**TRAINING REPORT**

**ON**

**“MAILING SYSTEM PROJECT”**

**SUBMITTED FOR DEGREE OF B.TECH**

**(COMPUTER ENGINEERING)**

**TO**

**Department of Computer Engineering**

**National Institute of Technology KURUKSHETRA**

**November-2016**

**Submitted By:-**

Vivek Bindal

Roll Number- 1130547

Section- CO-4

**ACKNOWLEDGEMENT**

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I would like to express my gratitude towards my parents & members of LINUX WORLD PVT. LTD. JAIPUR for their kind co-operation and encouragement which help me in completion of this project.

I would like to express my special gratitude and thanks to persons for giving me such attention and time.

My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

Vivek Bindal

1130547

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.NO | TITLE | PAGE NUMBER |
| 1. | Introduction | 6 |
| 2. | System Analysis | 10 |
| 3. | Project Model | 15 |
| 4. | System Implementation | 16 |
| 5. | Member Login or Register | 19 |
| 6. | Inbox | 21 |
| 7. | Compose | 22 |
| 8. | Sent | 23 |
| 9. | Drafts | 24 |
| 10. | Trash | 25 |
| 11. | Logout | 25 |

**1.INTRODUCTION**

**1.1** **ABOUT THE PROJECT**

**Project Title :** Letsmail.co.in

The Project titled as **“Letsmail.co.in”** is developed in Advanced Java as Front Tool and MYSQL 5.5 as Back End Server. The main aim of the project is to create a website to provide that provide voice mail , video mail and file attachment upto 50 MB to the registered members.

In this project people can register themselves and can login to the site and can access their mails. This will be a vastly complex Website development project which will take approximately 3 months to complete. The project will be split up into stages and documented thoroughly throughout.

**1.2 OBJECTIVE OF THE PROJECT**

In this site there is registration process for each user and they can login and can access the protected information of the site. The entire information available on the site is verified by the admin. This also site provides various features like:

* User friendly environment.
* Easy access of data.
* Easy maintenance.
* Providing better performance.
* Security of data.

**1.3 USERS OF THE PROJECT**

* Member

**Roles of Users:**

**Member:**

1. A member of the site can access emails.
2. Member can send file attachment upto 50 MB.
3. Member can contact through voice mail or video mail.

**2. SYSTEM ANALYSIS**

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

**2.1 PRELIMINARY INVESTIGATION**

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

* How present system works?
* What drawbacks are in the present system?
* What is their vision about the new system?
* What specific facilities they want from new system, which are currently not in scope of existing system?
* How data will flow in the system?
* Who will be authenticated to access data and his/her access rights?

**2.2** **DRAWBACKS OF EXISTING SYSTEM**

* E-mail limit was upto 250 per hour.
* Time consuming.
* File Attachment was upto 25 MB.
* Voice or Video Mailing was not available.

**2.3** **PROPOSED SYSTEM**

The aim of proposed system is to develop a system of improved facilities. The system provides voice and video mail feature and file attachment is extended upto 50 MB. The proposed system tries to extend older features... The proposed system helps the user to work in user friendly manner.

**2.4 EXPECTED ADVANTAGES OF PROPOSED SYSTEM**

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features:

* Ensure data accuracy’s.
* Minimize manual data entry.
* Minimum time needed for the various processing.
* Greater efficiency.
* Better service.
* User friendliness and interactive.

###### **2.5 SOFTWARE REQUIREMENT SPECIFICATION**

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional description, a representation of system behavior, an indication of performance requirement and design constraints, appropriate validation criteria, and other information pertinent to requirement.

**3.8.1** **Software and Hardware Specification**

**3.8.1.1 Minimum Hardware Specification**

Processor : Dual Core 2GHz

RAM : 512 MB

Hard disk : 80 GB

**3.8.1.2 Minimum Software Specification**

Operating System : Windows XP/Windows 7

Programming Language : JDK 1.6

Web Servers : Apache Tomcat 6.0

Database Tool (RDBMS) : My SQL 5.5 Sever

Web-Browser Program : IE6 above, Mozilla, Chrome

IDE Tool : NetBeans

External JAR Files : My-SQL Connector 5.1.13.jar   
 jstl.jar & standard.jar  
 mail.jar

**3.8.2 Technologies Used**

Programming Language **:** JDK1.6

Back End Tool (RDBMS) **:** My-SQL Server 5.5

IDE **:** NetBeans

Web-Server **:** Apache Tomcat-6

Client Side Tools **:** HTML/XHTML, CSS, Java Script, AJAX

Server Side Tools **:** Servlets, JSP, EL &JSTL, Java Mail-API, Connection

Pooling

Design Patterns **:** MVC (Model-View-Controller), Data Access Object

**3. PROJECT MODEL**

According to our project requirements, we have used WaterfallModel to develop our software.

Through this model we would be able to produce well-documented maintainable software in a manner that is very predictable and easy to understand.

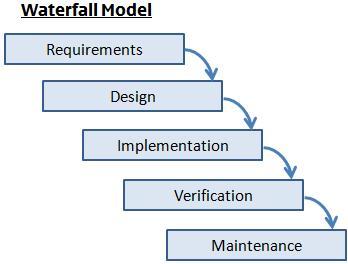
We are using this model because it reinforces the notion of “**Define before Design**” and **“Design before Code”** making it systematic approach.

**3.1 Waterfall Model**

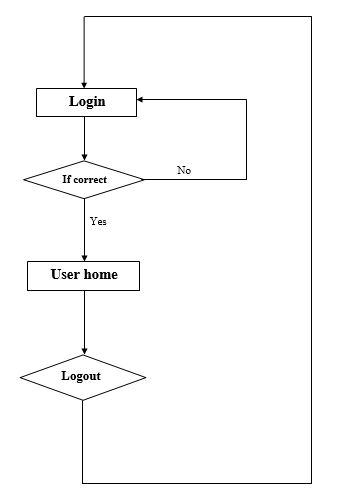
This is the most familiar model and consists of five phases given below –

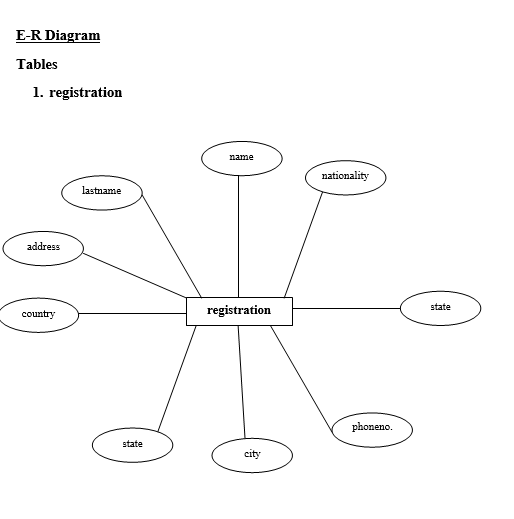
* Requirement Analysis and Specification
* Design
* Implementation and Unit Testing
* Operation and Maintenance

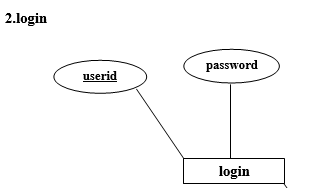
The phases always occur in this order and sequence. They must not overlap with one another. The developer must complete each phase before starting with the next one. It is called Waterfall model because its diagrammatic representation is similar to cascades of waterfall.

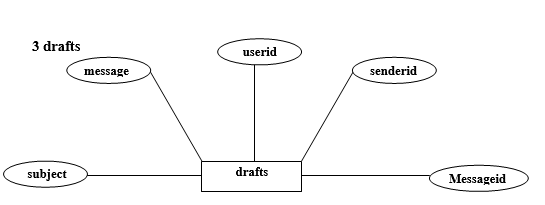


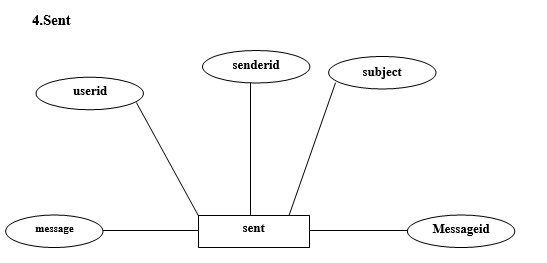
**3.2 Flow Charts**

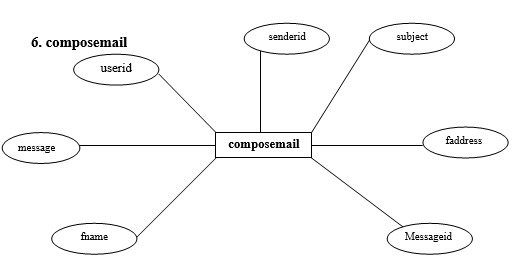
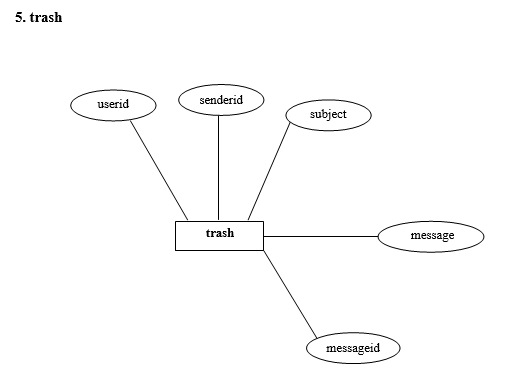
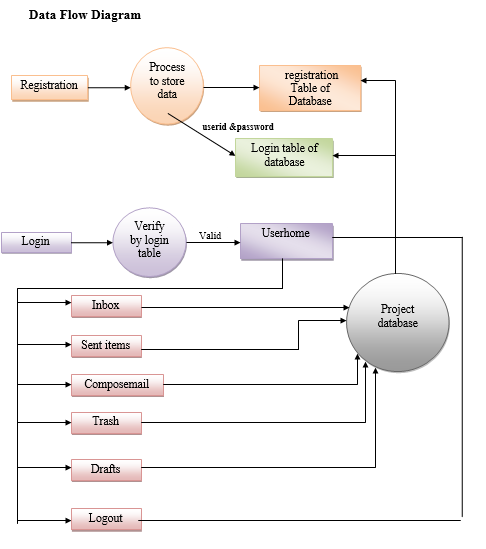










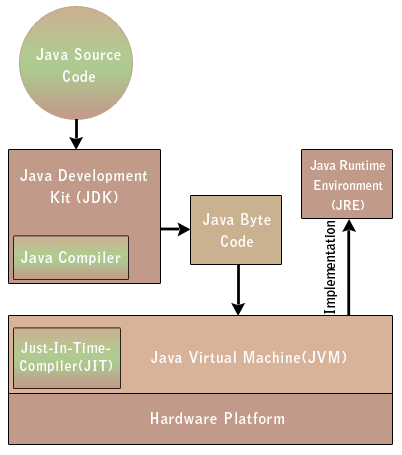
****

**SYSTEM IMPLEMENTATION**

**3.9 INTRODUCTION TO JAVA**

Java is a high-level, third generation programming language, like C, Fortran and many others. You can use Java to write computer applications that crunch numbers, process words, play games, store data or do any of the thousands of other things computer software can do. The Java language was developed at Sun Microsystems in 1991 as part of a research project to develop software for consumer electronics devices-television sets, VCRs, toasters, and the other sorts of machines you can buy at any department store. Java's goals at that time were to be small, fast, efficient, and easily portable to a wide range of hardware devices. Those same goals made Java an ideal language for distributing executable programs via the World Wide Web and also a general-purpose programming language for developing programs that are easily usable and portable across different platforms.

What's most special about Java in relation to other programming languages is that it lets you write special programs called *applets* that can be downloaded from the Internet and played safely within a web browser.



**3.9.1** **Features of Java**

### Java Is Object Oriented

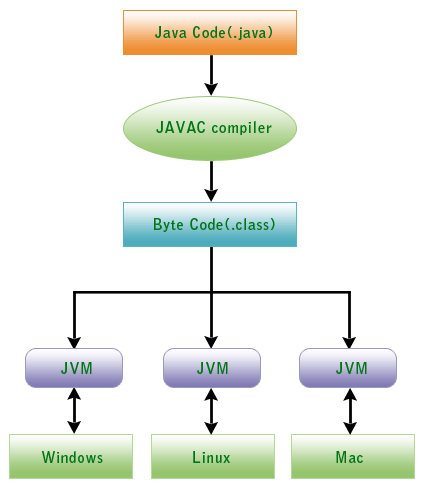
The object-oriented programming (OOP) technique is merely a way of organizing programs, and it can be accomplished using any language. Working with a real object-oriented language and programming environment, however, enables you to take full advantage of object-oriented methodology and its capabilities for creating flexible, modular programs and reusing code.

* **Java is Dynamic**

Java does not have an explicit link phase. Java source code is divided into .java files, roughly one per each class in your program. The compiler compiles these into .class files containing byte code. Each .java file generally produces exactly one .class file.

### Java Is Platform Independent

The ability of a program to move easily from one computer system to another-is one of the most significant advantages that Java has over other programming languages.



### Java Is Object Oriented

The object-oriented programming (OOP) technique is merely a way of organizing programs, and it can be accomplished using any language. Working with a real object-oriented language and programming environment, however, enables you to take full advantage of object-oriented methodology and its capabilities for creating flexible, modular programs and reusing code.

* **Java is Dynamic**

Java does not have an explicit link phase. Java source code is divided into .java files, roughly one per each class in your program. The compiler compiles these into .class files containing byte code. Each .java file generally produces exactly one .class file.

* **Java is High Performance**

Java byte codes can be compiled on the fly to code that rivals C++ in speed using a "just-in-time compiler." Several companies are also working on native-machine-architecture compilers for Java. These will produce executable code that does not require a separate interpreter, and that is indistinguishable in speed from C++.

* **Java is Multi-Threaded**

Java is inherently multi-threaded. A single Java program can have many different threads executing independently and continuously. Three Java applets on the same page can run together with each getting equal time from the CPU with very little extra effort on the part of the programmer.

* **Java is Garbage Collected**

You do not need to explicitly allocate or deallocate memory in Java. Memory is allocated as needed, both on the stack and the heap, and reclaimed by the garbage collector when it is no longer needed. There's no malloc(), free(), or destructor methods.

**3.9.2 Java Servlets**

A servlet is a Java programming language class used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers. Thus, it can be thought of as a Java Applet that runs on a server instead of a browser.

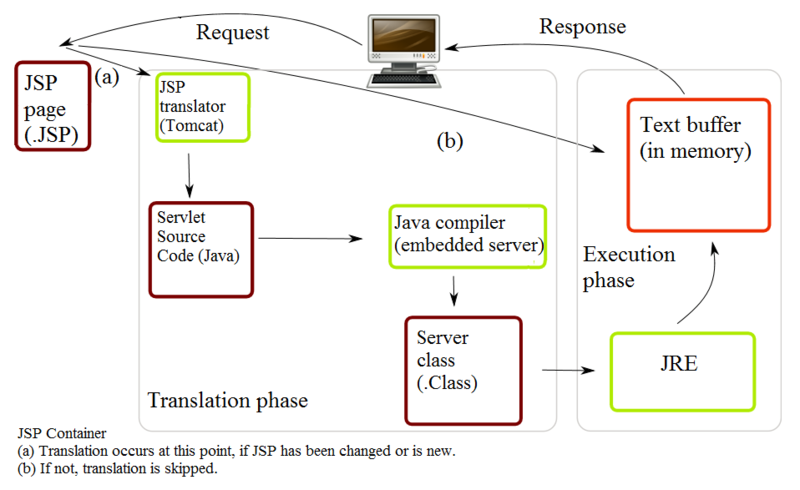
A Servlet is a Java class in Java EE that conforms to the Java Servlet API, a protocol by which a Java class may respond to requests. They are not tied to a specific client-server protocol, but are most often used with the HTTP protocol. Therefore, the word "Servlet" is often used in the meaning of "HTTP Servlet". Thus, a software developer may use a servlet to add dynamic content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as CGI and ASP.NET. Servlets can maintain state in session variables across many server transactions by using HTTP cookies, or URL rewriting.

To deploy and run a Servlet, a Web container must be used. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

The servlet API, contained in the Java package hierarchy javax.servlet, defines the expected interactions of the Web container and a servlet.

A Servlet is an object that receives a request and generates a response based on that request. The basic servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the servlet's configuration parameters and execution environment. The package javax.servlet.http defines HTTP-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application.

Servlets can be generated automatically from JavaServer Pages (JSP) by the JavaServer Pages compiler. The difference between Servlets and JSP is that Servlets typically embed HTML inside Java code, while JSPs embed Java code in HTML. While the direct usage of Servlets to generate HTML (as shown in the example below) is relatively rare nowadays, the higher level MVC web framework in Java EE (JSF) still explicitly uses the Servlet technology for the low level request/response handling via the FacesServlet. A somewhat older usage is to use servlets in conjunction with JSPs in a pattern called "Model 2", which is a flavor of the model-view-controller pattern.



### Life cycle of a servlet

1. The container calls the no-arg constructor.
2. The Web container calls the init() method. This method initializes the servlet and must be called before life of a servlet, the init() method is called only once.
3. After initialization, the servlet can service client requests. Each request is serviced in its own separate thread. The Web container calls the service() method of the servlet for every request. The service() method determines the kind of request being made and dispatches it to an appropriate method to handle the request. The developer of the servlet must provide an implementation for these methods. If a request for a method that is not implemented by the servlet is made, the method of the parent class is called, typically resulting in an error being returned to the requester.
4. Finally, the Web container calls the destroy() method that takes the servlet out of service. The destroy() method, like init(), is called only once in the lifecycle of a servlet.

**3.9.3 JSP (Java Server Page)**

Java Server Pages (JSP) is a Java technology that helps software developers serve dynamically generated web pages based on HTML, XML, or other document types. Released in 1999 as Sun's answer to ASP and PHP,JSP was designed to address the perception that the Java programming environment didn't provide developers with enough support for the Web.

To deploy and run, a compatible web server with servlet container is required. The Java Servlet and the Java Server Pages (JSP) specifications from Sun Microsystems and the JCP (Java Community Process) must both be supported by the container.

## Overview

Architecturally, JSP may be viewed as a high-level abstraction of Java servlets. JSPs are loaded in the server and are operated from a structured special installed Java server packet called a Java EE Web Application, often packaged as a file archive.

JSP allows Java code and certain pre-defined actions to be interleaved with static web markup content, with the resulting page being compiled and executed on the server to deliver an HTML or XML document. The compiled pages and any dependent Java libraries use Java bytecode rather than a native software format, and must therefore be executed within a Java virtual machine (JVM) that integrates with the host operating system to provide an abstract platform-neutral environment.

JSP syntax is a fluid mix of two basic content forms: *scriptlet elements* and *markup*. Markup is typically standard HTML or XML, while scriptlet elements are delimited blocks of Java code which may be intermixed with the markup. When the page is requested the Java code is executed and its output is added, in situ, with the surrounding markup to create the final page. JSPs must be compiled to Java bytecode classes before they can be executed, but such compilation is needed only when a change to the source JSP file has occurred.

Java code is not required to be complete (self contained) within its scriptlet element block, but can straddle markup content providing the page as a whole is syntactically correct (for example, any Java if/for/while blocks opened in one scriptlet element must be correctly closed in a later element for the page to successfully compile). This system of split inline coding sections is called *step over scripting* because it can wrap around the static markup by stepping over it. Markup which falls inside a split block of code is subject to that code, so markup inside an *if* block will only appear in the output when the *if* condition evaluates to true; likewise markup inside a loop construct may appear multiple times in the output depending upon how many times the loop body runs.

**3.10 MySQL SERVER 5.5**

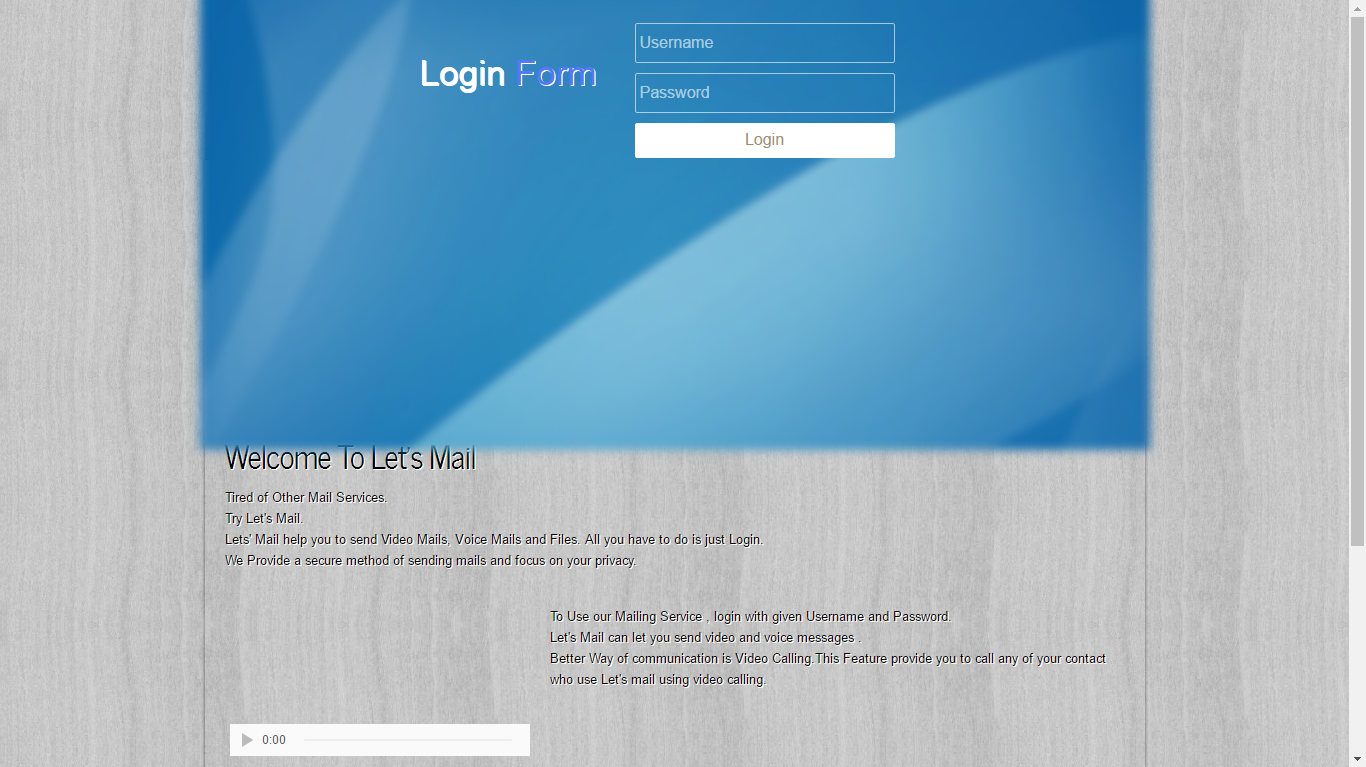
The MySQL Database powers the most demanding Web, E-commerce and Online Transaction Processing (OLTP) applications. It is a fully integrated transaction-safe, ACID compliant database with full commit, rollback and crash recovery and row level locking capabilities. MySQL delivers the ease of use, scalability, and performance that has made MySQL the world's most popular open source database. Some of the world's most trafficked websites like Facebook, Google, ticket master, and eBay rely on MySQL for their business critical applications.

|  |  |
| --- | --- |
| **Feature Benefits Description**tures & Benefits | |
| **Scalability and Flexibility** | Run anything from ...  Deeply embedded applications with a footprint of just 1MB, or  Massive data warehouses holding terabytes of information |
|  | |
| **High Performance** | Table and Index Partitioning  Ultra-fast load utilities  Distinctive memory caches  Full-text indexes, and more |
|  | |
| **High Availability** | Run high-speed master/slave replication configurations with Row-Based and Hybrid Replication  Specialized Cluster servers offering instant failover |
|  | |
| **Robust Transactional Support** | Complete ACID (atomic, consistent, isolated, durable) transaction support  Unlimited row-level locking  Distributed transaction capability, and  Multi-version transaction support |
|  | |
|  | |
| **Strong Data Protection** | Powerful mechanisms for ensuring only authorized users have access  SSH and SSL support safe and secure connections  Powerful data encryption and decryption functions |
|  | |
| **Comprehensive Application Development** | Support for stored procedures, triggers, functions, views, cursors, ANSI-standard SQL, and more  Plug-in libraries to embed MySQL database support into nearly any application |
|  | |
| **Management Ease** | Use Event Scheduler automatically schedule common recurring SQL-based tasks to execute on the database server  Average time from software download to complete installation is less than fifteen minutes |
|  | |
|  | |
| **Lowest Total Cost of Ownership** | Save on database licensing costs and hardware expenditures, all while cutting systems downtime |

**4. MEMBER LOGIN OR REGISTER**

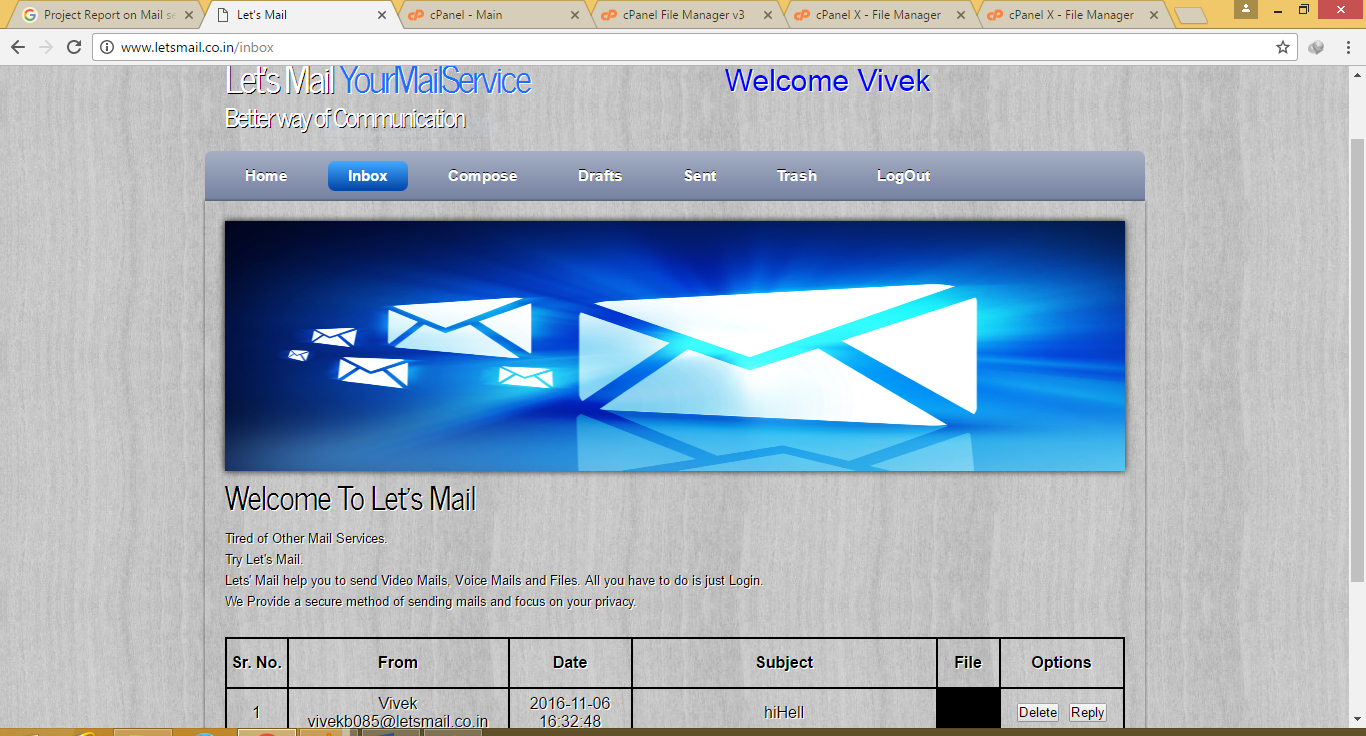
If user have username or password then he can login and use his mail service otherwise he has to register first.On registeration an OTP will be generated to user’s given mobile no. to verify the user.Then after verification user’s account will be active and he can use mailing service. If the user is not logged in, all the links will be hidden from the user except login , register and home page. After registeration all links will be shown.





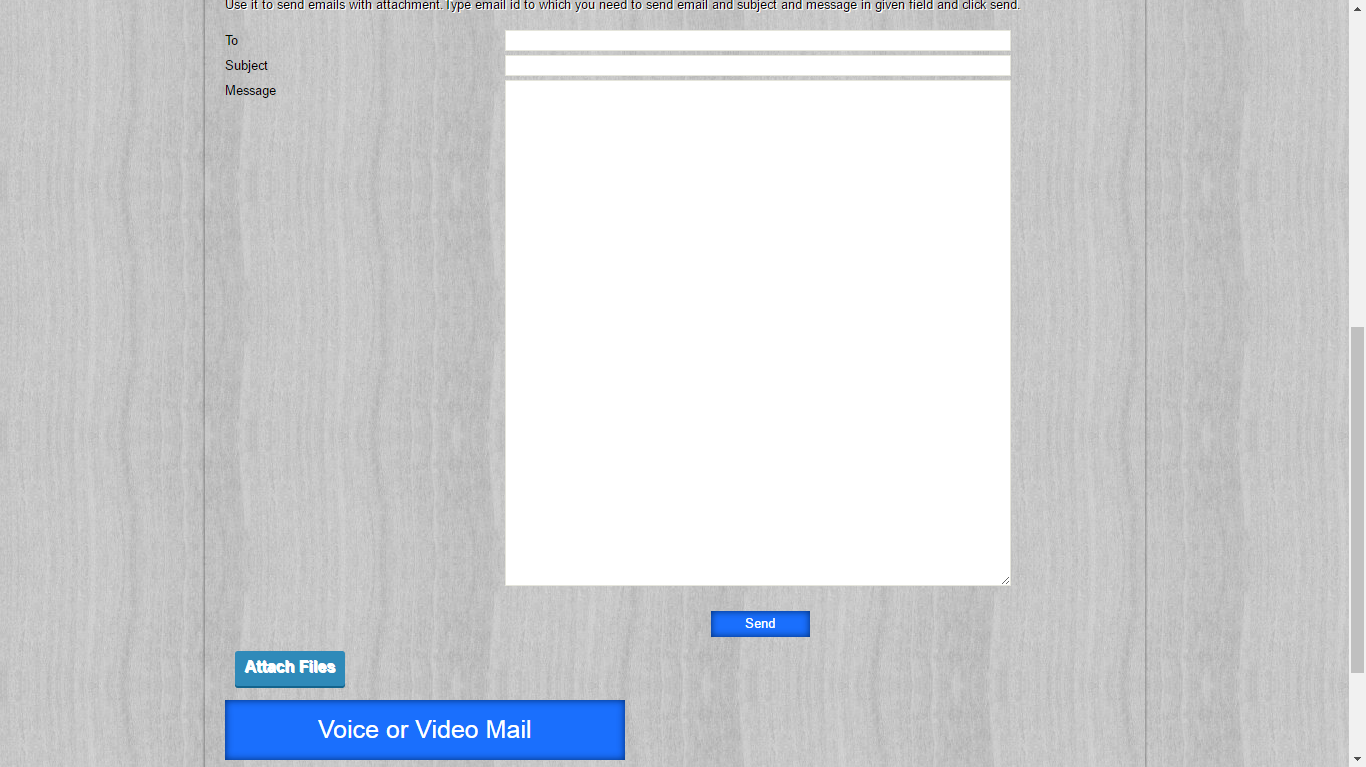
**5. INBOX**

In this part if there is any mail for user it checks for that mail and show that mail to the user. User can delete the mail if he wants.



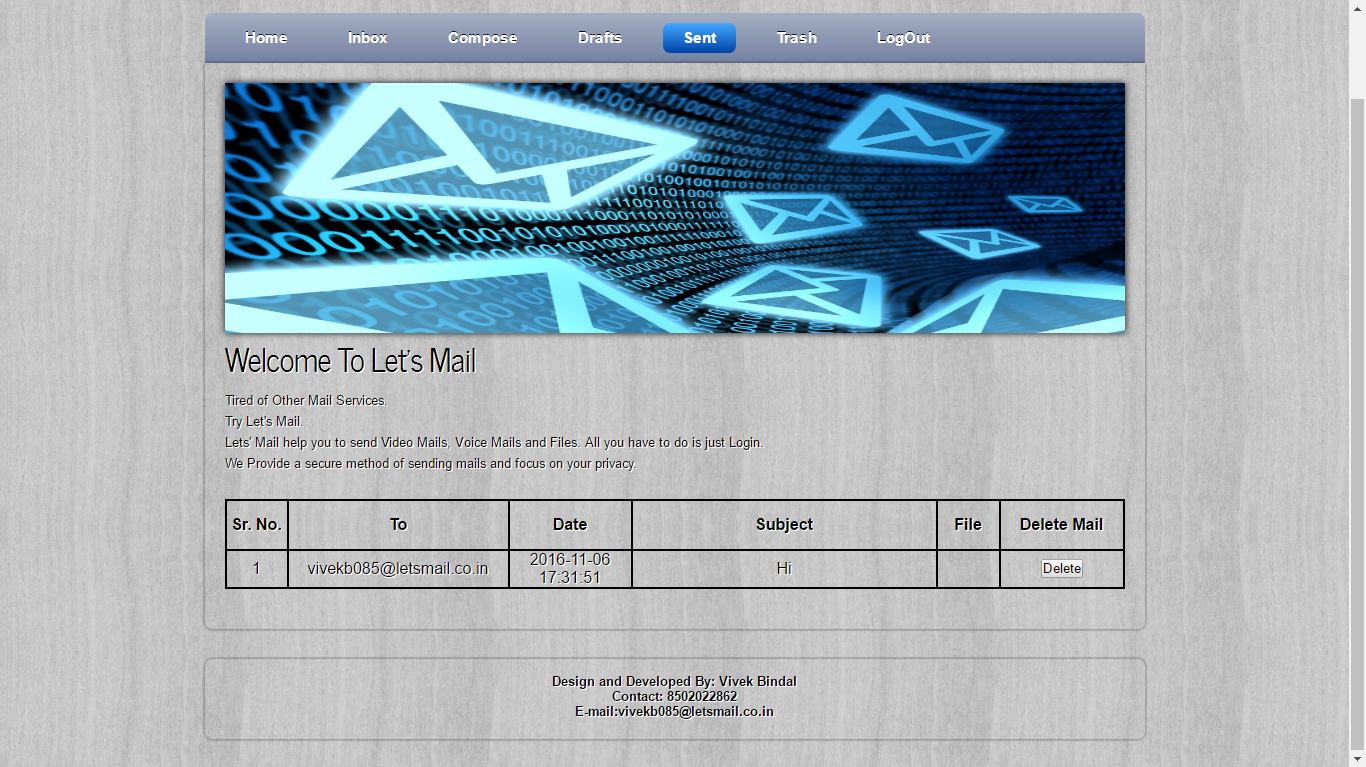
**6.COMPOSE**

In this part if user want to send a message he types the email address, subject , message. It provide file attachment upto 50 MB. User can also send voice or video mail to another user. At one time only 20 files can be sent.



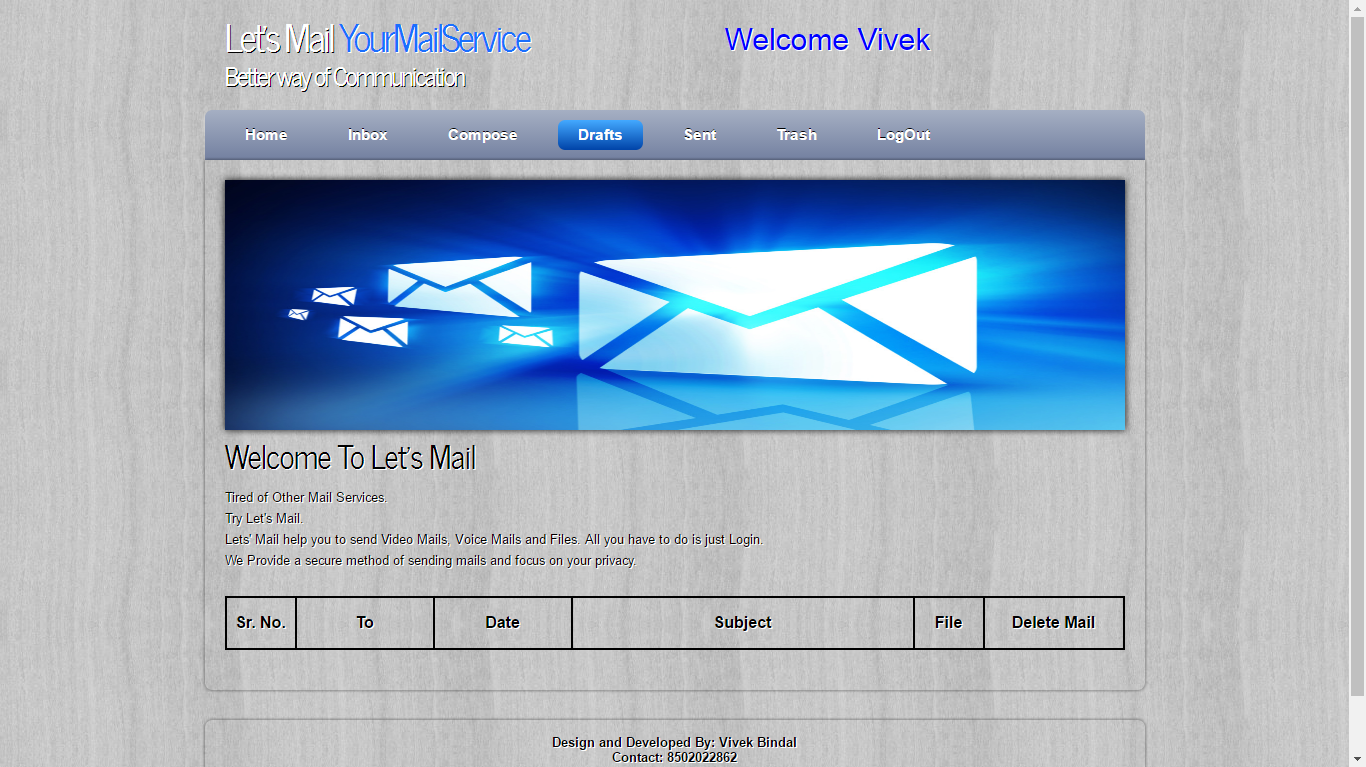
**7.SENT**

In this part all the message sent by the users are shown. User can delete those messages.Send files are also there. If user want to download he can click file name.



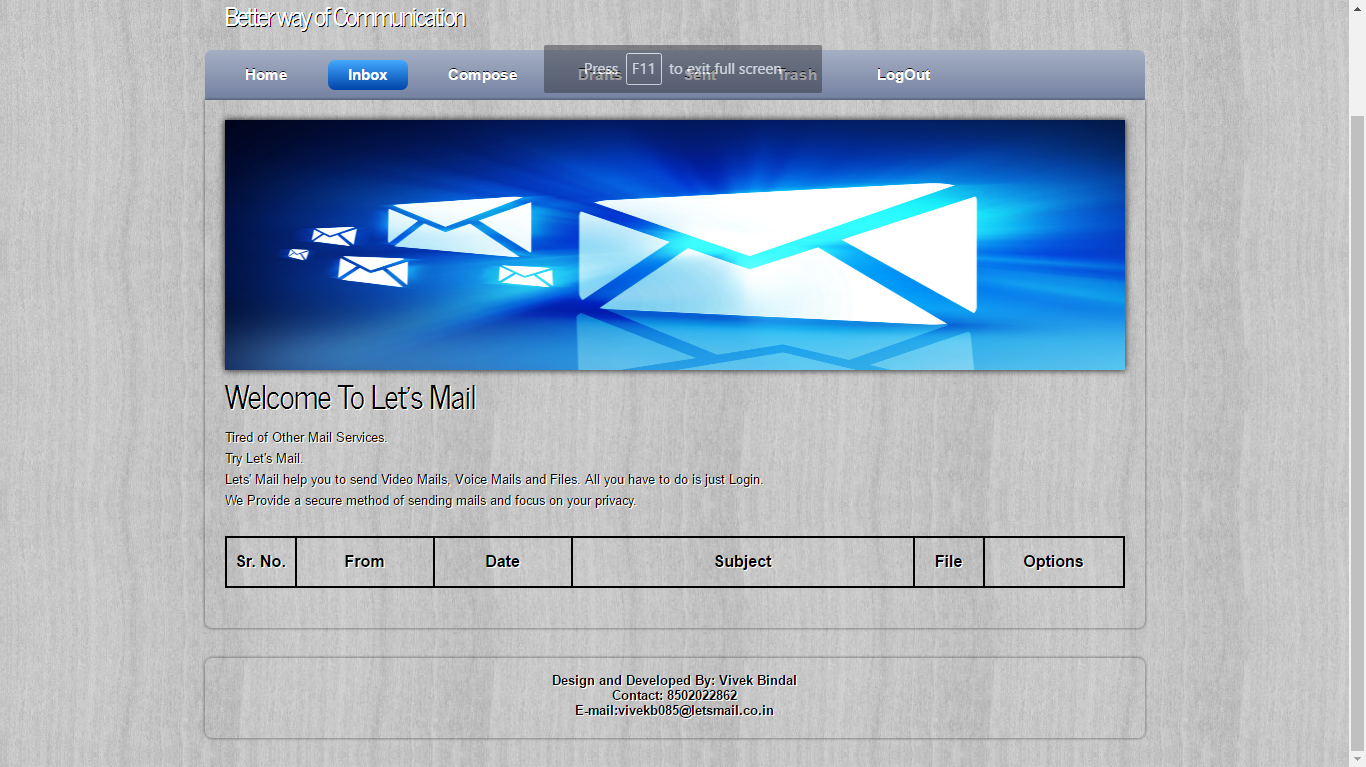
**8.DRAFTS**

If a user sends a message and it fails or upload any file and then refresh the page then all those messages and files are stored in drafts.



**9.TRASH**

All the message deleted from inbox goes to trash from where they can be permanently deleted.



**10.LOGOUT**

In this part session is terminated.