**1. INTRODUCTION**

**1.1 GENERAL**

Back-propagation is an effective method for learning neural networks and has been widely used in various applications. The accuracy of the learning result, despite other facts, is highly affected by the volume of high quality data used for learning. As compared to learning with only local data set, collaborative learning improves the learning accuracy by incorporating more data sets into the learning process: the participating parties carry out learning not only on their own data sets, but also on others’ data sets. With the recent remarkable growth of new computing infrastructures such as Cloud Computing, it has been more convenient than ever for users across the Internet, who may not even know each other, to conduct joint/collaborative learning through the shared infrastructure.

Despite the potential benefits, one crucial issue pertaining to the Internet-wide collaborative neural network learning is the protection of data privacy for each participant. In particular, the participants from different trust domains may not want to disclose their private data sets, which may contain privacy or proprietary information, to anybody else. In applications such as healthcare, disclosure of sensitive data, e.g., protected health information (PHI)[2], is not only a privacy issue but of legal concerns according to the privacy rules

**1.2 CHALLENGES**

Theoretically, secure multi-party computation (SMC) can be used to solve problems of this kind. But the extremely high computation and communication complexity of SMC, due to the circuit size, usually makes it far from practical even in the two-party case. In order to provide practical solutions for privacy preserving back-propagation neural (BPN) network learning,three main challenges need to be met simultaneously:1) To protect each participant’s private dataset and intermediate results generated during the BPN network learning process, it requires secure computation of various operations, e.g. addition, scalar product and the nonlinear sigmoid function, which are needed by the BPN network algorithm; 2) To ensure the practicality of the proposed solution, the computation/communication cost introduced to each participant shall be affordable.In order to accommodate a large range of collaborative learning, the proposed solution shall consider system scalability. In particular, it shall be able to support an arbitrary number of participants without introducing tremendous computation/communication costs to each participant. 3) For collaborative training, the training data sets may be owned by different parties and partitioned in arbitrary ways rather than a single way of partition.

**1.3 OBJECTIVE**

To aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable

**1.4 EXISTING SYSTEM**

Existing schemes supporting this kind of collaborative learning are either limited in the way of data partition or just consider two parties. There lacks a solution that allows two or more parties, each with an arbitrarily partitioned data set, to collaboratively conduct the learning.

**1.4.1 Overall Diagram**

**Input Data Encrypted**

Encrypt

Cloud server

**Decrypted**

Verify & Decrypt

**Output**

Fig 1.1: Existing System Diagram

**1.4.2 Existing System**

* Secure Multi-Party computation(SMC)

**1.4.3 Drawbacks in Existing System**

* The participants from different trust domains may not want to disclose their private data sets, which may contain privacy or proprietary information, to anybody else.

**1.5 PROPOSED SYSTEM**

In this work, we address this open problem by incorporating the computing power of the cloud. The main idea of our scheme can be summarized as follows: each participant first encrypts her/his private data with the system public key and then uploads the cipher texts to the cloud; cloud servers then execute most of the operations pertaining to the learning process over the cipher texts and return the encrypted results to the participants; the participants jointly decrypt the results with which they update their respective weights for the BPN network. During this process, cloud servers learn no privacy data of a participant even if they collude with all the rest participants.

**1.5.1 Overall Diagram**

Trusted authority

Cloud servers

Fig 1.2: Proposed System Diagram

**1.5.2 Proposed Algorithm**

* Back- Propagation Neural Network Learning Algorithm

**1.5.3 Advantages in Proposed System**

* Provides privacy preservation for multi-party collaborative BPN network learning over arbitrarily partitioned data.
* To support multi-party secure scalar product and introduce designs that allows decryption of arbitrary large messages.

**1.6 COMPARISON OF EXISTING AND PROPOSED SYSTEM**

|  |  |
| --- | --- |
| **Existing System** | **Proposed System** |
| supporting this kind of collaborative learning are either limited in the way of data partition  or just consider two parties. | Lacks a solution that allows two or more parties, each with an arbitrarily partitioned data set, to collaboratively conduct the learning. |
| The User cannot encrypts his/her private data locally, The cloud system only encrypts data and sore in the database | Each party encrypts his/her private data locally and uploads the ciphertexts into the cloud. The cloud then executes most of the operations pertaining to the learning algorithms over ciphertexts without knowing the original private data. |
| Offloading the expensive operations to the cloud, the computation cost is maximum to the participants. | Securely offloading the expensive operations to the cloud, we keep the computation and communication costs on each party minimal and independent to the number of participants. |
| Our scheme can be implemented with difficult communication protocol design needs to communicate with the cloud server. | Our scheme can be implemented with easy communication protocol design since each party only needs to communicate with the cloud server. |
| Our scheme can efficiently handle only single large dataset for learning, which is compared to the linearly increasing computation/ communication cost of existing works. | Our scheme can efficiently handle the relative large dataset for learning, which is compared to the linearly increasing computation/ communication cost of existing works. |

**1.7 SCOPE OF THE PROJECT**

To aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable.

**2. LITERATURE SURVEY**

**2.1 RELATED WORK**

Several privacy preserving BPN network learning schemes have been proposed recently.Schlitter[19] introduces a privacy preserving BPN network learning scheme that enables two or more parties to jointly perform BPN network learning without disclosing their respective private data sets. But the solution is proposed only for horizontal partitioned data.

Moreover, this scheme cannot protect the intermediate results, which may also contain sensitive data, during the learning process. Chen et. al.[6] proposes a privacy preserving BPN network learning algorithm for twoparty scenarios. This scheme provides strong protection for data sets including intermediate results. However,it just supports vertically partitioned data. To overcome this limitation, Bansal et. al.[4] enhanced this scheme and proposed a solution for arbitrarily partitioned data.Nevertheless, this enhanced scheme, just like [6], was proposed for the two-party scenario. Directly extending them to the multi-party setting will introduce a computation/communication complexity quadratic in the number of participants. In practical implementation, such a complexity represents a tremendous cost on each party considering the already expensive operations on the underlying groups such as Elliptic Curves. However, [4] just considers the two-party scenario though it supports arbitrarily partitioned dataset. To our best knowledge, none of existing schemes have solved all these challenges at the same time. There still lacks an efficient and scalable solution that supports collaborative BPN network learning with privacy preservation in the multi-party setting and allows arbitrarily partitioned datasets.

**1.Title:** Privacy Preserving Back-Propagation Neural Network Learning Over Arbitrarily Partitioned Data

**Author:** Ankur Bansal, Tingting Chen, Sheng Zhong.

**Year :** 2011

**Description:**

Neural Networks have been an active research area for decades. However, privacy bothers many when the training dataset for the neural networks is distributed between two parties, which is quite common nowadays. Existing cryptographic approaches such as secure scalar product protocol provide a secure way for neural network learning when the training dataset is vertically partitioned. In this paper we present a privacy preserving algorithm for the neural network learning when the dataset is arbitrarily partitioned between the two parties. We show that our algorithm is very secure and leaks no knowledge (except the final weights learned by both parties) about other party’s data. We demonstrate the efficiency of our algorithm by experiments on real world data.

**2.Title:** Evaluating 2-DNF Formulas on Ciphertexts.

**Author:** Dan Boneh, Eu-Jin Goh, Kobbi Nissim.

**Year :** 2005

**Description:**

Let be a 2-DNF formula on boolean variables x1, . . . , xn 2 {0, 1}. We present a Homomorphic public key encryption scheme that allows the public evaluation of given an encryption of the variables x1, . . . , xn. In other words, given the encryption of the bits x1, . . . , xn, anyone can create the encryption of (x1, . . . , xn). More generally, we can evaluate quadratic multi-variate polynomials on cipher texts provided the resulting value falls within a small set. We present a number of applications of the system: In a database of size n, the total communication in the basic step of the Kushilevitz-Ostrovsky PIR protocol is reduced from An efficient election system based on homomorphic encryption where voters do not need to include non-interactive zero knowledge proofs that their ballots are valid. The election system is proved secure without random oracles but still efficient .A protocol for universally verifiable computation.

**3.Title:** Privacy Preserving Back-Propagation Neural Network Learning.

**Author:** Tingting Chen, Sheng Zhong.

**Year** :2009

**Description:**

With the development of distributed computing environment, many learning problems now have to deal with distributed input data. To enhance coorperations in learning, it is important to address the privacy concern of each data holder by extending the privacy preservation notion to original learning algorithms. In this paper, we focus on preserving the privacy in an important learning model, multi-layer neural networks. We present a privacy preserving two-party distributed algorithm of back-propagation which allows a neural network to be trained without requiring either party to reveal her data to the other.We provide complete correctness and security analysis of our algorithms. The effectiveness of our algorithms is verified by experiments on various real world datasets.

**4.Title:** Over encryption: Management of Access Control Evolution on Outsourced Data.

**Author:**  Sabrina De Capitani di Vimercati, Sara Foresti, Sushil Jajodia.

**Year :** 2007

**Description:**

Data outsourcing is emerging today as a successful paradigm allowing users and organizations to exploit external services for the distribution of resources. A crucial problem to be addressed in this context concerns the enforcement of selective authorization policies and the support of policy updates in dynamic scenarios .In this paper, we present a novel solution to the enforcement of access control and the management of its evolution.Our proposal is based on the application of selective encryptionas a means to enforce authorizations. Two layers of encryption are imposed on data: the inner layer is imposed by the owner for providing initial protection, the outer layer is imposed by the server to reflect policy modifications. The combination of the two layers provides an efficient and robust solution. The paper presents a model, an algorithm for the management of the two layers, and an analysis to identify and therefore counteract possible information exposure risks.

**5. Title:** An Empirical Study of Learning Speed in Back-Propagation Networks.

**Author:** Scott E. Fahlman

**Year :** 1988

**Description:**

Most connectionist or "neural network" learning systems use some form of the back-propagation algorithm.However, back-propagation learning is too slow for many applications, and it scales up poorly as tasks become larger and more complex. The factors governing learning speed are poorly understood. I have begun a systematic,empirical study of learning speed in backprop-like algorithms, measured against a variety of benchmark problems.The goal is twofold: to develop faster learning algorithms and to contribute to the development of a methodology that will be of value in future studies of this kind. This paper is a progress report describing the results obtained during the first six months of this study. To date I have looked only at a limited set of benchmark problems, but the results on these are encouraging: I have developed a new learning algorithm that is faster than standard backprop by an order of magnitude or more and that appears to scale up very well as the problem size increases. This research was sponsored in part by the National Science Foundation under Contract Number EET-8716324 and by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 4976 under Contract F33615-87- C-1499 and monitored by the Avionics Laboratory, Air Force Wright Aeronautical Laboratories, Aeronautical Systems Division (AFSC), Wright-Patterson AFB, OH 45433-6543. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of these agencies or of the U.S. Government.

**6.Title:** Training a SVM-Based Classifier In Distributed Sensor Networks.

**Author:**K. Flouri1, B. Beferull-Lozano2, and P. Tsakalides

**Year:** 2006

**Description:**

The emergence of smart low-power devices (motes), which have micro-sensing, on-board processing, and wireless communication capabilities, has impelled research in distributed and on-line learning under communication constraints. In this paper, we show how to perform a classification task in a wireless sensor network using distributed algorithms for Support Vector Machines (SVMs), taking advantage of the sparse representation that SVMs provide for the decision boundaries. We present two energy-efficient algorithms that involve a distributed incremental learning for the training of a SVM in a wireless sensor network, both for stationary and non-stationary sample data (concept drift). Through analytical studies and simulation experiments, we show that the two proposed algorithms exhibit similar performance to the traditional centralized SVM training methods, while being much more efficient in terms of energy cost.

**7.Title:** Protocols for Secure Computations

**Author:** Andrew c.yao.

**Year:** 1982

**Description:**

Two millionaires wish to know who is richer; however,they do not want to find out inadvertently any additional information about each other’s wealth. How can they carry out such a conversation? This is a special case of the following general problem. Suppose m people wish to compute the value of a function f(x1, x2, x3, . . . , xm), which is an integer-valued function of m integer variables xi of bounded range. Assume initially person Pi knows the value of xi and no other x’s. Is it possible for them to compute the value off, by communicating among themselves, without unduly giving away any information about the values of their own variables? The millionaires’ problem corresponds to the case when m = 2 and f(x1, x2) = 1 if x1 < x2, and 0 otherwise. In this paper, we will give precise formulation of this general problem and describe three ways of solving it by use of one-way functions (i.e., functionswhich are easy to evaluate but hard to invert). These results have applications to secret voting, private querying of database, oblivious negotiation, playing mental poker, etc. We will also discuss the complexity question “How many bits need to be exchanged for the computation”,and describe methods to prevent participants from cheating. Finally, we study the question “What cannot be accomplished with one-way functions”. Before describing these results, we would like to put this work in perspective by first considering a unified view of secure computation in the next section.

**8. Title:** Handwritten Digit Recognition With a Back Propagation.

**Author:** Y.Le Cun,B.Boser,J.S.Denker,D.Henderson

**Year:** 1990

**Description:**

We present an application of back propagation network to written digit recognition minimal processing of the data was required,but the architechture of the network was highly constrained and specifically designed for the task the input of the network consist of normalized images of isolated digits.

**9.Title:** Handbook of Mathematical Functions

**Author:** M.Abramowitz and I.A.Stegun

**Year:**1964

**Description:**

The present handbook has been designed to provide scientific investigatiors with a comprehensive and self-contained summary of the mathematical functions that rise in physical and engineering problems.The book contains numerical tables and by giving large collection of mathematical properties of the tabulated functions.

**10.Title:** Handwritten digit recognition with a back-propagation network

**Author:**L.Cun,B.Boser,J.S.Denker,D.Henderson,R.E.Howard,W.Hubbard,L.D.jackel

**Year:**1990

**Description:**

The main point of this paper is to show that large back-propagation network can be applied to real image recognition problems without a large,complex preprocessing stage requiring detailed engineering.Unlike most previous work on this subject,the learning network is directly fed with images,rather than feature vectors,thus demonstarting the ability of BP networks to deal with large amountsof low level information

**11.Title:**A Public key cryptosystem and a signature scheme based on discrete logarithms

**Author:**T.El.Gamal

**Year:**1985

**Description:**

A new signature scheme is proposed together with an implementation of the Diffie-Hellmen key distribution scheme that achieves a publi key cryptosystem.The security of both system relies on the difficulty of computing discrete algorithm over finite fields

**2.2 CLOUD COMPUTING**

Cloud computing is basically an Internet-based network made up of large numbers of servers - mostly based on open standards, modular and inexpensive.  Clouds contain vast amounts of information and provide a variety of services to large numbers of people. The benefits of cloud computing are **Reduced Data Leakage, Decrease evidence acquisition time, they eliminate or reduce service downtime, they Forensic readiness, they Decrease evidence transfer time The main factor to be discussed is security of cloud computing, which is a risk factor involved in major computing fields.**

Imagine yourself in the world where the users of the computer of today’s internet world don’t have to run, install or store their application or data on their own computers, imagine the world where every piece of your information or data would reside on the Cloud (Internet).

As a metaphor for the Internet, "the cloud" is a familiar cliché, but when combined with "computing", the meaning gets bigger and fuzzier. Some analysts and vendors define cloud computing narrowly as an updated version of utility computing: basically virtual servers available over the Internet. Others go very broad, arguing anything you consume outside the firewall is "in the cloud", including conventional outsourcing.

Cloud computing comes into focus only when you think about what we always need: a way to increase capacity or add capabilities on the fly without investing in new infrastructure, training new personnel, or licensing new software. Cloud computing encompasses any subscription-based or pay-per-use service that, in real time over the Internet, extends ICT's existing capabilities.

Cloud computing is at an early stage, with a motley crew of providers large and small delivering a slew of cloud-based services, from full-blown applications to storage services to spam filtering. Yes, utility-style infrastructure providers are part of the mix, but so are SaaS (software as a service) providers such as Salesforce.com. Today, for the most part, IT must plug into cloud-based services individually, but cloud computing aggregators and integrators are already emerging.

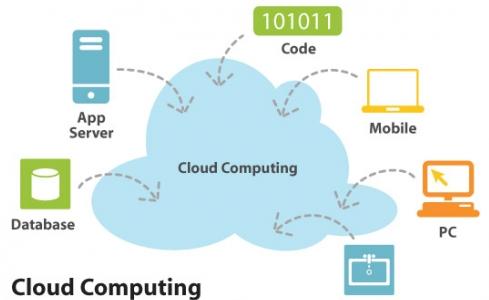


Fig 2.1: Cloud Computing

User of the cloud only care about the service or information they are accessing - be it from their

PCs, mobile devices, or anything else connected to the Internet - not about the underlying details

of how the cloud works.”

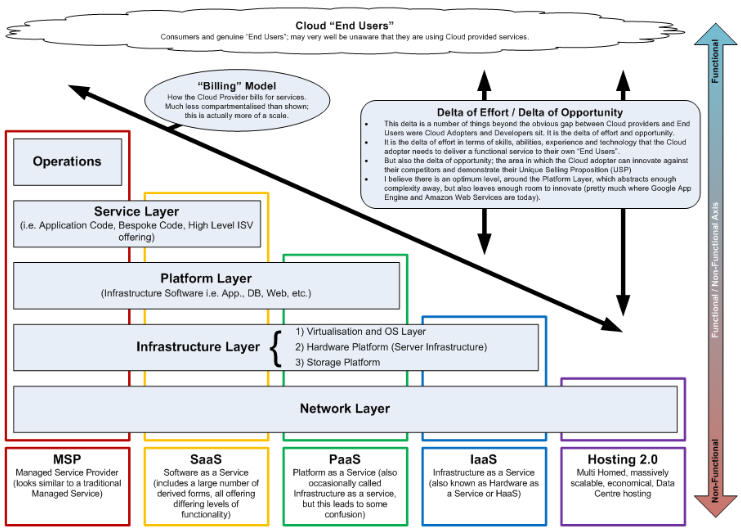
**2.2.1 Cloud Computing Architecture**

**Cloud architecture:** The systems architecture of the software systems involved in the delivery of cloud computing, comprises hardware and software designed by a cloud architect who typically works for a cloud integrator. It typically involves multiple cloud components communicating with each other over application programming interfaces, usually web services.

This closely resembles the UNIX philosophy of having multiple programs doing one thing well and working together over universal interfaces. Complexity is controlled and the resulting systems are more manageable than their monolithic counterparts.

Cloud architecture extends to the client, where web browsers and/or software applications access cloud applications.

Cloud storage architecture is loosely coupled, where metadata operations are centralized enabling the data nodes to scale into the hundreds, each independently delivering data to applications or user.

****

**Fig 2.2: Cloud Computing Architecture**

The Cloud Computing is driving in two types of categories .They are as follows:

1.Customer perspective

2.Vendor perspective

**Customer perspective**:

* In one word: economics
* Faster, simpler, cheaper to use cloud computation.
* No upfront capital required for servers and storage.
* No ongoing for operational expenses for running datacenter.
* Application can be run from anywhere.

**Vendor perspective:**

* Easier for application vendors to reach new customers.
* Lowest cost way of delivering and supporting applications.
* Ability to use commodity server and storage hardware.
* Ability to drive down data center operational cots.
* Computer hardware (Dell, HP, IBM, Sun Microsystems)
  + Storage (Sun Microsystems, EMC, IBM)
  + Infrastructure (Cisco Systems)
* Computer software (3tera, Hadoop, IBM, RightScale)
  + Operating systems (Solaris, AIX, Linux including Red Hat)
  + Platform virtualization (Citrix, Microsoft, VMware, Sun xVM, IBM)

## 2.2.2 Type of services:

## These services are broadly divided into three categories:

## Infrastructure-as-a-Service (IaaS)

## Platform-as-a-Service (PaaS)

## Software-as-a-Service (SaaS).

### Infrastructure-as-a-Service (IaaS):

[Infrastructure-as-a-Service](http://en.wikipedia.org/wiki/Infrastructure_as_a_service)(IaaS) like Amazon Web Services provides virtual servers with unique

IP addresses and blocks of storage on demand. Customers benefit from an [API](http://en.wikipedia.org/wiki/API) from which they

can control their servers. Because customers can pay for exactly the amount of service they use,

like for electricity or water, this service is also called utility computing.

### Platform-as-a-Service (PaaS):

[Platform-as-a-Service](http://en.wikipedia.org/wiki/Platform_as_a_service)(PaaS) is a set of software and development tools hosted on the provider's

servers. Developers can create applications using the provider's APIs. Google Apps is one of the

most famous Platform-as-a-Service providers. Developers should take notice that there aren't any

interoperability standards (yet), so some providers may not allow you to take your application and

put it on another platform.

### Software-as-a-Service (SaaS):

### [Software-as-a-Service](http://en.wikipedia.org/wiki/Software_as_a_service) (SaaS) is the broadest market. In this case the provider allows the

### customer only to use its applications. The software interacts with the user through a user

### interface. These applications can be anything from web based email, to applications like

### [Twitter](http://en.wikipedia.org/wiki/Twitter) or [Last.fm](http://en.wikipedia.org/wiki/Last.fm).

## 2.2.3 Types by visibility:

## 

Fig 2.3: Types of visibility

### Public cloud:

Public cloud or external cloud describes cloud computing in the traditional mainstream sense,

whereby resources are dynamically provisioned on a fine-grained, self-service basis over the

Internet, via [web applications](http://en.wikipedia.org/wiki/Web_application)/[web services](http://en.wikipedia.org/wiki/Web_service), from an off-site third-party provider who shares

[resources](http://en.wikipedia.org/wiki/Multitenancy) and bills on a fine-grained [utility computing](http://en.wikipedia.org/wiki/Utility_computing) basis.

### Hybrid cloud:

A hybrid cloud environment consisting of multiple internal and/or external providers[]](http://en.wikipedia.org/wiki/Cloud_computing#cite_note-54#cite_note-54) "will be

typical for most enterprises". A hybrid cloud can describe configuration combining a local device,

such as a [Plug computer](http://en.wikipedia.org/wiki/Plug_computer) with cloud services. It can also describe configurations combining virtual

and physical, [colocated](http://en.wikipedia.org/wiki/Colocation_centre) assets—for example, a mostly virtualized environment that requires physical

[servers](http://en.wikipedia.org/wiki/Servers), [routers](http://en.wikipedia.org/wiki/Routers), or other hardware such as a network appliance acting as a firewall or spam filter

### Private cloud:

Private cloud and internal cloud are [neologisms](http://en.wikipedia.org/wiki/Neologisms) that some vendors have recently used to describeofferings that emulate cloud computing on private networks. These typically [virtualisation](http://en.wikipedia.org/wiki/Platform_virtualization) products claim to "deliver some benefits of cloud computing without the pitfalls",capitalising on data security, corporate governance, and reliability concerns. They have been criticized on the basis that users "still have to buy, build, and manage them" and as such do notbenefit from lower up-front capital costs and less hands-on management[, essentially "[lacking]the economic model that makes cloud computing such an intriguing concept".

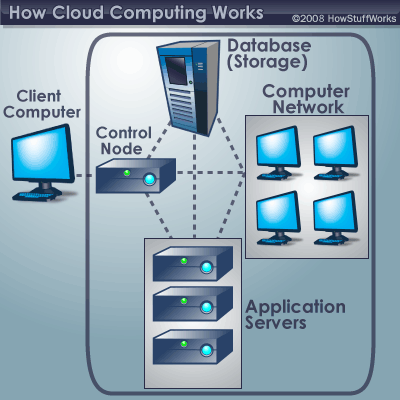
While an analyst predicted in 2008 that private cloud networks would be the future of corporate IT, there is some uncertainty whether they are a reality even within the same firm. Analysts also claim that within five years a "huge percentage" of [small and medium enterprises](http://en.wikipedia.org/wiki/Small_and_medium_enterprises) will get most of their computing resources from external cloud computing providers as they "will not have economies of scale to make it worth staying in the IT business" or be able to afford private clouds. Analysts have reported on [Platform's](http://en.wikipedia.org/wiki/Platform_Computing) view that private clouds are a stepping stone to external clouds, particularly for the financial services, and that future datacenters will look like internal clouds.

**2.2.4 How does cloud computing work?**

Supercomputers today are used mainly by the military, government intelligence agencies,

Universities and research labs, and large companies to tackle enormously complex calculations for such tasks as simulating nuclear explosions, predicting climate change, designing airplanes, and analyzing which proteins in the body are likely to bind with potential new drugs.

Cloud computing aims to apply that kind of power—measured in the tens of trillions ofcomputations per second—to problems like analyzing risk in financial portfolios, deliveringpersonalized medical information, even powering immersive computer games, in a way that userscan tap through the Web. It does that by networking large groups of servers that often uselow-cost consumer PC technology, with specialized connections to spread data-processingchores across them. By contrast, the newest and most powerful desktop PCs process only about3 billion computations a second. Let's say you're an executive at a large corporation. Yourparticular responsibilities include making sure that all of your employees have the righthardware and software they need to do their jobs. Buying computers for everyone isn't enough -- you also have to purchase software or **software licenses** to give employees the tools they require. Whenever you have a new hire, you have to buy more software or make sureyour current software license allows another user. It's so stressful that you find it difficult to go.



**2.2.5 A typical cloud computing system**­

Soon, there may be an alternative for executives like you. Instead of installing a suite of Software for each computer, you'd only have to load one application. That application would allow workers to log into a Web-based service which hosts all the programs the user would need for his or her job. Remote machines owned by another company would run everything from e-mail to word processing to complex data analysis programs. It's called **cloud computing**, and it could change the entire computer industry. In a cloud computing system, there's a significant workload shift. Local computers no longer have to do all the heavy lifting when it comes to running applications. The network of computers tha tmake up the cloud handles them instead. Hardware and software demands on the user's side decrease. The only thing the user's computer needs to be able to run is the cloud computing system's **interface software**, which can be as simple as a Web browser, and the cloud's network takes care of the rest.There's a good chance you've already used some form of cloud computing. If you have ane-mail account with a Web-based e-mail service like Hotmail, Yahoo! Mail or Gmail, then you've had some experience with cloud computing. Instead of running an e-mail program on your computer, you log in to a Web e-mail account remotely. The software and storage for your account doesn't exist on your computer -- it's on the service's computer cloud.

**3. REQUIREMENT ENGINEERING**

**3.1 HARDWARE REQUIREMENTS**

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the system do and not how it should be implemented.

**Hardware**

PROCESSOR : PENTIUM IV 2.6 GHz, Intel Core 2 Duo.

RAM : 512 MB DD RAM

MONITOR : 15” COLOR

HARD DISK : 40 GB

**3.2 SOFTWARE REQUIREMENTS**

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team’s progress throughout the development activity.

Front End : J2EE (JSP, SERVLET)

Back End : MS SQL 2000/05

Operating System : Windows 07

IDE : Net Beans, Eclipse

**3.3 FUNCTIONAL REQUIREMENTS**

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behaviour, and outputs. The proposed system is achieved by implementing the evolving agent behaviour classification and distribution.

**3.4 NON-FUNCTIONAL REQUIREMENTS**

**3.4.1 Efficiency:**

The approach we present is generalizable to all kinds of user behaviors represented by a sequence of events.

**4. PROJECT DESCRIPTION**

**4.1 GENERAL**

In this paper, we aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable.

**4.2 PROBLEM DEFINITION**

In this work, we address this open problem by incorporating the computing power of the cloud. The main idea of our scheme can be summarized as follows: each participant first encrypts her/his private data with the system public key and then uploads the cipher texts to the cloud; cloud servers then execute most of the operations pertaining to the learning process over the cipher texts and return the encrypted results to the participants; the participants jointly decrypt the results with which they update their respective weights for the BPN network. During this process, cloud servers learn no privacy data of a participant even if they collude with all the rest participants.

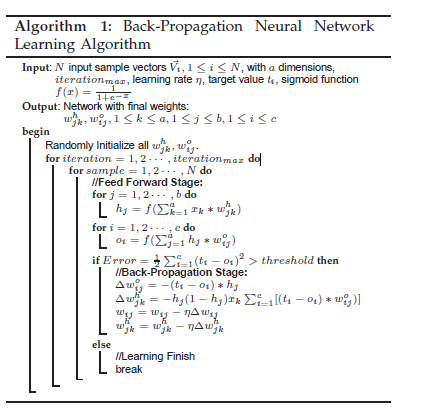
**4.3 METHODOLOGY**

* **Technique Used:**

**Back Propagation Neural Network Learning Algorithm:**

In this project, we aim at enabling multiple parties to jointly conduct BPN network learning without revealing their private data. The input data sets owned by the parties can be arbitrarily partitioned. The computational and communicational costs on each party shall be practically efficient and the system shall be scalable.

To introduce cloud based privacy preserving multi-party BPN network learning algorithm over arbitrarily partitioned data.



In our Project we use **Front End** as Java and **Back End** as a SQL Server 2005.

* **Jdk 1.7:**

In our project we are using java to design the application process. Java contains swing packages that are used to design the view page easily. Since java is an open source and platform independent this makes the application more flexible.

* **SQL Server 2005:**

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are different workloads (ranging from small applications that store and retrieve data on the same computer, to millions of users and computers that access huge amounts of data from the Internet at the same time).

**4.4 TECHNOLOGIES USED**

In this Project

* Collections
* JSP
* Servlet
* Thread
* SQL Server
* **JSP:**

In our project we are using JSP to design the application process. JSP pages are using to develop the form pages like login and user registration pages. it means it is mainly useful for user Interaction development. And some static content of html pages to jsp pages for dynamic content.

* **Servlet:**

In our project we are using servlet to control the application process. Servlet is the center of our application because all the controlling part will be monitoring by the Servlet only. It means Servlet takes requests and matches for suitable jsp’s and it is also useful for database controlling.

* **Collections:**

The Java Collections, Java API's provide Java developers with a set of classes and interfaces that makes it easier to handle collections of objects. Collections are useful to storing the values of a page as input to the backend and from backend to the front end carrying through network. Some of them are Array List, Hash Map.

* **Thread:**

In this project threading concept is very important. Why because, to increase the speed of the service process when more number of users interact. Thread is a independent part of the same program it may execute independently.

## Creating threads:

Java's creators have graciously designed two ways of creating threads: implementing an interface and extending a class. Extending a class is the way Java inherits methods and variables from a parent class. In this case, one can only extend or inherit from a single parent class. This limitation within Java can be overcome by implementing interfaces, which is the most common way to create threads. (Note that the act of inheriting merely allows the class to be run as a thread. It is up to the class to start() execution, etc.).

Interfaces provide a way for programmers to lay the groundwork of a class. They are used to design the requirements for a set of classes to implement. The interface sets everything up, and the class or classes that implement the interface do all the work. The different set of classes that implement the interface have to follow the same rules

* **SQL Server:**

Inour project we are using a backend as SQL Server 2005. Here we are create and maintaining the tables which are having values used for our processes. We are maintaining the registration table, parties private information, records details table etc.

**5. DESIGN ENGINEERING**

**5.1 SYSTEM DESIGN**

System Design sits at the technical kernel of the software engineering process The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customers view into a finished software product or system. Software Design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system-one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure ,program structure and procedural details are developed, reviewed and documented. System Design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities-architectural design, data structure design, interface design and procedural design.

**5.2 MODULES**

The different kinds of modules are:

* User Interface Design
* Arbitary Partitioned Data with Key Generation
* BGN Homomorphic Encryption
* Secure Scalar and Sharing Data with cloud
* Analysis

1. Security Analysis

2. Numerical Analysis

**5.2.1 User Interface Design:**

Trusted Authority

user

user

Login to user

Cloud Server

Fig 5.2.1 : User Interface Design

To connect with Cloud .user must give their username and password then only they can able to connect the server. If the user already exits directly can login into the server else user must register their details such as username, password, Email id, City and Country into the server. Server will create the account for the entire user to maintain upload and download rate. Name will be set as user id. . Logging in is usually used to enter a specific page, which trespassers cannot see. Once the user is logged in, the login token may be used to track what actions the user has taken while connected to the site.

**5.2.2 Arbitrary Partitioned Data with key Generation:**

User

Random Key Generation

Cloud Server

Fig 5.2.2: Arbitrary Partitioned Data With Key Generation

We have generated key for user security. This key have gave to individual participating parties. This key has used to enter the cloud server. This key value did not know another user. Keys were sending secretly to individual user.

**5.2.3 BGN Homomorphic Encryption:**

Trusted Authority

Key Value

Product and Addition operation

Encryption

Fig 5.2.3: BGN Homomorphic Encryption

Homomorphic encryption enables operations on plaintexts to be performed on their respective cipher texts without disclosing the plaintexts. We have introduced a public-key ‘doubly homomorphic’ encryption scheme (called ‘BGN’ for short), which simultaneously supports one multiplication and unlimited number of addition operations. BGN Homomorphic encryption algorithm to support the multi-party scenario, which can be used as an independent solution for other related applications BGN algorithm just supports one step multiplication over cipher text, the intermediate results.

**5.2.4 Secure Scalar and Sharing Data with Cloud:**

Trusted Authority

Key Value

User u3

User u2

User u1

U3 Data

U2 Data

U1 Data

Fig 5.2.4: Secure scalar and sharing data with cloud

We have proposed multiple parties to perform secure scalar product and homomorphic addition operations on cipher texts using cloud computing. We have introduced designs that allow decryption of arbitrary large messages. The product and addition computation used to help of cloud. To support consecutive multiplications, the parties need to decrypt the intermediate results first. Decryption of the intermediate values will reveal these values to the parties, which may have privacy implications and shall be avoided. To protect these intermediate results (scalar products or sum), which enables each participating party to get a random value of the intermediate result without knowing its actual value.

**5.2.5 Analysis:**

* **Security Analysis:**

Propose an algorithm that allows multiple parties to perform secure scalar product and homomorphic addition operations on cipher texts using cloud computing. Party encrypts her/his data with the system public key and uploads the cipher texts to the cloud. The cloud servers compute the sum of original messages based on their cipher texts.

Organization 1

Organization 2

Fig 5.2.5: Security Analysis

* **Numerical Analysis:**

In practice, however, it is hard to guarantee that the final results (numbers) are always small enough for the Pollard’s lambda method to efficiently decrypt. This is either because the numbers contained in the vectors are too large, or the vectors are too long (of high dimension). To overcome this limitation, we propose to let the data holders divide the numbers, if they are large, into several numbers, and the cloud then decrypt the smaller ”chunks” with which the final result can be recovered. The decryption process can be parallelized for efficiency.

Cloud Provider

Access data

Shared encrypted data

Fig 5.2.6 Numerical Analysis

**Given Input Expected Output:**

**1. User Interface:**

Input: Username and password

Output: Validation

**2. Arbitrary Partitioned data with key Generation:**

Input: Username and password

Output: Key generation

**3. BGN Homomorphic Encryption;**

Input: Plaintext,generated key value

Output: Ciphertext

**5.3 UML DIAGRAMS:**

Design Engineering deals with the various UML [Unified Modelling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering. Design is the means to accurately translate customer requirements into finished product.

A model is a simplification of reality.We build models of complex system because we cannot comprehend such a system in entirety.

Architects design buildings,builders use the design to create buildings.The more complicated the building,the more critical communication between architect and builder.Blueprints are the standard graphical languagethat both architects and builders must learn as a part of their trade.

The UML is applicable to object oriented prblem solving.Any one interested in UML the construction of a model.A model is a abstraction of the underlying problem.The domain is the actual world from where the problem comes.

**Different Types Of Modeling Techniques used in UML:**

**1. Structural Modeling:** It consist of - Class Diagram

Object Diagram

**2.Behavioral Modeling:** It consist of - Use case Diagarm

Activity Diagram

State Chart Diagram

Interaction Diagram

a. Sequence Diagram

b. Collaboration Diagram

**3. Architectural Modeling:** It consist of-Component Diagram

Deployment Diagram

**5.3.1 Class Diagram:**

A Class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing the systems classes, their attributes, and the relationships between the classes. **Class Diagram**  is **UML**  [structure diagram](http://www.uml-diagrams.org/uml-24-diagrams.html#structure-diagram) which shows structure of the designed system at the level of [classes](http://www.uml-diagrams.org/class-diagrams.html#class) and [interfaces](http://www.uml-diagrams.org/class-diagrams.html#interface), shows their [features](http://www.uml-diagrams.org/uml-core.html#feature), [constraints](http://www.uml-diagrams.org/constraint.html?context=class-diagrams) and relationships -[associations](http://www.uml-diagrams.org/association.html?context=class-diagrams), [generalizations](http://www.uml-diagrams.org/generalization.html?context=class-diagrams), [dependencies](http://www.uml-diagrams.org/dependency.html), etc.

****

Fig 5.3.1 : Class Diagram

Parties have the username and password these are all used to enter login then trusted authority generate the key for individual party then it will send to the party.at the time upload the chipper text to cloud. the cloud check the user key value and then the cloud allow to download the desired file

**5.3.1 Object Diagram:**

Object diagrams are also closely linked to class diagrams. Instance of Cloud service provider having some behavior attribute which is stored in cloud service provider.

Username()Password()

Encrption()

Key Generation()

Store the File()

File Upload()

Check the user key value()

Fig 5.3.2: Object Diagram

**5.3.3 UseCase Diagram:**

**Use Case Diagram** are usually referred to as [behavior diagrams](http://www.uml-diagrams.org/uml-25-diagrams.html#behavior-diagram) used to describe a set of actions ([use cases](http://www.uml-diagrams.org/use-case.html)) that some system or systems ([subject](http://www.uml-diagrams.org/use-case-subject.html)) should or can perform in collaboration with one or more **external user** of the system ([actors](http://www.uml-diagrams.org/use-case-actor.html)). Each use case should provide some observable and valuable result to the actors or other stakeholders of the system.

Fig 5.3.3 : Usecase Diagram

Parties first login in to the system .trusted authority maintain all secrets about participating party the trusted authority generate the key using random key generation and send to the particular party.at the time the trusted authority encrypt the file using BGN homomorphic then upload the chipper text to cloud. The parties were entering in to the cloud using the key values.then downloading the desired file.

**5.3.4 Statechart Diagram:**

It describes different states of a component in a system. The states are specific to a component/object of a system. A Statechart diagram describes a state machine. Now to clarify it state machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.



Fig 5.3.4: State chart Diagram

Users enter into login state and generate the key from the trusted party and send it to the user at the same time we encrypt it the upload the file to cloud the cloud check the key value the allow to download

**5.3.5: Activity Diagram:**

Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deals with all type of flow control by using different elements like fork, join etc.



Fig 5.3.5:Activity Diagram

Enter into login state and generate the key from the trusted party and send it to the user at the same time we encrypt it the upload the file to cloud the cloud check the key value the allow to download.

**5.3.6 Sequence Diagram :**

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart.



Fig 5.3.6: Sequence Diagram

Above Diagram tells about the different sequence enter into login state and generate the key from the trusted party and send it to the user at the same time we encrypt it the upload the file to cloud the cloud check the key value the allow to download

**5.3.7 Collaboration Diagram:**

Collaboration diagram is generated based on the flow of sequence between the objects Collaboration diagrams represent a combination of information taken from class, sequence, and use case diagrams describing both the static structure and dynamic behavior of a system

****

Fig 5.3.7: Collaboration Diagram

**5.3.8 Component Diagram:**

A component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems



Fig 5.3.8: Component Diagram

Enter into login state and generate the key from the trusted party and send it to the user at the same time we encrypt it the upload the file to cloud the cloud check the key value the allow to download

**5.4 DATA FLOW DIAGRAM**

Data flow diagram is a pictorial representation of the system in which flow of data from one process to other process is described.

Data flow diagrams are of two types:

**1.Physical Data Flow Diagrams:**

These are implementation- dependent i.e they show the actual devices,departments,people etc involved in the sytem.

**2. Logical Data Flow Diagrams:**

These diagrams describe the system independently of how it is actually implemented,they show what takes places,rather than how an activity is accomplished.

The DFD is intended to represent information flow but it is not a flow chart and it is not intended decision making,flow of control,loops and other procedural aspects of the system.DFD is a useful graphical tool and is applied to the earlier stages of requirement analysis.It may be further refined at prelliminary design stage and is used as mechanism for top level structural design for software

The DFD drawn first at a preliminary level is further expanded into greater details:

* The context diagram is decomposed and represented with multiple rectangles
* Each of these rectangles may be decomposed further and given as more detailed DFD

In the DFD there are four symbols

1. A square define a source or destination of system data

2. An arrow identifies data flow.It is the pipeline through which iinformation flows.

3. A circle or bubble represents a process that transforms incoming data flow into outgoing data flows.

4.An open rectangle is a data store,data at rest or a temporary repository of data.

Trusted Authority

Key generation

gene

Server

Download File

Encryption

File Upload

Fig 5.4.1: Zero Level Data Flow Diagram

User

KeyValue

Trusted Authority

Upload File

Download file

Database

Fig 5.4.2: First Level Data Flow Diagram

**Explanation:**

User enter into the system using username and passwordtheTrusted Host will be generate key and send it to the user then the trusted host encrypt the original data and upload the cloud the has check the key value like user key value same as the trust host key value then it allow to download the desired file then the user downloading the desired file.

**5.5 E-R DIAGARM**

**Login**

Valid user

Key Value

login

Encryption

Record Upload

Fig 5.5: E-R Diagram for Login

Login entity having two attributes like username and password. This is used to validate the user. It will check the user is valid or not the User get the key value the the trusted host upload the desired file to cloud the cloud check the user key and trusted host key then it will allow to download the desired file

**5.6 SYSTEM ARCHITECTURE**



Sending information

**Explanation:**

The systems architect establishes the basic structure of the system, defining the essential core design features and elements that provide the framework. The systems architect provides the architects view of the users' vision. Above diagram user first login to the account then he can download a file which are available in Cloud.

**5.7 DATA DICTIONARY**

The most imporatant aspect of the system in data design.The data must be organized according to the system requirements.

The database approach is used to store and organize the data in developing the system.The database is an integrated collection of data stored in different types of tables.

Some general objectives in establishing a database are as follows:

* Integarting all data
* Incorporate Updations easily
* Provide data security from unauthorized users

**Table Name 1: Admin Login**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| Username | VARCHAR(20) | Name of the Admin |
| Password | VARCHAR(20) | Password of the Admin |

**Table Name 2: Company or Hospital Login**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| Company Name | VARCHAR(20) | Name of the company |
| Company ID | NUMBER(5) | ID of the company |

**6. IMPLEMENTATION**

**DEVELOPMENT TOOLS**

**6.1 GENERAL**

Implementation is the process of having systems personnel check out and put new equipment into use, train users, install the new application depending on the size of the organization that will be involved in using the application and the risk associated with its use, systems developers may choose to test the operation in only one area of the firm, say in one department or with only one or two persons. Sometimes they will run the old and new systems together to compare the results. In still other situation, developers will stop using the old system one-day and begin using the new one the next. As we will see, each implementation strategy has its merits, depending on the business situation in which it is considered. Regardless of the implementation strategy used, developers strive to ensure that the system’s initial use in trouble-free.

Once installed, applications are often used for many years. However, both the organization and the users will change, and the environment will be different over weeks and months. Therefore, the application will undoubtedly have to be maintained; modifications and changes will be made to the software, files, or procedures to meet emerging user requirements. Since organization systems and the business environment undergo continual change, the information systems should keep pace. In this sense, implementation is ongoing process.

This chapter is about the software language and the tools used in the development of the project. The platform used here is JAVA. The Primary languages are JAVA,J2EE . In this project J2EE is chosen for implementation.

**6.2 FEATURES OF JAVA**

**6.2.1 The Java Framework**

**Java** is a programming language originally developed by James Gosling at Microsystems and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is general-purpose, concurrent, class-based, and object-oriented, and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere".

Java is considered by many as one of the most influential programming languages of the 20th century, and is widely used from application software to web applications the java framework is a new platform independent that simplifies application development internet. Java technology's versatility, efficiency, platform portability, and security make it the ideal technology for network computing. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

**6.2.2 Objectives of Java**

To see places of Java in Action in our daily life, explore java.com.

## Why Software Developers Choose Java:

Java has been tested, refined, extended, and proven by a dedicated community. And numbering more than 6.5 million developers, it's the largest and most active on the planet. With its versatility, efficiency, and portability, Java has become invaluable to developers by enabling them to:

* Write software on one platform and run it on virtually any other platform
* Create programs to run within a Web browser and Web services
* Develop server-side applications for online forums, stores, polls, HTML forms processing, and more
* Combine applications or services using the Java language to create highly customized applications or services
* Write powerful and efficient applications for mobile phones, remote processors, low-cost consumer products, and practically any other device with a digital heartbeat

## Some Ways Software Developers Learn Java:

Today, many colleges and universities offer courses in programming for the Java platform. In addition, developers can also enhance their Java programming skills by reading Sun's java.sun.com Web site, subscribing to Java technology-focused newsletters, using the Java Tutorial and the New to Java Programming Center, and signing up for Web, virtual, or instructor-led courses.

**Object Oriented Language:**

To be an Object Oriented language, any language must follow at least the four characteristics.

**1. Inheritance**   :It is the process of creating the new classes and using the behavior of the existing classes by extending them just to reuse  the existing code and adding addition a features as needed.

**2. Encapsulation:** It is the mechanism of combining the information and providing the abstraction.

**3. Polymorphism:** As the name suggest one name multiple form, Polymorphism is the way of providing the different functionality by the functions having the same name based on the signatures of the  methods.

**4. Dynamic binding:** Sometimes we don't have the knowledge of objects about their specific types while writing our code. It is the way of providing the maximum functionality to a program about the specific type at runtime.

**6.2.3 Features of JAVA:**

* Simple
* Secure
* Portable
* Object Oriented
* Robust
* Mulithreaded

**Simple:**

It is simple for professional programmer to learn and theycan use it effectively.If we already know object oriented programming learning java is very easy.It inherits syntax from C and object oriented features from C++.So if users know C/C++ then it will be easy way to do effective java programming.

**Secure:**

As we know many people are affected by virus when they download an executable file or program.Rather than virus programs we have malicious programs that collect private information,such as credit card number,bank account balance and password by searchung the contents of your compters local file syatem.Java has a better answer for this effects i.e FIREWALL between networked application and your computer.

**Portable:**

In Internet the programs have to be dynamically downloaded to all the various types of platforms.For this purpose java programs will genrerate a byte code.Byte code is highly optimized set of instructionsto be executed by java run system,which is called a JVM.

**Object Oriented:**

Java is a purely object oriented.The object model in java is simple and easy to extend,while simple types,such as integers,are kept as high performance non-objects

**Robust:**

The ability to create robust programs was given a high priority in the design of java.To gain realiability,java restricts you in a few key areas,to force to you find your mistakes early in program development.At sane time java frees you from having to worrry about many of the most common causes of programming errors.Because java is strictly a typed language,it checks your code at compile time.Java is robust for two reasons:they are Memory Management and mishandled exceptional task in traditional programming environments.

**6.2.4 Java Environment**

Java environment includes a large number of development tools and hundreds of classes and methods. The development tools are part of the system known as Java Development Kit (JDK) and the classes and methods are part of the Java Standard Library (JSL), also known as the Application Programming Interface

**Java Development Kit**

The Java Development Kit comes with a collection of tools that are used for developing and running Java programs. They include:

1. Applet viewer (for viewing Java applets)
2. Javac (Java Compiler)
3. Java (Java Interpreter)
4. Javap (Java Disassembler)
5. Javah (for C header files)
6. Javadoc (for creating HTML documents)
7. Jdb (Java debugger)

**Application Programming Interface**

The Java Standard Library (or API) includes hundreds of classes and methods grouped into several functional packages. Most commonly used packages are:

* **Language Support Package:** A collection of classes and methods required for implementing basic features of Java.
* **Utilities Package:**  A collection of classes to provide utility functions such as date and time functions.
* **Input/output Package:** A collection of classes required for input/output manipulation.
* **Networking Package:** A collection of classes for communicating with other computers via Internet.
* **AWT Package:** The Abstract Window Tool Kit package contains classes that implements platform-independent graphical user interface.
* **Applet Package:** This includes a set of classes that allows us to create **Java** applets.

**6.3 HTML**

The Hypertext Mark up Language (HTML) is a simple markup language used to create hypertext documents that are portable from one platform to another. HTML documents are SGML documents with the generic Semantics that are appropriate for representing the information from wide range of applications. Html version 3.2

A set of instruction embedded in a document is called the mark up language. These instructions describe what the document text means and how is should look like in a display hyperlink specification language that define the syntax and placement of special embedded directories that are not displayed by a web browser, but it tell how to displays the content of the documents including text, images and other supported media.

Web site is a collection of pages, publications and documents that reside on the web server. While these page publications and a document as a formatted in any single format. You should use html for Home page and all primary pages and the site. This will enable the millions of web users to easily access and to take advantage of your web site. Html Documents are platform independent, if they created properly you can move home page to any server platform.

#### 6.3.1 Basic Markup Tags

An HTML documents starts with <HTML> tag. This element tells the browser that the file contains HTML coded information. HTML document contain two parts namely the head and body

HEAD

**<HEAD>…</HEAD>**

This is the first part of HTML document, which contain description of the HTML page.

#### TITLE

**<TITLE>…</TITLE>**

These tags are embedded within the Head tags. Each HTML page should have a short and descriptive title. The browser at the top usually displays the title.

**A Title:**

* Is used in the indexes as well as in browser’s history list and bookmarks.
* Cannot contain any formatting, images, or links to other pages.
* Can have animated titles.

**Example**

<HTML>

<HEAD>

<TITLE>this is an html title</TITLE>

</HEAD>

…

</HTML>

**Some popular browsers:**

* Netscape Navigator
* Netscape Communicator
* Internet Explorer 3.0
* Internet Explorer 4.0

**Advantages:**

An HTML document is small and hence easy to send over the net. It is small because it does not include format information.HTML documents are cross platform compatible and device independent.

**6.4 JAVA SCRIPTING LANGUAGE**

Java script is a scripting language developed jointly by sun Netscape and is meant for the WWW. A scripting language is a simple script based programming language designed to enable programming to write useful programs quickly. A script is similar to a macro, which tells a program how to perform a specific procedure. As you go through this chapter we will bet a better understand of what a scripting is how Java Script brings about interactive wed pages with HTML.

**Java Script is embedded into HTML**

Java Script code usually embedded into HTML document and is executed within them. By it self-JavaScript has no user interface. It rallies on HTML to provide the mean of interaction with the user. Most of JavaScript object have HTML by providing events to HTML tags and provide event driven code to execute it.

# Java Script is Browser Dependent

JavaScript depends on the web browser to support it. If the browser doesn’t support it, JavaScript code will be ignored. Internet Explorer 3.0 and Netscape Navigator 2.0 onwards support JavaScript.

**Java Script is an Interpreted Language**

Java script is interpreted at runtime by the browser before it is executed. It is not completed into a separate program like a.exe but remains part of the HTML file.

**Java Script is a loosely type Language**

Java script is very flexible compared to java. We need not specify the data type of a variable while declaring it. Also we need not declare variable explicitly. It is perfectly legal to declare variable as when we required them.

**Java Script is an object based language**

Java Script is an object-based language. We can work with objects that encapsulate data and behavior. However JavaScript object model is instanced-based and there is no inheritance. This is basic difference between an object oriented and objects based language.

**Java Script is not Java**

Java applet is stored in a separate file and connected to HTML file through the <applet> tag, and it strongly typed, object oriented compiled language. JavaScript is loosely typed object based, interpreted language meant to create script. JavaScript can be used to

* Enhance Html pages.
* Develop client side application.
* Built to a certain extend client/server web application.
* Create extension to a web server.
* Provide database connectivity without using CGI.

**6.4.1 Client Side Framework**

The client side framework include the following

* **Web Browser**

Html client extension (java applets, active x controls and Netscape plugging) scripts language (JavaScript) JavaScript role in web application development.

* **Client Side Application**

JavaScript has good capabilities when working with Html tags & associated objects compared to java. For certain cases JavaScript provides a programming backbone with which to develop application.

* **Data Validation**

JavaScript provides the means for basic data validation before it is sent to the server. Whether the values entered are correct or not or whether all the fields in a form are filled out or not can be checked before sending data to web server, if JavaScript is not used then data is sent to web server, and the web server would response with a message that the data sent to it is incorrect or incomplete Thus JavaScript ensures data validation and also reduces the network traffic.

* **Identifiers**

Represent variable method or objects. Must start with a letter or under score (\_) subsequent character can be numbers or alphabets.

* **Keywords**

Break, if, this, continue, in time, else, return, but, false, new, while, for, null, with, function.

* **Literals**
* Integer literals.
* Floating-point literals.
* Boolean literals.
* String literals.
* Special literals.
* Special Character.

A function is declared in the <HEAD> part of the Html Script so that the browser loads the function before the body executes it.

**6.4.2 Java Script Object Model**

JavaScript is an object-based language. It has no inheritance. The relationship between objects at different levels is not of ancestor descent but of container. When an object properties or methods are referenced is used to denote ownership.

**6.4.3 Document Object**

Document object is the most important as the document object is responsible for all the actual contents displayed on a given page. Using document object we can display dynamic Html pages. Also all typical interface elements of a web application are contained in the document. A common use of document object is generating Html pages through JavaScript. This is done using write () or writeln () methods.

**6.5 Java Server Page**

Java Server Pages is a simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and a mature re-usable component model.

* **Portability:**

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation and management of the Java Server Pages lifecycle and its interaction with associated components.

The JSP engine for a particular server might be built-in or might be provided through a 3rd –party add-on. As long as the server on which you plan to execute the Java Server Pages supports the same specification level as that to which the file was written, no change should be necessary as you move your files from server to server. Note, however, that instructions for the setup and configuration of the files may differ between files.

**6.5.1 Composition**

It was mentioned earlier that the Java Server Pages architecture could include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans and Serves. As the default scripting language, Java Server Pages use the Java Programming language. This means that scripting on the server side can take advantage of the full set of capabilities that the Java programming language offers.

**6.5.2 Processing**

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. It may have associated components in the form of. class, .jar, or .ser files- -or it may not. The use of components is not required.

The Java Server Pages file has a .jsp extension to identify it to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a servlet on the server side. The servlet that is generated, outputs real content in straight HTML for responding to the customer. Because it is standard HTML, the dynamically generated response looks no different to the customer browser than a static response.

**6.5.3 Access Models**

A Java Server Pages file may be accessed in at least two different ways:

A client request comes directly into a Java Server Page.

**Browser**

##### JSP

Request

Response

Fig 6.1: Client request into JSP

In this scenario, suppose the page accessed reusable Java Bean components that perform particular well-defined computations like accessing a database. The result of the Bean’s computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client. A request comes through a servlet.

###### Browser

#### Servlet

JSP

Request

Response

Result Bean

JDBC

Database

Fig 6.2: Handling the response to the client

The Servlet generates the dynamic content. To handle the response to the client, the Servlet creates a Bean and stores the dynamic content (sometimes called the result *set*) in the Bean. The Servlet then invokes a Java Server Page that will present the content along with the Bean containing the generated from the Servlet.

**6.6 WHAT IS JDBC?**

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for "Java Database Connectivity".) It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it isn't necessary to write one program to access a Sybase database, another program to access an Oracle database, another program to access an Informix database, and so on. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. And, with an application written in the Java programming language, one also doesn't have to worry about writing different applications to run on different platforms. The combination of Java and JDBC lets a programmer write it once and run it anywhere.

Java being robust, secure, easy to use, easy to understand, and automatically downloadable on a network, is an excellent language basis for database applications. What is needed is a way for Java applications to talk to a variety of different databases. JDBC is the mechanism for doing this.

JDBC extends what can be done in Java. For example, with Java and the JDBC API, it is possible to publish a web page containing an applet that uses information obtained from a remote database. Or an enterprise can use JDBC to connect all its employees (even if they are using a conglomeration of Windows, Macintosh, and UNIX machines) to one or more internal databases via an intranet. With more and more programmers using the Java programming language, the need for easy database access from Java is continuing to grow.

MIS managers like the combination of Java and JDBC because it makes disseminating information easy and economical. Businesses can continue to use their installed databases and access information easily even if it is stored on different database management systems. Development time for new applications is short. Installation and version control are greatly simplified. A programmer can write an application or an update once, put it on the server, and everybody has access to the latest version. And for businesses selling information services, Java and JDBC offer a better way of getting out information updates to external customers.

JDBC does the following things:

* Establish a connection with a database
* Send SQL statements
* Process the results.

The following code fragment gives a basic example of these three steps:

Class.forName (“oracle.jdbc.driver.OracleDriver”);

Connectioncon=DriverManager.getcConnection(“jdbc:oracle:thin:@192.168.0.2:1521:tit”,”scott”,”tiger”);

Statement stmt=con.createStatement ();

**Connection**

A connection object represents a connection with a database. A connection session includes the SQL statements that are executed and the results that are returned over the connection. A single application can have one or more connections with a single database, or it can have connections with many different databases.

#### Opening Connection

The standard way to establish a connection with a database is to call the method *DriverManager.getConnection*. This method takes a string containing a URL. The Driver Manager class, referred to a the JDBC management layer, attempts to locate a driver than can connect to the database represented Driver classes, and when the method get Connection is called, it checks with each driver in the list until it finds one that can connect uses this URL to actually establish the connection.

The standard syntax for JDBC URLs is:

Jdbc: < sub protocol>: < sub name>

A JDBC URL has three parts, which are separated by colons:

* jdbc-the protocol. The protocol in a JDBC URL is always jdbc.
* <Sub protocol>-usually the driver or the database connectivity mechanism, which may be supported by one or more drivers. A prominent example of a sub protocol name is “oracle”, which has been reserved for URLs that specify “thin”-style data source names.
* <Sub name>- a way to identify the database. The sub names can vary, depending on the sub protocol, and it can have a sub name with any internal syntax the driver writer chooses. The point of a sub name is to give enough information to locate the database.

**Sending Statement:**

Once a connection is established, it is used to pass SQL statements to its underlying database. JDBC does not put any restrictions on the kinds of SQL statements that can be sent; this provides a great deal of flexibility, allowing the use of database-specific statements or even non-SQL statements. It requires, however, that the user be responsible for making sure that the underlying database can process the SQL statements being sent and suffer the consequences if it cannot.

**Driver Manager:**

The Driver Managerclass is the management layer of JDBC, working between the user and the drivers. It keeps track of the drivers that are available and handles establishing a connection between a database and the appropriate driver. It addition, the driver managerclass attends to things like driver login time limits and the printing of log and tracing messages. The only method in this class that a general programmer needs to use directly is DriverManager.getConnection. As its name implies, this method establishes a connection to a database.

**JDBC Driver Types**

The JDBC drivers that we are aware of at this time fit into one of four categories:

1. **JDBC-ODBC bridge plus ODBC driver:**

The JavaSoft bridge product provides JDBC access via ODBC drivers. Note that ODBC binary code, and in many cases database client code, must be loaded on each client machine that uses this driver. As a result, this kind of driver is most appropriate on a corporate network where client installations are not a major problem, or for application server code written in Java in a three-tier architecture.

1. **Native-API partly-Java driver:**

This kind of driver converts JDBC calls into calls on the client API for Oracle, Sybase, Informix, DB2, or other DBMS. Note that, like the bridge driver, this style of driver requires that some binary code be loaded on each client machine.

1. **JDBC-Net pure Java driver:**

This driver translates JDBC calls into a DBMS- independent net protocol, which is then translated to a DBMS protocol by a server. This net server middleware is able to connect its pure Java clients to many different databases. The specific protocol used depends on the vendor. In general, this is the most flexible JDBC alternative. It is likely that all vendors of this solution will provide products suitable for Intranet use. In order for these products to also support Internet access, they must handle the additional requirements for security, access through firewalls, and so on, that the Web imposes. Several vendors are adding JDBC drivers to their existing database middleware products.

1. **Native-protocol pure Java driver:**

This kind of driver converts JDBC calls into the network protcol used by DBMSs directly. This allows a direct call from the client machine to the DBMS server and is a practical solution for Intranet access. Since many of these protocols are proprietary, the database vendors themselves will be the primary source, and several database vendors have these in progress.

###### The J2EE Technologies:

* Enterprise JavaBeans (EJBs)
* Java Remote Method Invocation and RMI-IIOP
* Java Naming Directory Interface (JNDI)
* Java Database Connectivity (JDBC)
* Java Transaction (JTs)
* Java Messaging Service (JMS)
* Java Servlets
* Java Server Pages (JSP)
* Java IDL
* Java Mail
* J2EE Connector Architecture (JCA)
* The Java API for XML Parsing (JAXP)
* The Java Authentication and Authorization Service (JAAS)

# 6.7 Database Management System

A database management system (DBMS)consists of a collection of interrelated data and a set of programs to access the data. The collection of data is usually referred to as the database. A Database system is designed to maintain large volumes of data. Management of data involves:

* Defining the structures for the storage of data
* Providing the mechanisms for the manipulation of the data
* Providing for the security of the data against unauthorized access

**Users of the DBMS:**

Broadly, there are three types of DBMS users:

* The application programmer
* The end user
* The database administrator (DBA)

The application programmer writes application programs that use the database. These programs operate on the data in the database. These operations include retrieving information, inserting data, deleting or changing data.

The end user interacts with the system either by invoking an application program or by writing their queries in a database query language. The database query language allows the end user to perform all the basic operations (retrieval, deletion, insertion and updating) on the data.

The DBA has to coordinate the functions of collecting information about the data to be stored, designing and maintaining the database and its security. The database must be designed and maintained to provide the right information at the right time to authorized people. These responsibilities belong to the DBA and his staff.

**Advantages Of A DBMS**

The major advantage that the database approach has over the conventional approach is that a database system provides centralized control of data. Most benefits accrue from this notion of centralized control.

###### Redundancy can be controlled:

Unlike the conventional approach, each application does not have to maintain its own data files. Centralized control of data by the DBA avoids unnecessary duplication of data and effectively reduces the total amount of data storage required. It also eliminates the extra processing necessary to trace the required data in a large mass of data present. Any redundancies that exist in the DBMS are controlled and the system ensures that these multiple copies are consistent.

* **Inconsistency can be avoided:**

Since redundancy is reduced, inconsistency can also be avoided to some extent. The DBMS guarantee and that the database is never inconsistent, by ensuring that a change made to any entry automatically applies to the other entries as well. The process is known as propagating update.

* **The data can be shared:**

A database allows the sharing of data under its control by any number of application program or users. Sharing of data does not merely imply that existing applications can share the data in the database, it also means that new applications can be developed to operate using the same database.

* **Standards can be enforced:**

Since there is centralized control of data, the database administrator can ensure that standards are maintained in the representation of the stored data formats. This is particularly useful for data interchange, or migration of data between two systems.

* **Security restrictions can be applied:**

The DBMS guarantees that only authorized persons can access the database. The DBA defines the security checks to be carried out. Different checks can be applied to different operations on the same data. For instance, a person may have the access rights to query on a file, but may not have the right to delete or update that file. The DBMS allows such security checks to be established for each piece of data in the database.

* **Integrity can be maintained:**

Centralized control can also ensure that adequate checks are incorporated in the DBMS to provide data integrity. Data integrity means that the data contain in the database is both accurate and consistent. Inconsistency between two entries can lead to integrity problems. However, even if there is no redundancy, the data can still be inconsistent. For example a student may have enrolled in 10 courses in a semester when the maximum number of courses one can enroll in is 7. Another example could be that of a student enrolling in a course that is not being offered that semester. Such problems can be avoided in a DBMS by establishing certain integrity checks to be carried out whenever any update operation is done. These checks can be specified at the database level, besides the application programs.

* **Data Independence**

In non-database systems, the requirement of the application dictates the way in which the data is stored and the access techniques. Besides, the knowledge of the organization of the data, the access techniques are built into the logic and code of the application. These systems are data dependent. Consider this example, suppose the university has an application that processes the student file. For performance reason, the file is indexed on the roll number. The application would be aware of the existing index, and the internal structure of the application would be built around this knowledge. Now consider that the some reason, the file is to index on the registration data. In this case it is impossible to change the structure of the stored data without affecting the application too. Such an application is a data dependent one.

It is desirable to have data independent applications. Suppose two applications X and Y need to access the same file. However both the applications require a particular field to be stored in different formats. Application X requires the field “customer balance” to be stored in decimal format, while the application Y requires it to be stored in binary format. This would pose a problem in an old system. In a DBMS differences may exist in the way that data is actually stored, and the way that it is seen and used by a given application.

# 6.7.1 Database Models:

**1.Hierarichal Model:**

IBM introduced this model in the Information Management System (IMS) development in 1968. This mode is like a hierarchical tree structure, used to construct a hierarchy of records in the form of nodes and branches. The data elements present in the structure have a Parent-child relationship. Closely related information in the parent child structure is stored together as a logical unit. A present unit may have many child units, but a child is restricted to have only one parent.

In order to gain an understanding of the various database models, we take an example of a sample database consisting of supplier, parts and shipments. In this view, data is represented by a simple tree structure, with parts superior to suppliers. The user sees four individual trees, of hierarchical occurrence one of each part. Each tree consists of one part record occurrence, together with a set of subordinate supplier record occurrence, one of each supplier of the part. Each supplier occurrence includes the corresponding shipment quantity. Note that the set of supplier occurrences for a given part may contain any number of members, including zero.

**The Drawbacks of this model are:**

1. The Hierarchical structure is not flexible enough to represent all the relationship proportions, which occur in the real world.
2. It cannot demonstrate the over all data model for the enterprise because of the non-availability of actual data at the time of designing the data model.
3. It cannot represent the Many-to-Many relationship.
4. The Hierarchical model is used only when the concerned data has a clearly hierarchical character with a single root, for e.g. the DOS directory structure.

**2.Network Model:**

It is an improvement on the hierarchical model. Here multiple parent child relationships are used. Rapid and easy access to data is possible in this model due to multiple access paths to the data elements.

In this model, as in the hierarchical approach, records and links represent data. However, a network is a more general structure than a hierarchy because a given record occurrence may have any number of immediate superiors and dependents. We can have more than one, unlike hierarchical. The network approach thus allows us to model a many to many correspondence. In addition to the record types representing the suppliers and parts themselves, we introduce a third type a record, which we will call the connector.

A connector occurrence, represents the association between one superior and one part, and contains data describing the association. All connector for given supplier or part are placed on a chain starting at and returning to that supplier or part.

Transaction is maintained using pointers and having to trace the pointers is the drawback of this design.

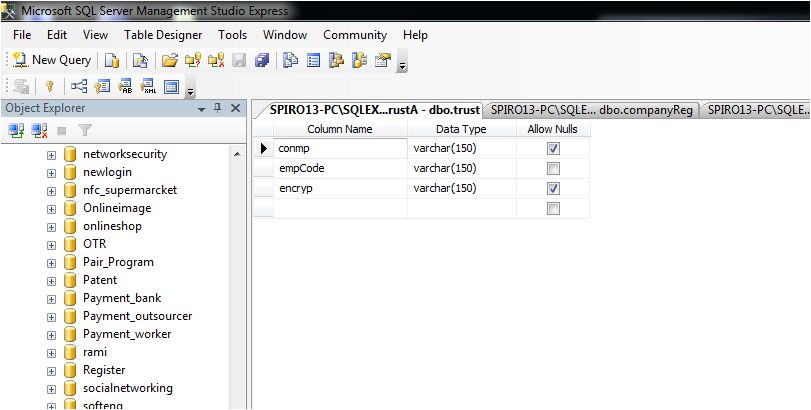
**3.Relational Model**

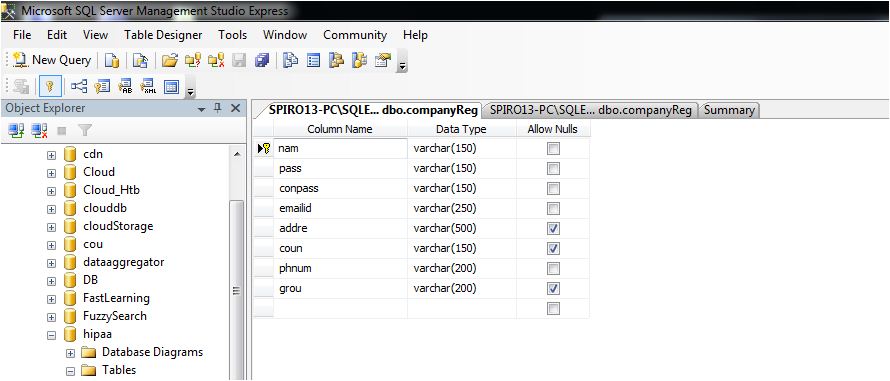
* Does not maintain physical connection between relations.
* Data is organized in terms of rows and columns in a table.
* The position of a row and / or column in a table is of no importance
* The intersection of a row and column must give a single value.
* All values appearing in the columns are derived from the underlying domain.

# Features Of RDBMS:

* The ability to create multiple relations and enter data into them
* An interactive query language
* Retrieval of information stored in more than one table

**Database table structure:**

****

****

The above data structure showing the table of the company registration ,the field names and the data type

**6.8 SERVLETS**

**6.8.1 Introduction**

The Java web server is JavaSoft's own web Server. The Java web server is just a part of a larger framework, intended to provide you not just with a web server, but also with tools. To build customized network servers for any Internet or Intranet client/server system. Servlets are to a web server, how applets are to the browser.

**6.8.2 About Servlets**

Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions, platform-specific APIs, and incomplete interfaces.

Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side - object byte codes that can be dynamically loaded off the net. They differ from applets in that they are faceless objects (without graphics or a GUI component). They serve as platform independent, dynamically loadable, pluggable helper byte code objects on the server side that can be used to dynamically extend server-side functionality.

For example, an HTTP Servlets can be used to generate dynamic HTML content. When you use Servlets to do dynamic content you get the following advantages:

* They’re faster and cleaner than CGI scripts
* They use a standard API (the Servlets API)
* They provide all the advantages of Java (run on a variety of servers without needing to be rewritten).

**6.8.3 Attractiveness of Servlets:**

There are many features of Servlets that make them easy and attractive to use. These include:

* Easily configured using the GUI-based Admin tool
* Can be loaded and invoked from a local disk or remotely across the network.
* Can be linked together, or chained, so that one Servlets can call another Servlets, or several Servlets in sequence.
* Can be called dynamically from within HTML pages, using server-side include tags.
* Are secure - even when downloading across the network, the Servlets security model and Servlets sandbox protect your system from unfriendly behavior.

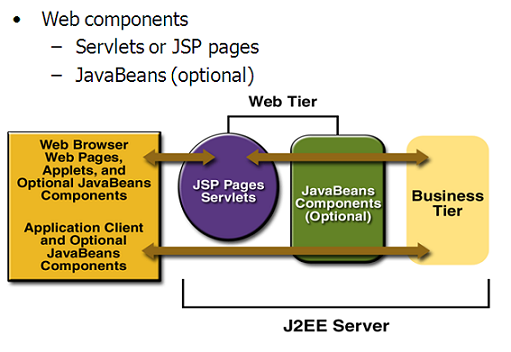
**Advantages of the Servlet API**

One of the great advantages of the Servlet API is protocol independence. It assumes nothing about:

* The protocol being used to transmit on the net
* How it is loaded
* The server environment it will be running in

These qualities are important, because it allows the Servlet API to be embedded in many different kinds of servers. There are other advantages to the Servlet API as well. These include:

* It’s extensible - you can inherit all your functionality from the base classes made available to you.
* It’s simple, small, and easy to use.



**6.8.4** **Features of Servlets:**

* Servlets are persistent. Servlet are loaded only by the web server and can maintain services between requests.
* Servlets are fast. Since Servlets only need to be loaded once, they offer much better performance over their CGI counterparts.
* Servlets are platform independent.
* Servlets are extensible. Java is a robust, object-oriented programming language, which easily can be extended to suit your needs
* Servlets are secure.
* Servlets can be used with a variety of clients.

**6.8.5 Loading Servlets:**

**Servlets can be loaded from three places**

From a directory that is on the CLASSPATH. The CLASSPATH of the JavaWebServer includes service root/classes/ which is where the system classes reside.

From the <SERVICE\_ROOT /Servlets/ directory. This is \*not\* in the server’s class path. A class loader is used to create Servlets from this directory. New Servlets can be added - existing Servlets can be recompiled and the server will notice these changes.

From a remote location, for this a code base like http: // nine.eng / classes / foo / is required in addition to the Servlets class name. Refer to the admin GUI docs on Servlet section to see how to set this up.

**Loading Remote Servlets**

Remote Servlets can be loaded by:

1. Configuring the Admin Tool to setup automatic loading of remote Servlets
2. Setting up server side include tags in. shtml files
3. Defining a filter chain configuration

**Invoking Servlets**

A Servlet invoker is a Servlet that invokes the "service" method on a named Servlet. If the Servlet is not loaded in the server, then the invoker first loads the Servlet (either from local disk or from the network) and the then invokes the "service" method. Also like applets, local Servlets in the server can be identified by just the class name. In other words, if a Servlet name is not absolute, it is treated as local.

A client can invoke Servlets in the following ways**:**

* The client can ask for a document that is served by the Servlet.
* The client (browser) can invoke the Servlet directly using a URL, once it has been mapped using the [Servlet Aliases](../administration/servlet_alias.html) section of the admin GUI.
* The Servlet can be invoked through [server side include](ssinclude.html) tags.
* The Servlet can be invoked by placing it in the Servlets/ directory.
* The Servlet can be invoked by using it in a filter chain.

**7. TESTING**

**7.1 Software Testing Techniques:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, designing and coding.

**7.1.1Testing Objectives:**

1. Testing is process of executing a program with the intent of finding an error.
2. A good test case design is one that has a probability of finding an as yet undiscovered error.
3. A successful test is one that uncovers an as yet undiscovered error.

These above objectives imply a dramatic change in view port.

Testing cannot show the absence of defects, it can only show that software errors are present.

**7.1.2 Test Case Design:**

Any engineering product can be tested in one of two ways:

**1.White Box Testing:**

This testing is also called as glass box testing. In this testing, by knowing the specified function that a product has been designed to perform test can be conducted that demonstrates each function is fully operation at the same time searching for errors in each function. It is a test case design method that uses the control structure of the procedural design to derive test cases. Basis path testing is a white box testing.

**Basis Path Testing:**

1. Flow graph notation
2. Cyclomatic Complexity
3. Deriving test cases
4. Graph matrices

Control Structure Testing:

1. Condition testing
2. Data flow testing
3. Loop testing

**2.Black Box Testing:**

In this testing by knowing the internal operation of a product, tests can be conducted to ensure that “ all gears mesh”, that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

The steps involved in black box test case design are:

1. Graph based testing methods
2. Equivalence partitioning
3. Boundary value analysis
4. Comparison testing

**7.2 Software Testing Strategies:**

A software testing strategy provides a road map for the software developer. Testing is a set of activities that can be planned in advance and conducted systematically. For this reason a template for software testing a set of steps into which we can place specific test case design methods should be defined for software engineering process. Any software testing strategy should have the following characteristics:

1. Testing begins at the module level and works “outward” toward the integration of the entire computer based system.
2. Different testing techniques are appropriate at different points in time.
3. The developer of the software and an independent test group conducts testing.
4. Testing and Debugging are different activities but debugging must be accommodated in any testing strategy.

**1.Unit Testing:** Unit testing focuses verification efforts in smallest unit of software design (module).

1.Unit test considerations

2.Unit test procedures

**2.Integration Testing:**  Integration testing is a systematic technique for constructing the program structure while conducting tests to uncover errors associated with interfacing. There are two types of integration testing:

**a. Top-Down Integration:**

Top down integration is an incremental approach to construction of program structures. Modules are integrated by moving down wards throw the control hierarchy beginning with the main control module.

**b.Bottom-Up Integration:**

Bottom up integration as its name implies, begins construction and testing with automatic modules.

**3.Regression Testing:**

In this contest of an integration test strategy, regression testing is the re execution of some subset of test that have already been conducted to ensure that changes have not propagate unintended side effects.

**7.3 Validation Testing:**

At the culmination of integration testing, software is completely assembled as a package; interfacing errors have been uncovered and corrected, and a final series of software tests – *validation testing* may begin. Validation can be fined in many ways, but a simple definition is that validation succeeds when software functions in a manner that can be reasonably expected by the customer.

Reasonable expectation is defined in the software requirement specification – a document that describes all user-visible attributes of the software. The specification contains a section titled “Validation Criteria”. Information contained in that section forms the basis for a validation testing approach.

In the text boxes like name, address etc., only alphabets and number could be entered thus if the operator by mistake enters other special characters, it would not be entered.

In the text boxes like age, telephone number only numbers could be entered.

If the users do not fill any of the fields, which could not be empty, a message would be displayed asking to enter the required parameters.

When a user starts the applications, a login form will be displayed prompting to enter the username and password, if even any one of them is not matched with the details stored in the database, the user will be warned to re-enter the correct details.

While entering the details of new customer, the customer number which cannot be null value will be automatically generated which is one greater than the highest number existing previously.

When the details of one customer are modified even if one parameter is missed a message will be displayed asking to enter complete details.

**7.3.1**  **Validation Test Criteria:**

Software validation is achieved through a series of black-box tests that demonstrate conformity with requirement. A test plan outlines the classes of tests to be conducted, and a test procedure defines specific test cases that will be used in an attempt to uncover errors in conformity with requirements. Both the plan and procedure are designed to ensure that all functional requirements are satisfied; all performance requirements are achieved; documentation is correct and human-engineered; and other requirements are met.

After each validation test case has been conducted, one of two possible conditions exist: (1) The function or performance characteristics conform to specification and are accepted, or (2) a deviation from specification is uncovered and a deficiency list is created. Deviation or error discovered at this stage in a project can rarely be corrected prior to scheduled completion. It is often necessary to negotiate with the customer to establish a method for resolving deficiencies.

**Configuration Review:**

An important element of the validation process is a *configuration review*. The intent of the review is to ensure that all elements of the software configuration have been properly developed, are catalogued, and have the necessary detail to support the maintenance phase of the software life cycle. The configuration review sometimes called an audit.

**7.4 Alpha and Beta Testing:**

It is virtually impossible for a software developer to foresee how the customer will really use a program. Instructions for use may be misinterpreted; strange combination of data may be regularly used; and output that seemed clear to the tester may be unintelligible to a user in the field.

When custom software is built for one customer, a series of acceptance tests are conducted to enable the customer to validate all requirements. Conducted by the end user rather than the system developer, an acceptance test can range from an informal “test drive” to a planned and systematically executed series of tests. In fact, acceptance testing can be conducted over a period of weeks or months, thereby uncovering cumulative errors that might degrade the system over time.

If software is developed as a product to be used by many customers, it is impractical to perform formal acceptance tests with each one. Most software product builders use a process called alpha and beta testing to uncover errors that only the end user seems able to find.

A customer conducts the alpha test at the developer’s site. The software is used in a natural setting with the developer “looking over the shoulder” of the user and recording errors and usage problems. Alpha tests are conducted in controlled environment.

The beta test is conducted at one or more customer sites by the end user of the software. Unlike alpha testing, the developer is generally not present. Therefore, the beta test is a “live” application of the software in an environment that cannot be controlled by the developer. The customer records all problems that are encountered during beta testing and reports these to the developer at regular intervals. As a result of problems reported during bets test, the software developer makes modification and then prepares for release of the software product to the entire customer base.

**8. RESULTS**

**8.1 General**

Cloud computing enables highly scalable services to be easily consumed over the Internet on an as-needed basis. A major feature of the cloud services is that users’ data are usually processed remotely in unknown machines that users do not own or operate.

**1.Company Registartion Page**

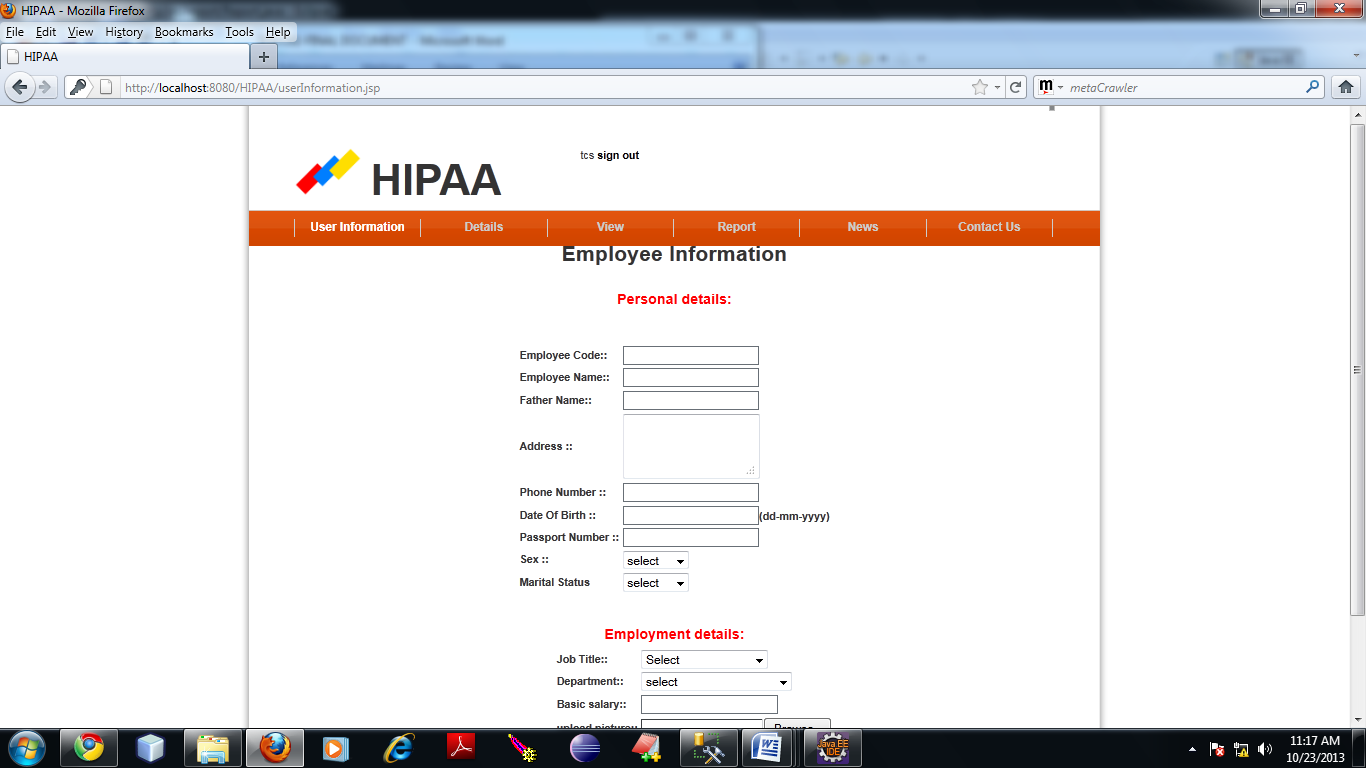
****

**Navigation**: open the eclipse and import the project to the workplace File Import

**Running the project**:Right click on the imported project file and selest Run on server option.The server will be started.

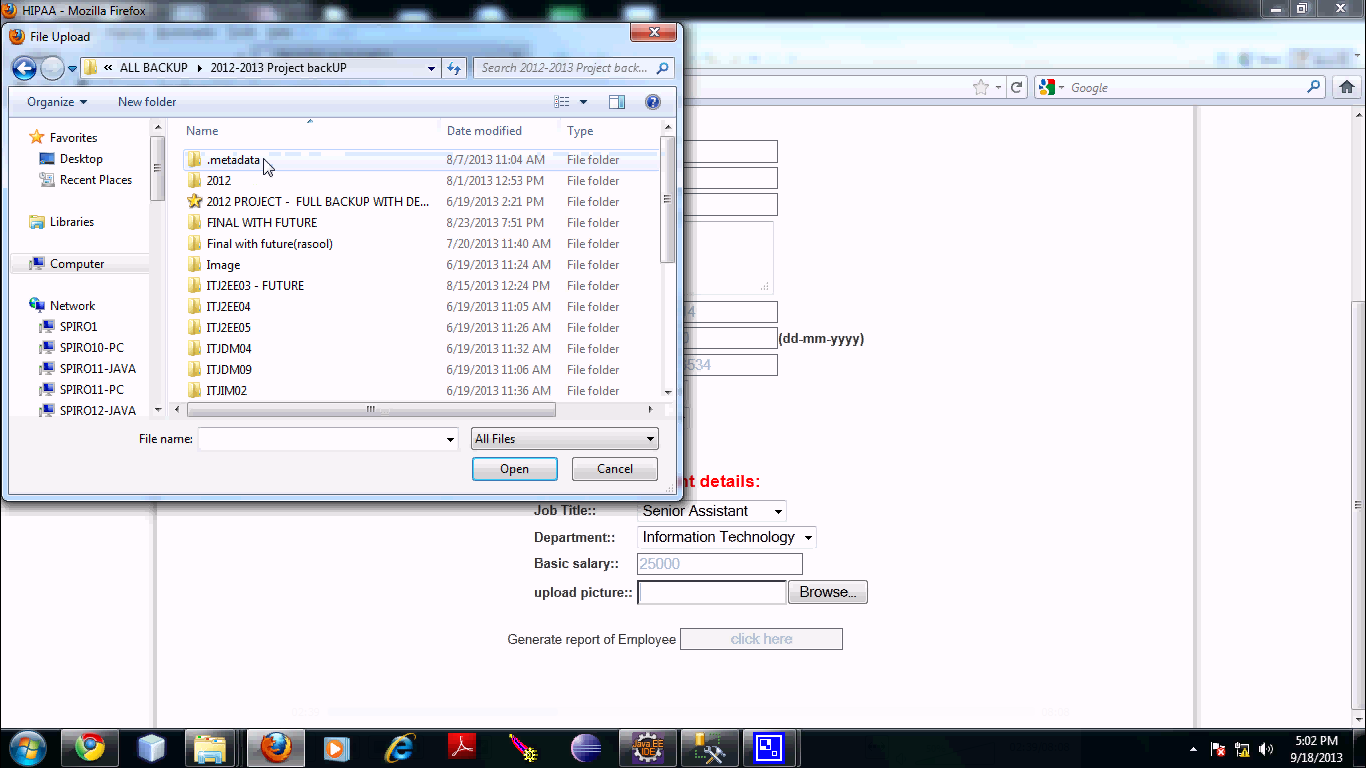
Then the home page HIPPA will be displayed,then click on registration option and Select either company or hospital .If we select Company Registartion ,then the above page will be displayed fill the form the company registration will be successful

**2. Employement information Page**

****

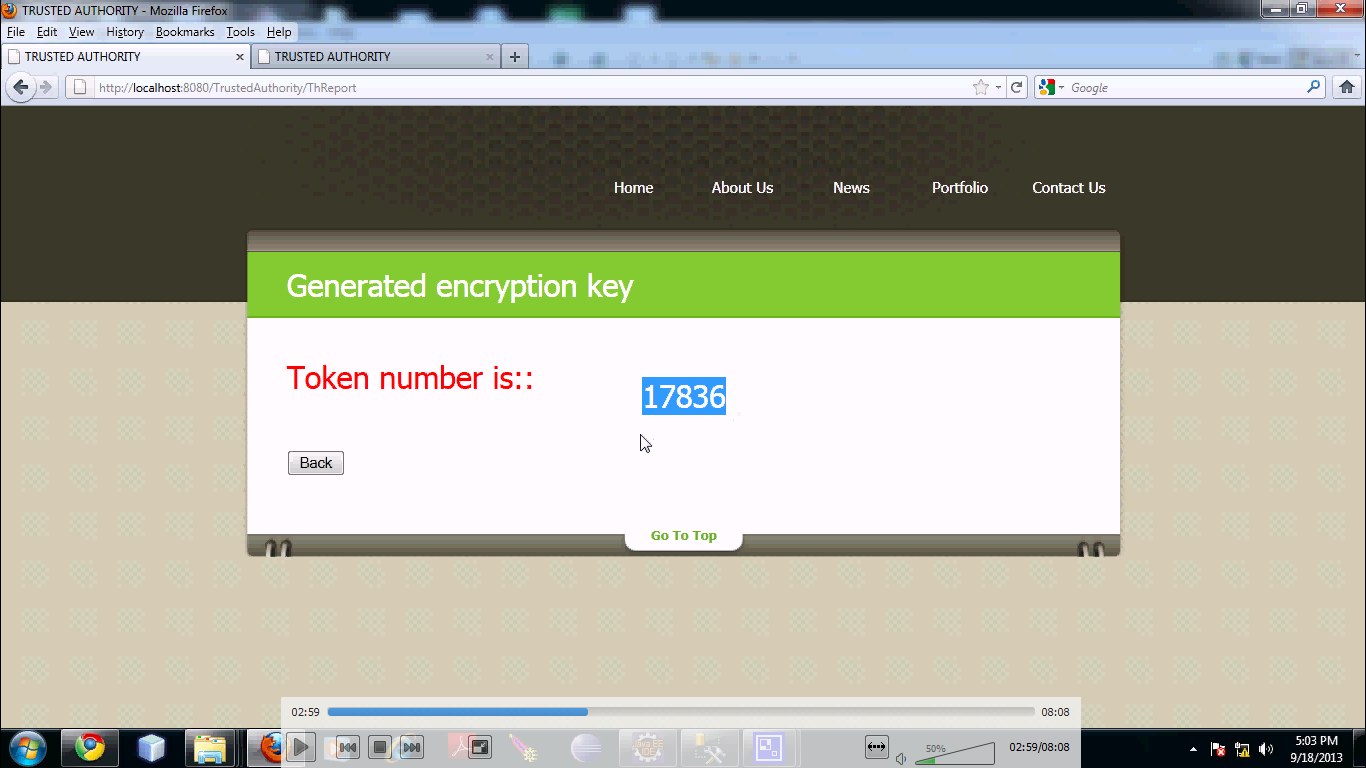
After completion of company and hospital registration.We have to login through company ID and name and update the employee information by clicking on user information.

**3. Uploading the picture of the employee**

****

To upload the picture of the employee click on the browse button and select the picture of the employee and select Generate report of Employee.

**4. Encryption Key Generation:**

****

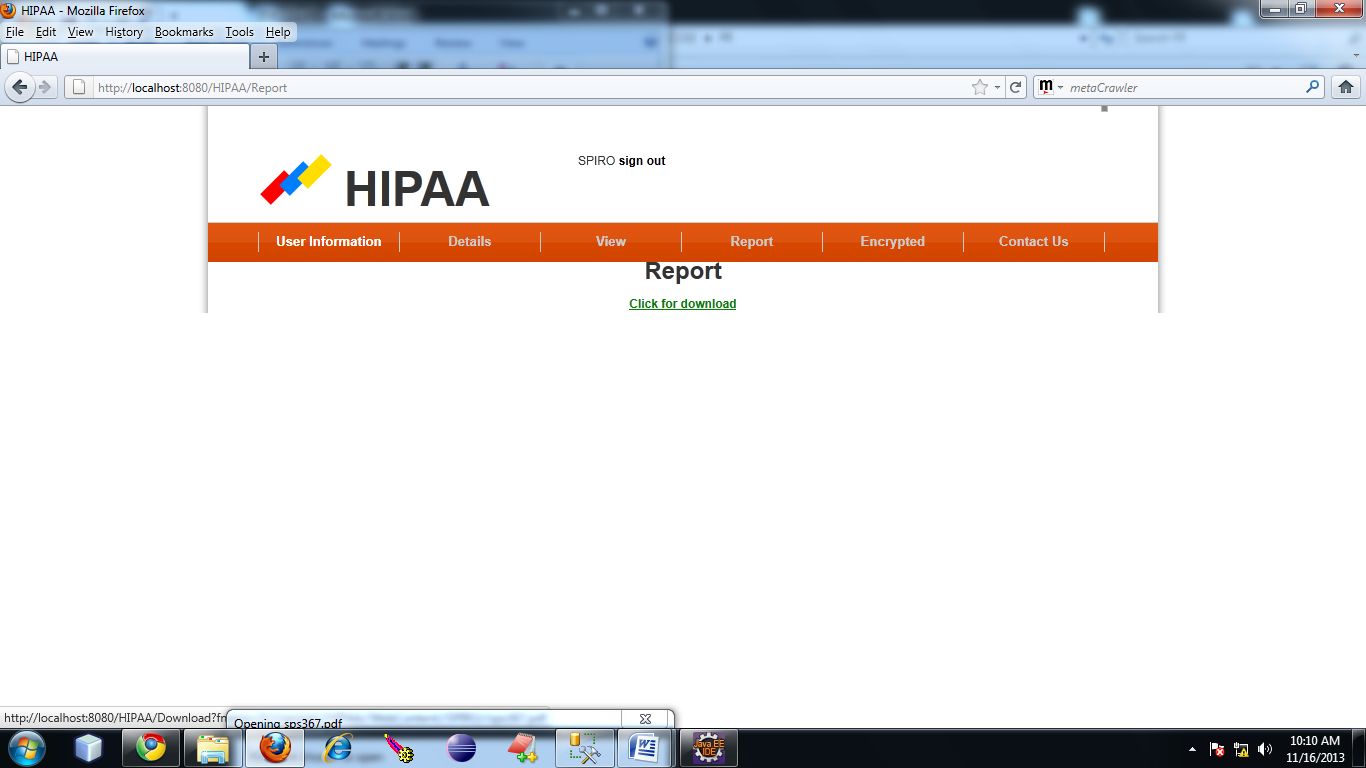
After selecting generate the report of the employee the encryption key will be generated

**5. Generating the encrypted report of the employee of the company**

****

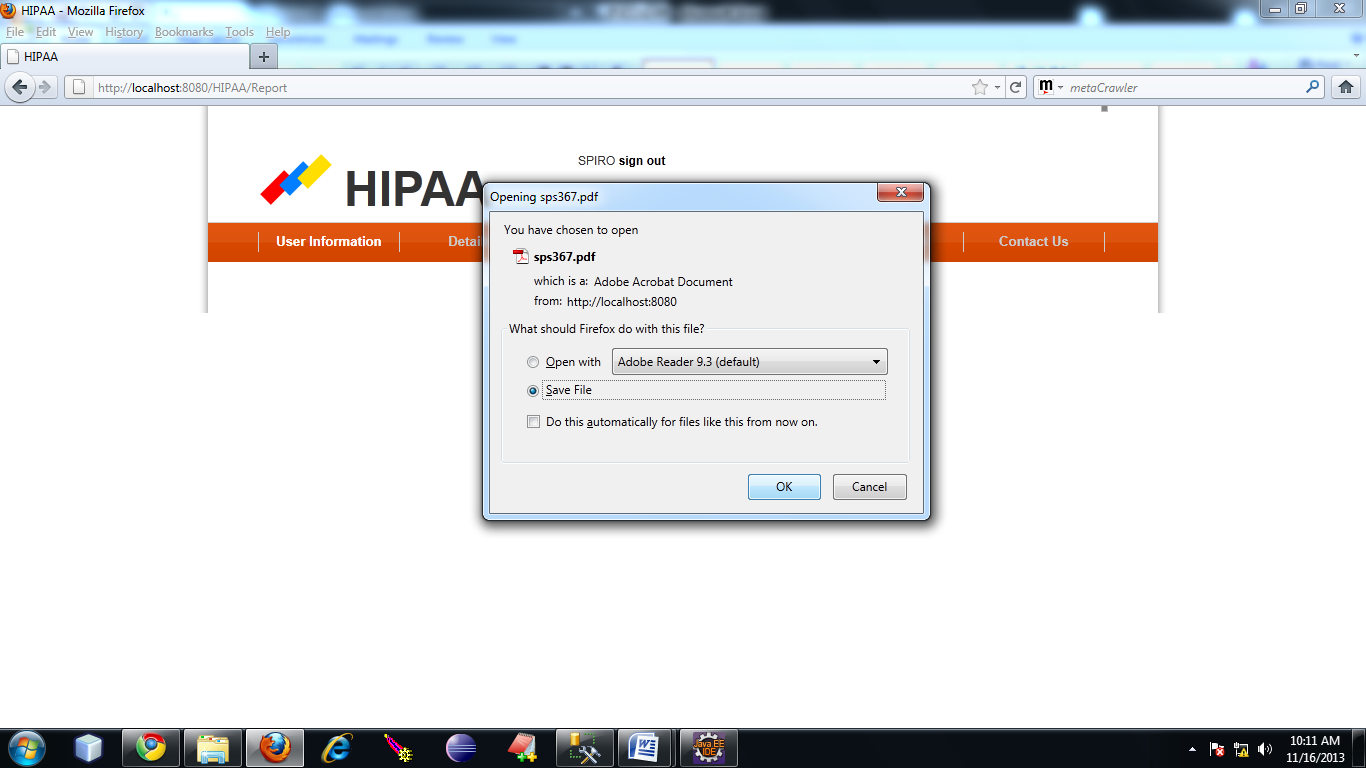
Then click on the report button on the home page and fill the details of company name,candidate name and candidate ID and click on submit button to generate the encrypted report of employee

**6. Downloading the Encrypted report of the Employee**

****

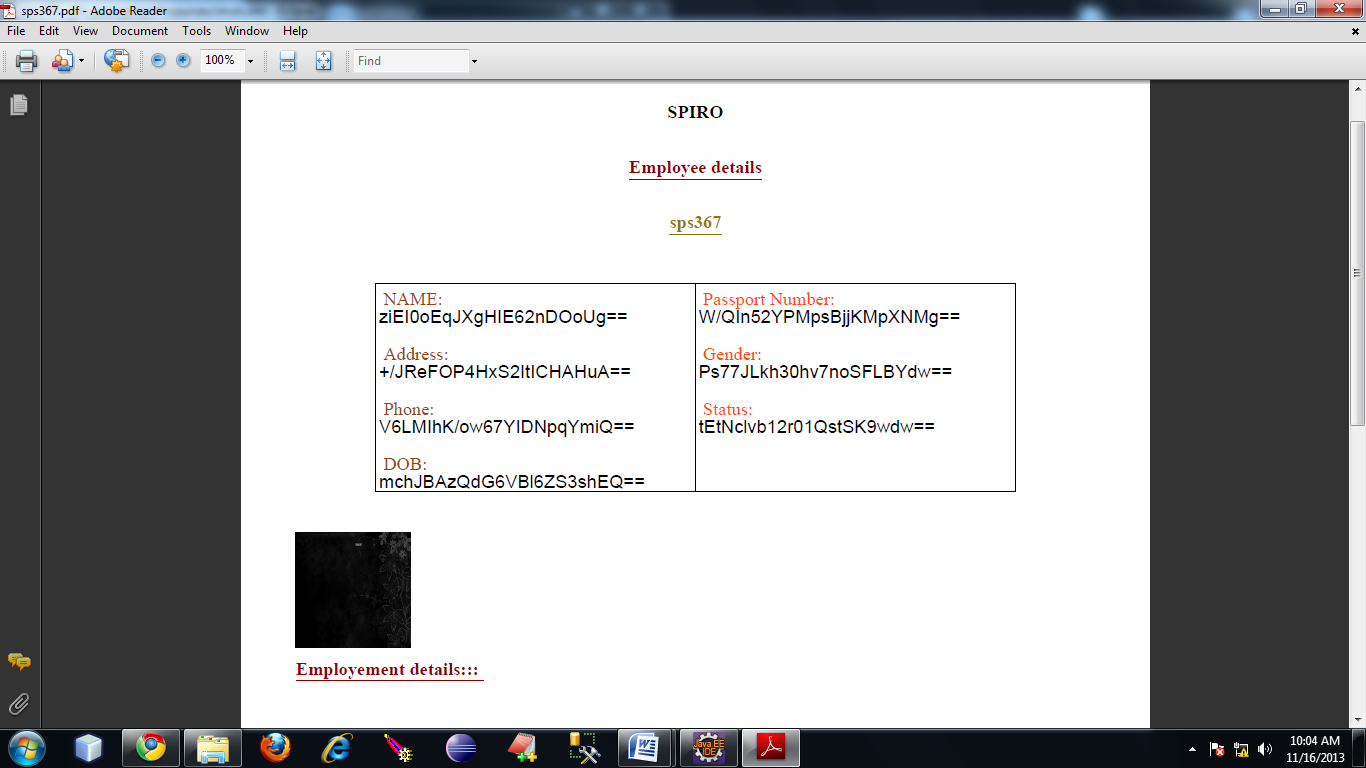
Then select on click for download to generate the encrypted report of the employee.

**7. Saving the Encrypted Report:**

****

Then save the generated encrypted report of the employee in a specified folder

**8.Encrypted Report:**

****

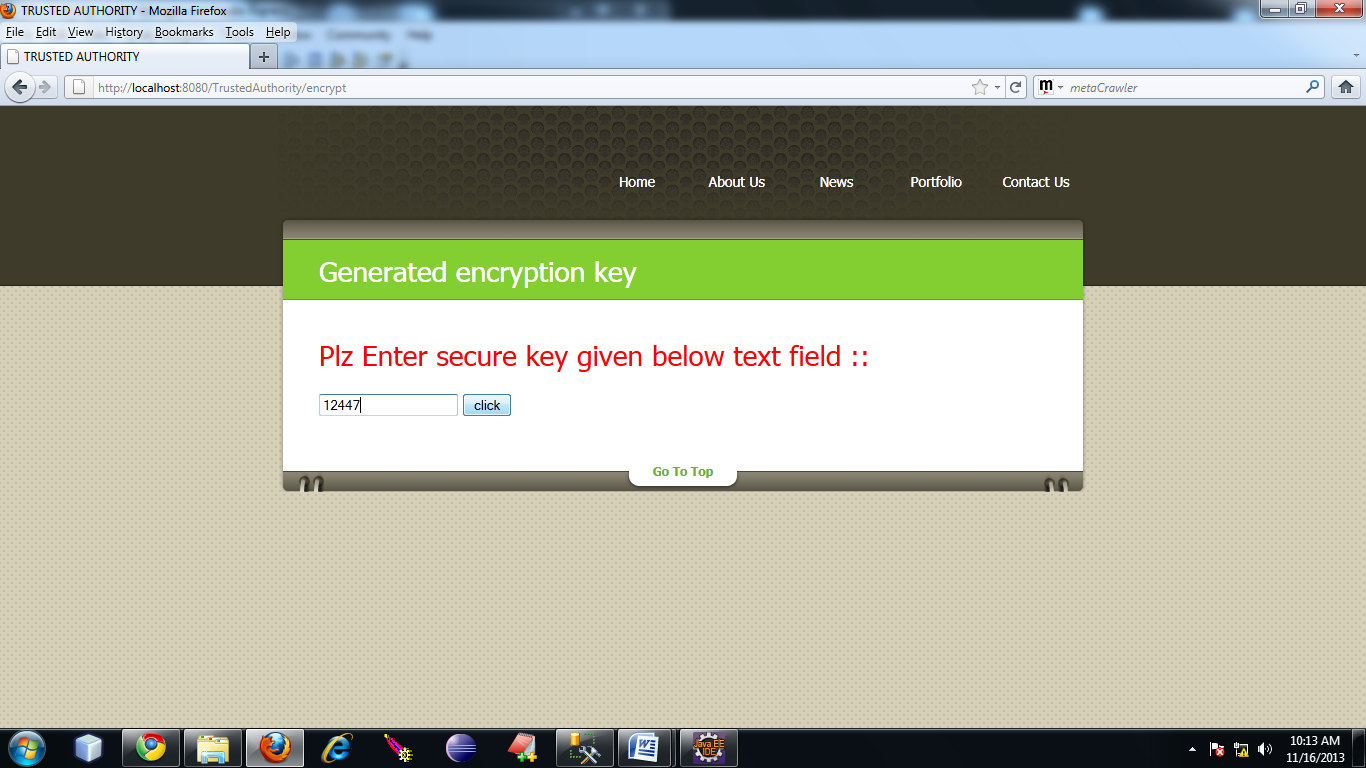
The encrypted form of the employee report will be in the above format.

**9. Browsing encrypted file:**

****

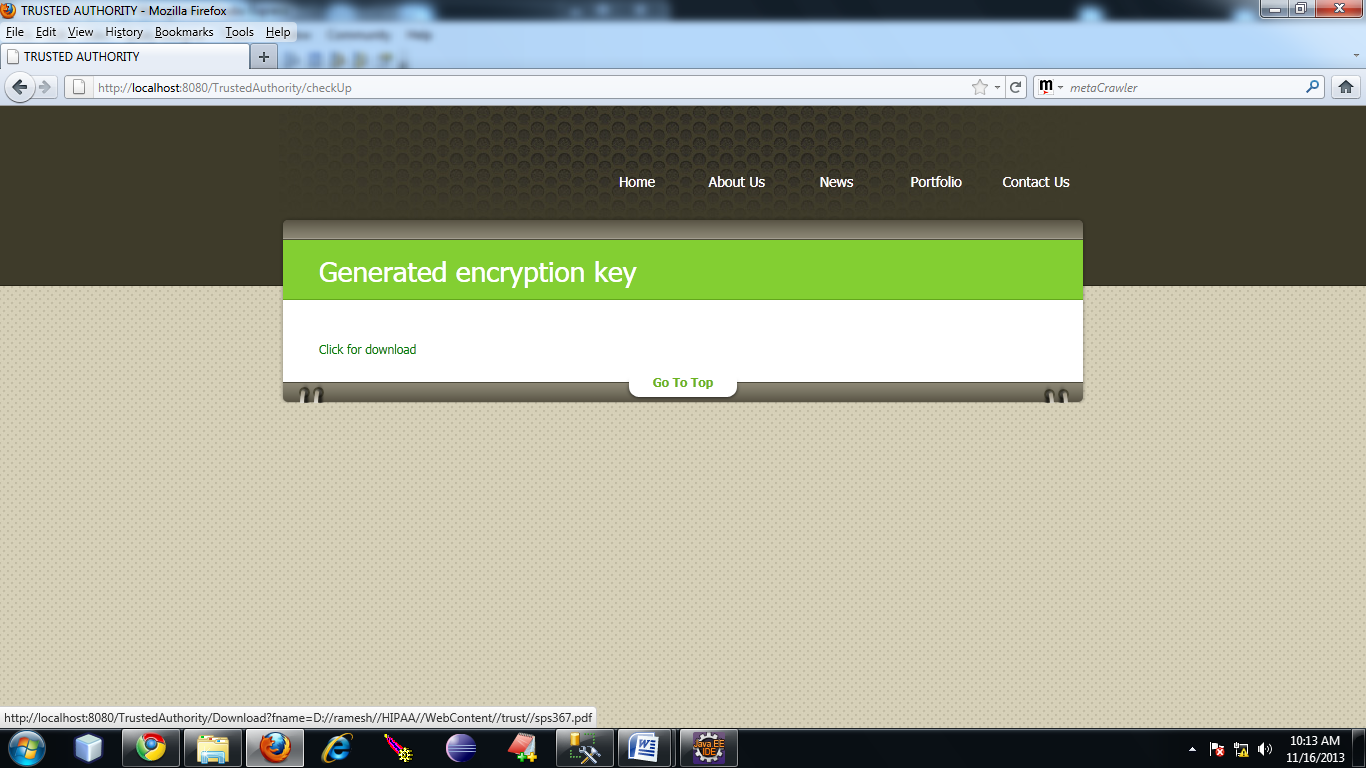
To decrypt the generated encrypted report select the encrypt option and browse for the encrypted file and select ok

**10. Entering the Encryption Key value:**

****

Enter the generated encryption key value to decrypt the generated report of the employee.

**11. Downloading the Decrypted report**

****

Then select on click for download to generate the decrypted report of the employee.

**12. Decrypted Employee report:**

****

The decrypted report of the employee will be in the above format.

**9. CONCLUSION AND FUTURE SCOPE**

**9.1 Application**

**Health Information**

The Office for Civil Rights enforces the HIPAA Privacy Rule, which protects the privacy of individually identifiable health information; the HIPAA Security Rule, which sets national standards for the security of electronic protected health information; the HIPAA Breach Notification Rule, which requires covered entities and business associates to provide notification following a breach of unsecured protected health information; and the confidentiality provisions of the Patient Safety Rule, which protect identifiable information being used to analyze patient safety events and improve patient safety.

**Health Insurance Portability and Accountability**

HIPAA provides rights and protections for participants and beneficiaries in group health plans. HIPAA includes protections for coverage under group health plans that limit exclusions for preexisting conditions; prohibit discrimination against employees and dependents based on their health status.

**9.2 Future Enhancement**

In real life, parties such as the government agents or organization alliances can be the TA. Although the existence of TA is helpful, we leave the completely distributed solution as a future work. One interesting future work is to enable multiparty collaborative learning without the help of TA.

**9.3 Conclusion**

In this work, we proposed the first secure and practical multi-party BPN network learning scheme over arbitrarily partitioned data. In our proposed approach, theparties encrypt their arbitrarily partitioned data and upload the ciphertexts to the cloud. The cloud can executemost operations pertaining to the BPN network learning algorithm without knowing any private information. The cost of each party in our scheme is independent to the number of parties. This work tailors the BGN homomorphic encryption algorithm to support the multi-party scenario, which can be used as an independent solution for other related applications. Complexity and security analysis shows that our proposed scheme is scalable, efficientand secure. One interesting future work is to enable multiparty collaborative learning without the help of TA.

**10. REFERENCES AND BIBLOGRAPHY**

[1] The health insurance portability and accountability act of privacy and security rules. url: <http://www.hhs.gov/ocr/privacy>.

[2] National standards to protect the privacy of personal health information. url: http://www.hhs.gov/ocr/hipaa/finalreg.html.

[3] M. Abramowitz and I. A. Stegun. Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables. Dover

books on mathematics. Dover, New York, 1964.

[4] A. Bansal, T. Chen, and S. Zhong. Privacy preserving backpropagation neural network learning over arbitrarily partitioned data. Neural Comput. Appl., 20(1):143–150, Feb. 2011.

[5] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-dnf formulas on ciphertexts. In Proceedings of the Second international conference on Theory of Cryptography, TCC’05, pages 325–341, Berlin, Heidelberg, 2005.

[6] T. Chen and S. Zhong. Privacy-preserving backpropagation neural network learning. Trans. Neur. Netw., 20(10):1554–1564, Oct. 2009.

[7] L. Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Handwritten digit recognition with a back-propagation network. In Advances in Neural Information Processing Systems, pages 396–404. Morgan Kaufmann, 1990.

[8] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Over-encryption: management of access control evolution on outsourced data. In Proceedings of the 33rd international conference on Very large data bases, VLDB ’07, pages 123–134. VLDB Endowment, 2007.

[9] T. El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In Proceedings of CRYPTO 84 on Advances in cryptology, pages 10–18, New York, NY, USA, 1985.

[10] S. E. Fahlman. Faster-learning variations on Back-propagation: An empirical study, pages 38–51. Morgan Kaufman, 1988.

[11] K. Flouri, B. Beferull-lozano, and P. Tsakalides. Training a svmbased classifier in distributed sensor networks. In *Proceedings of* *14nd European Signal Processing Conference*, pages 1–5, 2006.

[12] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[13] R. Grossman and Y. Gu. Data mining using high performance data clouds: experimental studies using sector and sphere. In *Proceedings of the 14th ACM SIGKDD international conference on* *Knowledge discovery and data mining*, KDD ’08, pages 920–927, New

York, NY, USA, 2008.

[14] R. L. Grossman. The case for cloud computing. *IT Professional*, 11(2):23–27, Mar. 2009.

[15] A. Inc. *Amazon Elastic Compute Cloud (Amazon EC2)*. Amazon Inc.,

http://aws.amazon.com/ec2/#pricing, 2008.

[16] R. Law. Back-propagation learning in improving the accuracy of neural network-based tourism demand forecasting. *Tourism* *Management*, 21(4):331–340, 2000.

[17] A. J. Menezes, P. C. V. Oorschot, S. A. Vanstone, and R. L. Rivest. Handbook of applied cryptography, 1997.

[18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Parallel distributed processing: explorations in the microstructure of cognition, vol. 1. chapter Learning internal representations by error propagation, pages 318–362. MIT Press, Cambridge, MA, USA,

1986.

[19] N. Schlitter. A protocol for privacy preserving neural network learning on horizontal partitioned data. In *Proceedings of the* *Privacy Statistics in Databases (PSD)*, Sep. 2008.

[20] S. Stolfo, A. L. P. S. Tselepis, A. L. Prodromidis, S. Tselepis, W. Lee, D. W. Fan, and P. K. Chan. Jam: Java agents for meta-learning over distributed databases. In *In Proc. 3rd Intl. Conf. Knowledge* *Discovery and Data Mining*, pages 74–81. AAAI Press, 1997.

[21] B. Yang, Y.-d. Wang, and X.-h. Su. Research and design of distributed neural networks with chip training algorithm. In *Proceedings of the First international conference on Advances in Natural* *Computation - Volume Part I*, ICNC’05, pages 213–216, Berlin,

Heidelberg, 2005.

[22] A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, SFCS’82, pages 160–164, Washington, DC, USA, 1982.

[23] J. Yuan and S. Yu. Privacy preserving back-propagation learning made practical with cloud computing. Securecomm’12, Pauda, Italy, Sep. 2012.

[24] S. Zang and S. Zhong. A privacy-preserving algorithm for distributed training of neural network ensembles. *To appear in Neural Computing and Applications*.