ISSN No:-2456-2165

Security of Health Related Sensitive Data using Modern Cryptography

[1]Dr. Preeti Patil, [2]Shubham Davate, [3]Rishikesh Jagtap, [4]Rishikesh Devkate, [5]Apoorv Gajbhe
[1]Head of Department of Information Technology, DY Patil College of Engineering, Akurdi
[2,3,4,5]Student, Department of Information Technology, DYPCOE Akurdi
Savitribai Phule Pune University, Pune, India

Abstract:- Every person on this earth lives with certain kind of lifestyle irrespective of health concern. Due to fast growing life-style, Nobody gives time health related problems. Whenever we visit hospital for minor health related problems and then later ignore precautions and treatment cause. But due to this anyone health can possibly reflected into major serious health issue. So it is very important for everyone to take care of their health first.

This project uses modern security technique for providing very useful and highly secure online medical science application for overcoming such facts. Here we have presented an idea of online scheme for sharing, transferring, storing and maintaining database related to person's health in very secure way. It will simply focus on maintaining integrity of data on cloud. For this, we have used technique of modern cryptography specifically a fully homomorphism encryption technique and applied for attributes based encrypted data of every patient. This is a head technique we are going to use which will actually encrypt on already encrypted data. Our scheme will also help the service for patient for finding best doctor for best treatment. Henceforth, our system will surely useful for medical science field for providing best treatment to patients and to give satisfaction for every individual in view of their health.

I. INTRODUCTION

As today's encryption system use partial homomorphic encryption and this encryption techniques are not fully secured to provide protection in quantum age so we proposed fully homomorphic encryption technique to provide complete protection to the data stored on third party server. Limitations of recent papers can be overcome by fully homomorphism. we can use this for more confidential and trust worthy transfer services like to maintain medical databases and to use paperless medical science field with fully homomorphism. Additionally record is maintained with double encrypted on this hub hard party without giving access to server so that avoids Man in Middle Attacks. When homomorphism has been applied on encrypted data old data get lost automatically which is stored on third party. If it can be used widely then it can be act as a professional or platform to connect patient and doctor in medical science field.

II. LITERATURE SURVEY AND EXISTING WORK

In today's quantum world most of the organizations keeps their data on cloud as cloud provide certain advantages over physical data storage. But this cloud data storage comes with the drawback of data security. Data security is growing and major issue in today's world as traditional security algorithms are vulnerable to security attacks. so to provide security many cryptographic algorithms are developed such as RSA algorithm, Pailliers Cryptosystem [1] etc. But these cryptosystems are gets easily hacked and data is available to attacker easily. Due to this Modern cryptography joined the field to provide more reliable and efficient solutions to securely stored data on cloud. In modern cryptography mainly Partial homomorphism[1][2] is the technique used to provide security to the cloud data. This partial homomorphism used technique of providing double encryption to the already encrypted data so that the data which stored on cloud gets double encrypted and it is extremely difficult for the attackers to retrieve original message from doubly encrypted data on cloud. So the now a day's organizations uses Partial homomorphism to secured data. In the reference paper [1] "Secure data storage into the cloud with homomorphic Encryption" published by Yasmina Bensitel and Rahal Romadi, author described the idea and algorithm of Partial homomorphic encryption. In partial homomorphism they used either used additive or multiplicative operations to stored data. Author used RSA algorithm for multiplicative partial homomorphism and also described the concept of "Somewhat Homomorphic Encryption"[1][2]. The drawbacks of this system is that they use of Partial Homomorphic Encryption instead of Fully Homomorphic Encryption and also Keys which are used for encrypting and decrypting cipher text are not secured and keys are stored as it is on cloud's database.

In the article [3] "A Guide to Fully Homomorphism Encryption" written by Christian A. Reuter, author represented the idea of fully homomorphic encryption and give brief description on how this encryption works. In fully homomorphic encryption author extend the scope of mathematical function to any function and used any mathematical operation on already encrypted data to provide double encryption. Due to this we can used any mathematical operation and combination of mathematical operations such as simultaneous multiplicative and additive operation to perform operation on encrypted data to double encrypt that data and stored on cloud. Also author gives the idea of function encryption and obsfunction[3][4]. The problem found in this system is that the algorithm is inefficient because of large overhead produced by mathematical operations.

III. SYSTEM ANALYSIS AND PROPOSED ARCHITECTURE

➢ *Fully Homomorphism*

Fully homomorphic encryption (FHE) is also called as the holy grail of cryptography, used to solve IT related problem regarding trust and security. Fully Homomorphic Encryption is termed as revolution in the field of cryptography which extends the scope of computations and perform operations on already encrypted data[2][3][5].That is fully homomorphism allows arbitrary computations on encrypted data. In partial Homomorphism Scheme user is not able to use the data he must download the data and perform the computations locally, with fully homomorphic encryption the cloud can do computations on behalf of the user and gives encrypted result[4].

➢ *Proposed System*

As Partial Homomorphism is not that much reliable and inefficient we proposed algorithm and system for Fully Homomorphism. We used concept of Fully Homomorphism in the field of Medical Science to securely stored Health related data on cloud. In this system we keep Patients health related sensitive data on cloud as well as we keep feedback system for doctor as patient gives reviews to doctor as per treatment. Also in existing system keys which are used for encryption and decryption are not stored securely so in our approach we applied encryption to keys as well so that keys are also stored securely on cloud in separate database. In this system we provide location system through which Patient can search Doctor easily in his area also Patient can search Doctor based on Doctor's specialization. In this application we provide double encryption to Patients database as well as Doctors database. So by this we can securely stored Patients database on cloud and also Doctors database related to Review System is also stored on cloud. And in our system Patients database is updated and altered by Doctor, only when Patient provide his secret key to decrypt that double encryption. And Patient can give review to Doctor when Doctor provides his secret key to decrypt the double encryption.

So, in our system there are three databases are required to stored data related to Doctor, Patient and Keys. Patient's database is used to stores the data related to Patient's sensitive health related information along with Patient's registration information and these data is stored with homomorphism so that third party cannot access original data and only patient has right to access his/her information. Doctor's database is used to stored Doctor's registration information as well as Doctor's review related information which is also stored with homomorphism encryption and only Patient can give review to Doctor with respect to treatment so that it will be beneficiary for Patient to find Best Doctor based on Specialization, Location and Review. Third database is related to Keys which are used for decryption of homomorphism encryption. In our system we are also providing protection to Keys by using double encryption and stored these Keys on global database in doubly encrypted format[6].

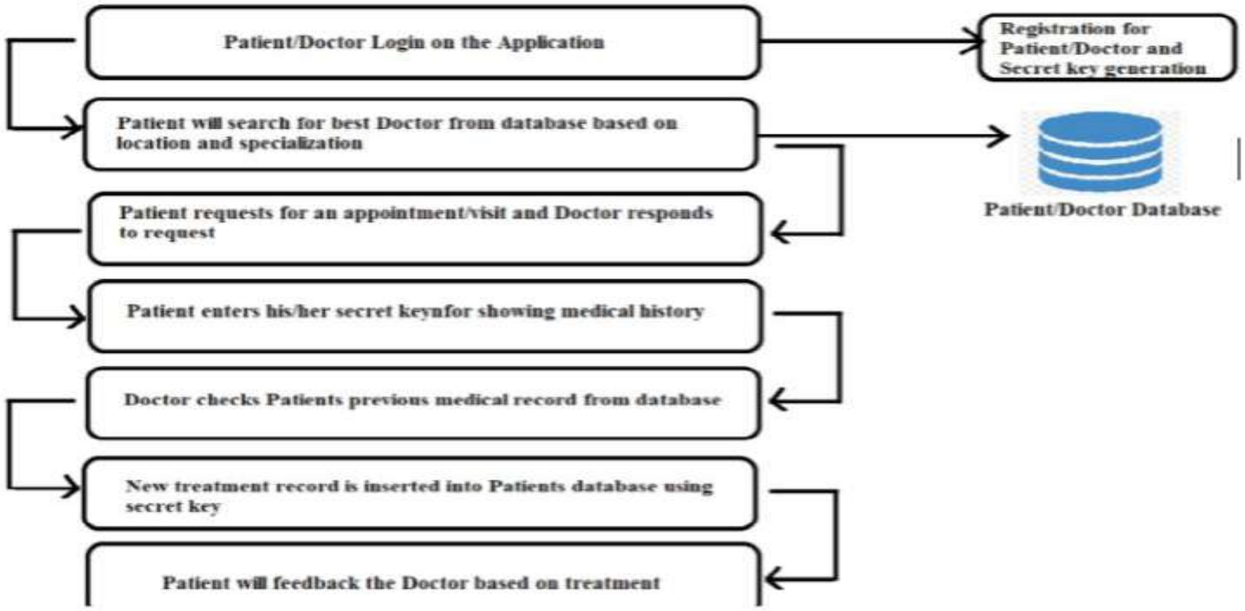
Patient/Doctor Login on the Application
Registration for Patient/Doctor and Secret key generation
Patient will search for best Doctor from database based on location and specialization
Patient/Doctor Database
Patient requests for an appointment/visit and Doctor responds to request
Patient enters his/her secret keynfor showing medical history
Doctor checks Patients previous medical record from database
New treatment record is inserted into Patients database using secret key
Patient will feedback the Doctor based on treatment

Fig 1

Above figure will explain the architecture of how our application will work in real time. This diagram shows how medical health care databases of patient can store on cloud with fully homomorphism. When Patient/Doctor use this application for the first time, they will have to register themselves on the application so that from the attributes they put on registration form from that attribute(password) we can generate Patient(PID) and Doctor(DID) registration number. This PID and DID are used to generate Keys which are useful to encrypt and decrypt the data. So, when Patient physically available for treatment at that time he used his/her secret keys to decrypt his/her records and show them to respective Doctor for best treatment. Like Patient, Doctor should also register himself/herself for getting their DID (Doctor's Identity) to generate Secret Key which he/she used when Patient is physically present at their place. Due to this Patient can give Doctor ratings based on his/her treatment which will help to improve Doctor's ratings. Also, Patient can search for Doctor based on Patient's respective area as well as Doctor's Specialization and Rating. Due to Secret Keys only Doctor can update Patient's medical records based on treatment by using Patient's Secret Key when patient will physically present at Doctor's place. And by using Doctor's Secret Key Patient can give Ratings to the Doctor based on Doctor's treatment. Thus, above steps shows our approach to implement our system and how our system will work in real time scenario of Doctor and Patient to secured Doctor's and Patient's data on cloud(Global Server).

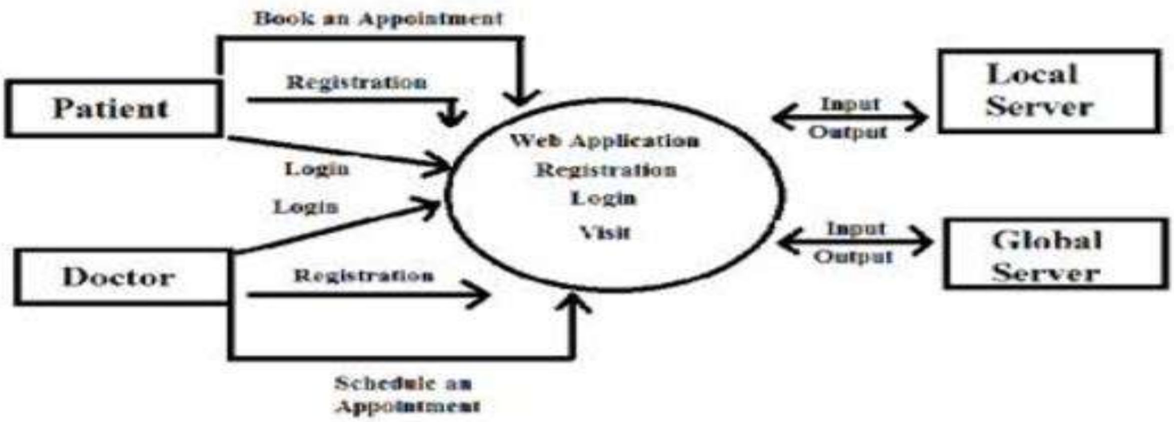

Fig 2

It is the part of system analysis. In that we show that how the data is flow. Here we show that how Patient and Doctor perform various processes on Web applications and how Global Server stores data.

IV. ALGORITHM AND MATHEMATICAL MODEL

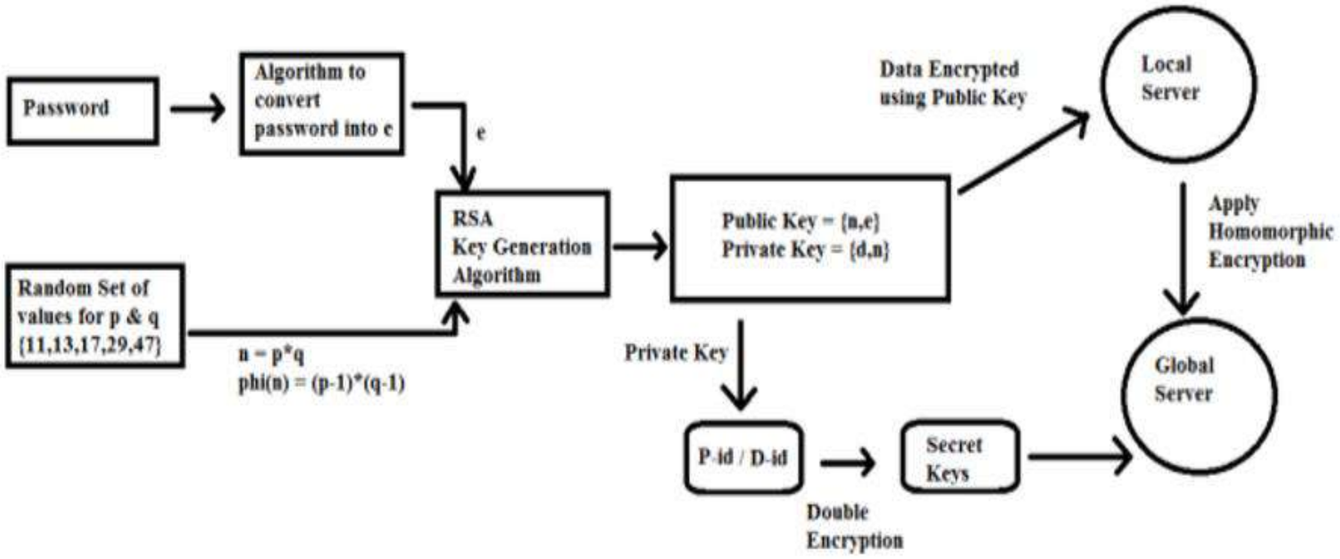
Password
Algorithm to convert password into e
e
RSA Key Generation Algorithm
Random Set of values for p & q {11,13,17,29,47}
n = p*q
phi(n) = (p-1)*(q-1)
Public Key = {n,e}
Private Key = {d,n}
Data Encrypted using Public Key
Local Server
Apply Homomorphic Encryption
Global Server
Private Key
P-id / D-id
Secret Keys
Double Encryption

Fig 3

- The data which is already encrypted stored on Local Server is again encrypted with homomorphic technique and stored on Global Server.
- When Doctor and Patient registered themselves on web server they are free to set their own password.
- After registration this information is stored on Local Server in encrypted form by using RSA algorithm. Steps:

✓ Firstly, we have set of random prime values at backend in the database say p= {11, 17, 23, 29, 47}

✓ Create random p and q prime numbers selected from stored database.

✓ For generation of keys, let calculate n as, n = p*q; phi(n) = (p-1) * (q-1);

✓ Then by using password entered during registration, we first convert this password into a prime numbers which can be relatively prime as, {pwd} -> e ,such that 1<e<phi(n)

✓ After evaluating the value of term e we get the Public and Private Key which are used for encryption and decryption respectively.

✓ Public Key = PK (n, e) and Private Key = SK (d, n).

✓ Public Key (pk) is used for encrypting the information and stored it on Local Server.

✓ Then Private Key i.e. Pid is used for decryption of information which is stored on Local server.

✓ This Private Key is Stored in separate Database Local Server.

- Due to separate value of term e and random selection of values of p and q it will be hard for attacker to retrieve data stored on Local Server.
- After this the data stored on Local Server is stored on Global Server with double encryption.

➢ *Encryption Process*

When Patient/Doctor registers himself on application at that time he/she is free to set the password of his/her choice. Then, by using this password as ‘e’ and choosing the value of ‘p’ and ‘q’ public key and private key is generated using RSA algorithm. This public key is used to encrypt the Patient’s/Doctor’s information and then homomorphism encryption is applied on that already encrypted data so that data should be doubly encrypted and stored on global server. Now on server side Database of Patient and database of doctors will be maintained in encrypted format along with the secret keys are also in encrypted format. So, now server doesn’t have any original data but only the encrypted data on it.

➢ *Decryption Process*

In our system, private key which is generated during RSA algorithm is also stored on global server by applying double encryption. So, the password which used during registration is first encrypted using RSA algorithm and then again this password is doubly encrypted to store on global server.

These singly and doubly encrypted keys are sent to corresponding registered user on his email. So, when Patient wants to see his treatment information as well as wants to book an appointment then he/she uses his/her singly encrypted key. When Patient will take an appointment for treatment and meets that Doctor physically at that time Patient uses his doubly encrypted key to decrypt the data which is stored on global server and Doctor can see the previous treatment records of that Patient and update and insert the data according to treatment.

V. SYSTEM DISIGN

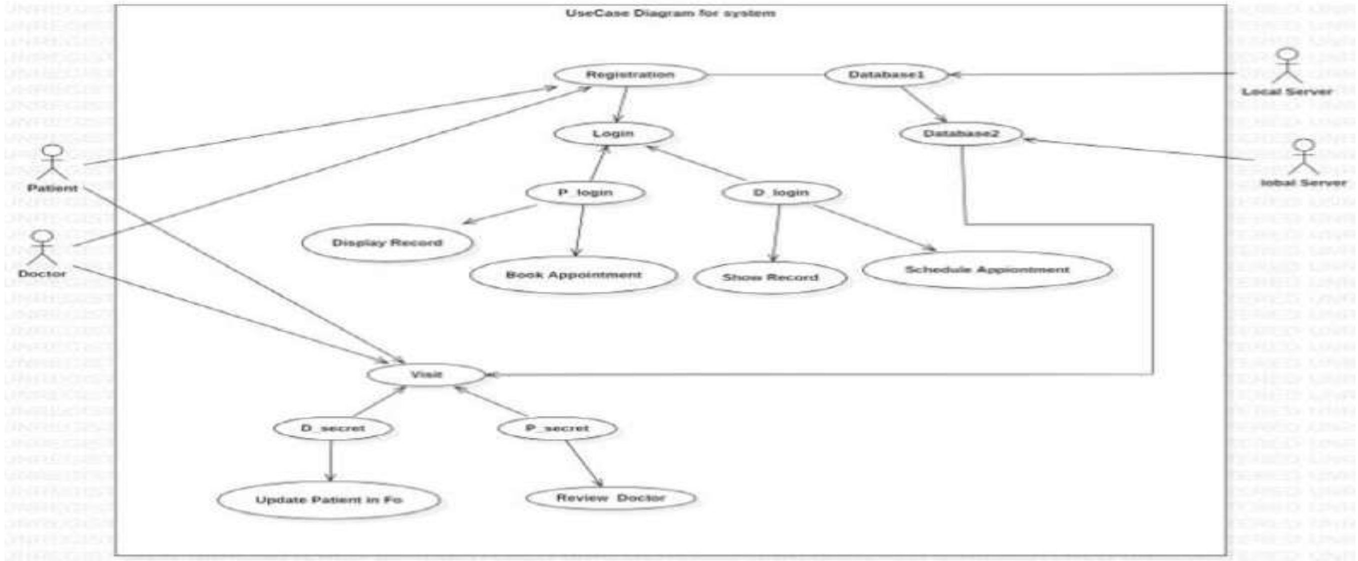

Fig 4:- Use Case Diagram

Use Case Diagram shows Patient and Doctor Interaction with Web Application. It shows how Patient and Doctor first registered themselves on Web Application and then that information is stored on Database1 at Local Server. After registration Patient and Doctor can Login into Application to perform various functions. At Visit Database2 comes into picture from which doubly encrypted information is decrypted into plain text by using Patient's and Doctor's Secret Keys.

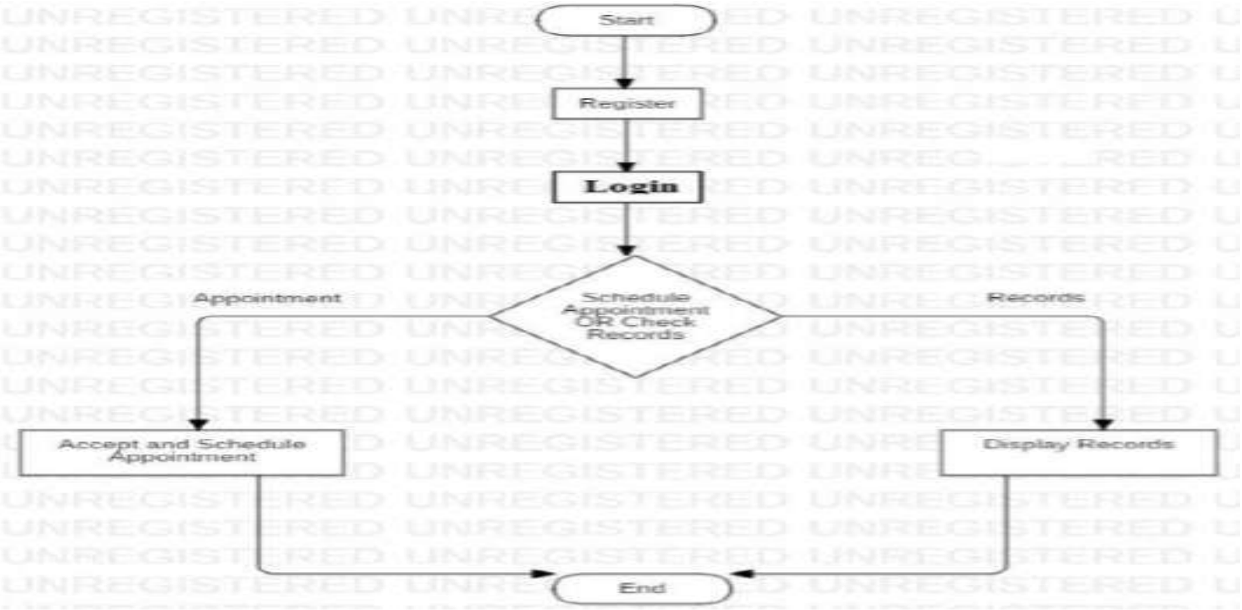
Start
Register
Login
Schedule Appointment OR Check Records
Appointment
Records
Accept and Schedule Appointment
Display Records
End
UNREGISTERED

Fig 5:- Activity Diagram of Doctor

Activity diagram show what activities are required to be done by doctor on web application and also show flow of activities from starting to the end.

VI. USER INTERFACE

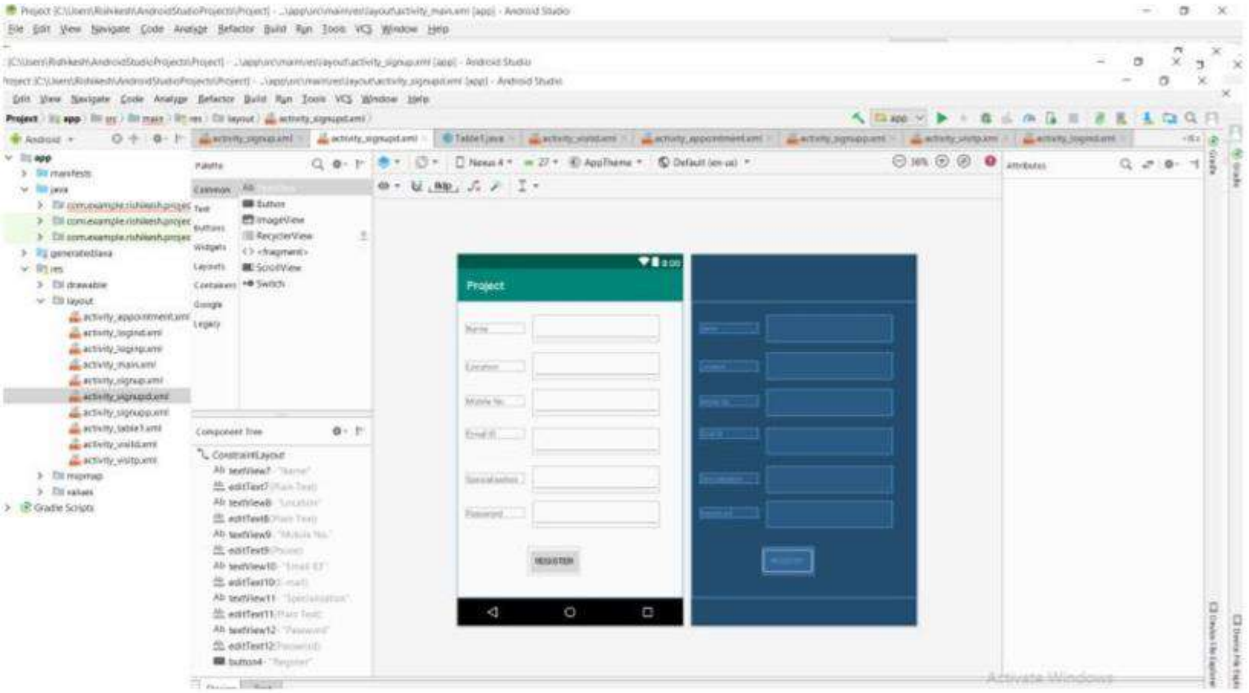

Fig 6

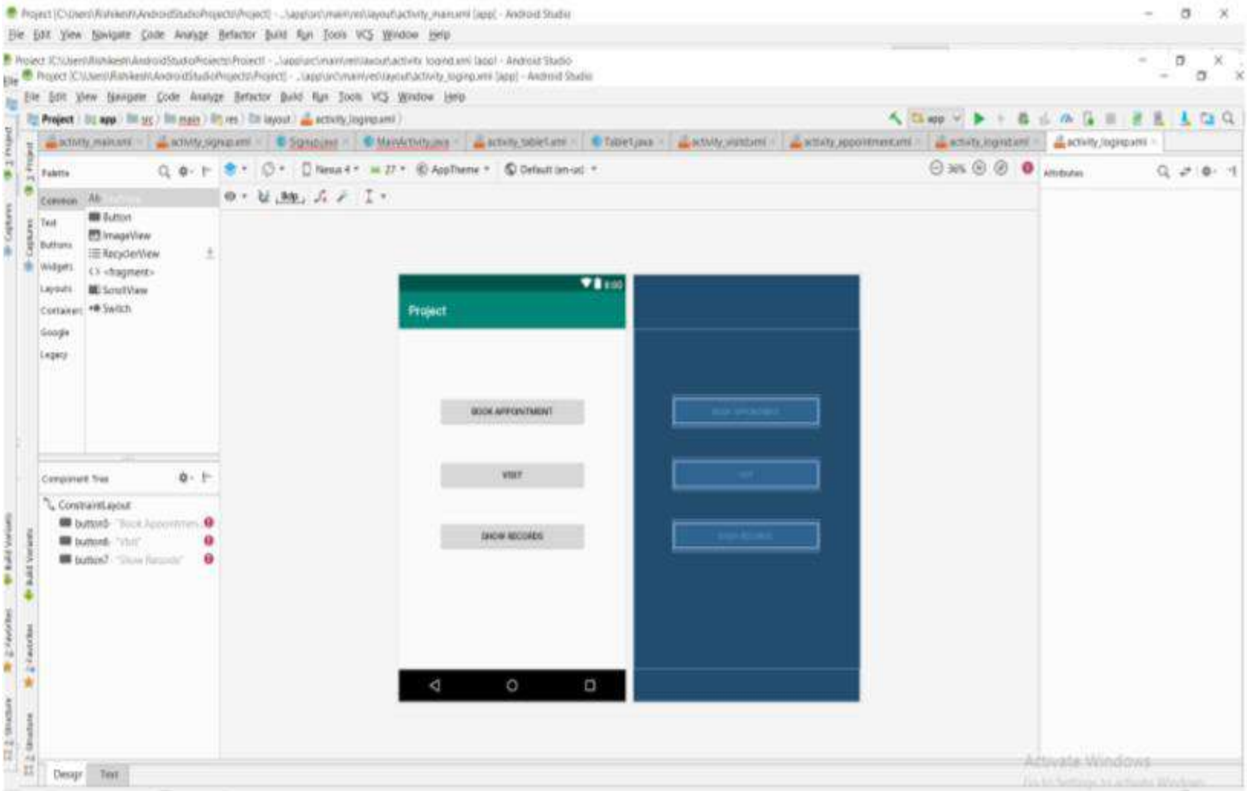
Project
BOOK APPOINTMENT
VISIT
SHOW RECORDS
Palette
Component Tree
ConstraintLayout
button5 - "Book Appointmen..."
button6 - "Visit"
button7 - "Show Records"
Design
Text

Fig 7

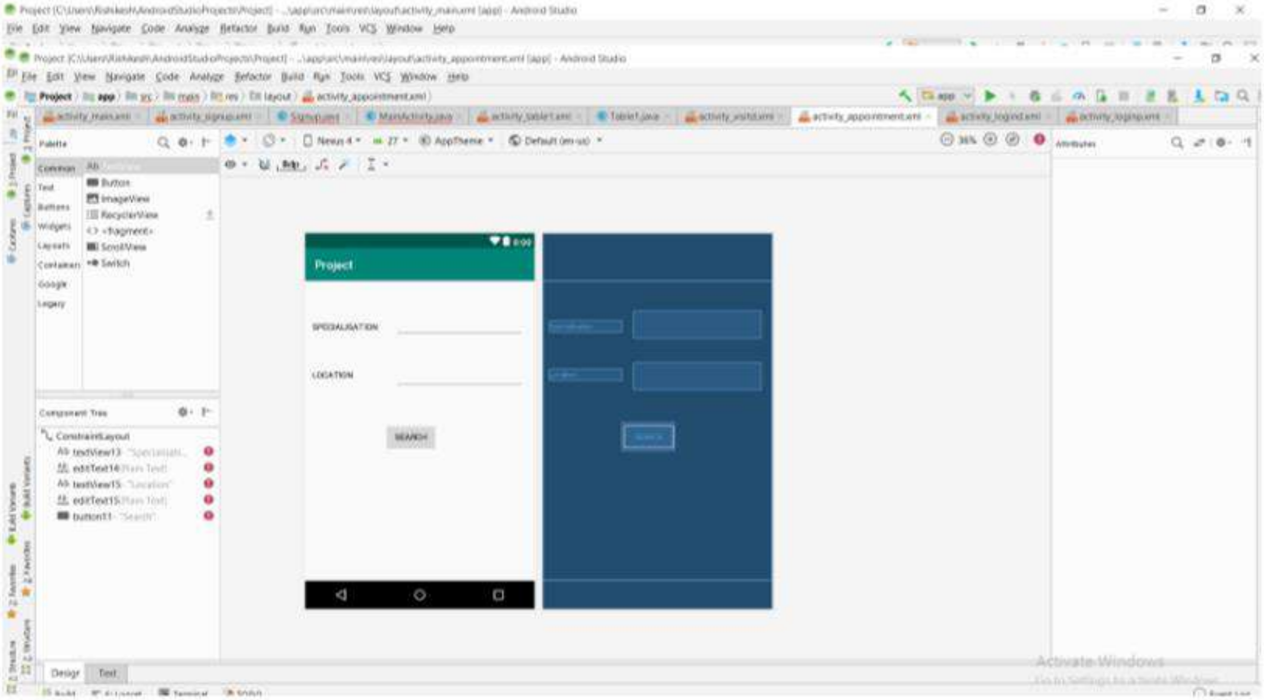
Project
SPECIALISATION
LOCATION
SEARCH
Palette
Component Tree
ConstraintLayout
textView13 - "Specialisati..."
editText14 (Plain Text)
textView15 - "Location"
editText15 (Plain Text)
button11 - "Search"
Design
Text

Fig 8

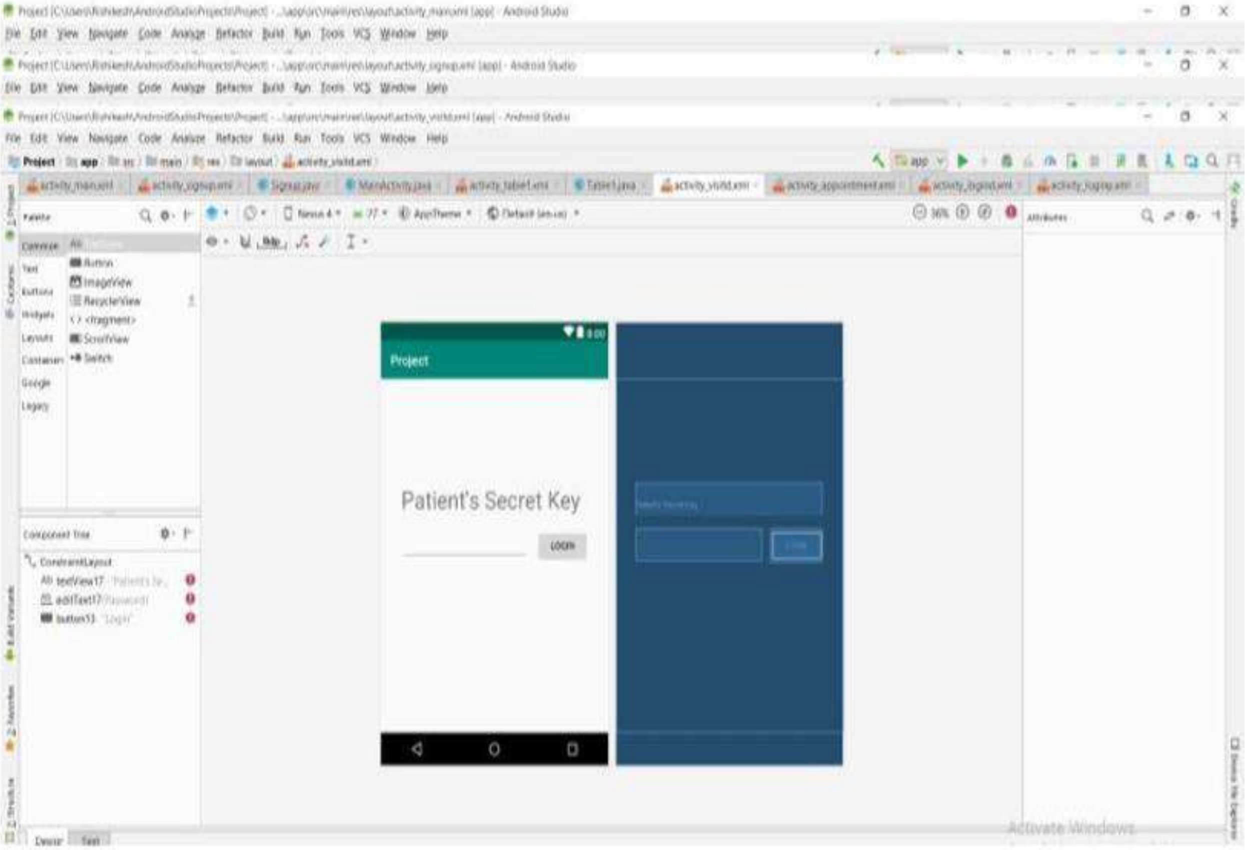

Fig 9

VII. ADVANTAGES

- The authenticity of data is not compromised and it can also be accessed by the authentic user.
- Using homomorphic encryption can avoid data breach of cloud data.
- Data security is maintained due to use of fully homomorphic encryption.
- Keys to access data on server are secured as we provide encryption to the keys also.
- Due to online data storage of medical science field traditional paper base system will get vanished.
- Due to review system patient can find best doctor within specific area.

VIII. FUTURE SCOPE

Fully Homomorphic Encryption is an advancement in the field of modern cryptography which overcomes the drawback of traditional cryptographic method and partial homomorphic encryption technique. In Fully Homomorphic encryption we can extend the scope of mathematical function so to make it extremely complex for attacker to decipher it and attack on user's data. We extend this concept in the field of medical science to secured Doctor's and Patient's database which will stored on cloud. This security technique can be used in various different applications related to cloud. Due to Fully Homomorphic Encryption, data which will be on cloud will always in as double encrypted format and original data is not present as it is on cloud. So, any type of application which stored their data on cloud can use Fully Homomorphic Encryption technique to secured their data on cloud.

IX. CONCLUSION

Some of the encryption systems have multiplicative or additive property and combining both gives fully homomorphic encryption. This allows performing multiplicative or additive operations on ciphertext data without having to decrypt it first. The homomorphic encryption is solution to reduce security flaws in the cloud. These techniques have more secured approach of hiding the data from external attacks. In this article, we show that partial homomorphism is easy and efficient to store and secure data on cloud. But due to new age quantum computers this encryption technique is not much efficient. So we show that fully homomorphism can fill all the security flaws of partial encryption and used to secure data on cloud. We also extend this concept in medical science field to achieve reliable and highly secured communication between doctor and patient.

REFERENCES

[1]. IEEE paper – Yasmina BENSITEL and Rahal ROMADI, "Secure data storage in the cloud with homomorphic encryption" 2016.
[2]. SPRINGER article – Diaz Perez,"A Brief Introduction to Modern Cryptography"2007.
[3]. IEEE paper – Colin Boyd and Christian A. Reuter, "A Guide to Fully Homomorphic Encryption" 2017.
[4]. THESIS – Zhenfei Zhang,"Revisiting Fully Homomorphic Encryption Schemes and Their Cryptographic Primitives" 2014.
[5]. WIKIPEDIA - Article on "Fully Homomorphism".
[6]. Sharddha Shelar, Deepali Rane,"Reducing Efforting in Healthcare System Using Secure Database".
[7]. P Wardlaw,"The RSA Public key Cryptosystem".