Analysis*

System analysis is an important activity that takes place when we are building a new system or changing existing one. Analysis helps to understand the existing system and the requirements necessary for building the new system. If there is no existing system, then analysis defines only the requirements.

One of the most important factors in system analysis is to understand the system and its problems. A good understanding of the system enables designer to identify and correct problems.

4.1 Entity-Relationship Diagrams

The Entity-Relationship Diagram depicts a relationship between data objects. The ERD is the notation that is used to conduct the data modeling activity. The attributes of each data object noted in the ERD can be described using a data object description.

At first a set of primary components are identified for ERD i.e. Data objects, Attributes, Relationships and Various type indicators. Data objects are represented by labeled rectangles. Relationships are indicated with labeled lines connecting objects.

Data modeling and the entity-relationship diagram provide the analyst with a concise notation for examining data with in the context of data processing application.

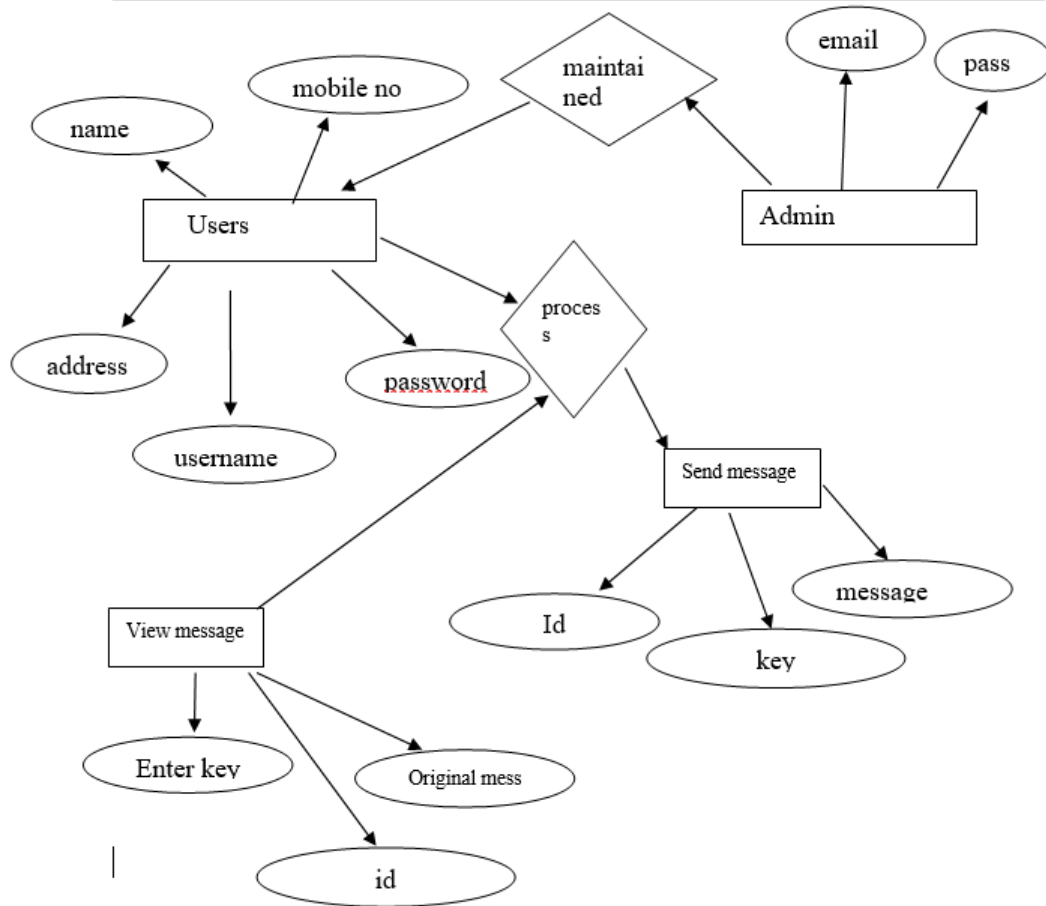


Fig4.1.1: E-R Diagram

The above diagram explains about the various database tables and the relation among those tables almost all the tables have one-to-many relation ship. By using the UserDB Database, we can select the information of each record from the database, Manpower Services will the precious time, money, energy and paper with flawless execution.

4.2 Data Flow Diagram (DFD)

A data flow diagram is a graphic description of the system or portion of a system. It is used to describe and analyze the movement of data through a system. Data flow diagrams are the center tool and the basic from which other components are developed.

The transaction of data from input to output, through process, may be described logically and independently of the physical components associated with the system. It consists of data flow process; sources, destination and stores all described through the use of easily understood symbols.

Physical Flow Diagrams:

There are implementation dependent views of current system showing what task is carried out how they are performed. These diagrams show the actual devices, departments, and people in system.

Logical Flow Diagrams:

As implementation-independent view of the system, focusing on the flow of data between processes without regard for the specific devices, storage location and people in the system.

At level 0 DFD, also called as the context diagram, represents the entire system as a single module with input and output data indicated by incoming and outgoing arrows respectively.

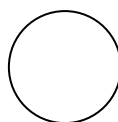
A level 1 DFD, also called as top-level DFD, represent the system with major modules and data stores. The other levels will show each module in the top-level DFD in a more detailed fashion.

Notations:

Data Flow: Data move in a specific direction from an origin to a destination. The data flow is a “packet” of data.



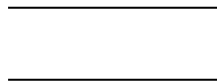
Process: People, procedures or devices that produce data. The physical component is not identified.



Source or Destination of data: External sources or destinations of data, which may be people or organizations or other entities.



Data Store: Here, the data referred by a process in the system.



Context Level DFD:

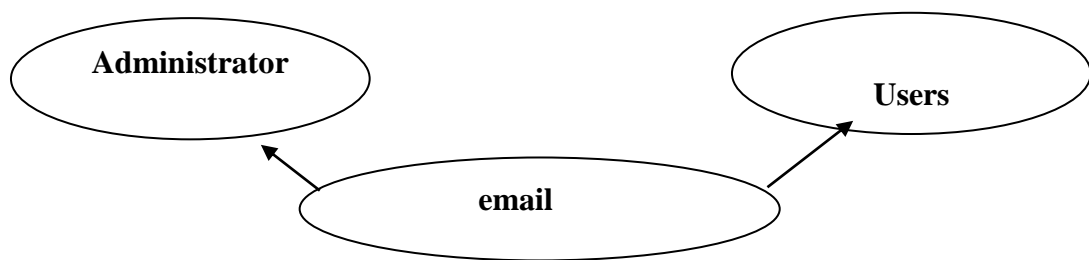


Figure 4.2.1: Context level Data Flow Diagram

1st Level DFD

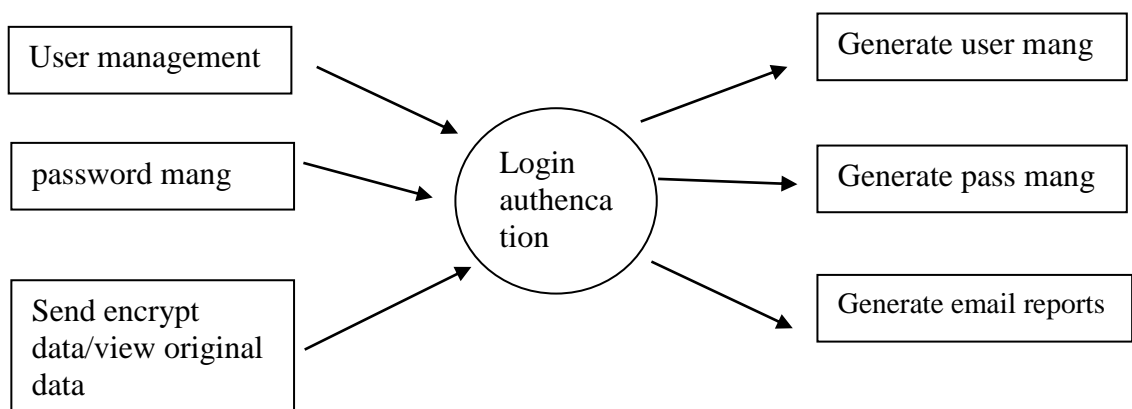


Figure 4.2.2: First level Data Flow Diagrams

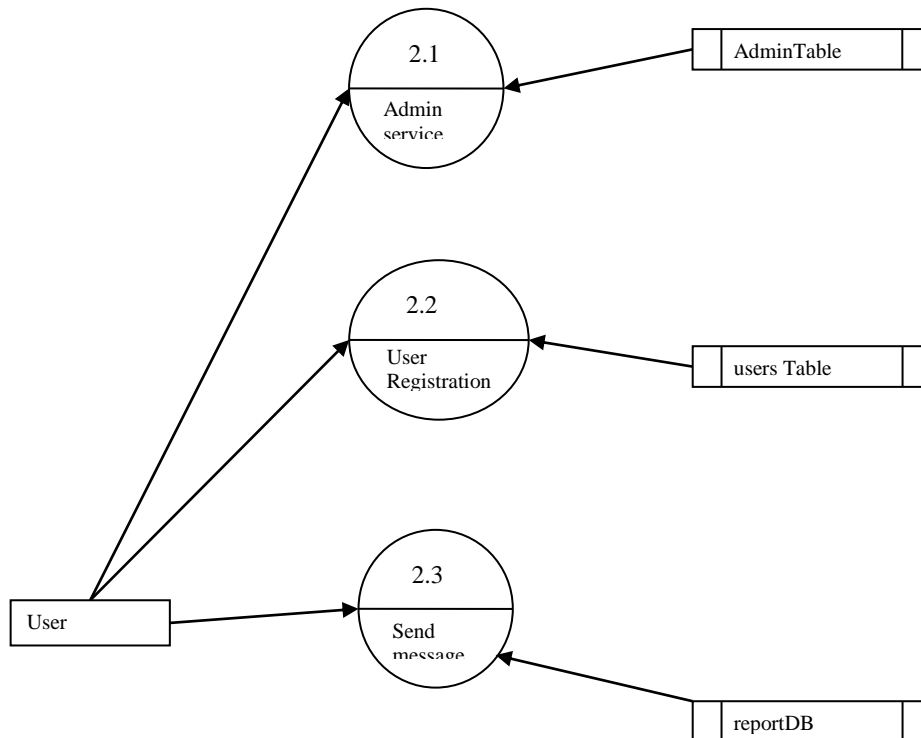


Figure 4.2.3: Second level Data Flow Diagram

4.3 Use case Diagrams

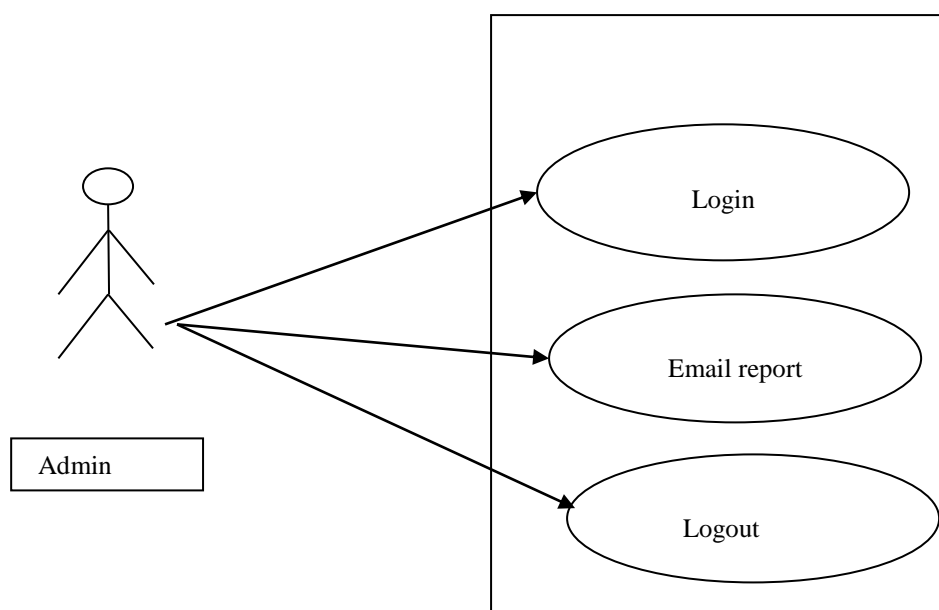


Figure 4.3.1: Use case diagram for Administrator

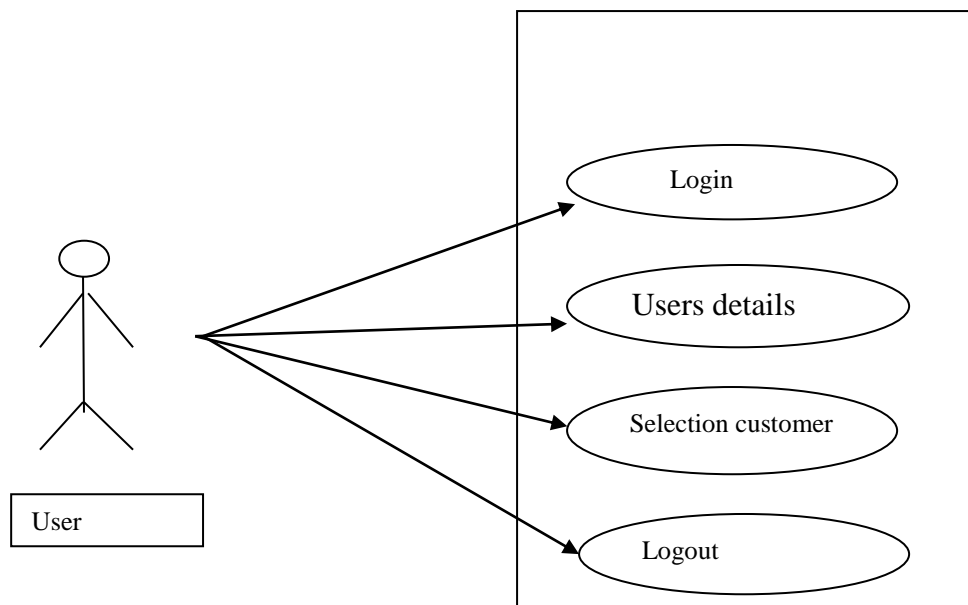


Figure 4.3.2: Use case diagram for User

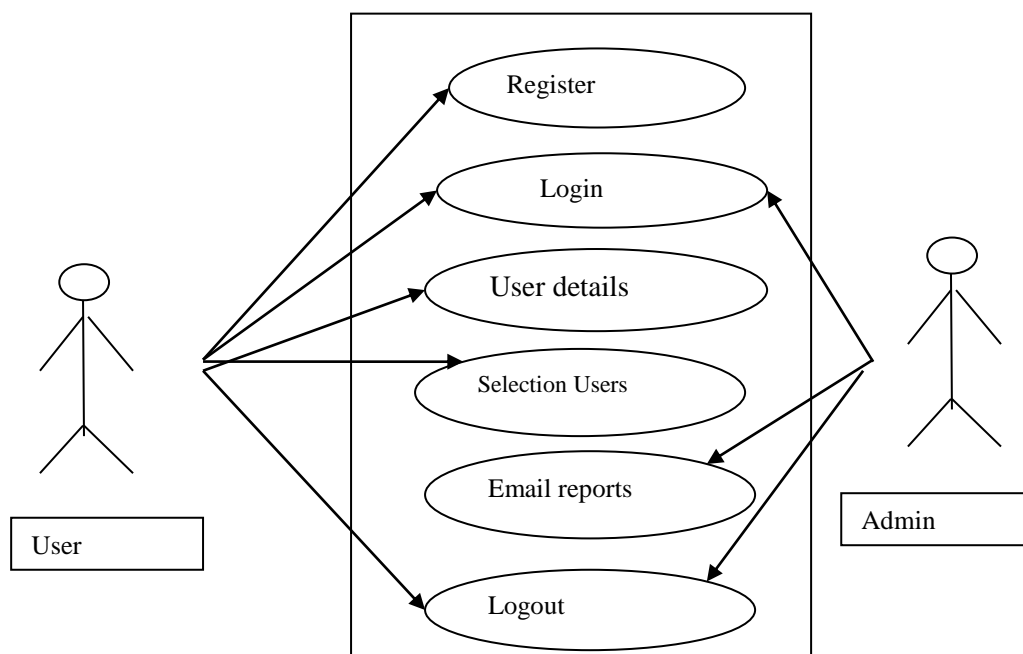


Figure 4.3.2: Use case diagram for User/Administrator

A use diagram is a graph of actors, a set of use cases enclosed by a system boundary, communication (participation) associations between the actors and the use cases, and generalization among the use cases.

A use case is shown as an ellipse containing the name of the use case. The name of the use case is placed below or inside the ellipse. Actors' names are use case names should follow the capitalization and punctuation guidelines of the model.

An actor is shown as a class rectangle with the label <<actor>>, or the label and a stick figure with the name of the actor below the figure. In the use case diagram they are the two actors like User and the Administrator.

The relationships are shown in a use case diagram:

1. Communication: The communication relationship of an actor in a use case is shown by connecting actor symbol to the use case symbol with a solid path. The actor is said to “communicate” with the use case.
2. Uses: A uses relationship between use cases is shown by a generalization arrow from the use case.
3. Extends: The extends relationship is used when you have one use case that is similar to another use case but does a bit more. In essence, it is like a subclass.

4.4 Activity Diagram

Activity diagram is an important diagram in UML to describe the dynamic aspects of the system. 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.

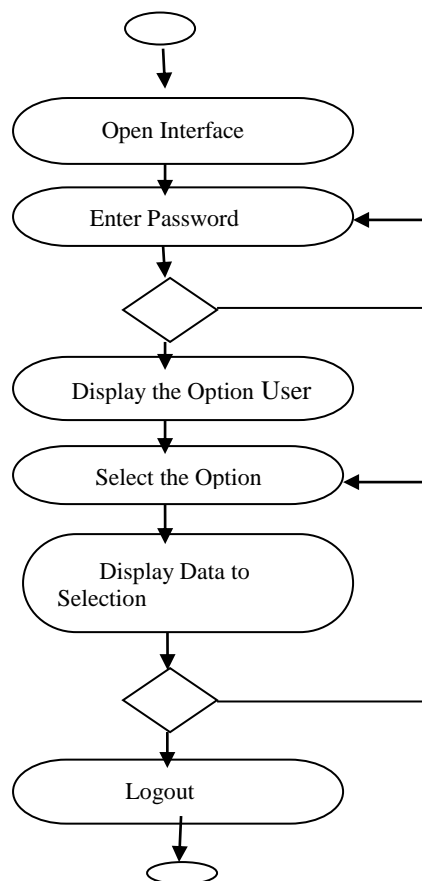


Figure 4.4. Activity diagram

An activity diagram is a variation or special case of a state machine, in which the states are activities representing the performance of operations and the transitions are triggered by the completion of the operations. The purpose of an activity diagram is to provide a view of flows and what is going on inside a use case or among the classes. However, activity diagram can also be used to represent a class's method implementation.

In this activity diagram first the Supplier/Admin opens the interface, and enters the password. If it is valid go to the display options phase. If it is not valid reenter the correct password. In that display options select the option and do the work that option for example labour information etc. then appropriate screens will be displayed, if you want to more go to select options statements other wise logout the system.

4.5 Data Dictionary:

A Data dictionary as the name implies, is a repository of information about data. In some database systems, the stored definitions of data (called schemas) provide all necessary data dictionary information;

In others, the data dictionary is supplementary. The information in the data dictionary is about types of data and uses of data.

Data dictionary is used:

- ◆ To manage the details in large system.
- ◆ To communicate a common meaning for all system developers.
- ◆ To document the features of the system.
- ◆ To facilitate analysis of the details in order to evaluate characteristics and determine where system changes to be made.

Component may be a data field or user variable used in the program.

SL.NO.	Component Name	Description
1	Id	To Identify users
2	User full name	Name to identify user
3	User Addresss	Address of the user
4	Email	To identify User uniquely
5	Username	Name of the user
8	DateOfBirth	Data Of Birth of the User
9	Gender	Gender of the User
10	Mobilenumber	Mobile number of the User
11	Address	User current address
12	Password	Security for details
13	Adminusername	To identified Admin Uniquely
14	Adminpassword	Security to Admin details

Secure Email Service in Cloud By Using Modified Homomorphic Encryption

15	Usermail	Mail address to a customer
16	Key	Store the keys
17	Message	Text for user data

5. *System Requirements*

System requirements gives the idea about what are the necessary things that are needed for proposed system, which plays very important role in development of this project. This chapter deals with what are hardware components that are needed for the system, application software that are required for the development of the system.

The environment deals with the features of software PHP is used as the server side scripting language and MY SQL as a backend. Front end tools help to visualize the system through naked eyes while back end helps in activities which are unseen to the end user.

5.1: Hardware Requirements

Processor	: Intel Pentium III or Higher
RAM	: 4 GB Minimum
Hard Disk	: 250 GB Minimum

5.2: Software Requirements

Operating System	: Windows 7 or higher
Front-End	: PHP
Back-End	: MySQL Server

5.2.1 Features in PHP

1. Simple

It is very simple and easy to use, compare to other scripting language it is very simple and easy, this is widely used all over the world.

2. Interpreted

It is an interpreted language, i.e. there is no need for compilation.

3. Faster

It is faster than other scripting language e.g. asp and jsp.

4. Open Source

Open source means you no need to pay for use php, you can free download and use.

5. Platform Independent

PHP code will be run on every platform, Linux, Unix, Mac OS X, Windows.

6. Case Sensitive

PHP is case sensitive scripting language at time of variable declaration. In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive.

7. Error Reporting

PHP have some predefined error reporting constants to generate a warning or error notice.

8. Real-Time Access Monitoring

PHP provides access logging by creating the summary of recent accesses for the user.

9. Loosely Typed Language

PHP supports variable usage without declaring its data type. It will be taken at the time of the execution based on the type of data it has on its value.

MySQL Server

MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications.

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB,

which is a Swedish company. MySQL is becoming so popular because of many good reasons –

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to JSP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

MySQL Enterprise Transparent Data Encryption (TDE)

MySQL Enterprise Transparent Data Encryption (TDE) enables data-at-rest encryption by encrypting the physical files of the database. Data is encrypted automatically, in real time, prior to writing to storage and decrypted when read from storage.

MySQL Enterprise Backup

MySQL Enterprise Backup reduces the risk of data loss by delivering online "Hot" backups of your databases. It supports full, incremental and partial backups, Point-in-Time Recovery and backup compression.

MySQL Enterprise High Availability

MySQL InnoDB Cluster delivers an integrated, native, HA solution for your databases. It tightly integrates MySQL Server with Group Replication, MySQL Router, and MySQL Shell, so you don't have to rely on external tools, scripts or other components.

MySQL Workbench

It is a unified visual tool for database architects, developers, and DBAs. It provides data modeling, SQL development, database migration and comprehensive administration tools for server configuration, user administration, and much more.

6. *System Design*

6.1 Introduction

Design is the first step in the development phase for any system. It may be defined as the “process of applying various techniques and principles for the purpose of designing a device, a process, or a system”.

Software design is an iterative process through which requirements are translated into a ‘Blue Print’ for constructing the software. Preliminary design is concerned with the transformation of requirements into data and software architecture.

The design is a solution, a “how to” approach to the creation of a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended.

The **database design** transforms the information domain model created during analysis into the data structures that will be required to implement software.

The **architectural design** defines the relationship among major structural elements of the program

The **interface design** describes how the software communicates within itself, to systems that interoperate with it, and with humans who use it. An interface implements flow of information.

6.2 Design Principles

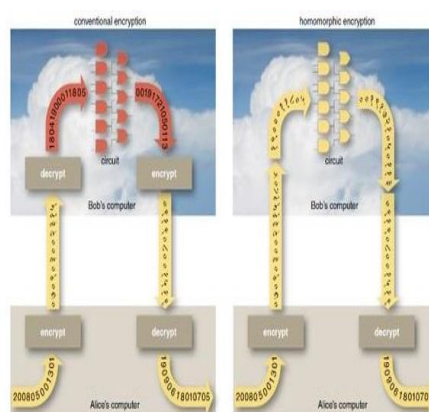
Basic design principles that enable the software engineer to navigate the design process are:

- The design should be traceable to the analysis model.
- The design should minimize the intellectual distance between the software and the problem, as it exists in the real world.
- The design should exhibit uniformity and integrity.
- The design should be structured to accommodate changes.
- The design is not coding, the coding is not a design.
- The design should be reviewed to minimize the conceptual errors.

6.3 Design Algorithm

Homomorphic cryptography systems square measure wont to perform operations on encrypted information while not knowing the key (without decrypted), the shopper is that the solely individual of the key. after we rewrite the results of the operation, it's a similar as if we have a tendency to had distributed the calculation on the information.

This algorithm is asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. **Public Key** and **Private Key**. As the name describes that the Public Key is given to everyone and



Private key is kept private.

Modified Homomorphic Encryption

Key Generation:

Step1: Generate prime number using sieve of Eratosthenes (let P and Q)

Step1.1: for all number $m=1,2,\dots,n$ if m is unmarked

Step1.2: add m to prime list

Step1.3: mark all its multiple, lesser or equal

Than $n(k*m \leq n, k \leq 2)$;

Step1.4: otherwise if m is marked then it is a composite number

Step2: p and q values from m.

Step3: find $\text{GCD}(pq, (p-1)(q-1))=1$

Step3.1: using euclidian algorithm

Step3.2: if $p < q$ exchange p and q

Step3.3: Divide a by q and get remainder r if $r=0$

Report q as the GCD

Step3.4: replace p and q and replace b by r

Step4: $n=pq$ and $\lambda = \text{lcm}(p-1)(q-1)$

Step5: g is $n+1$

Step6: $\mu = \lambda^{-1} \bmod n$

Now

Public key (n, g)

Private key (λ, μ)

Encryption Data

Step1: let m is message and r is a random number $m < n$ and $r < n$

Step2: $c = g^m \cdot r^n \bmod n^2$

Decryption Data

Step1: $m = (c^\lambda \bmod n^2)^\mu \bmod n$

6.4 Database Design

The goal of database design is to generate a set of relation schemes that allow us to store information without necessary redundancy and allows us to retrieve information easily. We can achieve optimization, ease of use in which data is stored in the form of tables and there exists a relation between or among tables.

The design objectives must be:

- To reduce redundancy.
- To arrive at loss-less join.
- To reduce the time as compared to the present system.
- To reduce the number of errors.

6.4.1 Normalization

Normalization of relation schema is done to eliminate insertion and deletion anomalies that exist in databases. Normalization is a step-by-step reversible process of converting given collection of relations to some desirable form in which the relations have a progressively simpler and regular structure.

The objectives of Normalization are:

- To make it feasible to represent any relation in the database.
- To obtain powerful retrieval algorithms based on a simpler collection of relational operations.
- To free relations from undesirable insertions, update and deletion dependencies.

A relation R is said to be in 1NF if all underlying domains contain atomic values only.

A relation R is said to be in 2NF if and only if it is in 1NF and every non-key attribute is non-transitively dependent on the primary key.

A relation R is said to be in 3NF if it is in 2NF and its non-key attribute is non-transitively dependent on its primary key.

All the tables that have been designed for developing this system follow 2nd Normalization form.

6.4.2 Database Tables

Table 1: **Table Name: Register**

Field Name	Data Type	Size	Constraints
Id	Int	10	Primary key
Email	username	30	Not Null
Password	Varchar	100	Not Null
Dateofbirth	Date	10	Not Null
Firstname	Varchar	20	Not Null
Lastname	Varchar	20	
Mobileno	Int	14	Not Null
Gender	Varchar	20	Not Null
Address	Varchar	40	Not Null

Table 2 **Table name: Sent**

Field Name	Data Type	Size	Constraints
Id	Int	10	Primary key
Email	Varchar	100	Not null
Sender	Varchar	100	Not null
Message	Varchar	150	Not Null
Key	Varchar	255	Not Null

Table3 **Table Name: Admin**

Field Name	Data Type	Size	Constraints
Adminid	Varchar	30	Not null
Password	Varchar	30	Not Null

Table4 **Table name: key Generations**

Field Name	Data Type	Size	Constraints
------------	-----------	------	-------------

Id	Int	10	Not Null
Public key	Varchar	255	Not Null
Private key	Varchar	255	Not null
Secret key	Varchar	255	Not null

6.5 Module Design

Manpower Services is designed to develop as product with following architecture and components.

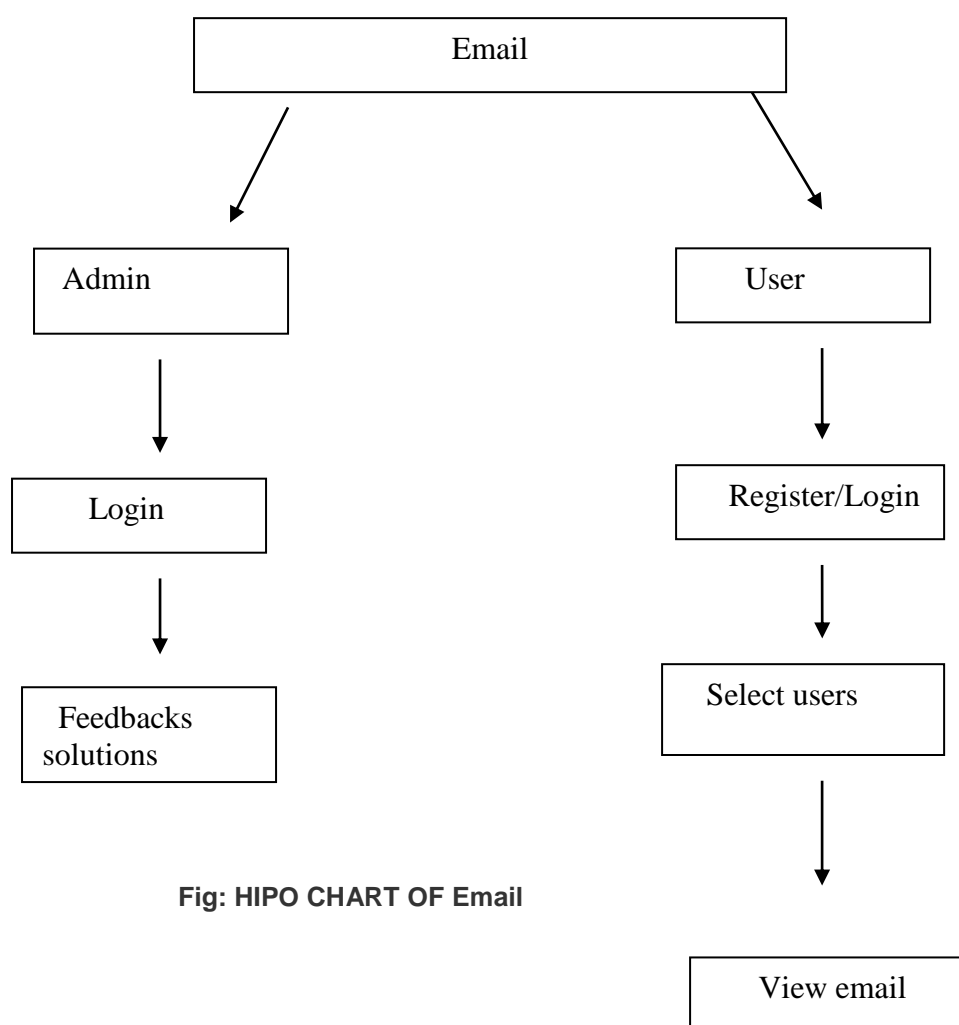


Fig: HIPO CHART OF Email

The project Manpower Services has divided into following modules

1. User Details Module
2. User Selection Module
3. Send/View module

Modules are designed keeping in view of developing a product based system which can be used with any organization having functionality similar to the User services.

User details:

It contains User information. This application provides all information about User details like User name, User number and location ,User address etc.

User Selection:

It contains a input field throw the using values (eg., eid ,Name, Address) we can select the needed User.

Send/View module:

In this module user can send message meanwhile we will get message to view ,Here we can use some keys for providing security

All the above modules are operated by Admin, Supplier who are password protected/Authenticated strictly. System usage is allowed only when the users are logged successfully. Important transactions are stored in Database securely along with person currently logged in information.

7. *System Testing*

Introduction:

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

Psychology of Testing

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Testing is the process of executing a program with the intent of finding errors.

Testing Objectives:

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time.

- Testing is a process of executing a program with the intent of finding an error.
- A successful test is one that uncovers an as yet undiscovered error.
- A good test case is one that has a high probability of finding error, if it exists.
- The tests are inadequate to detect possibly present errors.
- The software more or less confirms to the quality and reliable standards.

Testing case design:

A rich variety of test case design methods have evolved for software. These methods provide the developer with a systematic approach to testing. More important, methods provide a mechanism that can help to ensure the

completeness of tests and provide the highest likely hood of uncovering errors in software.

Any engineered product can be tested in one of the two ways:

1. Knowing the specified function that a product has been designed to Perform.
2. Knowing the internal workings of a product.

Testing case design:

Step 1: Select primes $p=11$, $q=13$.

Step 2: $n = pq = 143$ and $g=n+1=144$

Step 3: $\mu=42$ and $\mu=\lambda^{-1} \bmod n$

Step 5: Public key = $(n, g) = (143, 144)$

Private key = $(\lambda, \mu) = (120, 42)$.

Step 4: $c=g^m \cdot r^n \bmod n^2$

$$=144^{42} \cdot 23^{143} \bmod 143^2$$

$$=9637$$

This is actually the smallest possible value for the modulus n for which this Algorithm works. Now say we want to encrypt the message $m = 47$,

Hence the cipher text $c = 9637$

Step 6: To check decryption we compute $\mu=L(9637^{120} \bmod 143^2) \cdot (120^{-1} \bmod 143) \bmod 143$ $m=42$.

A rich variety of test case design methods have evolved for software. These methods provide the developer with a systematic approach to testing. More important, methods provide a mechanism that can help to ensure the completeness of tests and provide the highest likely hood of uncovering errors in software.

Any engineered product can be tested in one of the two ways:

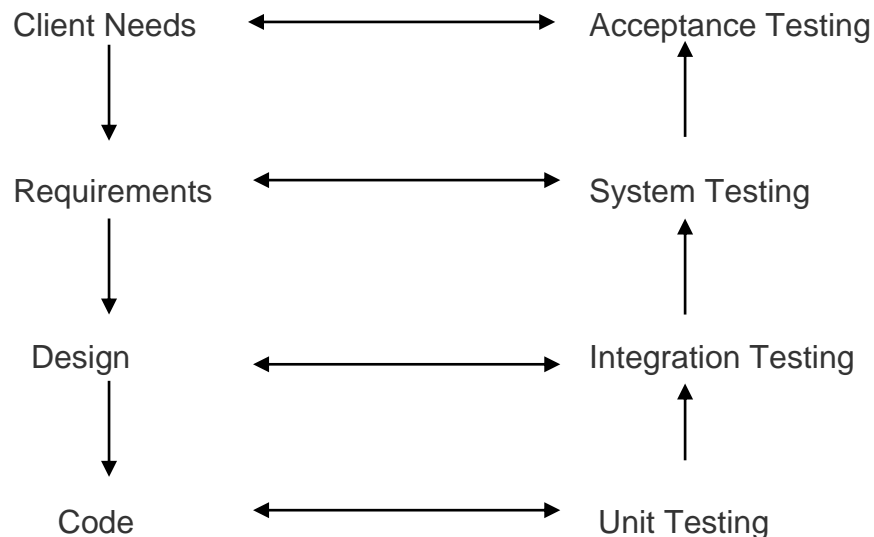
1. Knowing the specified function that a product has been designed to

Perform.

2. Knowing the internal workings of a product.

Levels of Testing:

In order to uncover the errors present in different phases we have the concept of levels of testing. The basic levels of testing are



7.1 White Box Testing

This is a unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors. We tested step wise every piece of code, taking care that every statement in the code is executed at least once. The white box testing is also called Glass Box Testing.

We have generated a list of test cases, sample data which is used to check all possible combinations of execution paths through the code at every module level.

7.1.1 Cyclomatic Complexity:

It is Software metric that provides a quantitative measure of the logical complexity of a program.

7.2 Black Box Testing

This testing method focuses on the functional requirements of the software. Here each module will be treated as a black box that will take some input and generate output. Output for a given set of input combinations are forwarded to other modules.

Black box testing attempts to find the following types of errors

- Incorrect or missing functions.
- Interface errors.
- Errors in data structures or external database access.
- Performance errors
- Initialization errors and termination errors.

All the forms have been executed and verified .Based on some sample input data, the generated output is verified whether the system is providing better results or not.

7.2.1. Boundary value Analysis:

A greater number of errors occur at the boundaries of the input domain. The BVA has been developed as a testing technique. To test BVA, we take one form and apply the conditions

DATE:

Test case: - Date will be selected from the user and if the user select any invalid date then it should display message like “Select Valid Date”.

PERIODS TAKEN:

Test case: - The checkboxes will be in enable state if the attendance has not been entered already, else the checkboxes will be in disabled state.

7.3 Unit testing:

Unit testing focuses on the verification of code produced during coding phase and hence the goal is to test the interface in order to ensure that information property flows in and out of the program under unit test.

7.4 Integration Testing:

The goal here is to see whether the modules are integrated properly, the emphasis being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions.

In this project the main system is formed by integrating all the modules. When integrating all the modules, we have checked whether the integration effects working of any of the services by giving different Combinations of inputs with which the two services run perfectly before Integration.

7.5 Validation Testing:

Validation testing demonstrates tractability of software requirements. Validation succeeds when the software functions in expected manner. The major elements of this process are alpha and beta testing along with configuration reviews.

In the present system, validations are been written for email, password and proper error message's are displayed when any validation error occurs. Validation's such as a Text Field should accept only Character data but not any other Characters.

7.6 Client Server Testing:

In general, the testing of Client/Server software occurs at three different levels:

- Individual client applications are connected a “disconnected mod”, the operation of the server and the underlying network are not considered.

- The client software and associated server applications are tested in concert, but network operations are not explicitly exercised.
- The complete C/S architecture, including network operation and performance, is tested.

The following are the testing approaches for C/S applications.

Application Function Tests:

Here the functionality of client applications is tested. In essence, the application is tested in stand-alone fashion in an attempt to uncover errors in its operation.

Server Tests:

The coordination and data management functions of the server are tested. Server performance is also considered.

Database Tests:

The accuracy and integrity of data stored by the server is tested. Transactions posted by client applications are examined to ensure that data are properly stored, updated, and retrieved.

Transaction Tests:

A series of tests are created to ensure that each class of transactions is processed according to requirements.

Network communication Tests:

These tests verify that communication among the nodes of the network occurs correctly and that message passing, transactions, and related network traffic occur without error.

Client Server testing the major test which has been performed for the proposed system. All the tests, which were mentioned above, are done for the proposed system.

7.7 Test Cases for Login Form

Test cases should be checked for both positive and Negative test cases of the Login page as below:

- **Positive Test Case**

1) Verify the Correct username, Correct password - Login Successfully.

- **Negative Test Cases**

1) Verify the Incorrect username, incorrect password- Can't Login

2) Verify the Incorrect username, incorrect password- Can't Login

3) Verify valid username and empty password. -Can't Login

4) Verify empty username and valid password. - Can't Login

5) Verify some password(can be a registered/unregistered)- Can't Login

6) Verify case changed username /password.- Can't Login

7) Verify registered user's login id and password -Can't Login

8) Verify registered username and password.- Can't Login

9) Verify to enter disable(Blocked) email address.- Can't Login

10) Verify to unverified Email address. - Can't Login

Test Scenarios of a User Form:

1. Verify that the User form contains id , email, Address, key Details.
2. Verify that tab functionality is working properly or not
3. Verify that Enter/Tab key works as a substitute for the Submit button
4. Verify that all the fields such as Username, First Name, Last Name, Password and other fields have a valid placeholder
5. Verify that the labels float upward when the text field is in focus or filled (In case of floating label)

6. Verify that all the required/mandatory fields are marked with * against the field
7. Verify that clicking on submit button after entering all the mandatory fields, submits the data to the server
8. Verify that system generates a validation message when clicking on submit button without filling all the mandatory fields.
9. Verify that entering blank spaces on mandatory fields lead to validation error
10. Verify that clicking on submit button by leaving optional fields, submits the data to the server without any validation error
11. Verify that case sensitivity of User name (usually Username field should not follow case sensitivity – 'saikrishna' & 'SAIKRISHNA' acts same)
12. Verify that system generates a validation message when entering existing username
13. Verify that the character limit in all the fields (mainly username and password) based on business requirement
14. Verify that the username validation as per business requirement (in some application, username should not allow numeric and special characters)
15. Verify that the validation of all the fields are as per business requirement
16. Verify that the validation of phone no by entering incorrect values other than phone nos.
17. Verify that the validation of numeric fields by entering alphabets and characters
18. Verify that leading and trailing spaces are trimmed after clicking on submit button
19. Verify that the password is in encrypted form when entered
20. Verify whether the password and confirm password are same or not

8. Implementation

Implementation is the process of converting a new or revised system design into an operational one. Apart from planning, the major tasks of preparing for implementation or education and training of users. Implementation includes following activities:

- Obtaining and installing the system hardware
- Providing user access to the system
- Creating and updating the database
- Training the users on the new system
- Documenting the system for its users
- Evaluating the operation and use of the system

Implementation Methods

There are four basic methods of implementation:

- ✓ Direct conversion
- ✓ Parallel conversion
- ✓ Pilot conversion
- ✓ Phasing conversion

Direct Conversion:

Description:

In this method the new one replaces the old system. This makes organization to fully rely on the new system.

Advantages:

This method forces users to make the new system work. There are immediate benefits from new methods and controls.

Disadvantages:

There is no other system to fall back on if difficulties arise with new system.
This method also requires most careful planning.

Algorithm Implementation:

//html code

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>

<head>
<title>Lively Chat a Corporate Category Bootstrap Responsive Web
Template | Sign Up :: W3layouts </title>
<!--/tags -->
<script type="application/x-javascript">
        addEventListener("load", function () {
            setTimeout(hideURLbar, 0);
        }, false);

        function hideURLbar() {
            window.scrollTo(0, 1);
        }
    </script>
    <!--//tags -->
<link href="css/bootstrap.css" rel="stylesheet" type="text/css"
media="all" />
<link href="css/style.css" rel="stylesheet" type="text/css"
media="all" />    <link href="css/font-awesome.css"
rel="stylesheet">
        <!-- //for bootstrap working -->
<link
href="//fonts.googleapis.com/css?family=Raleway:300,300i,400,400i,50
0,500i,600,600i,700,700i,800" rel="stylesheet">
<link
href='//fonts.googleapis.com/css?family=Lato:400,100,100italic,300,3
00italic,400italic,700,900,900italic,700italic' rel='stylesheet'
```



```
type='text/css'><link
href="//fonts.googleapis.com/css?family=Source+Sans+Pro:300,300i,400
,400i,600,600i,700" rel="stylesheet">
<script language="JavaScript" type="text/javascript"
src="code/jsbn.js"></script>
<script language="JavaScript" type="text/javascript"
src="code/jsbn2.js"></script>
<script language="JavaScript" type="text/javascript"
src="code/prng4.js"></script>
<script language="JavaScript" type="text/javascript"
src="code/rng.js"></script>
<script language="JavaScript" type="text/javascript"
src="code/rsa.js"></script>
<script language="JavaScript" type="text/javascript"
src="code/rsa2.js"></script>
<script language="JavaScript">
varn="a5261939975948bb7a58dffe5ff54e65f0498f9175f5a09288810b8975871e
99\naf3b5dd94057b0fc07535f5f97444504fa35169d461d0d30cf0192e307727c06
\n5168c788771c561a9400fb49175e9e6aa4e23fe11af69e9412dd23b0cb6684c4\n
c2429bce139e848ab26d0829073351f4acd36074eafd036a5eb83359d2a698d3";
vare="10001";vard="8e9912f6d3645894e8d38cb58c0db81ff516cf4c7e5a14c7f
1eddb1459d2cded\n4d8d293fc97aee6aefb861859c8b6a3d1dfe710463e1f9ddc72
048c09751971c\n4a580aa51eb523357a3cc48d31cfad1d4a165066ed92d4748fb65
71211da5cb1\n4bc11b6e2df7c1a559e6d5ac1cd5c94703a22891464fba23d0d9650
86277a161";varp="d090ce58a92c75233a6486cb0a9209bf3583b64f540c76f5294
bb97d285eed33\naec220bde14b2417951178ac152ceab6da7090905b478195498b3
52048f15e7d";varq="cab575dc652bb66df15a0359609d51d1db184750c00c6698b
90ef3465c996551\n03edbf0d54c56aec0ce3c4d22592338092a126a0cc49f65a4a3
0d222b411e58f";
var
dmp1="1a24bca8e273df2f0e47c199bbf678604e7df7215480c77c8db39f49b000ce
2c\nf7500038acfff5433b7d582a01f1826e6f4d42e1c57f5e1fef7b12aabc59fd25
";
var
dmq1="3d06982efbbe47339e1f6d36b1216b8a741d410b0c662f54f7118b27b9a4ec
9d\n914337eb39841d8666f3034408cf94f5b62f11c402fc994fe15a05493150d9fd
";
var
coeff="3a3e731acd8960b7ff9eb81a7ff93bd1cfa74cbd56987db58b4594fb09c09
```

```
084\ndb1734c8143f98b602b981aaa9243ca28deb69b5b280ee8dcee0fd2625e53250";
```

```
function do_encrypt() {
    var before = new Date();
    var rsa = new RSAKey();
    rsa.setPublic(n,e);
    var res = rsa.encrypt(document.rsatest.t2.value);
    var after = new Date();
    if(res) {
        document.rsatest.t2.value = linebrk(res, 64);
        do_status("Encryption Time: " + (after - before) + "ms");
    }
}
</script>
</head>

<body>
    <!-- header -->
    <div class="header" id="home">
        <div class="top_menu_w3layouts">

<div class="header_left">
            <ul>
                <li><i class="fa fa-map-marker"
aria-hidden="true"></i> 1143 New York, USA</li>
                <li><i class="fa fa-phone" aria-
hidden="true"></i> +(010) 221 918 811</li>
                <li><i class="fa fa-envelope-o"
aria-hidden="true"></i> <a
href="mailto:info@example.com">info@example.com</a></li>
            </ul>
        </div>

        <div class="clearfix"> </div>
    </div>

    <div class="content white">
        <nav class="navbar navbar-default"
role="navigation">
```

```

<div class="container">
    <div class="navbar-header">
        <button type="button"
class="navbar-toggle" data-toggle="collapse" data-target="#bs-
example-navbar-collapse-1">
            <span class="sr-only">Toggle
navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand"
href="index.html">
            <h1><span class="fa
fa-comments-o" aria-hidden="true"></span>Lively Smail
<label>For Customer Support</label></h1>
            </a>
        </div>
        <!--/.navbar-header-->
        <div class="collapse navbar-
collapse" id="bs-example-navbar-collapse-1">
            <nav>
                <ul class="nav
navbar-nav">
                    <li><a
href="index.html">Home</a></li>
                    <li><a
href="login_action.php">Send Mail</a>
                    </li>
                    <li><a
href="mail.php">View Mail's</a>
                    </li>
                    <li><a
href="/mail/demo/key_gen.php">Key Generation</a></li>
                    <li><a
href="blog.html">Blog</a></li>
                    <li><a
href="mail.html">Mail Us</a></li>
                </ul>
            </nav>

```

```

        </div>
        <!--/.navbar-collapse-->
        <!--/.navbar-->
    </div>
</nav>
</div>
</div>
<!-- banner -->
<div class="banner_inner_content_agile_w3l">
    <p><h1 style="text-align: center;color:
white;">Welcome </h1></p>
</div>
<!--//banner -->
<!--/w3_short-->
<div class="services-breadcrumb_w3ls">
    <div class="inner_breadcrumb">

        <ul class="short">
            <li><a
href="index.html">Home</a><span>|</span></li>
            <li><a
href="index.html">logout</a><span>|</span></li>
            <li><a href=""><?php echo
$_SESSION["mail"];?></a></li>
        </ul>
    </div>
</div>
<div class="container">
    <div class="row">
        <form action="sent.php" name="rsatest" method="post">
            <table class="table table-hover">
                <tr><th><input type="hidden" name="t0"
value='<?php echo $_SESSION["mail"];?>'></th></tr>
                <tr><th>Enter Mail</th><th><input
type="text" name="t1"></th></tr>
                <tr><th>Enter Message</th><th><textarea
cols="50" rows="7" name="t2"></textarea></th></tr>
                <tr><th>Enter Key</th><th><input type="text"
onblur="do_encrypt();" name="t3">
            </table>
        </form>
    </div>
</div>

```

```
        <input type="submit" name="" value="Send"
class="btn btn-primary">
        <br><br>
    </form>
</div>
</div>

<div class="footer_wthree_agile">
    <p>All Copy Rights by Sai</p>

</div>

<!-- //footer -->
<!-- js -->
<script type="text/javascript" src="js/jquery-
2.1.4.min.js"></script>
<script>
    $('ul.dropdown-menu li').hover(function () {
        $(this).find('.dropdown-menu').stop(true,
true).delay(200).fadeIn(500);
    }, function () {
        $(this).find('.dropdown-menu').stop(true,
true).delay(200).fadeOut(500);
    });
</script>
<!-- password-script -->
<script type="text/javascript">
    window.onload = function () {

        document.getElementById("password1").onchange =
validatePassword;

        document.getElementById("password2").onchange =
validatePassword;
    }

    function validatePassword() {
        var pass2 =
document.getElementById("password2").value;
```

```
        var pass1 =
            document.getElementById("password1").value;
            if (pass1 != pass2)

                document.getElementById("password2").setCustomValidity("
Passwords Don't Match");
            else

                document.getElementById("password2").setCustomValidity('
');

                //empty string means no validation error
            }
        </script>
        <!-- //password-script -->

        <script type="text/javascript"
src="js/bootstrap.js"></script>
    </body>

</html>
```

//PHP Code For View Data

```
<?php
    session_start();
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Dashboard</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.
css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.
js"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.js
"></script>
```

```
<style type="text/css">

    table tr td:last-child a{
        margin-right: 15px;
    }
    body {margin:0;font-family:Arial}

</style>
<script type="text/javascript">
    $(document).ready(function() {
        $('[data-toggle="tooltip"]').tooltip();
    });
</script>
</head>
<body>

    <div>
        <div class="container-fluid">
            <div class="row">
                <div class="col-md-12">
                    </div>
                    <?php
                        // Include config file
                        require_once "admin/employeeer-config.php";
                        // Attempt select query execution
                        $email=$_SESSION["mail"];

                        $sql = "SELECT * FROM sent where
email='$email'";

                        if($result = mysqli_query($link, $sql)){
                            if(mysqli_num_rows($result) > 0){
                                echo "<table class='table table-bordered
table-striped'>";

                                    echo "<thead>";
                                    echo "<tr>";
                                        echo "<th>id</th>";
                                        echo "<th>email</th>";
                                        echo "<th>Sender</th>";
```

```

                                echo "<th>message</th>";
                                echo "<th>Secret Key</th>";
                                echo "<th>Operation's</th>";
                                echo "</tr>";
                                echo "</thead>";
                                echo "<tbody>";
                                while($row =
mysqli_fetch_array($result)){
                                echo "<tr>";
                                echo "<td>" . $row['id'] .
                                "</td>";
                                echo "<td>" . $row['email']
                                "</td>";
                                echo "<td>" . $row['Sender']
                                "</td>";
                                echo "<td><a href='view.php'
                                style='text-decoration:none;color:black;'>" . $row['message'] .
                                "</a></td>";
                                echo "<td>" .
                                $row['secret_key'] . "</td>";
                                echo "<td>";
                                echo "<a
                                href='employeeer-view.php?id=". $row['id'] ."' title='View Record'
                                data-toggle='tooltip'><span class='glyphicon glyphicon-eye-
                                open'></span></a>";
                                echo "<a
                                href='employeeer-updates.php?id=". $row['id'] ."' title='Update
                                Record' data-toggle='tooltip'><span class='glyphicon glyphicon-
                                pencil'></span></a>";
                                echo "<a
                                href='employeeer-delete.php?id=". $row['id'] ."' title='Delete
                                Record' data-toggle='tooltip'><span class='glyphicon glyphicon-
                                trash'></span></a>";
                                echo "</td>";
                                echo "</tr>";
                                }
                                echo "</tbody>";
                                echo "</table>";
                                // Free result set
```



```
        mysqli_free_result($result);
    } else{
        echo "<p class='lead'><em>No records
were found.</em></p>";
    }
    } else{
        echo "ERROR: Could not able to execute $sql.
" . mysqli_error($link);
    }

    // Close connection
    mysqli_close($link);
?>

</div>
</div>
</div>
</div>

<!-- //footer -->
<script>
function myFunction() {
    var x = document.getElementById("myTopnav");
    if (x.className === "topnav") {
        x.className += " responsive";
    } else {
        x.className = "topnav";
    }
}
</script>

</body>
</html>
```

9. *Conclusions*

The Modified homomorphic encryption algorithm we used in this project for that we can get security in cloud email service. We can do encryption by using a public key when the time send email meanwhile we can do decryption by using a private key when the time views the email in the cloud. We gathered all the requirements by studying the existing system and we have analyzed the system in terms of the tasks that it is performing and the types of users using the system and the problems that are to be overcome in the proposed system.

All the requirements that are gathered in the Analysis phase are given a basic structure by following the Design Principles in the Design phase and the data from the analysis stage is converted in to design in the form of interfaces.

In the next stage, the coding is done following coding standards and the testing of the system has been performed by different testing strategies and techniques.

Security of cloud computing supported absolutely Homomorphic coding may be a new conception of security that is change to produce the results of calculations on encrypted information while not knowing the raw entries on that the calculation was dispensed respecting the confidentiality of information. Our work is predicated on the appliance of absolutely Homomorphic coding to the safety of Cloud Computing: a) Analyze and improve the present cryptosystem to permit servers to perform numerous operations requested by the shopper. b) Improve the complexness of the Homomorphic coding algorithms and study the interval to requests in keeping with the length of the general public key

Finally we can conclude that this project is satisfying all the requirements of the users of the system and is satisfying all its objectives set at the time of the development of the system.

Appendices

APPENDIX-A: User Manual

User Manual is the guide to the users of the system. It paves a path to the corresponding user to help him how to proceed further in the proper understanding of the system. The Interfaces of the system gets familiar to the user, based on this manual only.

The first form is the login form where user has to enter his username and password; here the types of users are Administrator, Users.

If the person has connected as an administrator, he will have rights to authorize to create User-id and passwords, Change of Users details in Database. If the user has been connected as Administrator he can enter the Users details, and can view reports.

Administrator will have access for all the following forms.

- User Details Entry – To store the user details
- User Details delete – To delete the user details
- User Details update – To update the user details
- User Details view – To view the user details
- Help –Customer support for user

APPENDIX-B: Test Screens

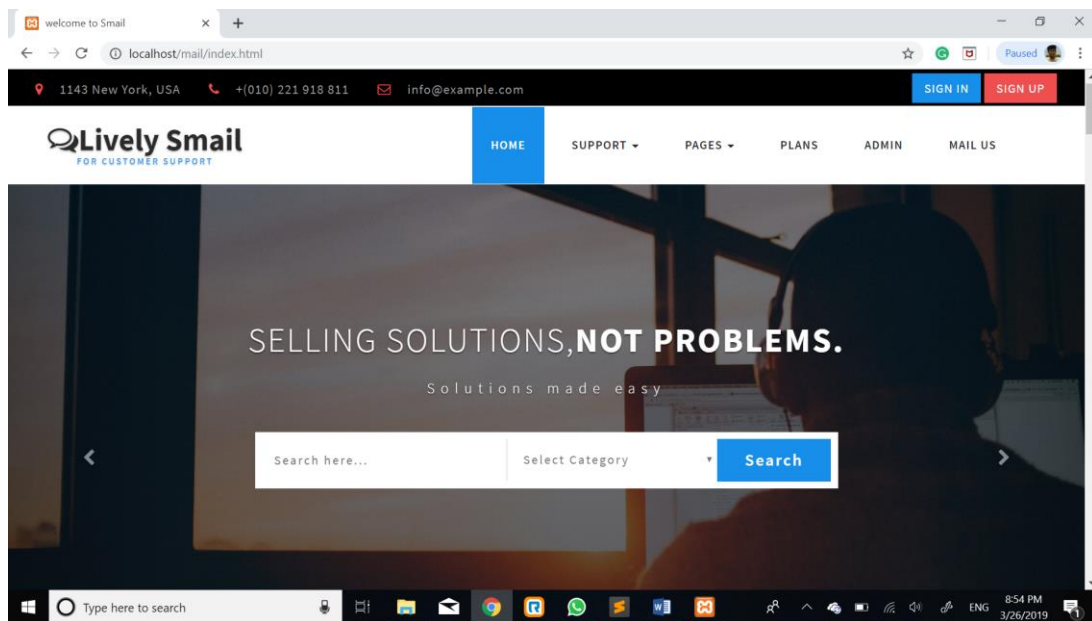
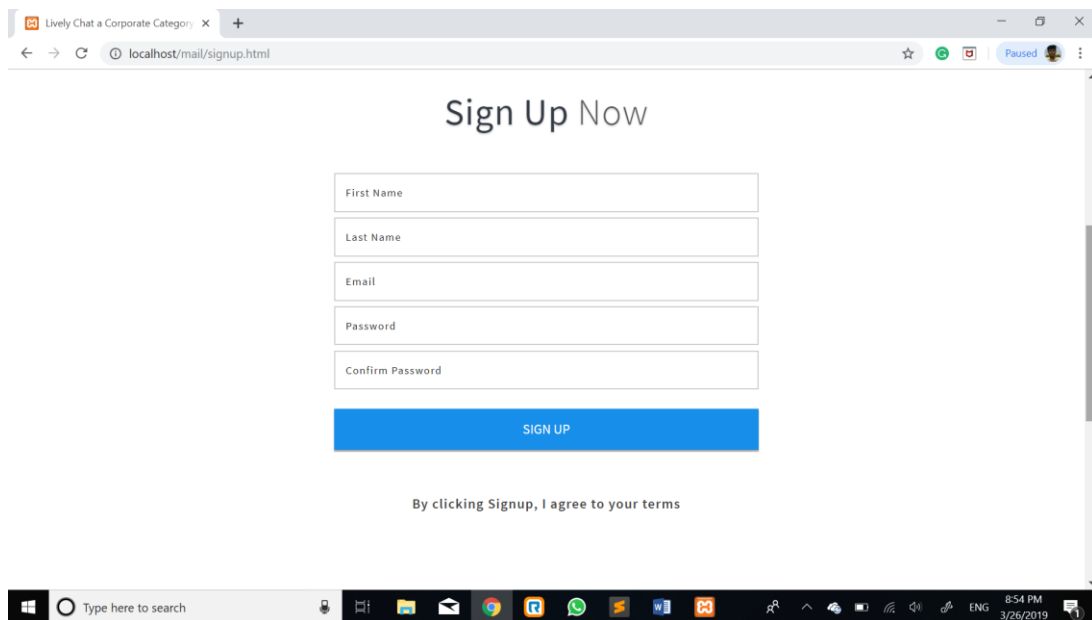
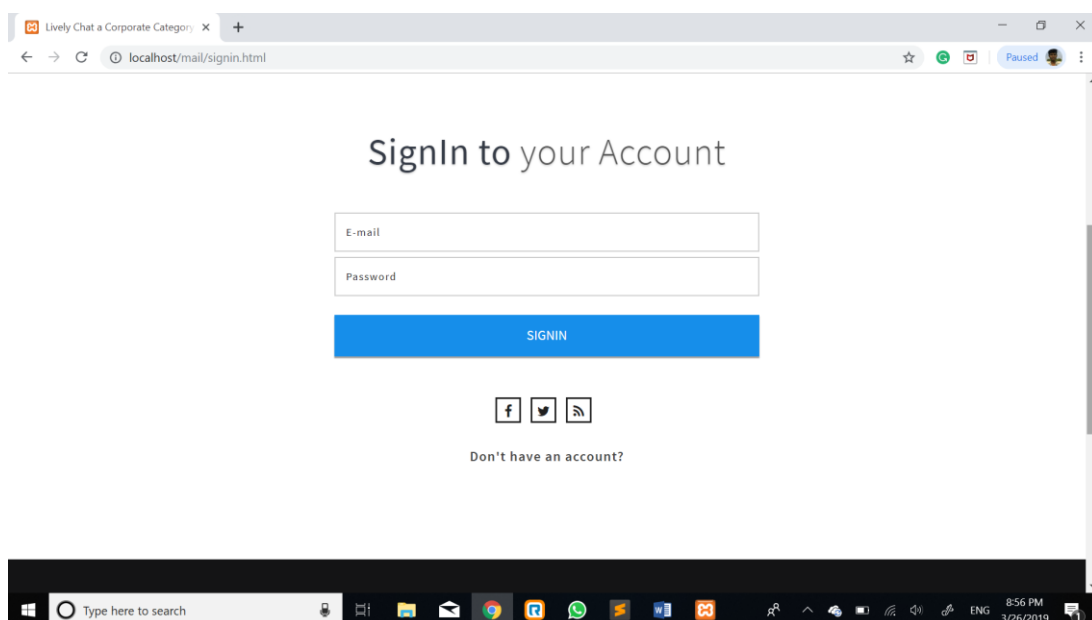


Fig: 1 Home Screen for Email



The screenshot displays a web browser window with a single tab titled 'Lively Chat a Corporate Category'. The address bar shows 'localhost/mail/signup.html'. The page content features a 'Sign Up Now' heading, followed by five input fields: 'First Name', 'Last Name', 'Email', 'Password', and 'Confirm Password'. A blue 'SIGN UP' button is positioned below these fields. At the bottom of the form, a text line reads 'By clicking Signup, I agree to your terms'. The Windows taskbar at the bottom includes a search bar, several application icons, and a system clock showing 8:54 PM on 3/26/2019.

Fig 2: User Details Registration Form



The screenshot shows a web browser window with a tab titled 'Lively Chat a Corporate Category'. The address bar displays 'localhost/mail/signin.html'. The page content includes a 'SignIn to your Account' heading, two input fields for 'E-mail' and 'Password', and a blue 'SIGNIN' button. Below the button are three social media icons (Facebook, Twitter, and RSS) and a link that says 'Don't have an account?'. The Windows taskbar at the bottom shows the search bar, application icons, and a system clock indicating 8:56 PM on 3/26/2019.

Fig 3: User Login Form

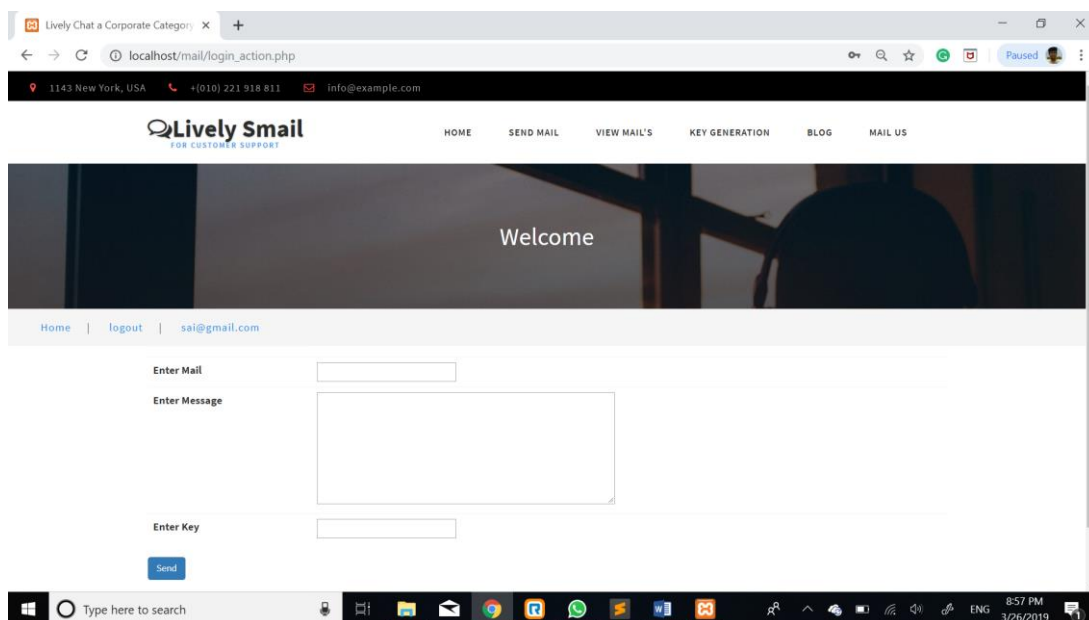


Fig 5: Sent Message

Secure Email Service in Cloud By Using Modified Homomorphic Encryption

The screenshot shows a web browser window with the address bar displaying 'localhost/mail/demo/key_gen.php'. The page contains a form for generating a key pair. At the top, there is a text input field labeled 'Number of bits' with the value '1024' and a blue button labeled 'Generate keypair'. Below this, there are two large text input fields: the top one is labeled 'public key:' and the bottom one is labeled 'private key:'. At the bottom of the form, there is a blue button labeled 'Submit'. A black banner at the bottom of the browser window reads 'All Copy Rights by Sai'. The Windows taskbar is visible at the bottom of the screen.

Fig 6: Key Generation Form

The screenshot shows a web browser window with the address bar displaying 'localhost/mail/login_action.php'. The page is for 'Lively Mail' and includes a navigation bar with links: HOME, SEND MAIL, VIEW MAIL'S, KEY GENERATION, BLOG, and MAIL US. Below the navigation bar is a large banner with the word 'Welcome' and a background image of a person. Under the banner, there is a login section with the text 'Home | logout | sai@gmail.com'. Below this, there are two input fields: 'Enter Mail' and 'Enter Message'. The Windows taskbar is visible at the bottom of the screen.

Fig 7: View Message

Enter A secret key KMMCG10

Original Data

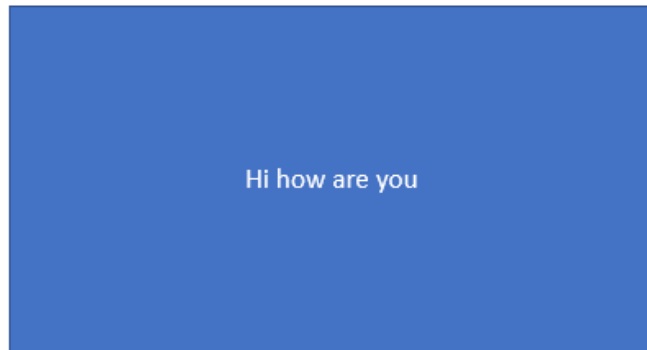


Fig 8: view Reports

APPENDIX-C: Base Paper

Homomorphic Encryption Method Applied to Cloud Computing

APPENDIX-D: Conference Paper

A Survey On Securing Cloud Service Data By Using Homomorphic Encryption

APPENDIX-E: Journal Paper

Secure Email Service in Cloud by using Modified Homomorphic Encryption Algorithm

Bibliography

- [1] Z. Brakerski, C. Gentry, and V. Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. *ACM Transactions on Computation Theory*, 6(3):13, 2014.
- [2] Z. Brakerski and V. Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE . *SIAM J. Comput.*, 43(2):831–871, 2014.
- [3] Pascal Paillier, “Public-key cryptosystems based on composite degree residuosity classes”, In 18th Annual Eurocrypt Conference (EUROCRYPT'99), Prague, Czech Republic, volume 1592, 1999
- [4] Vic (J.R.) Winkler, “Securing the Cloud, Cloud Computer Security, Techniques and Tactics”, Elsevier, 2011.
- [5] D. Boneh and al. Public key encryption with keyword search, proceedings of Eurocrypt 2004, LNCS 3027, pp. 506-522, 2004.
- [6]. E-Voting Simulator based on the Paillier Cryptosystem, Andreas Steffen, HSR Hochschule für Technik Rapperswil .
- [7].Public-key cryptosystems based on composite degree residuosity classes (1999), Pascal Paillier
- [8] M.J. Wiener, “Cryptanalysis of short RSA secret exponents,” *IEEE Transactions on Information Theory*, vol: 36, Issue: 3, pp: 553-558, May 1990.
- [9] C.-C. Yang, T.-S. Chang and C.-W. Jen, “**A new RSA** cryptosystem hardware design based on Montgomery's algorithm,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol: 45, Issue: 7, pp: 908-913, July 1998.
- [10] C.-H. Wu, J.-H. Hong, and C.-W. Wu, “RSA cryptosystem design based on the Chinese Remainder Theorem,” *Proceedings of the ASP-DAC 2001*, 30th Jan.-2nd Feb. 2001, pp: 391–395.
- [11] Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186, May. 1994.

- [12] RSA Security Inc., Crypto FAQ: Chapter 6: Laws concerning cryptography, 6.3. Patents on cryptography.
- [13] RSA Security Inc., Crypto FAQ: Chapter 2: Cryptography, 2.2. Simple applications of cryptography.
- [14] RSA Security Inc., Crypto FAQ: Chapter 4: Applications of Cryptography. 4.1 Key management, 4.1.2 General.
- [15] RSA laboratory bulletin number 13, A cost-based security analysis of symmetric and asymmetric key lengths. April 2000. Available: <http://www.rsasecurity.com/rsalabs/node.asp?id=2088>.
- [16] RSA Security Inc., “PKCS #1 v2.0 amendment 1: Multi-prime RSA,” July 2000. Available: <ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-1/pkcs-1v2-0a1.pdf>.
- [17] A. Krishnamurthy, Y. Tang, C. Xu and Y. Wang, “An efficient implementation of multi-prime RSA on dsp processor,” IEEE Int. Con. on Acoustics, Speech, & Signal Processing, Hongkong, China, vol. 2, April 2003, pp 413-416.
- [18] S. Yen, S. Kim, S. Lim and S. Moon, “RSA speedup with Chinese Remainder Theorem immune against hardware fault attack,” IEEE Transactions on computers, vol. 52, pp. 461-472, April 2003.
- [19] L. R. YU, “The generalization of the Chinese Remainder Theorem,” Acta Mathematica Sinica, English Series, vol. 18, pp. 532-538, July 2002.
- [20] J.-J. Quisquater and C. Couvreur, “Fast decipherment algorithm for RSA public-key cryptosystem,” Electronic Letters, vol. 18, no. 21, pp 905-907, Sept. 1982.
- [21] Dan Boneh, “Twenty years of attacks on the RSA cryptosystem,” 2000. Available: <http://crypto.stanford.edu/~dabo/papers/RSA-survey.pdf>.

REFERENCES:

1. <https://digitalguardian.com/blog/what-email-encryption>
2. <https://study.com/academy/lesson/what-is-email-encryption-definition-methods.html>