Index

**Chapter 1 Introduction1**

1.1 Need1

**Chapter 2 Review of Literature2-4**

2.1 Cloud Computing Components2

2.2 Cloud Service Providers Examples3

**Chapter 3 Report on Present Investigation (Existing System)5**

**Chapter 4 Aim and Objective6**

**Chapter 5 Problem Statement7**

**Chapter 6 Proposed System for Project8**

6.1 Data Integrity8

6.2 Data Intrusion8

6.3 Service Availability8

**Chapter 7 Requirement Analysis (SRS)9**

**Chapter 8 Scope (Feasibility of Project)10-11**

8.1 Operational Feasibility10

8.2 Technical Feasibility10

8.3 Economical Feasibility11

8.4 Legal Feasibility11

**Chapter 9 Methodology12-14**

9.1 Security in Cloud Services Is Based On The Following 12

9.2 Secret Sharing Strategy14

**Chapter 10 Design Details15-23**

10.1 Algorithm Used23

**Chapter 11 Implementation and Experimental Setup24-33**

11.1 Hardware and Software Requirement24

11.1.1 Hardware Requirements24

11.1.2 Software Requirements24

11.2 System Development Life Cycle25

11.2.1 The Iterative Model27

11.3 Gantt Charts31

**Chapter 12 Testing Technology34-37**

12.1 Unit Testing34

12.2 Integration Testing34

12.3 Functional Testing34

12.4 Performance Testing35

12.4.1 Expected Result35

12.4.2 Observation35

12.5 Load/Stress Testing35

12.5.1 Expected Result35

12.5.2 Observation35

12.6 Test Cases36

**Chapter 13 Result and Analysis38-42**

**Chapter 14 Conclusion43**

**Chapter 15 Reference44**

**Chapter 16 Acknowledgement45**

**Chapter 17 Paper Publication Details46**

**List of Figures**

Fig. 2.1 Cloud Computing Components2

Fig. 9.1 Cloud Computing Security13

Fig. 10.1 Context Level Diagram15

Fig. 10.2 (a) DFD Diagram15

Fig. 10.2 (b) DFD Diagram15

Fig. 10.3 Sequence Diagram16

Fig. 10.4 Use Case Diagram17

Fig. 10.4 (a) Admin Use Case18

Fig. 10.4 (b) User Use Case18

Fig. 10.5 (a) Admin Activity Diagram19

Fig. 10.5 (b) User Activity Diagram20

Fig. 10.6 System Architecture21

Fig. 10.7 Architectural Design22

Fig. 11.1 System Development Life Cycle25

Fig. 11.2 Iterative Model28

Fig. 11.3 Gantt Chart33

Fig. 13.1 Admin Login Panel38

Fig. 13.2 Manage/Add User38

Fig. 13.3 Manage Files39

Fig. 13.4 Upload Files39

Fig. 13.5 Share Files40

Fig. 13.6 View Shared Files40

Fig. 13.7 User Login Panel41

Fig. 13.8 E-Mail Containing Secret Key and Authentication Keys41

Fig. 13.9 View Shared Files42

Fig. 13.10 Download Shared Files42

**List of Tables**

Table 11.1 Project Plan Phase-131

Table 11.2 Project Plan Phase-232

Table 12.1 Test Case for Testing of Login Module36-37

Table 12.2 Test Case for Testing File Upload/Download Module37

**Abstract**

Now a day’s rapidly increased use of cloud computing in the many organization and IT industries and provides new software with low cost. So the cloud computing give us lot of benefits with low cost and of data accessibility through Internet. The ensuring security risks of the cloud computing is the main factor in the cloud computing environment, for example sensitive information with cloud storage providers may be entrusted. But, a single cloud provider is a less popular with customers due to risks service availability failure and possibly of malicious insiders in the ‘single cloud’.

This project surveys to many running research related project to single cloud security using Shamir‟s Secret Sharing algorithm and addresses possible solutions and methodology. Main focus of this project use of single cloud and data security and reduce security risks and affect the cloud computing user using Shamir‟s Secret sharing algorithm. It is a form of secret sharing, where a secret is divided into parts, which is giving each participant its own unique part, where some of the parts or all of them are required in order to reconstruct the secret. If we are going to Count all participants to combine together the secret might be impractical, and therefore sometimes the threshold scheme is used where any ‘k’ of the parts are sufficient to reconstruct the original secret.

**Chapter 1 Introduction**

**1.1 Need**

The cloud computing is a cost-effective, service availability, flexible and on demand service delivery platform for providing business through the internet. Cloud computing resources can be quickly extracted and effortlessly scaled with all the processes, services and applications provisioned on demand service despite the consequences of the user location or device. Hence, the opportunity for an organization to enhance their service deliverance efficiencies is achieved through cloud computing. The issues in cloud security series from substantial security of the cloud fixing and hardware infrastructure, through the architectural security of function and data deployments, to the actual security of the cloud framework in the presence of peripheral attacks and the mechanisms accessible to respond to and recuperate from these attacks. The use of cloud computing Subashini and Kavitha argue services for many reasons including because this service provide fast access the applications and reduce service costs. Cloud computing providers should address privacy and security as matter for higher and urgent priorities. The dealing with ‘single cloud’ providers is becoming less popular service with customers due to potential problems such as service availability failure for some time and malicious insider’s attacks in the single cloud. So now single cloud move towards ‘multi clouds’, ‘inter-clouds’, or ‘cloud of clouds’.

Aim of the Project, the data security aspect of cloud computing, data and information will be shared with a third party without any hacks. Every cloud users want to avoid un-trusted cloud provider for personal and important documents such as debit/credit cards details or medical report from hackers or malicious insiders is the importance. It supply secure cloud database that will prevent security risks [1]. We apply multi clouds concept using Shamir‟s Secret Sharing algorithm that is reduce risk of data intrusion and loss of service availability for ensuring data [3].

**Chapter 2 Review of Literature**

NIST describes cloud computing as “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”.

**2.1 Cloud Computing Components**

The cloud computing model consists of five characteristics, three delivery models, and four deployment models. The five key characteristics of cloud computing are: location-independent resource pooling, on-demand self-service, rapid elasticity, broad network access, and measured service. These five characteristics represent the first layer in the cloud environment architecture (see Figure 2.1).



Fig. 2.1 Cloud Computing Components

The three key cloud delivery models are infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). In IaaS, the user can benefit from networking infrastructure facilities, data storage and computing services. In other words, it is the delivery of computer infrastructure as a service. An example of IaaS is the Amazon web service. In PaaS, the user runs custom applications using the service provider’s resources. It is the delivery of a computing platform and solution as a service. An example of PaaS is GoogleApps. Running software on the provider’s infrastructure and providing licensed applications to users to use services is known as SaaS. An example of SaaS is the Salesforce.com CRM application. This model represents the second layer in the cloud environment architecture.

Cloud deployment models include public, private, community, and hybrid clouds. A cloud environment that is accessible for multi-tenants and is available to the public is called a public cloud. A private cloud is available for a particular group, while a community cloud is modified for a specific group of customers. Hybrid cloud infrastructure is a composition of two or more clouds (private, community, or public cloud). This model represents the third layer in the cloud environment architecture. Kamara and Lauter present two types of cloud infrastructure only, namely private and public clouds. The infrastructure that is owned and managed by users is in the private cloud. Data that is accessed and controlled by trusted users is in a safe and secure private cloud, whereas the infrastructure that is managed and controlled by the cloud service provider is in a public cloud. In particular, this data is out of the user’s control, and is managed and shared with unsafe and untrusted servers.

**2.2 Cloud Service Providers Examples**

In the commercial world, various computing needs are provided as a service. The service providers take care of the customer's needs by, for example, maintaining software or purchasing expensive hardware. For instance, the service EC2, created by Amazon, provides customers with scalable servers. As another example, under the CLuE program, NSF joined with Google and IBM to offer academic institutions access to a large-scale distributed infrastructure. There are many features of cloud computing. First, cloud storages, such as Amazon S3, Microsoft

SkyDrive, or NirvanixCLoudNAS, permit consumers to access online data. Second, it provides computation resources for users such as Amazon EC2. Third,

Google Apps or versioning repositories for source code are examples of online collaboration tools.

Cloud service providers should ensure the security of their customers’ data and should be responsible if any security risk affects their customers’ service infrastructure. A cloud provider offers many services that can benefit its customers, such as fast access to their data from any location, scalability, pay-for-use, data storage, data recovery, protection against hackers, on-demand security controls, and use of the network and infrastructure facilities. Reliability and availability are other benefits of the public cloud, in addition to low cost. However, there are also concerning issues for public cloud computing, most notably, issues surrounding data integrity and data confidentiality. Any customer will be worried about the security of sensitive information such as medical records or financial information.

**Chapter 3 Report on Present Investigation (Existing System)**

Due to data outsourcing and un-trusted cloud servers, the data access control becomes a challenging issue in cloud storage systems existing access control schemes are no longer applicable to cloud storage systems, because they either produce multiple encrypted copies of the same data or require a fully trusted cloud server [2][8].

**Disadvantage:**

However, cloud storage service separates the roles of the data owner from the data service provider, and the data owner does not interact with the user directly for providing data access service, which makes the data access control a challenging issue in cloud storage systems. Because the cloud server cannot be fully trusted by data owners, traditional server-based access control methods are no longer applicable to cloud storage systems [7][9].

**Chapter 4 Aim and Objective**

Cloud computing concept is relatively new concept but it is based on not so many new technologies. Many of the features that makes cloud computing attractive, however has to meet certain basic security criteria. In our project, we have briefed on various measure on cloud computing security challenges from single to multi clouds. While making a cloud secure, the following objectives are to be met:

* Understanding the cloud computing environment provided by the cloud service provider.
* The cloud computing solution should meet the basic security and privacy requirements of any firm deploying it.
* Maintain an account of the privacy of the cloud and data security and applications that are deployed in cloud computing environment.
* Data Integrity.
* Service Availability.
* The user runs customer applications using the service provider’s resources.

**Chapter 5 Problem Statement**

Even though Security, Privacy and Trust issues exists since the evolution of Internet, the reason why they are widely spoken these days is because of the Cloud Computing scenario. Any client/small organization/enterprise that processes data in the cloud is subjected to an inherent level of risk because outsourced services bypass the "physical, logical and personnel controls" of the user. When storing data on cloud, one might want to make sure if the data is correctly stored and can be retrieved later. As the amount of data stored by the cloud for a client can be enormous, it is impractical (and might also be very costly) to retrieve all the data, if one’s purpose is just to make sure that it is stored correctly.

Hence there is a need to provide such guarantees to a client. Hence, it is very important for both the cloud provider and the user to have mutual trust such that the cloud provider can be assured that the user is not some malicious hacker and the user can be assured of data consistency, data storage and the instance he/she is running is not malicious. Hence the necessity for developing trust models/protocols is demanding [4].

**Chapter 6 Proposed System for Project**

**6.1 Data Integrity:**

It is not an easy task to securely maintain all essential data where it has the need in many applications for clients in cloud computing. To maintain our data in cloud computing, it may not be fully trustworthy because client doesn't have copy of all stored data. But any authors don't tell us data integrity through its user. So we have to establish new proposed system for this using our data reading protocol algorithm to check the integrity of data before and after the data insertion in cloud. Here the security of data before and after is checked by client with the help of CSP using our "effective automatic data reading protocol from user as well as cloud level into the cloud" with truthfulness.

**6.2 Data Intrusion:**

The importance of data intrusion detection systems in a cloud computing environment. We find out how intrusion detection is performed on Software as a Service, Platform as a Service and Infrastructure as Service offerings, along with the available host, network and hypervisor-based intrusion detection options. Attacks on systems and data are a reality in the world we live in. Detecting and responding to those attacks has become the norm and is considered due diligence when it comes to security.

**6.3 Service Availability:**

Service availability is most important in the cloud computing security. Amazon already mentions in its licensing agreement that it is possible that the service might be unavailable from time to time. The user’s web service may terminate for any reason at any time if any user’s files break the cloud storage policy. In addition, if any damage occurs to any Amazon web service and the service fails, in this case there will be no charge to the Amazon Company for this failure. Companies seeking to protect services from such failure need measures such as backups or use of multiple providers.

**Chapter 7 Requirement Analysis (SRS)**

Requirements analysis is the process of analysing the information needs of the end users, the organizational environment and any systems presently being used thereby developing the functional requirements of a system that can meet the needs of the users. Also, the requirements should be recorded in a document, email, user interface. The requirements documentation should be referred to throughout the rest of the system development process to ensure the developing project aligns with the needs and requirements.

**Chapter 8 Scope (Feasibility of Project)**

The very first phase in any system developing life cycle is preliminary investigation. The feasibility study is a major part of this phase. A measure of how beneficial or practical the development of any information system would be to the organization is the feasibility study.

The feasibility of the development software can be studied in terms of the following aspects:

1. Operational Feasibility.

2. Technical Feasibility.

3. Economical feasibility.

4. Legal Feasibility

**8.1 Operational Feasibility:**

The site will reduce the time consumed to maintain manual records and is not tiresome and cumbersome to maintain the records. Hence operational feasibility is assured.

**8.2 Technical Feasibility:**

* At least 2GHz Pentium Processor or Intel compatible processor.
* At least 1 GB RAM.
* Broadband Internet Connection.
* A mouse or other pointing device.
* At least 50 GB free hard disk space.
* Microsoft Internet Explorer 8 or higher.

**8.3 Economical Feasibility:**

Once the hardware and software requirements get fulfilled, there is no need for the user of our system to spend for any additional overhead.

For the user, the web site will be economically feasible in the following aspects:

* The web site will reduce a lot of paper work. Hence the cost will be reduced.
* Our web site will reduce the time that is wasted in manual processes.
* The storage and handling problems of the registers will be solved.

**8.4 Legal Feasibility:**

The licensed copy of the required software is quite cheap and easy to get. So from legal point of view the proposed system is legally feasible.

**Chapter 9 Methodology**

Cloud customers may form their expectations based on their past experiences and organizations needs. They are likely to conduct some sort of survey before choosing a cloud service provider. Customers are expected also to do security checks that are centred on three security concepts: confidentiality, integrity and availability. On the other hand, cloud service providers may promise a lot to entice a customer to sign a deal, but some gaps may manifest later as overwhelming barriers to keep their promises. Many potential cloud customers are well aware of this, and certainly, still sitting on the sidelines. They will not undertake cloud computing unless they get a clear indication that all gaps are within acceptable limits. All relevant information are visualized into cloud computing security in a snapshot which is presented in following figure. We organized cloud computing security into three sections: security categories, security in service delivery models and security dimensions [6].

**9.1 Security in cloud services is based on the following:**

* Strong network security is possible around the service delivery platform
* Data encryption: for data in transit (particularly over wide area networks), and sometimes stored data, but it cannot be applied to data in use
* Access controls to ensure that only authorized users gain access to applications, data and the processing environment and is the primary means of securing cloud-based services



Fig. 9.1 Cloud Computing Security

Service providers are able to inspect activity in their environment and provide reports to clients.

Logs need to be carefully constructed to appraisal the actions of their system administrators and other restricted users or risk producing reports that mix events relating to different customers of the service.

Both the organizations seeking cloud solutions and the service providers have to ensure cloud security is addressed. Some of the measures to ensure security in cloud are good governance, compliance, privacy, Identity and Access Management (IAM), Data protection, Availability, Business Continuity and Disaster Recovery plans etc.

**9.2 Secret Sharing Strategy**

Simply storing the information on multiple clouds solves the problem of data availability, but what about security? Having multiple copies of data into different clouds it will just create multiple gates for intruders to hack in. Therefore, we need to make sure that the data shipped to multiple clouds is safer that it was on a single cloud. This is when we apply the secret sharing algorithm presented by Adi Shamir. Invented in 1979, the algorithm has occupied a huge place in the area of cryptography. The author discussed the issue of information distribution with the aim of showing that there is an orthogonal approach which is based on information distribution instead of encryption. The need of a secure communication between two endpoints challenged most of the work on data security. The mathematical evolution behind the algorithm is more complicated, that’s where the secret sharing algorithm of Shamir lies – in its simplicity of implementation. Shamir’s secret sharing or secret splitting represents a way for distributing a secret among a group of n participants, each of whom is allocated a part of the secret, in our case, a piece of data. The strong point of this method is that the secret can be reconstructed only when a predefined number of shares are combined together; individual shares are of no use on their own, so anyone with fewer than t out of n shares has no extra information about the secret than someone with 0 shares. For example, consider a secret sharing scheme in which our information to be protected is “academia”. This word is divided into the shares: A person with 0 shares knows that the word consists of eight letters. He would have to guess the word from 268 = 208 billion possible combinations.

If he has one share, then the interval is narrowing down to 266 = 308 million combinations, and so on. Thus, an user with fewer than t shares is able to reduce the problem of obtaining the inner secret without first needing to obtain all the necessary shares.

**Chapter 10 Design Details**

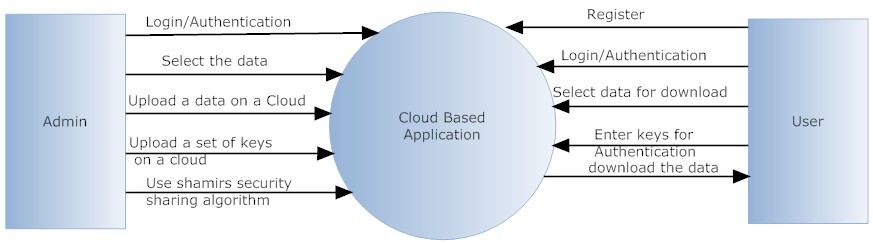


Fig 10.1 Context Level Diagram

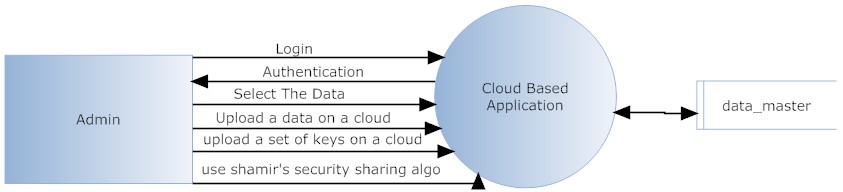


Fig 10.2 (a) DFD Diagram

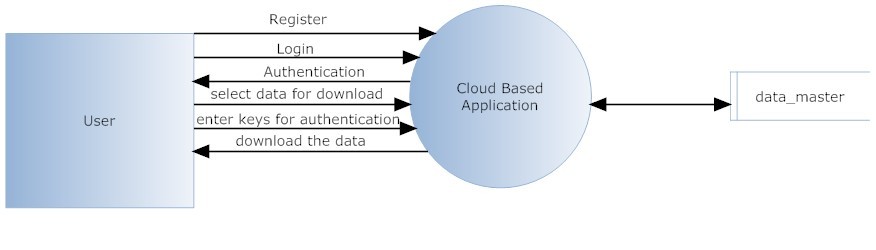


Fig 10.2 (b) DFD Diagram

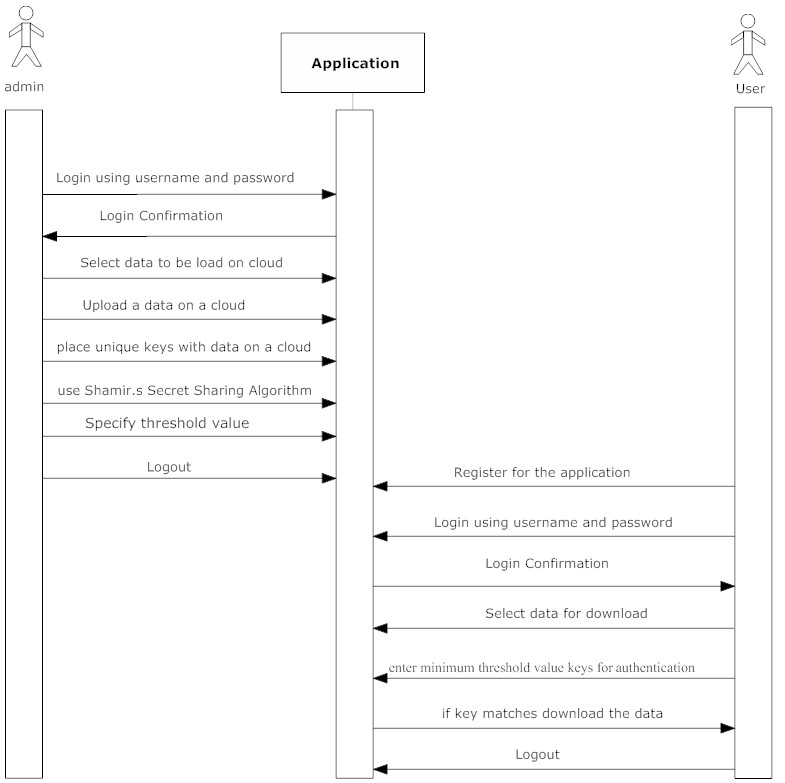
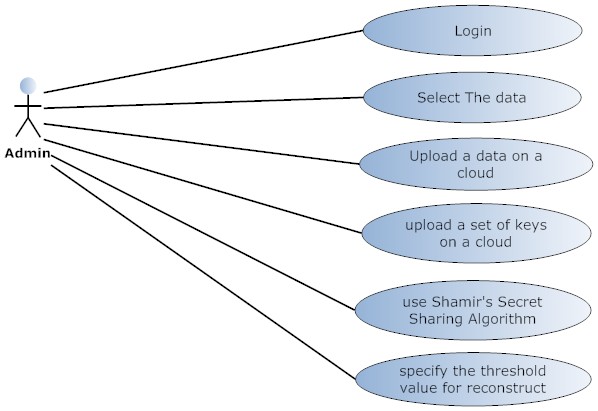
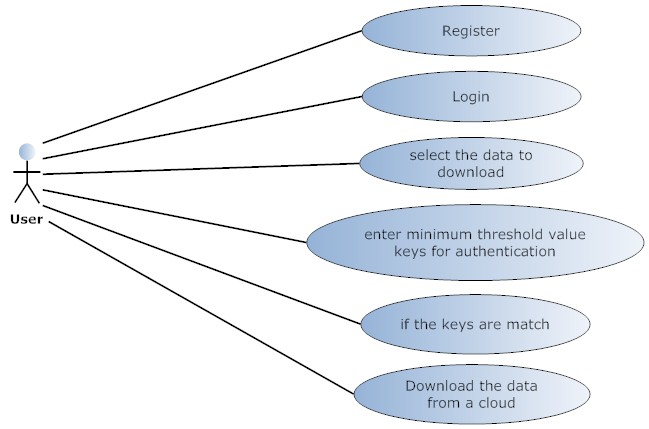


Fig 10.3 Sequence Diagram

 Fig 10.4 Use Case Diagram

 Fig 10.4 (a) Admin Use Case

 Fig 10.4 (b) User Use Case

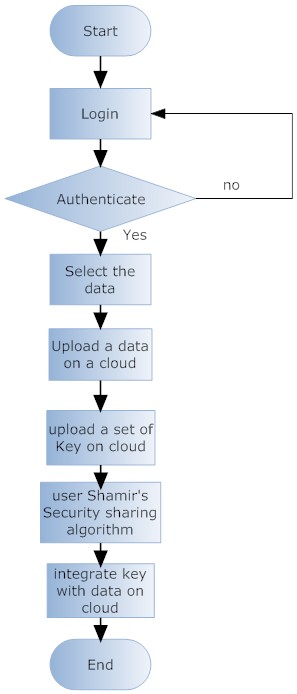


Fig 10.5 (a) Admin Activity Diagram

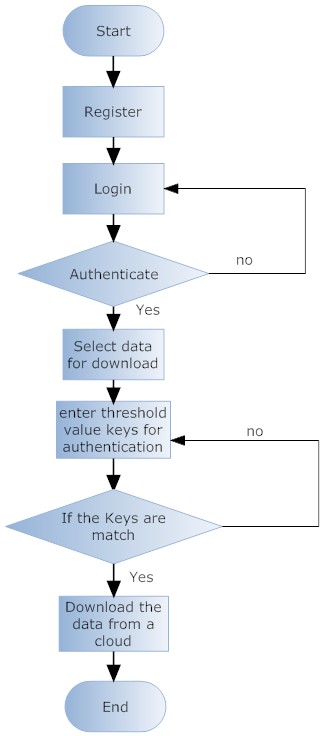


Fig 10.5 (b) User Activity Diagram

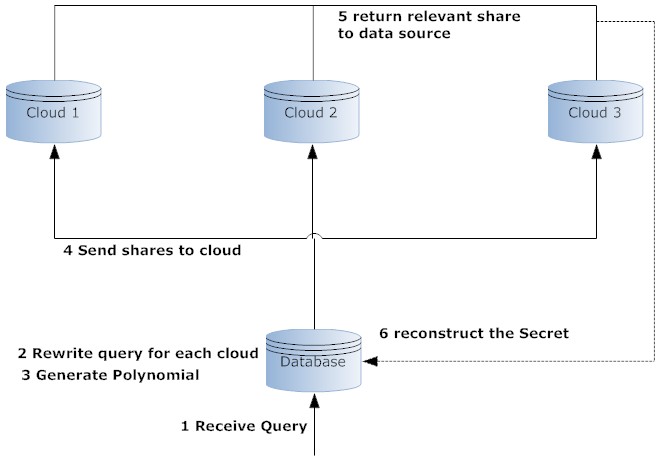


Fig 10.6 System Architecture

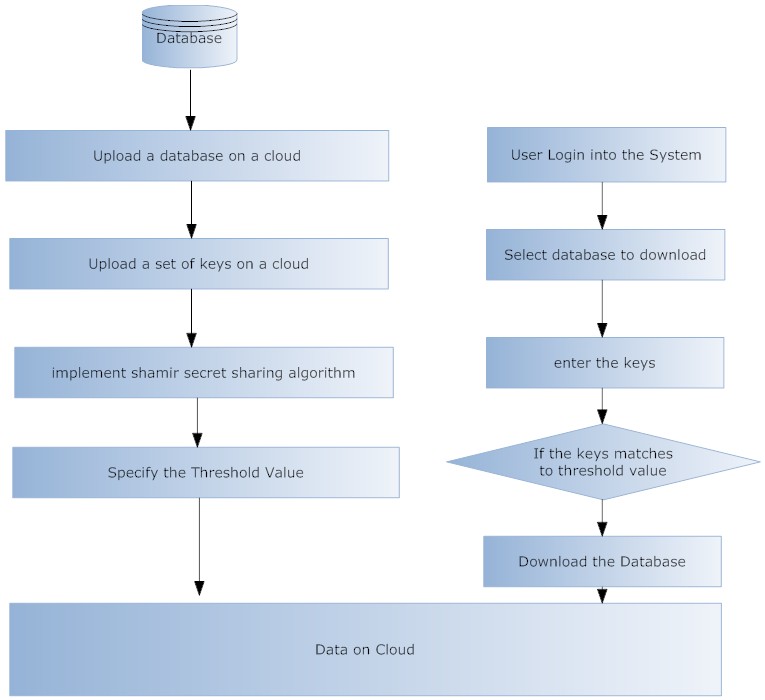


Fig 10.7 Architectural Design

**10.1 Algorithms / Methods Used**

Goal is to divide some data D (e.g., the safe combination) into n pieces D1, D2….Dn in such a way that:

* Knowledge of any k or more D pieces makes D easily computable.
* Knowledge of any k -1 or fewer pieces leaves D completely undetermined (in the sense that all its possible values are equally likely).

This scheme is called (k,n) threshold scheme. If k=n then all participants are required together to reconstruct the secret.

* Suppose we want to use (k,n) threshold scheme to share our secret S where k < n.
* Choose at random (k-1) coefficients a1,a2,a3…ak-1 , and let S be the a0



Construct n points (i, f(i)) where i=1,2…..n

Given any subset of k of these pairs, we can find the coefficients of the polynomial by interpolation, and then evaluate a0=S, which is the secret.

* Let S=1234
* n=6 and k=3 and obtain random integers
* a1=166 and a2=94



* Secret share points
* (1,1494),(2,1942)(3,2598)(4,3402)(5,4414)(6,5614)
* We give each participant a different single point (both x and f(x)) [4], [5].

**Chapter 11 Implementation and Experimental Set Up**

**11.1 Hardware and Software Requirement**

**11.1.1 HARDWARE REQUIREMENTS:**

1 GB RAM.

200 GB HDD.

Intel 1.66 GHz Processor Dual Core

**11.1.2 SOFTWARE REQUIREMENTS:**

Windows XP, Windows 7, 8

Visual Studio 2010

MS SQL Server 2008

**11.2 System Development Life Cycle**

The System Development Life Cycle is the process of developing information systems through investigation, analysis, design, implementation, and maintenance. The System Development Life Cycle (SDLC) is also known as Information Systems Development or Application Development.

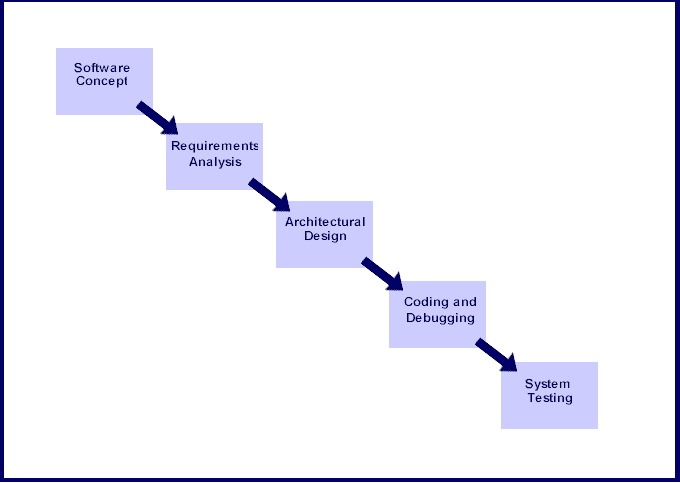


Fig. 11.1 Software Development Life Cycle

Steps involved in the System Development Life Cycle:

Below are the steps involved in the System Development Life Cycle. Each phase within the overall cycle may be made up of several steps.

**Step 1: Software Concept**

The first step is to identify a need for the new system. This will include determining whether a business problem or opportunity exists, conducting a feasibility study to determine if the proposed solution is cost effective, and developing a project plan.

This process may involve end users who come up with an idea for improving their work. Ideally, the process occurs in tandem with a review of the organization's strategic plan to ensure that IT is being used to help the organization achieve its strategic objectives. Management may need to approve concept ideas before any money is budgeted for its development.

**Step 2: Requirements Analysis**

Requirements analysis is the process of analyzing the information needs of the end users, the organizational environment, and any system presently being used, developing the functional requirements of a system that can meet the needs of the users. Also, the requirements should be recorded in a document, email, user interface storyboard, executable prototype, or some other form. The requirements documentation should be referred to throughout the rest of the system development process to ensure the developing project aligns with user needs and requirements.

Professionals must involve end users in this process to ensure that the new system will function adequately and meets their needs and expectations.

**Step 3: Architectural Design**

After the requirements have been determined, the necessary specifications for the hardware, software, people, and data resources, and the information products that will satisfy the functional requirements of the proposed system can be determined. The design will serve as a blueprint for the system and helps detect problems before these errors or problems are built into the final system. Professionals create the system design, but must review their work with the users to ensure the design meets users' needs.

**Step 4: Coding and Debugging**

Coding and debugging is the act of creating the final system. This step is done by software developer.

**Step 5: System Testing**

The system must be tested to evaluate its actual functionality in relation to expected or intended functionality. Some other issues to consider during this stage would be converting old data into the new system and training employees to use the new system. End users will be key in determining whether the developed system meets the intended requirements, and the extent to which the system is actually used.

**Step 6: Maintenance**

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

There are various software process models like:-

* Prototyping Model
* RAD Model
* The Spiral Model
* The Waterfall Model
* The Iterative Model

Of all these process models we’ve used the Iterative model (The Linear Sequential Model) for the development of our project.

**11.2.1 The Iterative model**

The waterfall model derives its name due to the cascading effect from one phase to the other as is illustrated in Figure 11.2. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase.

This model is sometimes referred to as the linear sequential model or the software life cycle.

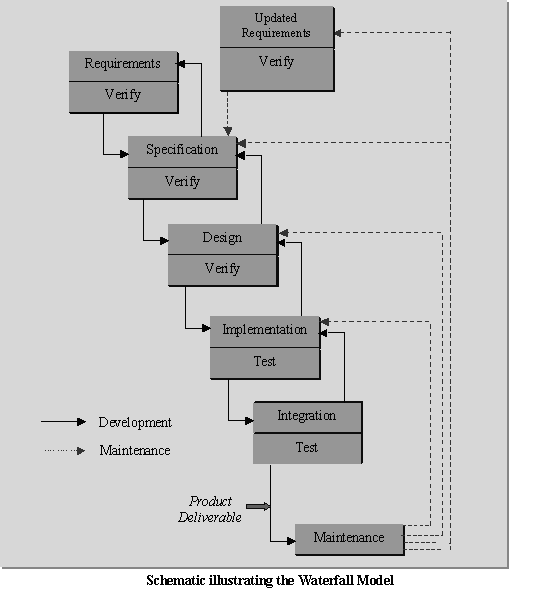


Fig. 11.2 Iterative Model

The model consists of six distinct stages, namely:

1. In the requirements analysis phase
   1. The problem is specified along with the desired service objectives (goals)
   2. The constraints are identified
2. In the specification phase the system specification is produced from the detailed definitions of (a) and (b) above. This document should clearly define the product function.
3. In the system and software design phase, the system specifications are translated into a software representation. The software engineer at this stage is concerned with:

* Data structure
* Software architecture
* Algorithmic detail
* Interface representations

The hardware requirements are also determined at this stage along with a picture of the overall system architecture. By the end of this stage should the software engineer should be able to identify the relationship between the hardware, software and the associated interfaces. Any faults in the specification should ideally not be passed ‘down stream.

1. In the implementation and testing phase stage the designs are translated into the software domain

* Detailed documentation from the design phase can significantly reduce the coding effort.
* Testing at this stage focuses on making sure that any errors are identified and that the software meets its required specification.

1. In the integration and system testing phase all the program units are integrated and tested to ensure that the complete system meets the software requirements. After this stage the software is delivered to the customer [Deliverable – The software product is delivered to the client for acceptance testing.]
2. The maintenance phase the usually the longest stage of the software. In this phase the software is updated to:

* Meet the changing customer needs
* Adapted to accommodate changes in the external environment
* Correct errors and oversights previously undetected in the testing phases
* Enhancing the efficiency of the software

Observe that feed back loops allow for corrections to be incorporated into the model. For example a problem/update in the design phase requires a ‘revisit’ to the specifications phase. When changes are made at any phase, the relevant documentation should be updated to reflect that change.

**Advantages of the Iterative Model:**

* Testing is inherent to every phase of the Iterative model
* It is an enforced disciplined approach
* It is documentation driven, that is, documentation is produced at every stage

**Disadvantages of the Iterative Model:**

The waterfall model is the oldest and the most widely used paradigm. However, many projects rarely follow its sequential flow. This is due to the inherent problems associated with its rigid format. Namely:

* It only incorporates iteration indirectly, thus changes may cause considerable confusion as the project progresses.
* As The client usually only has a vague idea of exactly what is required from the software product, this IM has difficulty accommodating the natural uncertainty that exists at the beginning of the project.
* The customer only sees a working version of the product after it has been coded. This may result in disaster any undetected problems are precipitated to this stage.

**11.3 Gantt Chart**

The following table gives the project plan for the Phase 1 & 2 of our project:

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Description** | **Effort in person weeks** | **Deliverable** |
| Phase 1 |  |  |  |
| P1-01 | Requirement Analysis | 2 weeks | Requirement Gathering |
| P1-02 | Existing System Study & Literature | 3 weeks | Existing System Study & Literature |
| P1-03 | Technology Selection | 2 weeks | .NET |
| P1-04 | Modular Specifications | 2 weeks | Module Description |
| P1-05 | Design & Modeling | 4 weeks | Analysis Report |
|  | **Total** | **13 weeks** |  |

Table 11.1: Project Plan Phase-1

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Description** | **Effort in person weeks** | **Deliverable** |
| Phase 2 |  |  |  |
| P2-01 | Detailed Design | 2 weeks | LLD / DLD Document |
| P2-02 | UI and user interactions design | Included in above | UI document |
| P2-03 | Coding & Implementation | 12 weeks | Code Release |
| P2-04 | Testing & Bug fixing | 2 weeks | Test Report |
| P2-05 | Performance Evaluation | 4 weeks | Analysis Report |
| P2-06 | Release | Included in above | System Release |
|  | **Total** | **20 weeks** | Deployment efforts are extra |

Table 11.2: Project Plan Phase-2

The Gantt Chart Shows planned and actual progress for a number of tasks displayed against a horizontal time scale. It is effective and easy-to-read method of indicating the actual current status for each of set of tasks compared to planned progress for each activity of the set.

Gantt Charts provide a clear picture of the current state of the project.

**Analysis**

Information Gathering

Requirement Analysis

**Design**

**User Interface**

**Coding**

**Testing**

**Documentation**

MAR

FEB

JAN

DEC

NOV

OCT

SEP

AUG

JUL

Figure 11.3: Gantt Chart

**Chapter 12 Testing Technology**

System testing is a critical phase implementation. Testing of the system involves hardware devise and debugging of the computer programs and testing information processing procedures. Testing can be done with text data, which attempts to stimulate all possible conditions that may arise during processing. If structured programming Methodologies have been adopted during coding the testing proceeds from higher level to lower level of program module until the entire program is tested as unit. The testing methods adopted during the testing of the system were unit testing and integrated testing.

**12.1 UNIT TESTING:**

Unit testing focuses first on the modules, independently of one another, to locate errors. This enables the tester to detect errors in coding and logical errors that is contained within that module alone. Those resulting from the interaction between modules are initially avoided.

**12.2 INTEGRATION TESTING:**

Integration testing is a systematic technique for constructing the program structure while at the same time to uncover the errors associated with interfacing. The objective is to take unit-tested module and build a program structure that has been detected by designing. It also tests to find the discrepancies between the system and its original objectives. Subordinate stubs are replaced one at time actual module. Tests were conducted at each module was integrated. On completion of each set another stub was replaced with the real module.

**12.3 FUNCTIONAL TESTING:**

Functional testing is a technique in which all the functionalities of the program are tested to check whether all the functions that where proposed during the planning phases are full filled.

This is also to check that if all the functions proposed are working properly.

This is further done in two phases:

* One before the integration to see if all the unit components work properly
* Second to see if they still work properly after they have been integrated to check if some functional compatibility issues arise.

**12.4 PERFORMANCE TESTING:**

**12.4.1 Expected Result**

* The client should be able to connect to the server properly without any problems.
* The connection establishment between the mobile device and the server should take minimal time.
* The mobile device should be able receive data from the server uninterruptedly.
* Information provided by the application should be correct and as per the user’s need.

**12.4.2 Observation**

* Connection can be established easily provided that the server is on.
* The connection with the server takes time as it uses Internet connection.
* Receiving data from the server takes time.
* Information coming from the database is correct.

**12.5 LOAD / STRESS TESTING:**

**12.5.1 Expected Result**

* Response time should be unaffected irrespective of the no of users.
* The introduction of the newer clients should not make the server to work hap hazardously.
* Continuous use of the server by different clients should not result into the server getting slowed down.
* Response time should not be degraded if there is congestion in network.

**12.5.2 Observation**

* The speed of transmission was fine even when the newer clients were getting added. The response of the server was satisfying even with the introduction of newer client.
  1. **Test Cases**

**Pre-condition:** User has a valid login credentials. And then only user will be able to login and use the application.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item No** | **Test Condition** | **Operator Action** | **Input Specification** | **Output Specification (Expected Results)** | **Pass or Fail** |
| 1 | Successful Login. | 1. Insert Username and Password.  2. Press Login button. | Username and Password | 1. System validates the User ID and Password and depending on the user provides the Info for either Faculty or Student | Pass |
| 2 | Unsuccessful  Login due to  Incorrect password. | 1. Insert Username and Password.  2. Press Login button. | User ID and Password | 1. System validates the User ID and Password and  Pops up message “Invalid User ID or password”. | Pass |
| 3 | Unsuccessful  Login due to No Server Connectivity | 1. Insert User ID and Password.  2. Press the Login button. | User ID and Password | 1. System validates the User ID and Password on server machine that the user which has no started SQL Server started.  Pops up message “Invalid User ID or password”. | Pass |
| 4 | Unsuccessful  Login due to Blank Text boxes. | 1. Press the Login button. | Null Values | 1. System checks that text boxes are empty so pops up the message: “Username & password cannot be blank”. | Pass |

Table 12.1 Test Case for Testing of Login module

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item No** | **Test Condition** | **Operator Action** | **Input Specification** | **Output Specification(Expected Results)** | **Pass or Fail** |
| 1 | Successful  Resource Download | User maintains a Communication with Server | User Request for resource | File is downloaded in the user Machine | Pass |
| 2 | Successful  Resource Upload | Admin uploads resource | Admin selects file for uploads | File is Uploaded in the respected server folder | Pass |

Table 12.2 Test Case for Testing File Upload/Download module

**Chapter 13 Result and Analysis**

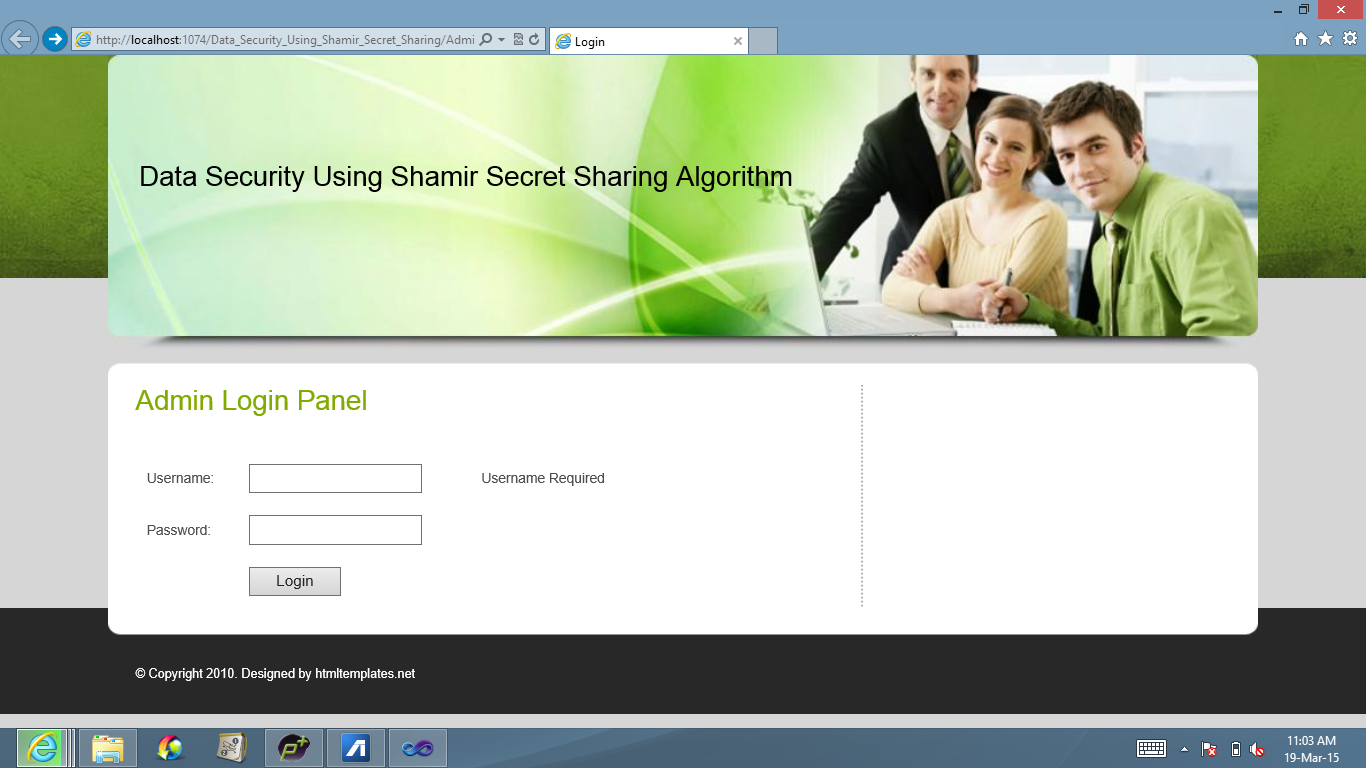


Fig. 13.1 Admin Login Panel

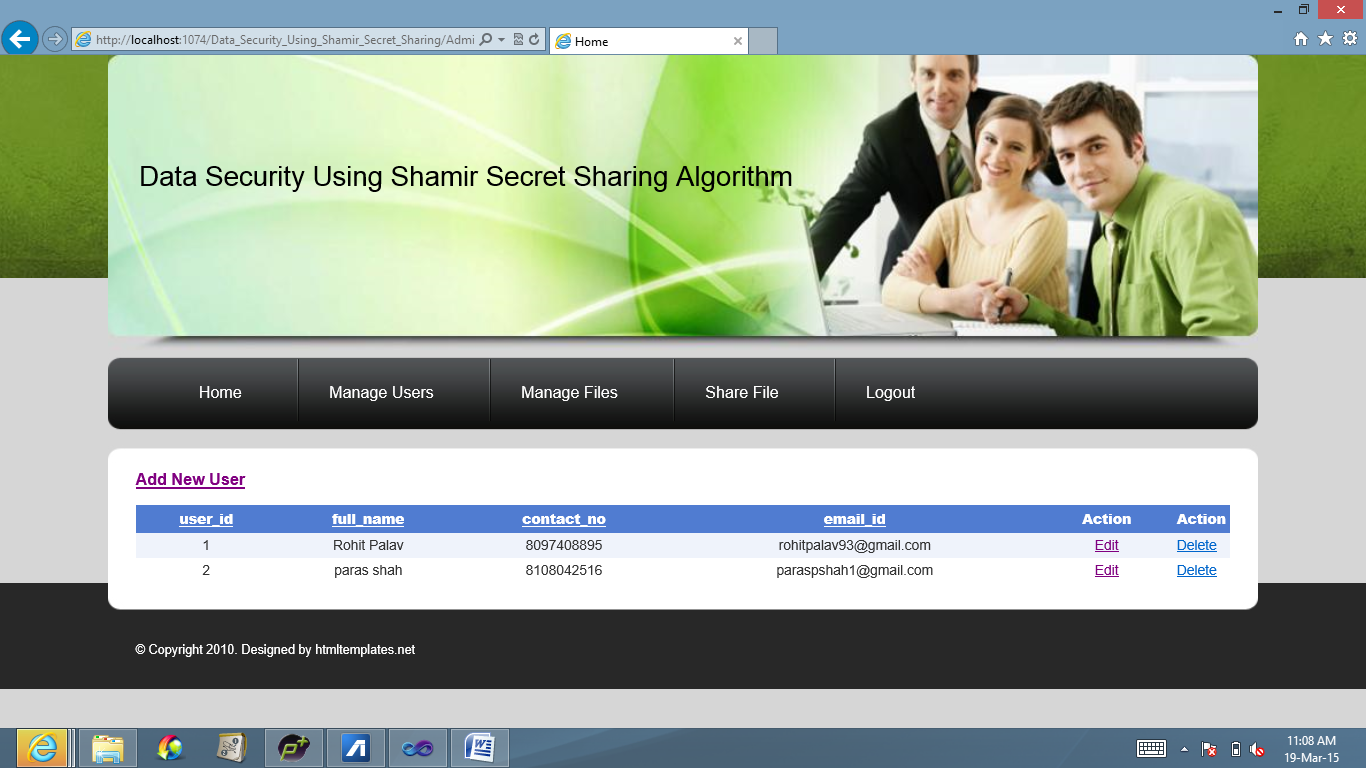


Fig. 13.2 Manage/Add Users

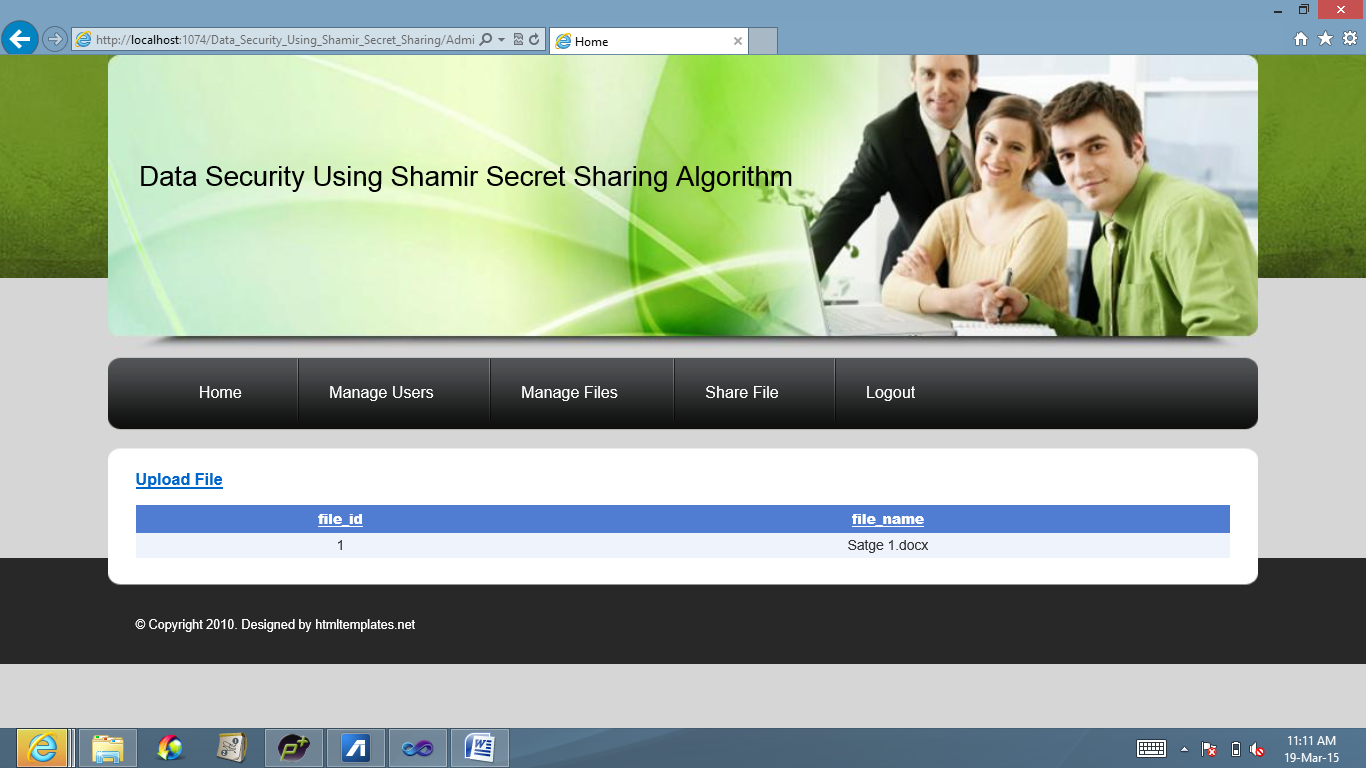


Fig. 13.3 Manage Files

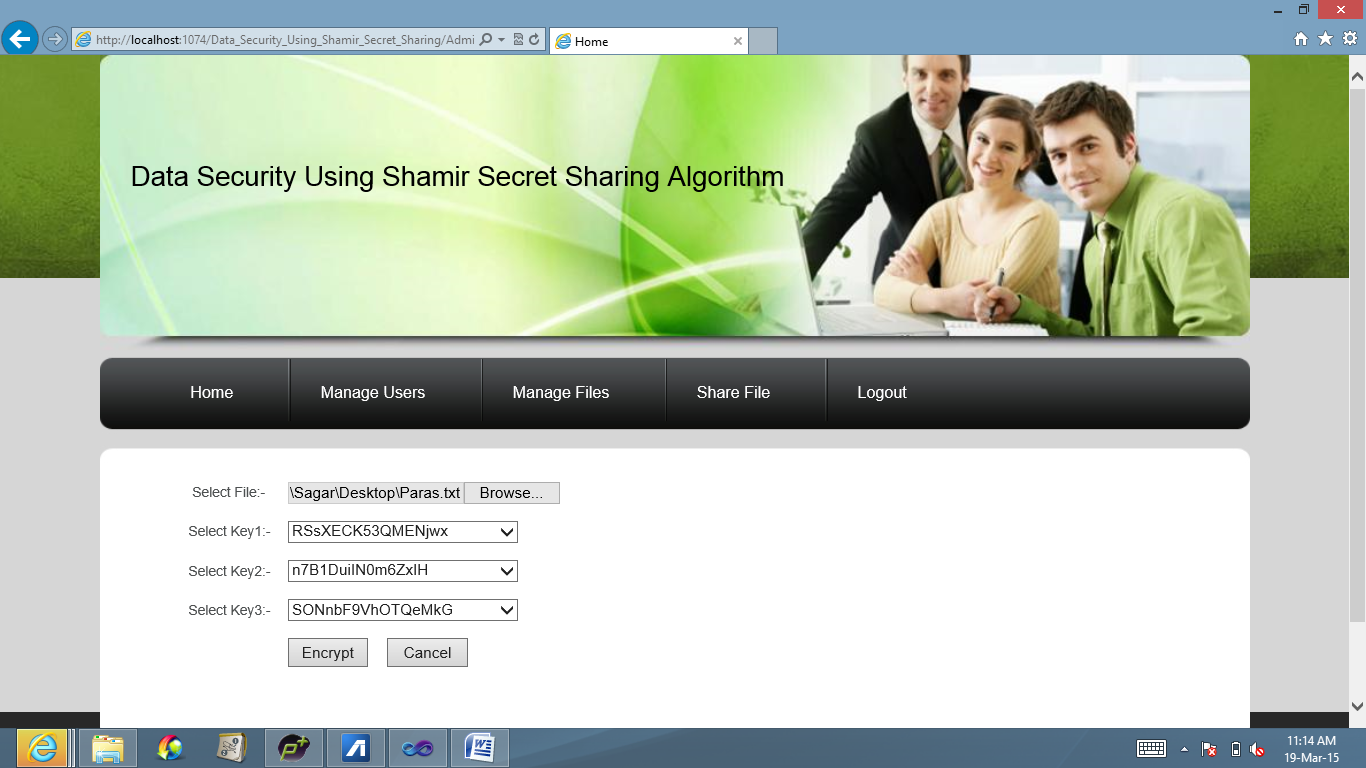


Fig. 13.4 Upload Files

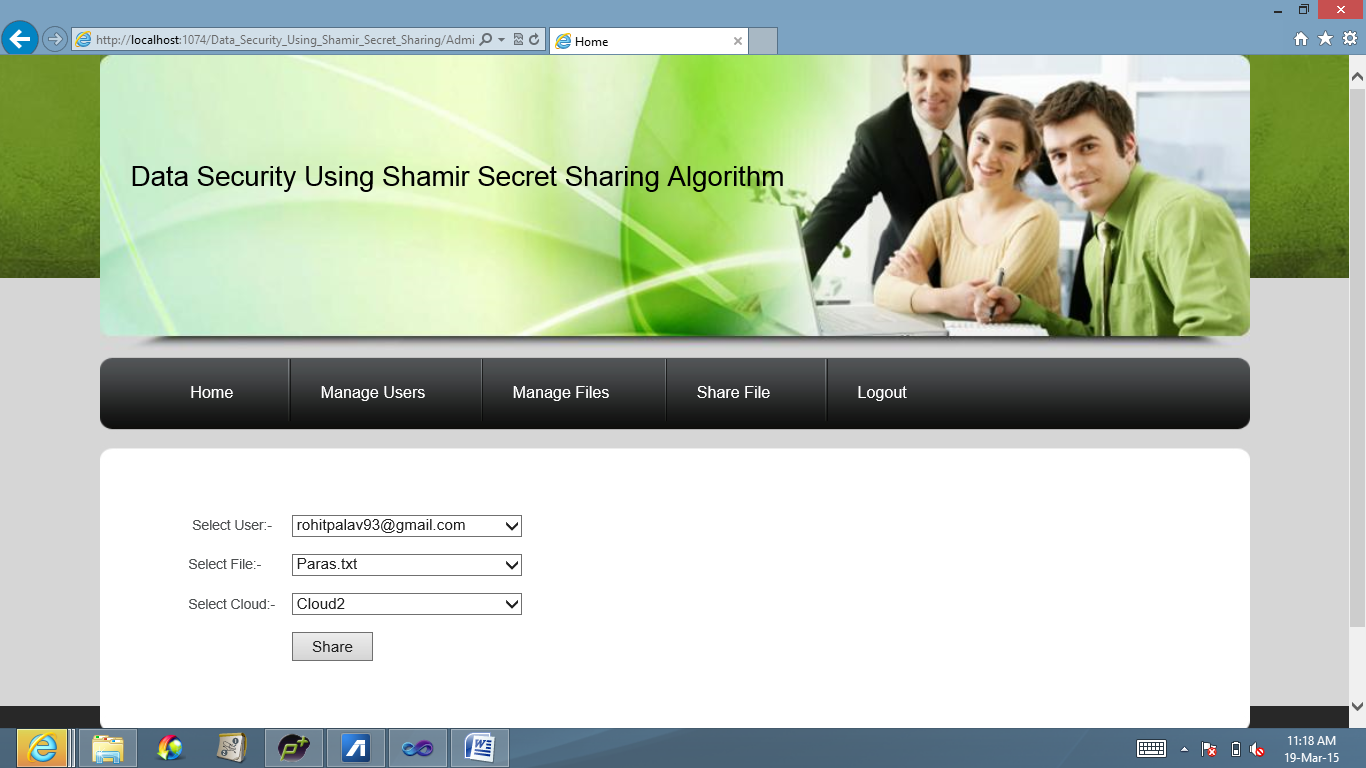


Fig. 13.5 Share File

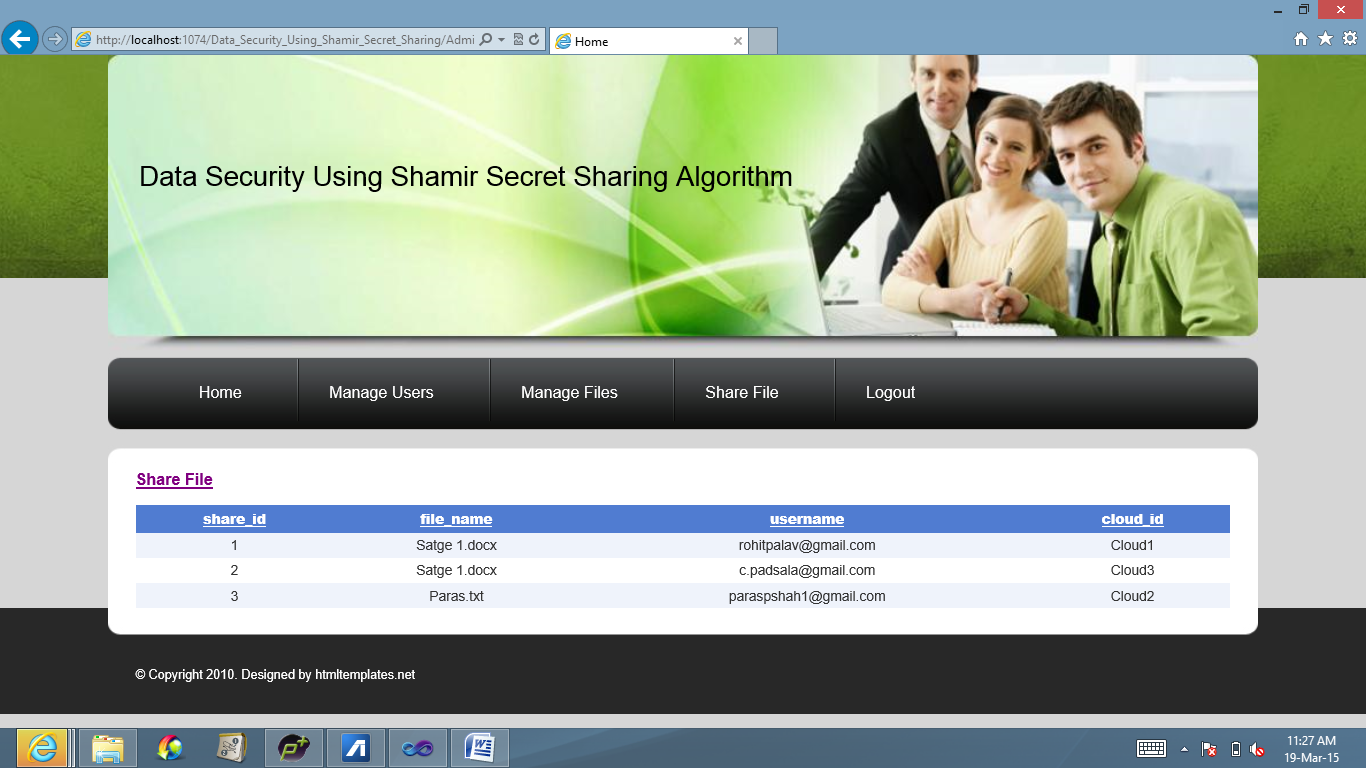


Fig. 13.6 View Shared File(s)

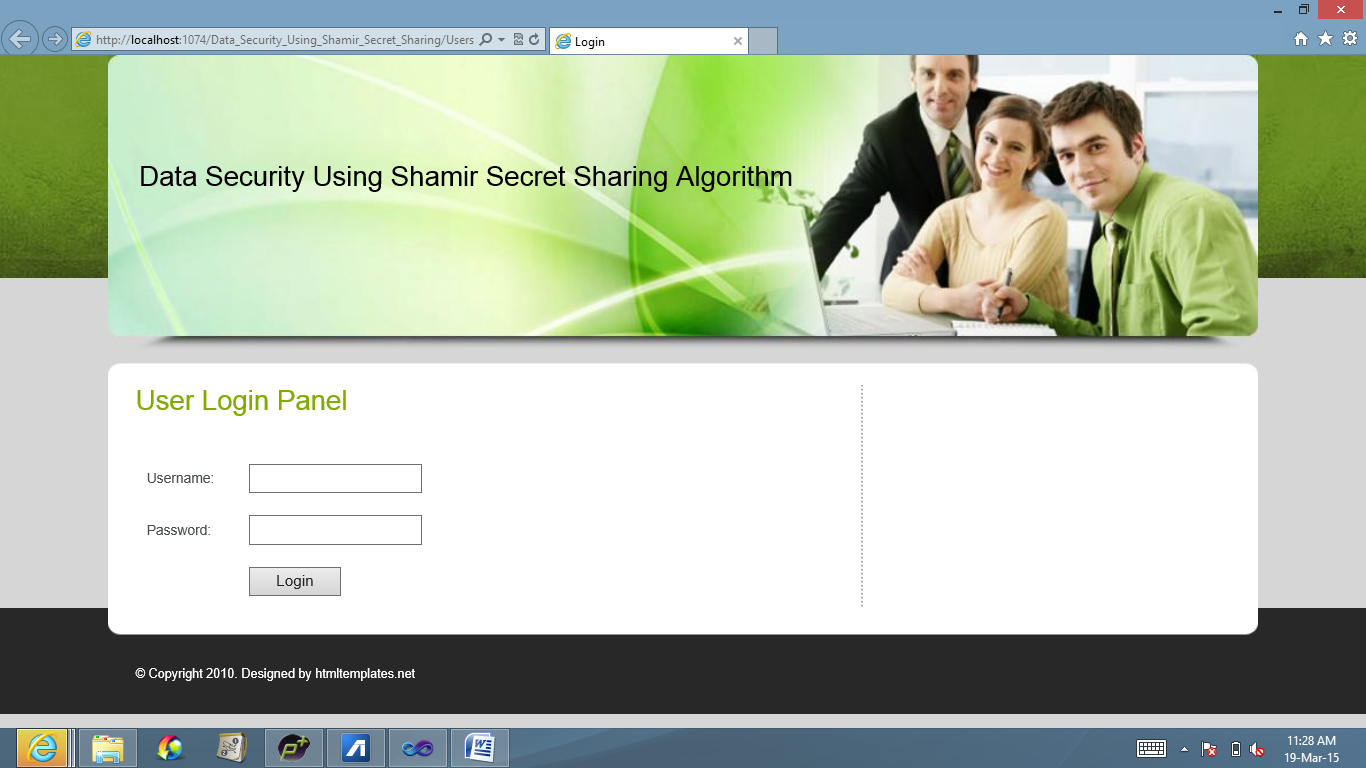


Fig. 13.7 User Login Panel

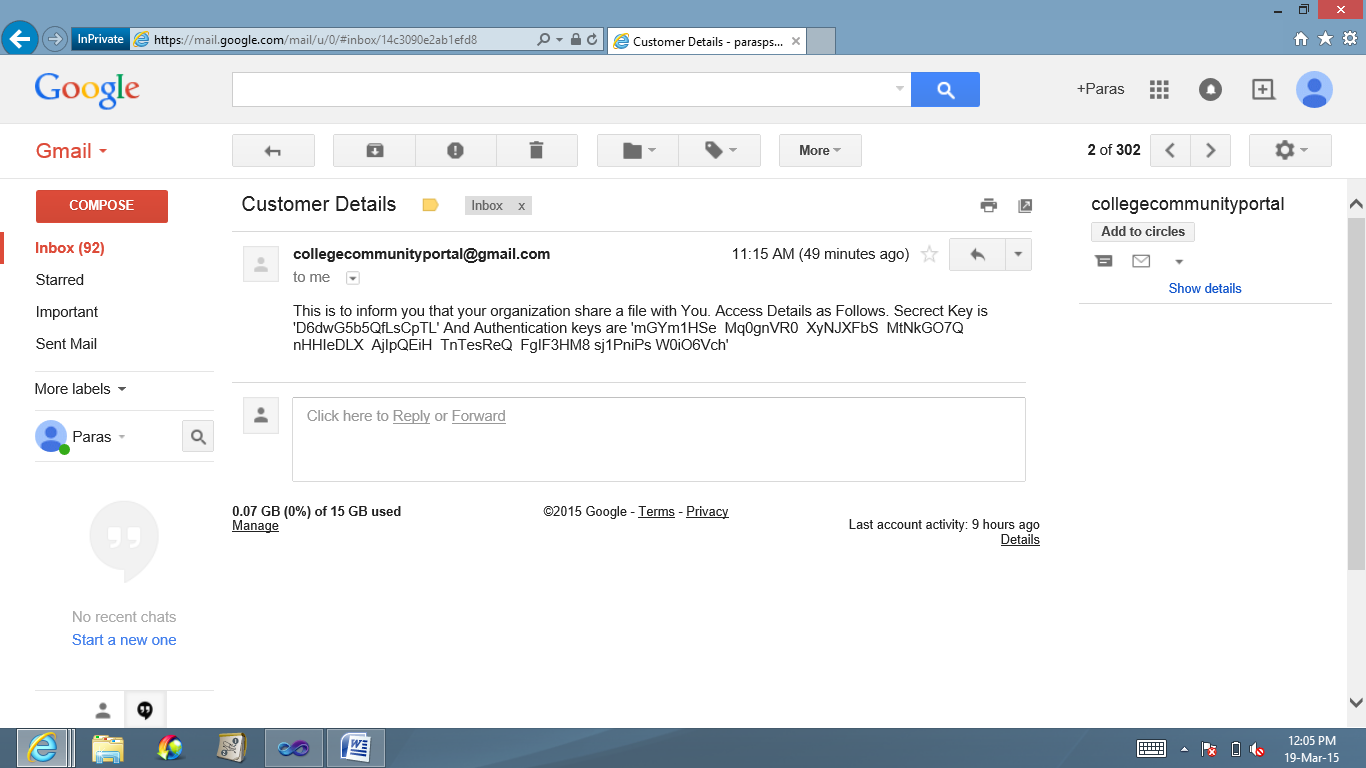


Fig. 13.8 E-Mail Containing Secret Key and Authentication Keys

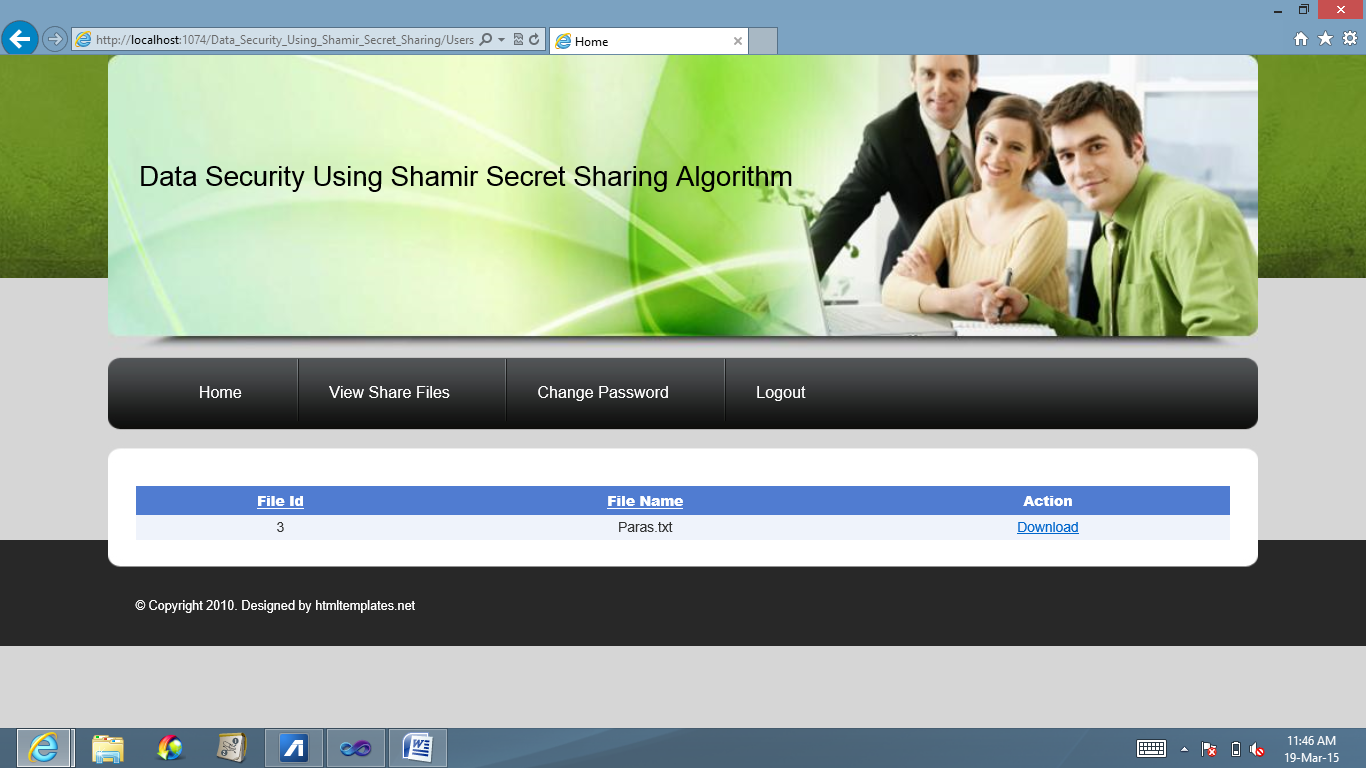


Fig. 13.9 View Shared Files

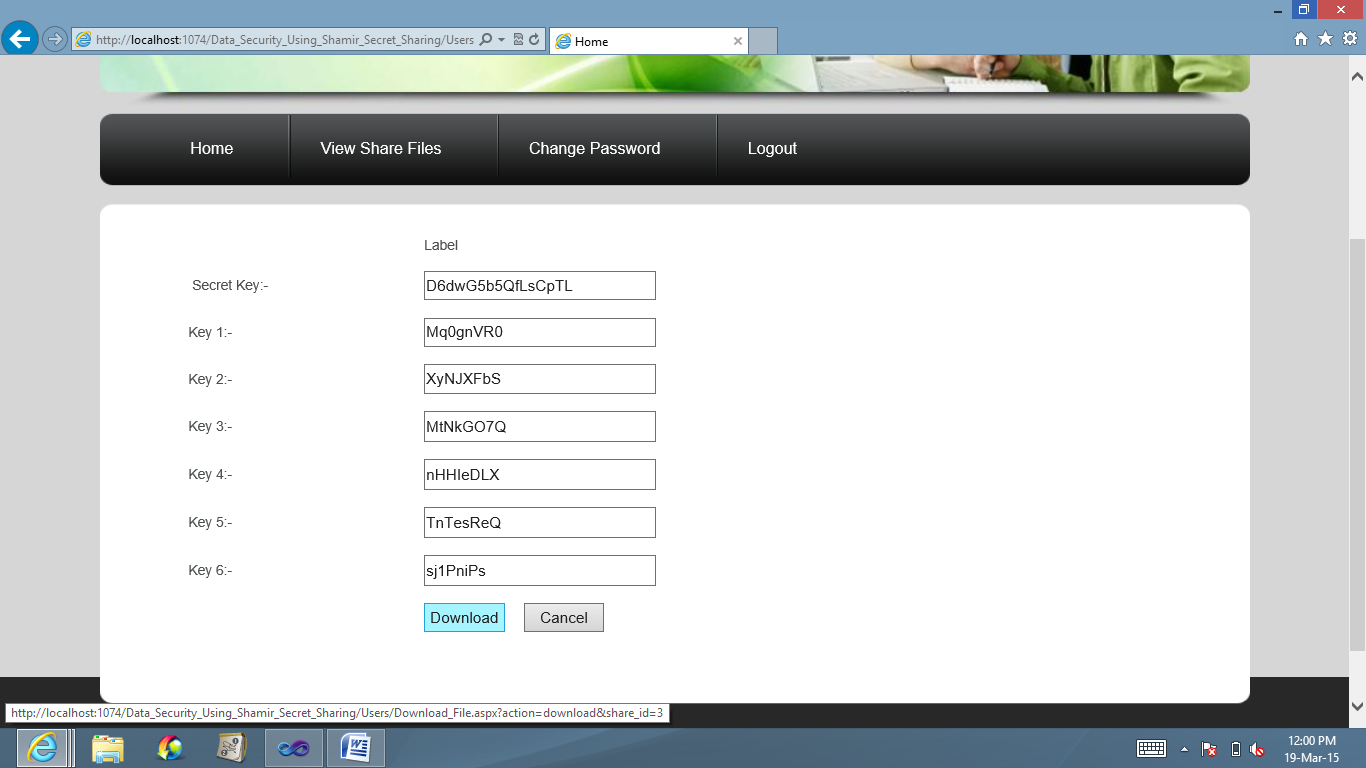


Fig. 13.10 Download Shared File

**Chapter 14 Conclusion**

The benefits of cloud computing are clear: minimizing the risk of physical infrastructure deployment, reducing cost of entry, reducing the execution and response time of applications, etc. Even though CC is extensively researched, security still represents the major issue of it. To this end, this project focuses on the issues related to the security aspects of cloud and aims at facilitating a new model which uses multiple cloud service providers (CSP) and Shamir’s secret sharing algorithm to prevent and overcome all the shortcomings of a single cloud model.

The use of multi-sharing techniques is considered novel. It is proven to be more secure, harder to compromise and superior to the encryption techniques due to their biggest limitation, large time for encryption and decryption. The purpose of this model is to reduce the security risks which occur in cloud computing. Also, we address the issues related to data integrity, confidentiality and service availability arguing why the multi-clouds model is superior to single cloud by giving relevant scenarios where the single cloud approach fails. Therefore, it is clear that storing the data over multi-clouds is efficient, so with the tools and functionality available today, we strongly believe that there is no excuse for not going the multi-cloud route.

**Chapter 15 References**

[1] Md Kausar Alam, Sharmila Banu K “An Approach Secret Sharing Algorithm in Cloud Computing Security over Single to Multi Clouds”

[2] Cloud Computing Security: From Single to Multi-Clouds, 2012 ,45th Hawaii International Conference on System Sciences.

[3] Review of methods for secret sharing in cloud Computing- “International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)”, Volume 2, Issue 1 January 2013

[4] Shamir, Adi (1979), "How to share a secret", Communications of the ACM 22 (11): 612–613

[5] Dawson, E.; Donovan, D. (1994), "The breadth of Shamir's secret-sharing scheme", Computers & Security 13: 69–78

[6] Hassan Takabi, James B.D., Joshi, Gail-Joon, Ahn, “Security and Privacy Challenges in Cloud Computing Environments”, University of Pittsberg, October 2010

[7] Kan Yang, Associate Member, IEEE, Xiaohua Jia, Fellow, IEEE, Kui Ren, Senior Member, IEEE, Bo Zhang, Member, IEEE, and Ruitao Xie, Student Member, IEEE. “DAC-MACS: Effective Data Access Control for Multiauthority Cloud Storage Systems.”

[8] D.boneh, R.canetti, S.halevi, J. Katz “Ciphertext security from identity-based encryption”

[9] Bethencourt, J., Sahai A. and Waters, B. 2007. “Cipher-text Policy Attribute-Based Encryption”

**Chapter 16 Acknowledgement**

It gives us great pleasure in presenting this project report titled:

“**Cloud Computing Security using Secret Sharing Algorithm**”

On this momentous occasion, we wish to express our immense gratitude to the range of people who provided invaluable support in the completion of this project. Their guidance and encouragement has helped in making this project a great success.

We express our gratitude to our project guide Prof. Sachin Sonawane, who provided us with all the guidance and encouragement and making the lab available to us at any time. We also would like to deeply express our sincere gratitude to Project coordinators.

We are eager and glad to express our gratitude to the Head of the Computer Engineering Dept. Prof. Mahendra Patil , for her approval of this project. We are also thankful to her for providing us the needed assistance, detailed suggestions and also encouragement to do the project.

We would like to deeply express our sincere gratitude to our respected principal Prof. Dr.Shrikant Kallurkar and the management of Atharva College of Engineering for providing such an ideal atmosphere to build up this project with well-equipped library with all the utmost necessary reference materials and up to date IT Laboratories

We are extremely thankful to all staff and the management of the college for providing us all the facilities and resources required.

**Chapter 17 Paper Publication Details**

Paper published in IJRASET journal in March 2015 issue.

1. Chirag Padsala, Rohit Palav, Paras Shah, Prof. Sachin Sonawane, “Survey of Cloud Security Techniques”, International Journal for Research in Applied Science and Engineering Technology (IJRASET), Website: http://www.ijraset.com/ (ISSN 2321 – 9653, Volume 3, Issue III, March 2015.