# INTRODUCTION

## Background

Mail server which helps to send mails through online with the help of unique Mail ID's. Here user can send mails within a fraction of second with a single click. Every mail the user sends it passes through mail servers. The person can send mail only from one companymailserver.com to another companymailserver.com.

There are many mail client servers in the Internet world like Gmail, Yahoo Mail, Outlook, etc. All the mail servers has basic features like Inbox, Outbox, Compose message, Sent mail, draft mail, Attachments, etc. The mail servers categorized into two categories, i.e; Outgoing mail server and Incoming mail server. The mails sending from SMTP address and for receiving messages it is using IMAP or POP address. SMTP communication mail server uses TCP port 25 AND POP server uses port 465. The email servers like Gmail, Yahoo Mail, Outlook mails are sending and receiving Emails in the non encrypted form which is not secure. Even mail servers uploaded content stores directly in the directory of the server without any file encryption. Currently some popular servers allows user to enable OTP method to secure the Email. It’s very easy to view those mails to hackers with the help of some software tools. Our secured mail server has client/server feature which helps users to send mail in a secured connection.

Our mail server project secures email account and its content. This Secured Mail Server which encrypts the Email messages to protect sensitive information from the hackers. In our project we have included OTP tool, Image encryption tool and password encryption tool to secure the Email data. Totally it allows organization to project overall access to multiple mail accounts.

We provide full-featured email server which supports S/MIME secure email (digital signing and message encryption using certificates) technology. Other encryption options include PGP and GNU Privacy Guard (GnuPG). This project can be used as Free and commercial basis. The PGP technology has ability to encrypt the message and in the same way it has option to decrypt the sent messages.

## Objectives

* + The main objective of this project is to develop a fully functional Mail System that allows the users to with others in a secured method.
  + Our secured Email server implements email security to secure user account and its email data from hackers.
  + Our Email security technology encompasses with multiple techniques to secure all kinds of Email Service.
  + Users can send and receive mails from the servers. Composed mail stores in sent box and received mail stores in the inbox.
  + Users login ID and passwords are stored in the database, Email Inbox module handles all the functions related mails like email forwarding, composing mails, email reply, view attachments, download attachments, etc in a secured way.
  + This system is reliable, cost effective and secure.
  + The sent mail directly sending to receivers Email ID in encrypted format.
  + Mail user needs to enter OTP (One time Password) to send and view the mail which is received by SMS.
  + The send can send text files inside the image.
  + User can select option of password before sending mail.
  + All the mails are sending under secured port in the encryption format. This mail server supports secured threading support which is automated. It handles the port connection and disconnection to a peer.
  + It supports all kinds of client and server port connections. The server port which accepts multiple connections and all the client socket connected to the server port.

## Purpose and Scope

### Purpose

Now a days, Internet and WWW become very popular communication media. These technology changing our daily life. The modern life activities are driven by internet. Email feature also popular in the latest technology which everyone use. The main purpose of developing this project is to provide high secured Email server for any organization. This system handles multiple users. This project allows users to register through online by entering their profile details. This project will hide users profile data and their email records. The login password also encrypted in this system. If the user is not using mail for 3-4 minutes the system will automatically logs out users account. This project is very helpful for many organizations to keep their Email data secure.

### Scope

The main scope of this project is to secure the Email account and Email data. The mails sent from Secured mail server can’t be read anyone except end user. This email server doesn’t required any anti-virus and spam detection softwares. This system scans everything before sending the mail. For every user the system suggests to enter strong password. The entered password encrypts and stores in the database. While sending mail this system sends notification of current location, Sent date, IP address of the user, etc.

The SECURED MAIL SERVER offers a secure 128 bit encryption to protect sensitive data from unauthorized access. An especially secured network is not required. Digital signatures protect the report from unauthorized manipulations and ensures the data integrity. The additional use of certificates makes sure the message was sent by you and can only be received by the authorized recipient. The report transmission is being logged and the message status is therefore always retraceable.

The TCP and UDP protocols use following ports for mail communication. The mail server consists of several components that work together to compose, send, receive, store and deliver a mail.

* File Transfer protocol (FTP) – Port 21
* TELNET – Port 23
* Simple Mail Transfer Protocol(SMTP) – Port 25
* HTTP – Port 80
* HTTPS – Port 443

# SURVEY OF TECHNOLOGIES

Survey of technologies signifies details of all related technologies that are necessary to complete this project. In this section, the descriptions of various technologies used in different parts of this system will be presented. This section also explains why the selected technologies have been chosen for this project.

## Hardware Interface

* **Operating system:** Windows XP or higher version
* **Hard disk:** Minimum 40 GB hard disk
* **RAM :** Minimum 1 GB RAM
* **Processor:** Intel Pentium or above

## Software Interface

* XAMPP 1.8.2
* Apache server
* PHP 5.4
* MYSQL server 5.5
* Adobe Dreamweaver CS 6.0
* Notepad++

## Programming language

* Front End: PHP
* Back End: MySQL

## Communication Interfaces

* The project shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.
* Server user internet connection to send and receive the mails from one server to another server.

## PHP

PHP is a script language and interpreter that is freely available and used primarily on Linux Web servers. PHP, originally derived from Personal Home Page Tools, now stands for PHP: Hypertext Preprocessor.

PHP executes on the server, while a comparable alternative, JavaScript, executes on the client. PHP is an alternative to Microsoft's Active Server Page (ASP) technology. As with ASP, the PHP script is embedded within a Web page along with its HTML. Before the page is sent to a user that has requested it, the Web server calls PHP to interpret and perform the operations called for in the PHP script.

An HTML page that includes a PHP script is typically given a file name suffix of ".php" ".php7," or ".phtml". Like ASP, PHP can be thought of as "dynamic HTML pages," since content will vary based on the results of interpreting the script.

## MySQL

MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. This tutorial will give you a quick start to MySQL and make you comfortable with MySQL programming.

## AJAX

AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.

* Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display.
* Conventional web applications transmit information to and from the sever using synchronous requests. It means you fill out a form, hit submit, and get directed to a new page with new information from the server.
* With AJAX, when you hit submit, JavaScript will make a request to the server, interpret the results, and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.
* XML is commonly used as the format for receiving server data, although any format, including plain text, can be used.
* AJAX is a web browser technology independent of web server software.
* A user can continue to use the application while the client program requests information from the server in the background.
* Intuitive and natural user interaction. Clicking is not required, mouse movement is a sufficient event trigger.
* Data-driven as opposed to page-driven.

## HTML

HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web.[3] Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

## CSS

CSS is used to control the style of a web document in a simple and easy way. CSS is the acronym for "Cascading Style Sheet". This tutorial covers both the versions CSS1,CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts.

## Javascript

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

## Bootstrap

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.

## Datatables

DataTables is a plug-in for the jQuery Javascript library. It is a highly flexible tool, based upon the foundations of progressive enhancement, and will add advanced interaction controls to any HTML table.

## Jquery

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

## Json

JSON or JavaScript Object Notation is a lightweight text-based open standard designed for human-readable data interchange. The JSON format was originally specified by Douglas Crockford, and is described in RFC 4627. The official Internet media type for JSON is application/json. The JSON filename extension is .json.

## Adobe Dreamweaver

Adobe Dreamweaver CC is a web design and development application that combines a visual design surface known as Live View and a code editor with standard features such as syntax highlighting, code completion, and code collapsing as well as more sophisticated features such as real-time syntax checking and code introspection for generating code hints to assist the user in writing code. Combined with an array of site management tools, Dreamweaver lets its users design, code and manage websites, as well as mobile content. Dreamweaver is an Integrated Development Environment (IDE) tool. You can live preview of changes for the frontend. Dreamweaver is positioned as a versatile web design and development tool that enables visualization of web content while coding.

## Apache server

Apache is the most widely used web server software. Developed and maintained by Apache Software Foundation, Apache is an open source software available for free. It runs on 67% of all webservers in the world. It is fast, reliable, and secure. It can be highly customized to meet the needs of many different environments by using extensions and modules. Most WordPress hosting providers use Apache as their web server software. However, WordPress can run on other web server software as well.

## MySQL Server

MySQL is the world's most popular open source database. With its proven performance, reliability, and ease-of-use, MySQL has become the leading choice of database for web applications of all sorts, ranging from personal websites and small online shops.

## Notepad++

Notepad++ is a free, open source application for writing text. A powerful editor full of options, Notepad++ is ideal for programming as well as normal text. Notepad++ is a free source code editor and Notepad replacement that supports several languages.

# REQUIREMENTS AND ANALYSIS

## Problem Definition

E-mail is now the most widely used method of communication between businesses and individuals; unfortunately, it is not the safest and most reliable in the current technology. Securing an email system is the main responsibility of the business. Virus-infected e-mails on network lines are the main reasons that affect the reliability of e-mail. Preventing e-mail messages from getting into the wrong hands can be avoided by encrypting the information being transmitted. With the help of digital certificate user can encrypt the mails.

Encrypting the connection prevents unauthorized users on the network from intercepting and capturing your login credentials and any email messages you send or receive as they leave your email provider’s server and travel from server to server around the Internet.

Encrypting email messages before they’re sent means that even if a hacker or anyone other than the intended recipient should intercept your email messages, they’re unreadable, and essentially useless.

Finally, if you store backed-up email messages in an email client, such as Microsoft Outlook, hackers may gain access despite password protection of your accounts and even your device. Email encryption ensures that even if access is obtained, the content of your email messages is unreadable.

## Requirement Specification

A software requirements specification is a comprehensive description of the intended purpose and environment for software under development. This document fully describes what the software will do and how it will be expected to perform.

### Modules of the software

The project has two types of user. They are Admin and users. The Secured mail server project has following modules:

#### Home page:

The home page has two options. New can register to the system and existing user can login to the system by entering login credentials.

* Login component
* Registration component
* Forgot password component

#### Registration module :

In the registration module the new user can register by entering registration details such as name, email id, password, confirm password, date of birth, Recovery mail, mobile number, captcha code, etc. After the successful registration the system redirects to inbox page.

* Registration component
* Mobile registration component

#### Login module:

In the login module the user can login to the website by entering login id and password. After successful login the page redirects to inbox page. If security is enabled then the system asks to enter OTP password to login the system. After entering valid OTP code the system redirects to inbox page.

* Login module

#### Forgot password module:

In the forgot password page the existing user can recover password by entering mobile number or recovery mail. After entering mobile number or recovery mail the the system request OTP code which is sent to Email or mobile phone. The user can change the password by entering new password and confirm password.

* Forgot password component
* Reset password component

#### Compose message

In the compose message user can write messages and they can upload files to the server. After composing message user can send to other mail ID. The unsent composed messages automatically stores in the draft folder.

* Compose message

#### OTP Login security

If the OTP login is enabled ,then the user needs to enter 6 digit OTP code to login the system. OTP login authentication passcode is sent through users phone to login and validate the user. User must register their mobile phone to enable OTP login.

* Login module
* OTP entry component

#### Mail encryption technology:

In the mail encryption technology it sends mail in encrypted form. This provides a high level of security by encrypting messages and digitally signing messages. The contents of a message are protected so that it can only be read by the intended recipient. A digital signature provides authentication of messages and it assures those messages hasn’t tampered.

* Mail Encryption component
* Mail composer

#### Stenography Technology:

This encrypts mails inside the images. This is an is another encryption technique that can be used along with cryptography as an extra-secure method in which to protect data. The mail contents are protected inside images. Only the end user can read messages by decrypting the images.

* Image encryption tool
* Mail composer

#### Inbox module

The inbox module contains received messages. In the first page user can view subject with sender, date and time. After clicking message the inbox displays complete message content with attachments.

* View message list
* View individual message

#### Sent Mail module

The sent mail interface is same as Inbox but here only the sent mail can be viewed. Deleted sent mail goes to draft folder.

* View sent message list
* View sent detailed message

#### Change password module:

Logged in user can change password by entering old password and new password. Password stores in encrypted format.

* Change password

#### Account module

In the account module the user can update their profile record. The system will ask to enter login password to update the record.

* Update profile
* Update password

#### Mail server

The project "Secured Mail Server" is divided in to three main components. i.e; Server component, Client component, Email Inbox component. Server accepts the connection from different clients through server socket class and all the details regarding user connection establishment, sending, receiving and termination is stored in the server. Users can connect to the server when server is active, Each user can send and receive mails, attachments from other users.

* Mail encryption
* Stegnography Technology
* Compose mail

#### Dashboard module

The administrator is the owner of the website who can view registered member records through online. Admin can create multiple employees.

* View users
* View accounts

#### Logout module

This module logs out user account. After the logout the system displays main page.

## Software and Hardware Requirements

### Hardware Requirements

* Operating system: Windows XP or higher version
* Hard disk: Minimum 40 GB hard disk
* RAM : Minimum 1 GB RAM
* Processor: Intel Pentium or above
* Keyboard and Mouse

### Software Requirements

* XAMPP 1.8.2
* Server : Apache server
* PHP 5.4
* Database server: MYSQL server 5.5
* IDE: Adobe Dreamweaver CS 6.0
* Notepad++
* Web browser: Mozilla Firefox OR Google Chrome

## Preliminary Product Description

Email encryption relies on a Public Key Infrastructure or PKI, in most cases, a combination of a private key (known only by you) and a public key (known only to those you choose to distribute it to or even made publicly available). Those sending emails that they want to encrypt would use the public key, while the intended recipient would use the private key to decrypt those messages into a readable format. In the PKI model, anyone can use a public key to encrypt email, but each encrypted message can only be decrypted by a unique private key.

Best practices for email encryption include consistently encrypting all messages you send and receive. Encrypting only email messages containing sensitive information raises a flag to hackers, pointing them directly to the messages that are most likely to contain valuable, sensitive information – the very information you’re trying to prevent outsiders from gaining access to in the first place.

When user encrypt all email messages as a standard practice, hackers wishing to access your personal information have a more substantial task in front of them. Decrypting email messages one-by-one in search of a single message containing sensitive information is a daunting and tedious task that even the most dedicated hackers may feel is not worth the effort.

Email process:

Now that you know the basics about incoming and outgoing mail servers, it will be easier to understand the role that they play in the emailing process. The basic steps of this process are outlined below for your convenience.

1. After composing a message and hitting send, your email client - connects to your domain's SMTP server. This server can be named many things; a standard example would be smtp.example.com.
2. Your email client communicates with the SMTP server, giving it your email address, the recipient's email address, the message body and any attachments.
3. The SMTP server processes the recipient's email address - especially its domain. If the domain name is the same as the sender's, the message is routed directly over to the domain's POP3 or IMAP server - no routing between servers is needed. If the domain is different, though, the SMTP server will have to communicate with the other domain's server.
4. In order to find the recipient's server, the sender's SMTP server has to communicate with the DNS, or Domain Name Server. The DNS takes the recipient's email domain name and translates it into an IP address. The sender's SMTP server cannot route an email properly with a domain name alone; an IP address is a unique number that is assigned to every computer that is connected to the Internet. By knowing this information, an outgoing mail server can perform its work more efficiently.
5. Now that the SMTP server has the recipient's IP address, it can connect to its SMTP server. This isn't usually done directly, though; instead, the message is routed along a series of unrelated SMTP servers until it arrives at its destination.
6. The recipient's SMTP server scans the incoming message. If it recognizes the domain and the user name, it forwards the message along to the domain's POP3 or IMAP server. From there, it is placed in a sendmail queue until the recipient's email client allows it to be downloaded. At that point, the message can be read by the recipient.

## Conceptual Models

### Data Flow Diagram

Data Flow Diagram (DFD) is the graphical representation of flow of data through the system. DFD’s can also be used for the visualization of data processing (structured design). On a DFD, data item flow from an external data source or an internal data store to an internal data store or external data sink, via an internal process.

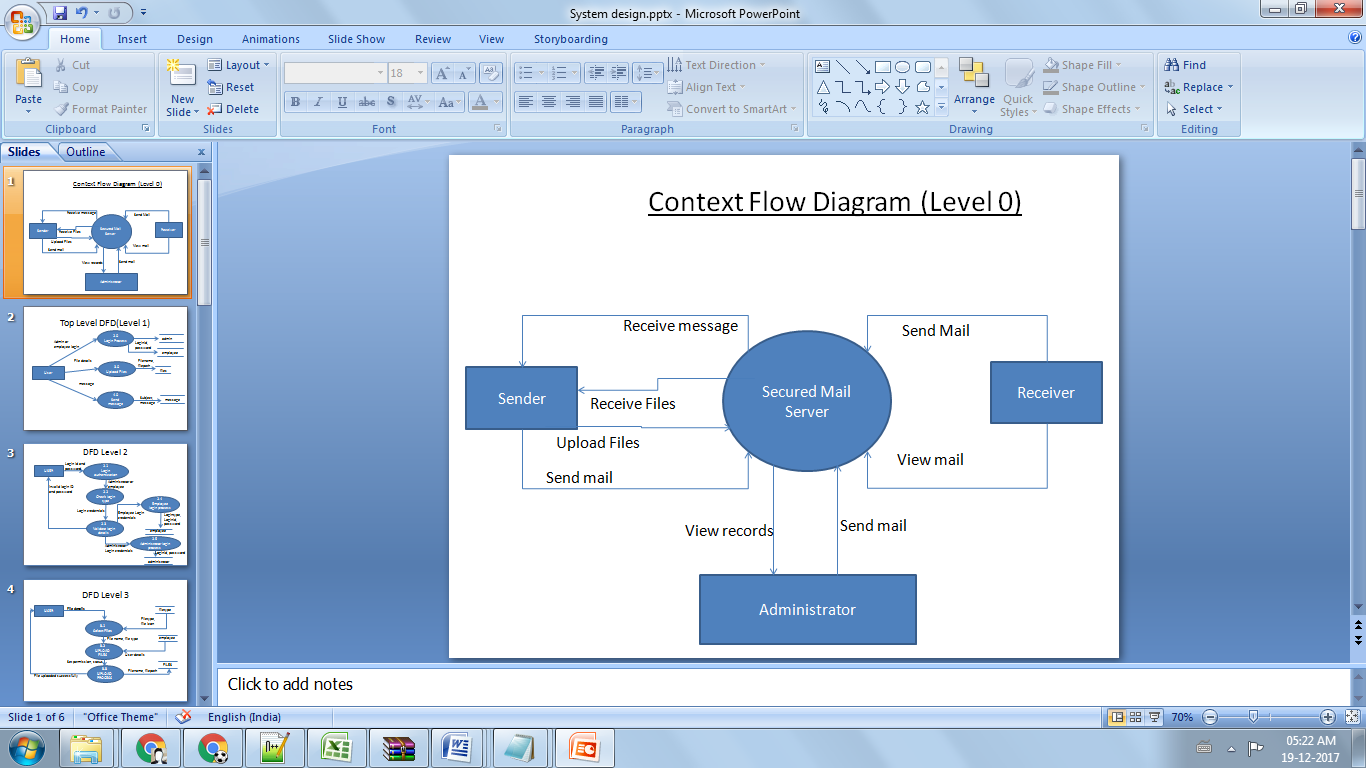
A DFD provides no information about the timing or the ordering of the processes or about the processing of the system in parallel or sequential terms. Therefore it is quite different from a flow chart, which shows the flow of control through an algorithm, allowing the reader to understand the operations performed, their order, the circumstances, but neither does it reveal the required input or the desired output of the system nor does it keep a track of flow of data or its storage.

**Notation**

|  |  |
| --- | --- |
| **NOTATION** | **DESCRIPTION** |
|  | A Circle represents a process or transform that is applied to data (or control) and changes it in some way. |
|  | A Rectangle is used to represent an external entity, that is a system element or another system that produces information for transformation by the software or receives information produced by the software. |
|  | An arrow represents one or more data items or data objects |
|  | The open box represents data store – stored information that is used by the software. |

#### Context Flow Diagram

A context Flow Diagram is a top level (also known as level 0) data flow diagram. It only contains one process node (process 0) that generalizes the function of the entire system in relationship to external entities. In context diagram, the entire system is treated as a single process and all its inputs, outputs, sinks and sources are identified and shown.The Context Flow Diagram of the proposed system is shown below.

****

### Top level DFD:

#### Level 2.1 DFD:

#### Level 2.2 DFD:

#### Level 2.3 DFD

**Level 2.4 DFD**

### Entity Relationship Diagram

An Entity-Relationship model is an abstract conceptual representation of structured data. Entity-Relationship modeling is a relational schema database modeling method, used in software engineering to produce a type of conceptual data model of a system. Diagrams created using this process are called Entity-Relationship Diagrams, or E-R Diagrams. A set of primary components are identified for the E-R Diagram: Data Object, Attributes, Relationships and various type indicators. The primary purpose of the E-R Diagram is to represent data objects and their relationships.

**Data Objects**: A data object is a representation of almost any of the information that must be understood by the software. Composite information refers to something that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence or an event, a role, an organizational unit, a place, or a structure. The first letter of an entity is always in upper case. The entity is represented by a rectangle.

**Attributes**: Attributes define the properties of a data object and take on one of three different characteristics. They can be used to (i) name an instance of the data object, (ii) describe the instance, or (iii) make reference to another instance in another table. In addition, one or more of these attributes must be defined as an identifier- that is, the identifier attribute becomes a “key” when we want to find an instance of the data object.

**Relationship**: The relationship is the interaction between the entities. A relationship is mainly represented by a diamond shape.

**Cardinality**: The data model must be capable of representing the number of occurences objects in a given relationship. The three main relationships are:

* One-to-One expressed as (1:1)
* One-to-Many expressed as (1:M)
* Many-to-Many expressed as (M:M)

### Notations

**Table 3.2 – ER Diagram notations**

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Symbol Name** | **Description** |
| **Entities** | | |
| ERD Symbols and Meaning - Entity | Entity | An entity is represented by a rectangle which contains the entity’s name. |
| ERD Symbols and Meaning - Weak Entity | Weak Entity | An entity that cannot be uniquely identified by its attributes alone. The existence of a weak entity is dependent upon another entity called the owner entity. The weak entity’s identifier is a combination of the identifier of the owner entity and the partial key of the weak entity. |
| ERD Symbols and Meaning - Associative Entity | Associative Entity | An entity used in a many-to-many relationship (represents an extra table). All relationships for the associative entity should be many |
| **Attributes** | | |
| ERD Symbols and Meaning - Attribute | Attribute | In the Chen notation, each attribute is represented by an oval containing attribute’s name |
| ERD Symbols and Meaning - Key attribute | Key attribute | An attribute that uniquely identifies a particular entity. The name of a key attribute is underscored. |
| ERD Symbols and Meaning - Multivalue attribute | Multivalued attribute | An attribute that can have many values (there are many distinct values entered for it in the same column of the table). Multivalued attribute is depicted by a dual oval. |
| ERD Symbols and Meaning - Derived attribute | Derived attribute | An attribute whose value is calculated (derived) from other attributes. The derived attribute may or may not be physically stored in the database. In the Chen notation, this attribute is represented by dashed oval. |
| **Relationships** | | |
| ERD Symbols and Meaning - Relationship | Strong relationship | A relationship where entity is existence-independent of other entities, and PK of Child doesn’t contain PK component of Parent Entity. A strong relationship is represented by a single rhombus |
| ERD Symbols and Meaning - Identifying Relationship | Weak (identifying) relationship | A relationship where Child entity is existence-dependent on parent, and PK of Child Entity contains PK component of Parent Entity. This relationship is represented by a double rhombus. |

### ER Diagram

### Database Design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in Data Definition Language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the table and views. In an object database, the entities and relationships map directly to object classes and name the relationships.

Below given are the tables designed for this system.

#### Database table – user

**Table Name :** user

**Description :** user table used to store user record

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| User\_id | INT | Primary Key AUTO INCREMENT |
| name | VARCHAR (15) | NOT NULL |
| emailid | VARCHAR(16) | NOT NULL |
| Password | Varchar(100) | NOT NULL |
| Dateofbirth | DATE | NOT NULL |
| mobilenumber | Varchar(15) | NOT NULL |
| recovermail | Varchar(50) | NOT NULL |
| address | Varchar(100) | NOT NULL |
| status | Varchar(10) | NOT NULL |

#### Database table – user

**Table Name :** userlogin

**Description :** This table stores users login history

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| Userlogin\_id | INT | Primary Key AUTO INCREMENT |
| Name | VARCHAR (15) | NOT NULL |
| Emailed | VARCHAR(16) | NOT NULL |
| Password | Varchar(100) | NOT NULL |
| Logindate | DATE | NOT NULL |

#### Database table – encryption

**Table Name :** encryption

**Description :** This table stores encryption details

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| encryptionid | INT | Primary Key AUTO INCREMENT |
| Encryptiontype | VARCHAR (15) | NOT NULL |
| encryptiondetail | TEXT | NOT NULL |
| status | Varchar(10) | NOT NULL |

#### Database table – mail

**Table Name :** mail

**Description :** This table stores mail details

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| mailid | INT | Primary Key AUTO INCREMENT |
| mailtype | VARCHAR (15) | NOT NULL |
| Senderid | VARCHAR(16) | NOT NULL |
| Receiverid | Varchar(100) | NOT NULL |
| messagedatetime | DATETIME | NOT NULL |
| Subject | VARCHAR(150) | NOT NULL |
| Message | TEXT | NOT NULL |
| Sentipaddress | VARCHAR(50) | NOT NULL |
| Encryptionid | INT(10) | NOT NULL |
| Encryptiontype | VARCHAR(15) | NOT NULL |
| Readstatus | VARCHAR(12) | NOT NULL |

#### Database table – admin

**Table Name :** admin

**Description :** This table stores admin details

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| Adminid | INT | Primary Key AUTO INCREMENT |
| Adminname | VARCHAR (15) | NOT NULL |
| Loginid | VARCHAR(16) | NOT NULL |
| Password | Varchar(100) | NOT NULL |
| Admintype | VARCHAR(25) | NOT NULL |
| Lastlogin | DATETIME | NOT NULL |

#### Database table – files

**Table Name :** file

**Description :** This table stores file records

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Constraints** |
| Fileid | INT | Primary Key AUTO INCREMENT |
| filename | VARCHAR (50) | NOT NULL |
| filetype | VARCHAR(15) | NOT NULL |
| filedescription | TEXT | NOT NULL |
| mailid | Int | NOT NULL |
| uploaddate | DATETIME | NOT NULL |

# Future scope of the project

* We can add video conference or live streaming features in the future
* Web file manager can be implemented in the future
* We can integrate this application with Android apps in future.
* Inclusion of features like themes to make the system more interactive.
* Inclusion of features of video chatting.
* Inclusion for feature of conferencing system.

# REFERENCES

* Website: [www.w3schools.com](http://www.w3schools.com)
* Website: www.tutorialspoint.com
* PankajJalote, An Integrated Approach To Software Engineering, 4ed,

Publication: Narosa Publishing House

* RamezElmasri and Shamkant B. Navathhe, Fundamentals Of Database Systems, 3ed, Benjamin/Cummings,1994
* Steven Holzner et al, Java 2 Programming- Black Book, 2ed,

Publication: Dreamtech Press

Table of Contents

[INTRODUCTION 1](#_Toc501456508)

[Background 1](#_Toc501456509)

[Objectives 1](#_Toc501456510)

[Purpose and Scope 2](#_Toc501456511)

[Purpose 2](#_Toc501456512)

[Scope 2](#_Toc501456513)

[SURVEY OF TECHNOLOGIES 4](#_Toc501456514)

[Hardware Interface 4](#_Toc501456515)

[Software Interface 4](#_Toc501456516)

[Programming language 4](#_Toc501456517)

[Communication Interfaces 4](#_Toc501456518)

[PHP 5](#_Toc501456519)

[MySQL 5](#_Toc501456520)

[AJAX 5](#_Toc501456521)

[HTML 6](#_Toc501456522)

[CSS 6](#_Toc501456523)

[Javascript 6](#_Toc501456524)

[Bootstrap 6](#_Toc501456525)

[Datatables 6](#_Toc501456526)

[Jquery 6](#_Toc501456527)

[Json 6](#_Toc501456528)

[Adobe Dreamweaver 7](#_Toc501456529)

[Apache server 7](#_Toc501456530)

[MySQL Server 7](#_Toc501456531)

[Notepad++ 7](#_Toc501456532)

[REQUIREMENTS AND ANALYSIS 8](#_Toc501456533)

[Problem Definition 8](#_Toc501456534)

[Requirement Specification 8](#_Toc501456535)

[Modules of the software 8](#_Toc501456536)

[Software and Hardware Requirements 11](#_Toc501456537)

[Hardware Requirements 11](#_Toc501456538)

[Software Requirements 11](#_Toc501456539)

[Preliminary Product Description 11](#_Toc501456540)

[Conceptual Models 13](#_Toc501456541)

[Data Flow Diagram 13](#_Toc501456542)

[Top level DFD: 15](#_Toc501456543)

[Entity Relationship Diagram 18](#_Toc501456544)

[Notations 19](#_Toc501456545)

[ER Diagram 21](#_Toc501456546)

[Database Design 21](#_Toc501456547)

[Future scope of the project 24](#_Toc501456548)

[REFERENCES 25](#_Toc501456549)