Reusable SMS/E-mail Web Client

By

C.Karthiga

Reg. No.: 71205621016

Of

Kumaraguru College of Technology

Coimbatore- 641006.

A PROJECT REPORT

Submitted to the

FACUTLY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

For the award of the degree

Of

MASTER OF COMPUTER APPLICATIONS

ANNA UNIVERSITY
CHENNAI 600 025

June, 2008

Kumaraguru College of Technology
Coimbatore – 641006

Department of Computer Science and Engineering

BONAFIDE CERTIFICATE

Certified that the project report titled "**Reusable SMS/Email Web Client**" is the bonafide work of the **Miss. Karthiga.C (Reg No.71205621016)**, who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

RKKavitha

GUIDE

Mrs.R.K.Kavitha, M.C.A., M.Phil

HEAD OF THE DEPARTMENT

Dr.M.Gururajan, Ph.D.,

Submitted to Project and Viva Examination held on 01.07.2008

at Kumaraguru College of Technology, Coimbatore.

INTERNAL EXAMINER

1/7/08

EXTERNAL EXAMINER

Infosys®

Infosys Technologies Limited
No 350, Hebbal Electronics City
Hootagalli, Mysore - 570 018
Tel: 91 821 240 4101 Fax: 91 821 240 4200
www.infosys.com

PROJECT COMPLETION CERTIFICATE

This is to certify that **Miss. C. Karthiga** of **Kumaraguru College of Technology,** Coimbatore, Tamil Nadu-641006, was associated with us as a project trainee to carry out her academic project for the partial fulfillment of award of Master of Computer Applications for the year 2008.

She has successfully completed the project titled **"Reusable SMS/Email Web Client"** as per the requirements.

Senior Project Manager
Infosys Technologies Limited,
Mysore

ABSTRACT

Functionality Overview

This Project "Reusable SMS / Email Web client" is an application which is used to automatically send SMS / E-Mail to the application users, customers, and clients when any business need or event arises. This project is a re-usable component and it can be incorporated with any application which needs to communicate any information to the application users or customers automatically when any event occurs or in any business need.

Send Email

Send Email is an important aspect of Reusable SMS /Email web client. In this the user can send Email to single or multiple persons according to the user need. If the user wants to send email to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

Send SMS

Send SMS is another important aspect of Reusable SMS /Email web client. In this the user can send SMS to single or multiple persons according to the user need. If the user wants to send SMS to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

View Report

View Report is another aspect of Reusable SMS /Email web client. In this the user can view the SMS sent report or email sent report by specifying from and to date. If the user wants to view SMS report, then the user needs to select SMS report check box, and specify the date. If the user wants to view email report, then the user needs to select email report check box, and specify the date. Then it will display the number of emails sent or SMS sent.

ACKNOWLEDGEMENT

First and foremost I thank God for his good will and blessings showered on me throughout the project. The success of this project needs cooperation and encouragement from different quarters. Words are inadequate to express my profound and deep sense of gratitude to those who helped me in bringing out this project successfully. First of all, I would like to exhibit my thanks from my deep heart to **my parents** who have been with me and the real source of my project.

I wish to express my deep unfathomable feeling of gratitude and indebtedness **to Dr.Joseph.V.Thanikal, M.E., Ph.D., PDF, CEPIT, Principal** – Kumaraguru College of Technology, Coimbatore for the successful completion of the project work. I am very gladly taking this opportunity to express a special word of thanks to **Dr.M.Gururajan, Ph.D, Head of the Department**, Kumaraguru College of Technology, Coimbatore for encouraging me to do this work.

I am very much indebted to **Mr.A.Muthukumar, M.C.A.** and **Mrs.R.K.Kavitha, MCA., M.Phil,** Kumaraguru College of Technology, Coimbatore for his complete assistance, guidance and support given to me throughout my project. It's always a pleasure and privileges to be associated with a prestigious outstanding esteemed organization "**Infosys Technologies Ltd**", Mysore. I am very happy and grateful to be a part of **Infosys.**

My hearty thanks to my Project Mentor **Mrs.Malavika.G.V** and my **E&R guide Mr.Praveen.A.Kulkarni** of **Infosys Technologies Ltd**, for their valuable guidance throughout the project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.INTRODUCTION

1.1. ORGANIZATION PROFILE

Infosys Technologies Ltd. (NASDAQ: INFY) was started in 1981 by seven people with US$ 250. Today is a global leader in the "next generation" of IT and consulting with revenues of over US$ 3 billion.

Infosys defines designs and delivers technology-enabled business solutions that help Global 2000 companies win in a Flat World. Infosys also provides a complete range of services by leveraging our domain and business expertise and strategic alliances with leading technology providers.

Infosys' service offerings span business and technology consulting, application services, systems integration, product engineering, custom software development, maintenance, re-engineering, independent testing and validation services, IT infrastructure services and business process outsourcing.

Infosys pioneered the Global Delivery Model (GDM), which emerged as a disruptive force in the industry leading to the rise of offshore outsourcing. The GDM is based on the principle of taking work to the location where the best talent is available, where it makes the best economic sense, with the least amount of acceptable risk.

Infosys has a global footprint with offices in 23 countries and development centers in India, China, Australia, the UK, Canada and Japan. Infosys has over 80,500 employees covering 66 nationalities. Infosys takes pride in building strategic long-term client relationships. Over 95% of our revenues come from existing customers

Vision

"To be a globally respected corporation that provides best-of-breed business solutions, leveraging technology, delivered by best-in-class people."

Mission

"To achieve our objectives in an environment of fairness, honesty, and courtesy towards our clients, employees, vendors and society at large."

Values

Infosys believe that the softest pillow is a clear conscience. The values that drive them underscore their commitment to:

- **Customer Delight:** To surpass customer expectations consistently
- **Leadership by Example:** To set standards in our business and transactions and be an exemplar for the industry and ourselves
- **Integrity and Transparency:** To be ethical, sincere and open in all our transactions
- **Fairness:** To be objective and transaction-oriented, and thereby earn trust and respect
- **Pursuit of Excellence:** To strive relentlessly, constantly improve ourselves, our teams, our services and products to become the best

A Magnet for the Best Global Talent

Fortune magazine identified Infosys among the top companies that "inspire, nurture and empower a new generation of global leaders." They are committed to remain among the industry's leading employers.

Quality Focus

'In God we trust, everyone else must come with data' is an oft-heard phrase at Infosys. Infosys constantly benchmark their services and processes against globally recognized quality standards. Their certifications include SEI-CMMI Level 5, CMM Level 5, PCMM Level 5, TL 9000 and ISO 9001-2000. In February 2007, Infosys BPO was certified for eSCM level 4.0, the eSourcing Capability Model for Service Providers developed by a consortium led by Carnegie Mellon University's Information Technology Services Qualification Centre.

Innovation, Speed and Excellence in Execution

They were one of the first companies to develop and deploy a global delivery model and attain SEI-CMMI Level 5 certification their offshore and onsite operations. They manage growth by investing in infrastructure and by rapidly recruiting, training and deploying new professionals. They have 44 global development centers, the majority of which are located in India. Infosys also have development centers in Australia, Canada, China, Japan, Mauritius, and at multiple locations in the United States and Europe. They invest in infrastructure and people to continue growing our business.

Industry Leadership

Infosys history is marked by a series of firsts. They were the first Indian company to list on a US stock exchange and the first Indian company to do a POWL in Japan. In December 2006, they became the first Indian company to be added to the NASDAQ-100 index and became the only Indian company to be part of any of the major global indices. More recently, Infosys was named among the 'Top 10 Companies for Leaders' by Fortune magazine. Infosys won the prestigious Global Most Admired Knowledge Enterprises (MAKE) award for the fourth year in succession

Beyond Business

- At Infosys, they believe that they must develop trust with the communities in which they operate to achieve longevity as a corporation. Through the Infosys Foundation, which receives a grant every year from Infosys (the last year's grant was US$ 3 million) they contribute to betterment of healthcare (hospitals, infrastructure), education (books, scholarships, refurbishment of infrastructure) and skills.

- Infosys emphasizes its commitment to investors through stringent corporate governance. Infosys was also among the first Indian companies to voluntarily comply with the US Generally Accepted Accounting Principles (GAAP) and now provides financial results in the GAAP of six countries.
- With employees from over 41 nationalities, Infosys has built an enduring value system based on openness, honesty, fairness and transparency, which has earned us the confidence and trust of our clients. We enjoy a +95% customer retention.
- Infosys has built one of the largest corporate education centers in the world. This 'finishing center', with an annual capacity of 15,000, provides engineering graduates who aspire to be employees with the equivalent of a Bachelor of Science degree in Computer Science from an American university.

Learn more about our initiatives to preserve arts and culture, encourage talent, support education and healthcare, create a sustainable culture of caring for the environment, and provide impetus to tomorrow's global companies.

1.2. PROBLEM DEFINITION

In the emerging world most of the daily transactions are converted into online / internet based / web based transactions. Now a days Email and SMS communications are becoming part of the day-to-day life in most of the countries.

The Proposed system "Reusable SMS / Email Web client" is an application which is used to automatically send SMS / E-Mail to the application users, customers, and clients when any business need or event arises. This project is a re-usable component and it can be incorporated with any application which needs to communicate any information to the application users or customers automatically when any event occurs or in any business need.

2. SYSTEM ANALYSIS

2.1. EXISTING SYSTEM ARCHITECTURE

Now-a-days, most of the existing systems, whenever the application wants to communicate any information to the application users or customers, an E-mail or SMS need to be manually prepared and send it to them through separate E-mail application or through Mobile Phone.

Problems in existing system

- In the existing systems, for a business need, it requires lots of manual effort to prepare the e-Mail / SMS and send them to the users or customers. This requires more time and effort to prepare, format the E-mail or SMS.
- In addition to that, it requires separate applications to send E-mail and SMS. If same information needs to be sent to multiple users with minimum customization, it requires tremendous effort when the number of users is more.

2.2. PROPOSED SYSTEM ARCHITECTURE

- The proposed system will be an integral component of the System, and the application will automatically generate / format the E-mail / SMS and send to customer whenever the system needs to communicate any information.
- The System will call the proposed application when any business information needs to be sent to application user. The application will receive the required information from the system. The SMS / Email Web Client, formats the Email or SMS and contacts the SMTP or SMS server and sends the information to the respective application user or customer.

2.3. USER INTERFACE REQUIREMENTS

Scope

The scope of this project limited to building of SMS / Email formatting and sending application. This re-usable component needs to be developed with a common interface. The common interface will be used by any system to communicate to the component to send E-mail or SMS.

This project scope includes customizing the SMS / E-mail application for a specific business or application. Also, the system needs to be enhanced to communicate to SMS / E-Mail web client by the standard interface. This project will not involve in development of any other part of the existing system.

Objective

The proposed application will be able to receive the recipient's information. Either Mobile phone number or an E-Mail address will be received from the System. The application will examine the recipient's information and identify weather it is Mobile phone number or E-Mail address.

The application will receive the name of the recipient or customer from the system. The application formats the information received based on the type of communication to the recipient / customer.

If mobile phone number is received, then the received information will be formatted as SMS. If the E-Mail address is received, then the received information will be formatted as E-Mail.

2.4. FUNCTIONAL REQUIREMENTS

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Non-functional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

Send Email

Send Email is an important aspect of Reusable SMS /Email web client. In this the user can send Email to single or multiple persons according to the user need. If the user wants to send email to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

Send SMS

Send SMS is another important aspect of Reusable SMS /Email web client. In this the user can send SMS to single or multiple persons according to the user need. If the user wants to send SMS to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

View Report

View Report is another aspect of Reusable SMS /Email web client. In this the user can view the SMS sent report or email sent report by specifying from and to date. If the user wants to view SMS report, then the user needs to select SMS report check box, and specify the date. If the user wants to view email report, then the user needs to select email report check box, and specify the date. Then it will display the number of emails sent or SMS sent.

2.5. SYSTEM FEATURES

Requirement #1 - Login

Purpose	Allows user to type username and password.
Inputs	User has to type the username and password to login into the application.
Processing	The system authenticates the username and password.
Outputs	The system will allow the user to select a choice.

Requirement #2 - Selection

Purpose	Allows user to select an option
Inputs	Select email, SMS or report from the choice provided.
Processing	Once the user selects the option, it will display the corresponding page to let the user to provide input.
Outputs	It should display the corresponding page to let the user to provide input.

Requirement #3 - send Email

Purpose	Allows user to provide email-id (s) and message as input.
Inputs	Provide email-id and message to be sent.
Processing	When the user finish providing input and press the send button, if valid, mail will be sent to the Recipient.
Outputs	Mail should be sent to the Recipient.

Requirement #4 - Send SMS

Purpose	Allows user to provide mobile number (s) and message as input.
Inputs	Provide Mobile number and message to be sent.
Processing	When the user finish providing input and press the send button, if valid, mail will be sent to the Recipient.
Outputs	Message should be sent to the Recipient.

Requirement #5 - View Email Report

Purpose	Allows user to select from and to date.
Inputs	User will select email check box to view Email report.
Processing	The system will connect to the database and provide the number of Emails sent within the specified date.
Outputs	User can be able to see the number of Emails sent.

Requirement #6 - View SMS Report

Purpose	Allows user to select from and to date.
Inputs	User will select SMS check box to view SMS report.
Processing	The system will connect to the database and provide the number of SMS sent within the specified date.
Outputs	User can be able to see the number of SMS sent.

Requirement #7 - Logout

Purpose	Allows user to safely logout of the Application.
Inputs	User has to select the logout button.
Processing	The system will end the session.
Outputs	The system will allow the user to logout.

Requirement #8 - Exit

Purpose	Allows user to exit from the Application.
Inputs	User has to select the exit button.
Processing	The system will end the session.
Outputs	The system will allow the user to exit from the Application.

2.6. NON FUNCTIONAL REQUIREMENTS

Performance

- Response time should be stable for any number records in the XLS File.
- The component should be able to retrieve data XLS File.
- Retrieval of data from databases or files should be faster.

Scalability

- The Component should work with large input data set or large number of records.
- The Component should be easily pluggable with other projects.

Error Logging

- If the user has selected some inappropriate operation, then the system should handle the error and display the appropriate error message.

Availability/Reliability

- The system should be ready for use at any specified time.
- The system or component should perform its required functions under stated conditions for a specified period of time.

3. DEVELOPMENT ENVIRONMENT

3.1. H/W ENVIRONMENT

The system should have following hardware requirements to execute the current application and the new application.

Application Server: An application to execute the current system and the new developed application.

Mail Server : A mail server required to fulfill the mail sending purpose of the new application.

Processor	: Pentium III 1.4 GHz or above
RAM	: 512 MB or above
Monitor	: 14" Wide
Keyboard	: 106 keys standard QWERTY Keyboard
Mouse	: Optical Scroll Mouse

This system works on the Windows XP environment.

3.2. S/W ENVIRONMENT

The system should have following software requirements to execute the current application and the new application.

Operating system	: Windows XP or Windows 2000 or Windows 2003 Server
Environment	: Java Enterprise edition Server
Programming Language	: Java Programming Language
Mail Server	: SMTP Server
SMS Server	: SMS Server compatible with GSM Mobile Phones
Development environment	: Integrated Development Environment (IDE) for Java programming.
Data base	: Any Relational Database (Oracle, DB2 etc)

3.3. NETWORK ENVIRONMENT

The new application should have connectivity to the following networks to send Emails and SMS Messages to the application users and customers.

1. WWW Internet connectivity with SMTP server
2. GSM Mobile network connectivity with SMS Server

3.4. SOFTWARE OVERVIEW

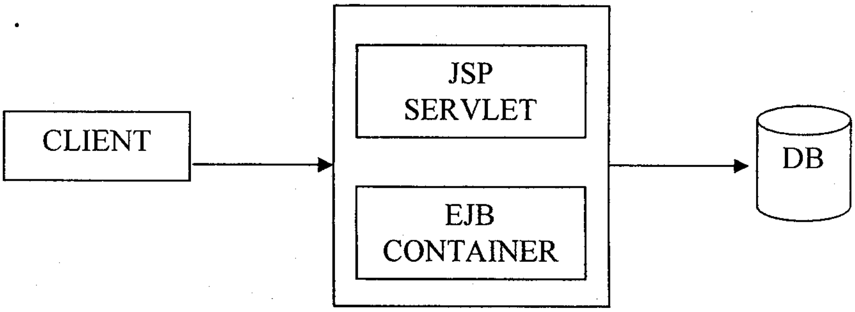
3.4.1. J2EE AND ITS FEATURES

Information technology departments had always sought ways to create cost- effective computer applications. One approach is client/server architecture, which uses a two-tier where client side software requests services from server side software. Common gateway Interface (CGI) technology was a solution that was adopted by many corporations. CGI technology consisted of a program that was callable by a browser whenever the appropriate hyperlink or submit action from a web form occurred.

CGI technology addressed the problem of interfacing web clients with the corporate infrastructure. But this technology was resource intensive and not scalable to meet the dramatic increase in the number of clients who needed to access corporate resources. Thus J2EE came in to existence.

Client/ Server architecture exploded from two-tier architecture to a multi-client architecture, where a client requests data to a server which generates request to other servers that are connected to a backbone network. Sun Microsystems has tried to define the function of a java application server clearly that all implementations play on the level field.

J2EE supports multi-tier architecture rather different from a standalone monolithic application. Applications that run in their servers typically have separate parts of clients, business logic and database. A tier is an abstract concept that defines a group of technologies that provide one or more services to its clients. A client is concerned about sending a request for service and receiving results from server. The middle tier is the J2EE server that does the calculations and processing. The third tier is the database server.



J2EE ARCHITECTURE

Fig 3.1

The applications are designed to handle thousands of users simultaneously 24 hours a day, 7 days a week without any downtime. J2EE is a versatile technology because application components built using J2EE are able to communicate with each other behind the scenes using standard communications methods such as HTTP, SSL, HTML, RMI, and IIOP. Java Beans, Java Servlets and Java Server Pages (JSP) are core components of J2EE.

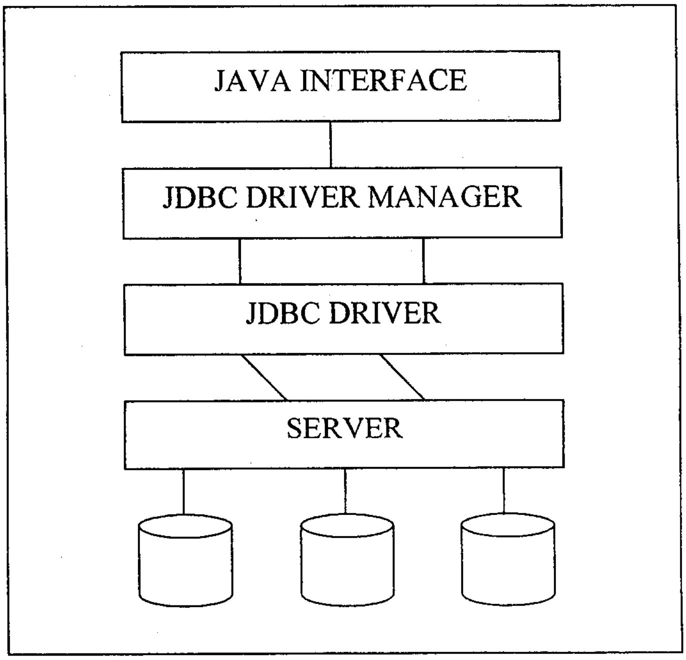
3.4.2. JAVA SERVER PAGES (JSP)

JSP makes the pages easier to edit with standard HTML authoring tools. JSP also enables java expressions and code to be intermixed with HTML tags and text. JSP is a server-side program that is similar in design and functionality to a java servlet. A JSP is a HTML page embedded with servlet code that is surrounded by <% and %> tags. The servlet code sometimes called Scriplet which uses implicit variables:

- request: the servlet request
- response: the servlet response
- out: the output writer
- in: the input reader

3.4.3. JAVA DATABASE CONNECTIVITY (JDBC)

JDBC is Sun Microsystems standard SQL database access interface providing uniform access to a wide range of relational database. It consists of a set of classes and interfaces in the java programming language.



JAVA DATABASE CONNECTIVITY

Fig 3.2

Establishing a connection involves two steps:

1. Loading the driver
2. Making the connection

Loading Driver and Making the Connection

A one-line code for JDBC-ODBC bridge driver is used for loading the driver. To connect to the DBMS an appropriate driver is used with the help of code. JDBC driver manager plays a key role in facilitating connection with the required Database by loading the appropriate driver. Java coding is written to accomplish these tasks.

4. SYSTEM DESIGN

4.1. PROCESS MODEL

4.1.1. CONTEXT DIAGRAM

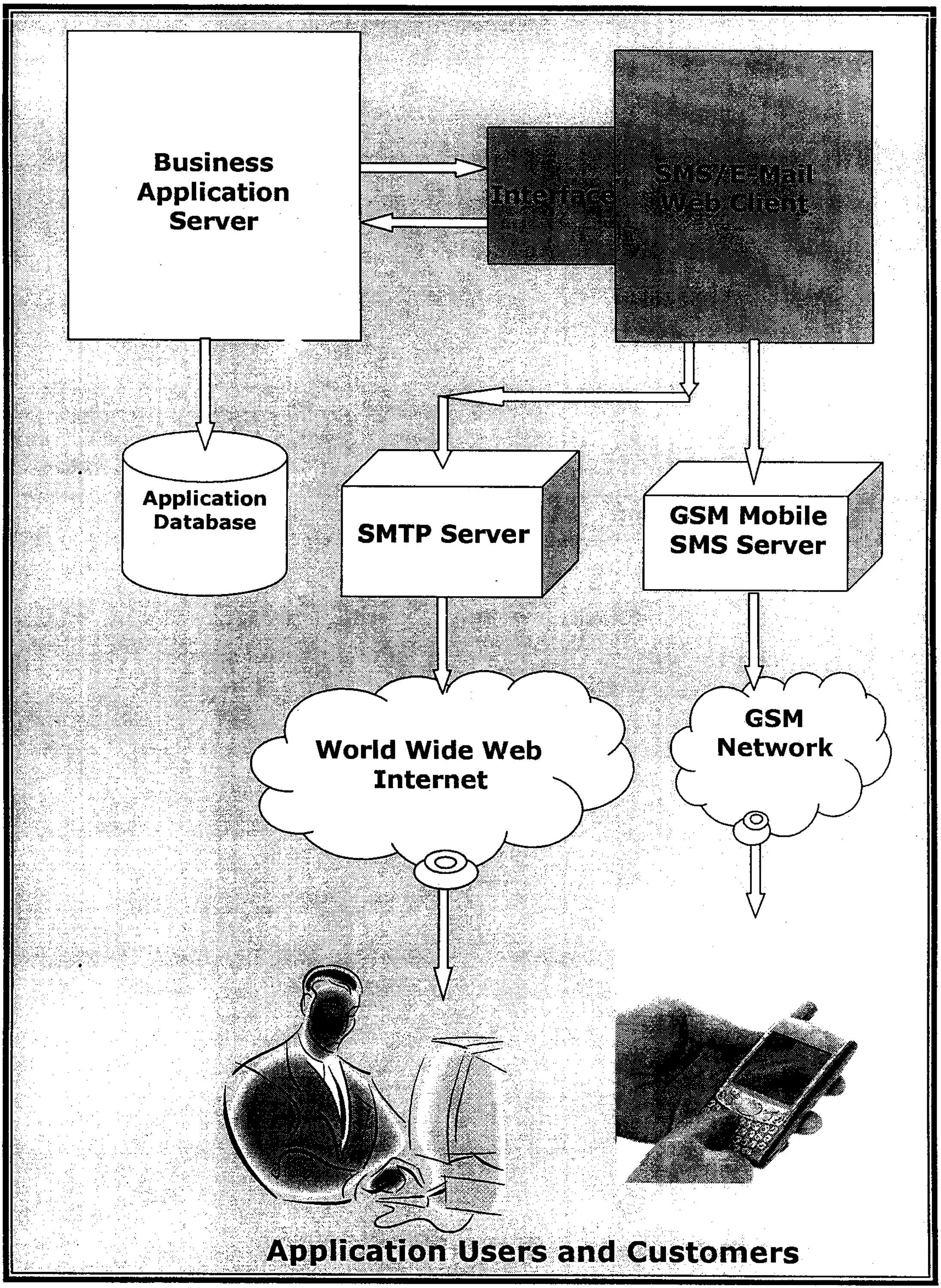
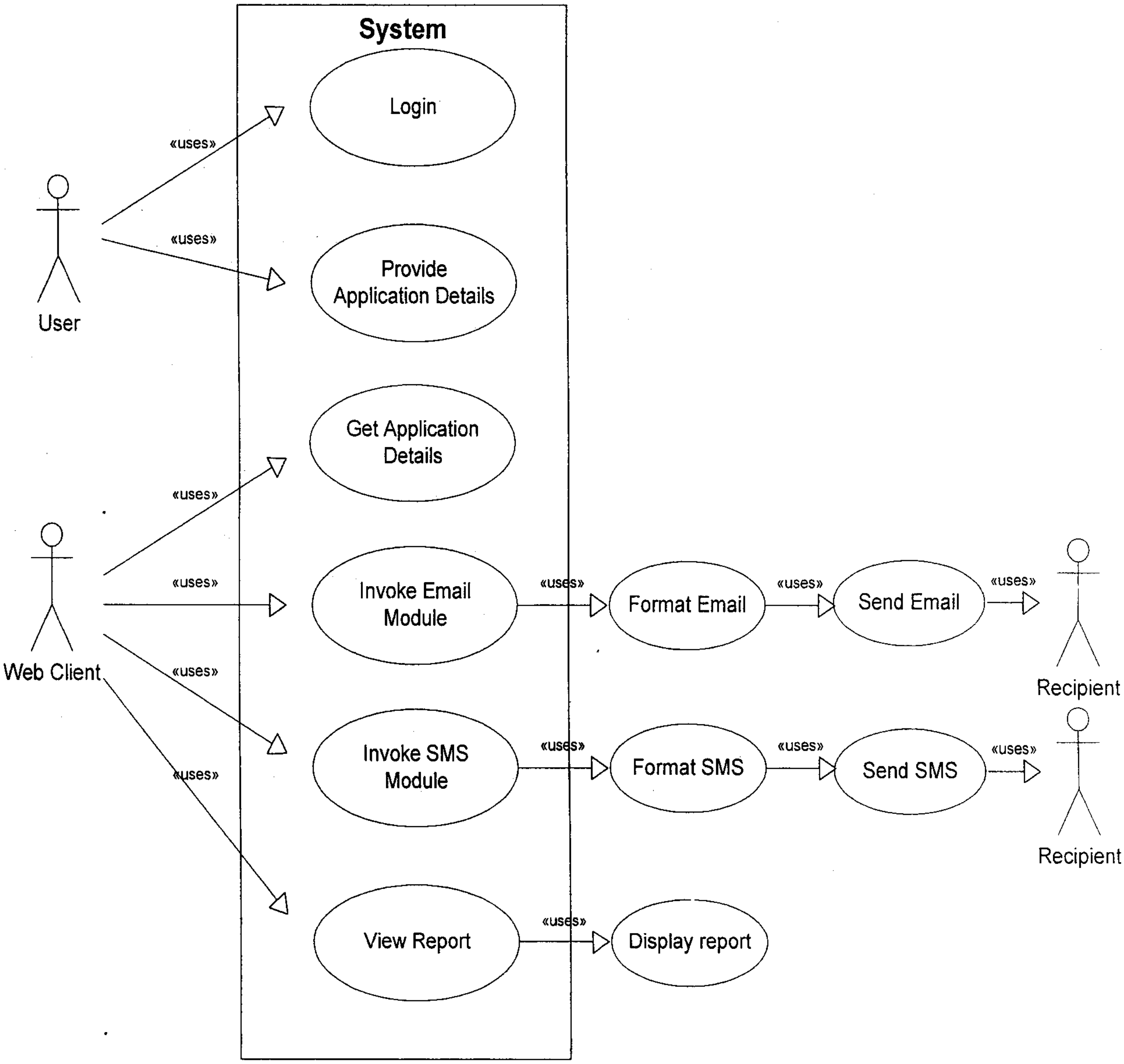


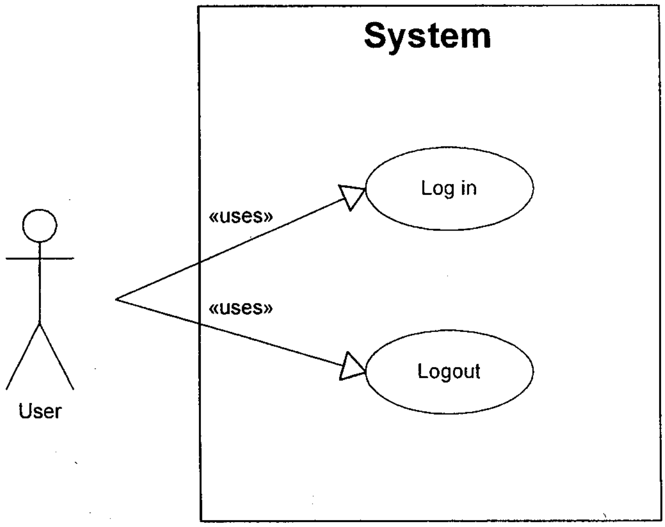
FIG 4.1

4.1.2. USE CASE DIAGRAM

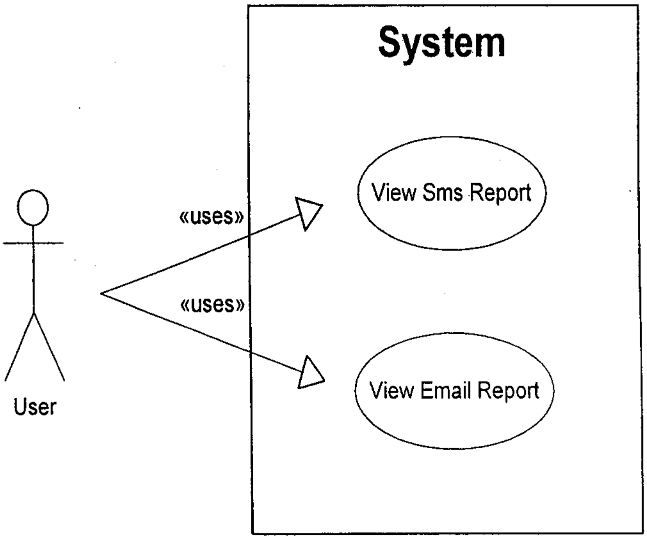
a. Use-Case Diagram: Reusable SMS/Email Web Client



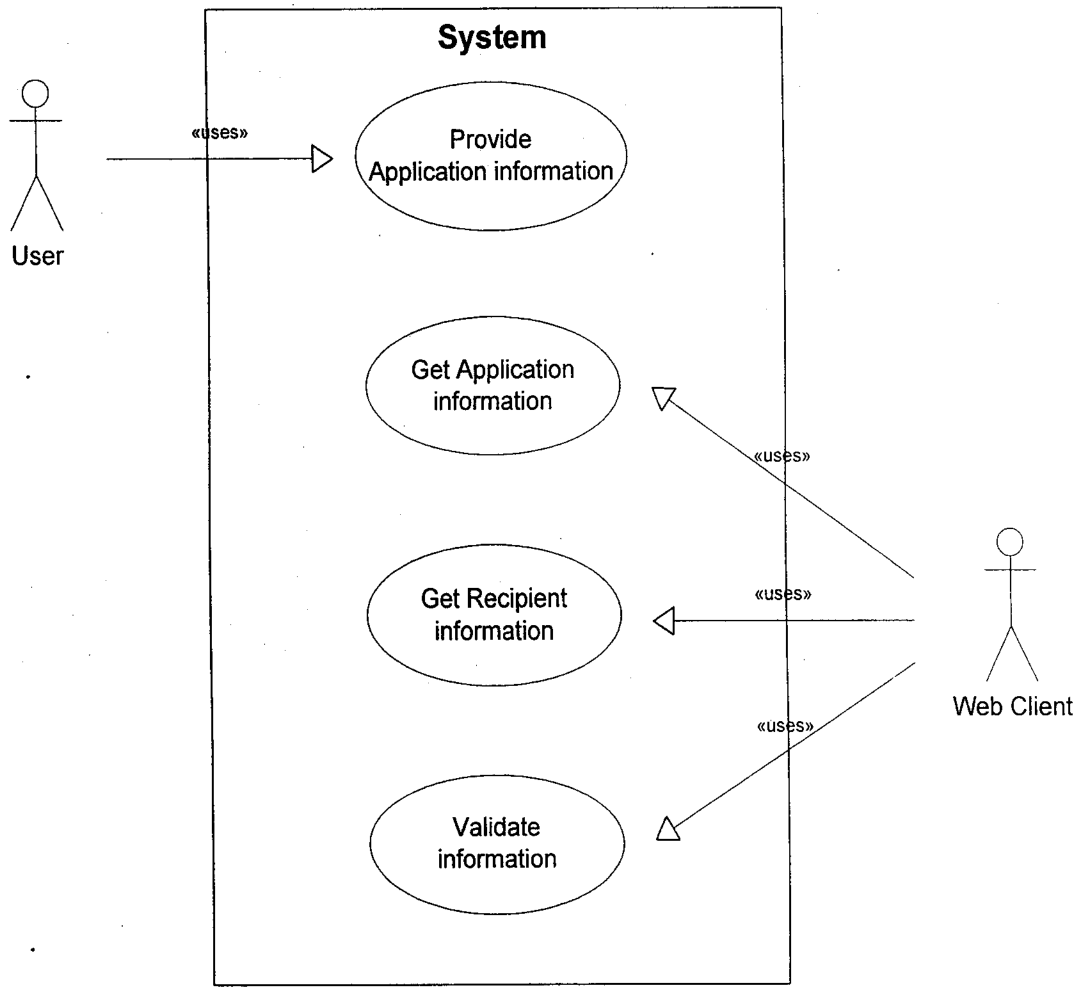
b. Use-Case Diagram: Login



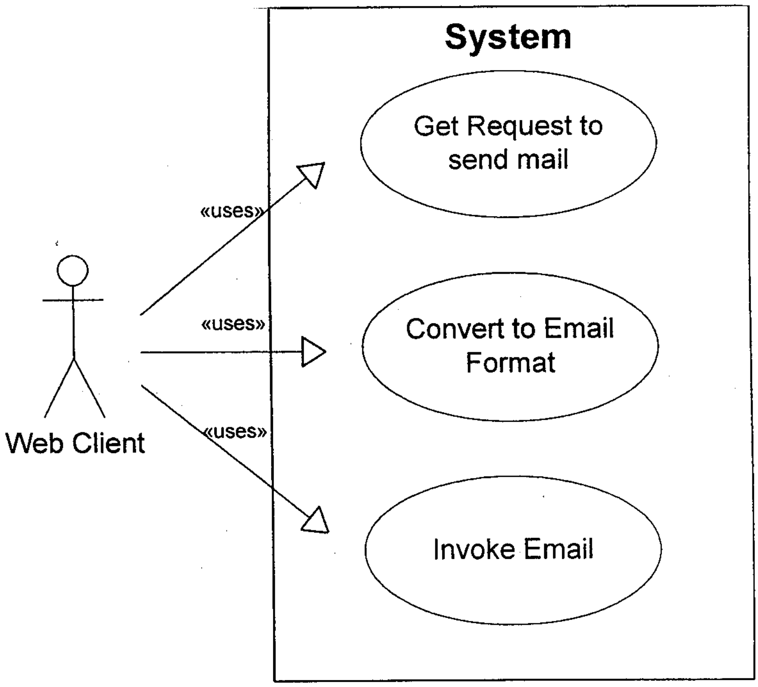
c. Use-Case Diagram: View Report



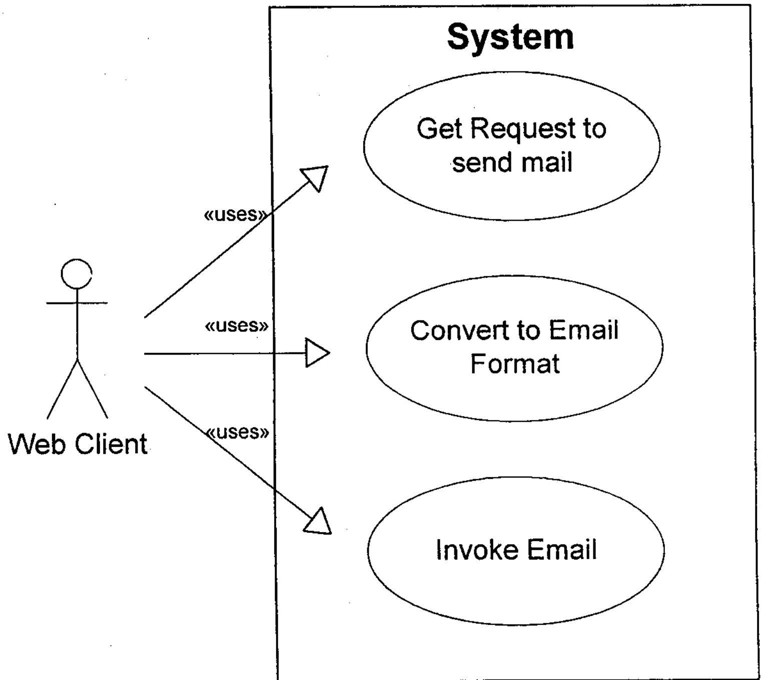
d. Use-Case Diagram: Provide Application information


System
«uses»
Provide Application information
User
Get Application information
«uses»
Get Recipient information
«uses»
Web Client
«uses»
Validate information

e. Use-Case Diagram: Send Email



f. Use-Case Diagram: Send SMS



4.1.3. USE-CASE DESCRIPTIONS

a. Description#1: Main Use Case

Use Case ID:	1
Use case Name:	Main Use case
Actors:	User, Web client.
Description:	The User will request the application to send an email, SMS or view report by providing the application and sender details. The application will validate and send an Email or SMS to the recipient.
Preconditions:	The Application should provide a valid recipient type.
Post conditions:	An Email or SMS will be sent to the Recipient.
Normal Flow:	1. User will provide the application and sender details. 2. Application will validate and call CheckRecipientType () method. 3. Message will be formatted according to the recipient type. 4. Application will send an Email or SMS to the recipient. 5. User can view report. 6. System exits use case.
Alternative flow:	Application will provide customized error messages.

b. Description#2: Login

Use Case ID:	2
Use case Name:	Login
Actors:	User.
Description:	The User will type the username and password to login into the system.
Preconditions:	The username and password should be existing.
Post conditions:	Display the options to the user.
Normal Flow:	1. The user can view the options and select one of the options to send Email, SMS or report. 2. The web client will display the corresponding page.
Alternative flow:	Application will provide customized error messages.

c. Description#3: View Report for Email

Use Case iD:	**3**
'Jse case Name:	Report
Actors:	User.
Description:	The User will select Email check boxes to view the corresponding report.
Preconditions:	The date provided should be valid.
Post conditions:	Display the details to the user.
Normal Flow:	Display the number of Email sent, to the user.
Alternative flow:	Application will provide customized error messages.

d. Description#4: View Report for SMS

Use Case ID:	**4**
Use case Name:	Report
Actors:	User.
Description:	The User will select SMS check boxes to view the corresponding report.
Preconditions:	The date provided should be valid.
Post conditions:	Display the details to the user.
Normal Flow:	3. Display the number of SMS sent, to the user.
Alternative flow:	Application will provide customized error messages.

e. Description#5: Send Email

Use Case ID:	5
Use case Name:	Sending Email
Actors:	Application.
Description:	The application will call the email sender method to send a email to the recipient.
Preconditions:	The CheckRecipientType () method should have return the value 1
Post conditions:	The application should call the Email Sender method.
Normal Flow:	1. Get the message to be formatted. 2. Format the message according to the SMTP server format 3. Call sender method to send email.
Alternative flow:	Application will provide customized error messages.

f. Description#6: Send SMS

Use Case ID:	6
Use case Name:	Sending SMS
Actors:	Application.
Description:	The application will call the SMS sender method to send a SMS to the recipient.
Preconditions:	The CheckRecipientType () method should have return the value 2
Post conditions:	The application should call the SMS Sender method.
Normal Flow:	1. Get the message to be formatted. 2. Format the message according to the GSM server format. 3. Call Sender method to send SMS.
Alternative flow:	Application will provide customized error messages.

4.2. INPUT DESIGN

Input design is the process of converting user originated inputs to a computer based format. input design is the part of the system design if it is incorrect then the processing and output will magnify the errors. Inaccurate input data is the most common cause of errors in data processing. The main objective of designing input focus on

- Ensuring the accuracy of the input data.
- Controlling the amount of input required.
- Avoid delayed response.
- Keeping process simple.
- Avoid errors.

The required inputs are stored in the form of tables. They may be numeric or alphanumeric input screen should be user friendly, so that every one can access the options without having the complete system knowledge.

4.3. OUTPUT DESIGN

The output must be provided in such a format that the user can understand. After analyzing the operations of the system, output information required for each jobs are determined. In addition to this, these outputs may be in format suitable to input for subsequent processing.

Output design refers to the results generated by the system. The output of a system can take many forms. The most common forms are reports, screen displays, printed form and graphical drawing forms.

The normal procedure in developing a system design is to design the output in detail first and then move back to the input.

4.4. DATA FLOW DIAGRAM

The data flow diagram is a graphical tool for requirement analysis. It depicts the information flow without any explicit representation. It is used to represent the system or software at any level of abstraction.

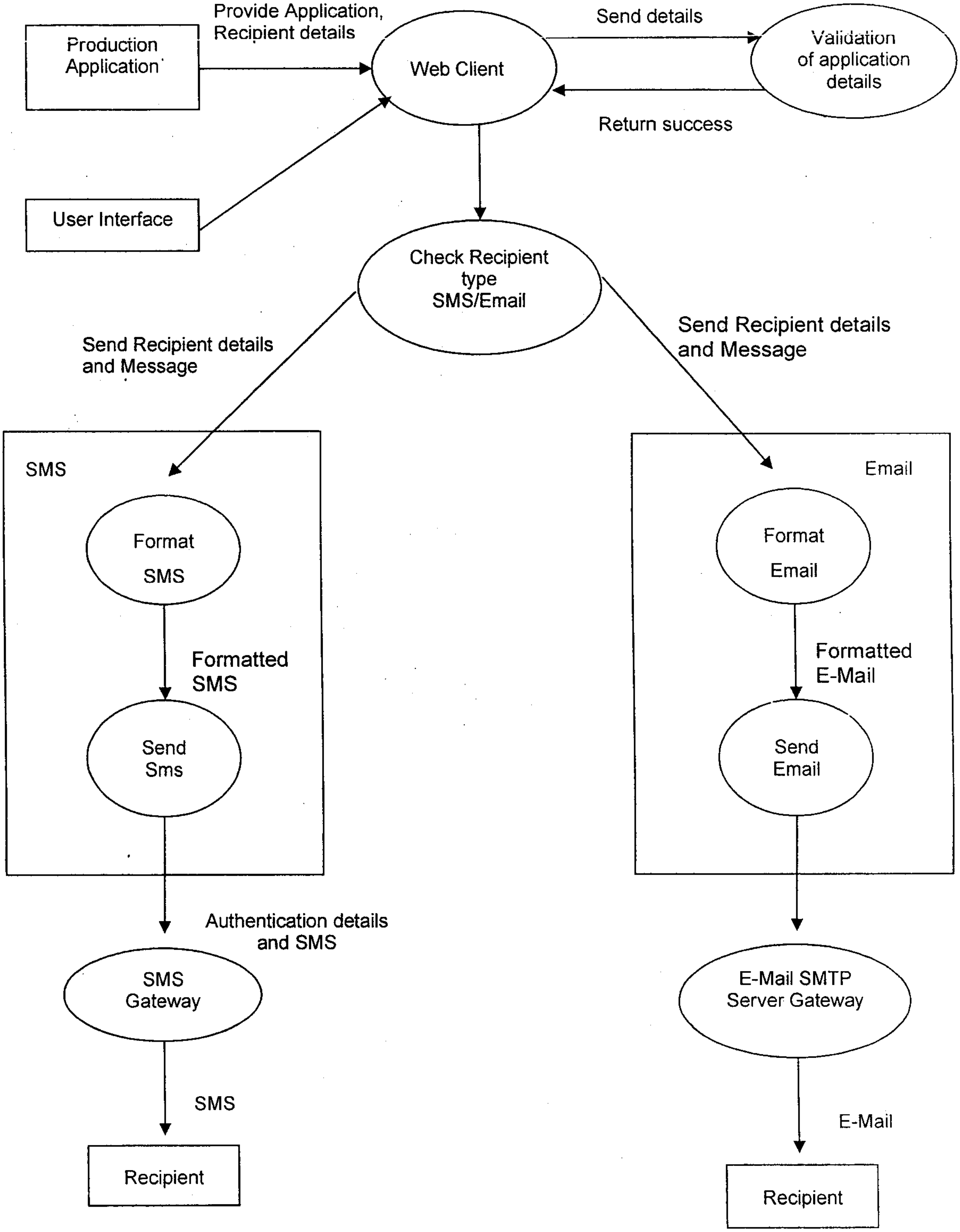

Production Application
Provide Application, Recipient details
Web Client
Send details
Validation of application details
Return success
User Interface
Check Recipient type SMS/Email
Send Recipient details and Message
Send Recipient details and Message
SMS
Format SMS
Formatted SMS
Send Sms
Email
Format Email
Formatted E-Mail
Send Email
Authentication details and SMS
SMS Gateway
E-Mail SMTP Server Gateway
SMS
E-Mail
Recipient
Recipient

Fig 4.2

4.5. DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. The design of the database is one of the most critical parts of design phase.

The details about the data relevant for the system are identified first. According to their relationship, tables are designed by following the standard database design methods. The data types for each data item in the tables are decided.

Normalization

The normalization simplifies the entities, removes the redundancies from the system data and finally builds a data structure, which is both flexible and adaptable to the system. It provides a systematic step-by-step approach towards this goal.

The normal forms applied is given below,

- First normal form
- Second normal form

The database is designed using the RDBMS concept there by enabling the sharing of data and was normalized to avoid redundancy. This leads to quicker application development with low maintenance cost.

a. **Table name: SMS**

Field Name	**Data Type**	**Length**	**Key**	**Description**
SNo	Number	3	P	SMS number
MobileNumber	Varchar2	10		Mobile number
Message	Varchar2	160		Message
Priority	Number	1		Priority of the message

b. **Table name: Login**

Field Name	**Data Type**	**Length**	**Key**	**Description**
Username	Varchar2	15	P	User name
Password	Varchar2	8		Password

c. **Table name: SMSReport**

Field Name	**Data Type**	**Length**	**Key**	**Description**
MobileNumber	Varchar2	10		Mobile number
SentDate	Date	8		Sent Date
Username	Varchar2	15	F	User name

d. **Table name: EmailReport**

Field Name	**Data Type**	**Length**	**Key**	**Description**
EmailId	Varchar2	50		Email Id
SentDate	Date	8		Sent date
Username	Varchar2	15	F	User name

5. ARCHITECTURAL DETAILS

5.1. PROGRAM DESIGN LANGUAGE

Pseudo Code for E-mail :

Email module

Email module is an important aspect of Reusable SMS /Email web client. In this the user can send Email to single or multiple persons according to the user need. If the user wants to send email to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

```
Public Boolean fnSendEmail(String ipAddress) {
Try {

        Store the ipaddress in a temporary variable

        Pass info to the mail server as Properties

        Define the local SMTP server as "host"

        Create the Session object

        Set debug to true

        Create a mime message object for the session

        IF From email id valid

                Set from email id to the message object
        IF multiple To email id
                Then separate the email ids
                IF To email id valid
                        Set To email id to the addRecipient method
                ELSE
                        Display error message

        IF multiple CC email id
                Then separate the email ids
                IF CC email id valid
                        Set CC email id to the addRecipient method
                ELSE
                        Display error message

        IF multiple BCC email id
                Then separate the email ids
                IF BCC email id valid
                        Set BCC email id to the addRecipient method
                ELSE
                        Display error message

        Add subject and message to be sent to the message object

        Pass the message object to the send method of Transport class

        Return true
    ELSE
        Return false
  }  Catch all the exceptions

}
```

Pseudo Code for SMS:

SMS Module

SMS module is another important aspect of Reusable SMS /Email web client. In this the user can send SMS to single or multiple persons according to the user need. If the user wants to send SMS to multiple persons, then the user needs to provide input from an excel file, or the user can enter directly in the provided field.

```
Public Boolean fnSendSMS (String mobileNumber, String message) {
Try {
        IF mobileNumber == "" OR message == ""){
                Display appropriate error message
                Return false;
        ELSE
        //      Get the SMS server address

                Set SMSHost to "10.66.110.138:8080/"

                Create an instance of HttpClient.

                Set URLEncoder.encode message type to "UTF-8"

                Create the url string

                Create the uri object from the url string. This method inserts escape
                characters where ever necessary

                Create a method instance with getEscapedURI.

                DefaultHttpMethodRetryHandler is created.
```

The first argument of the DefaultHttpMethodRetryHandler method indicates the number of retries and second indicates not to retry once message is successfully sent.

HttpClient is instructed to retry the method thrice even though the request may have already been processed and an exception occurred by setting the parameter RETRY_HANDLER.

```
Execute the method using Execute method and assign the status.
        IF status ! = HttpStatus.SC_OK)
                Display error message

        ELSE
                Set the status to true for successful request

}       Catch all the exceptions

Finally Release the connection.

        IF method != null
                Release Connection
        Return status;
}
```

Pseudo code for Report:

View Report Module

View Report module is used by the user to view the SMS sent report or email sent report by specifying from and to date. If the user wants to view SMS report, then the user needs to select SMS report check box, and specify the date. If the user wants to view email report, then the user needs to select email report check box, and specify the date. Then it will display the number of emails sent or SMS sent.

```
Public :nt report (Calendar dateFrom, Calendar dateTo, String table) throws
NullPointerExceptlon, SQLException, ClassNotFoundException, ParseException
{
Try {
        Establishing connection with database

        Create an instance for SimpleDateFormat

        Call create statement method

        Execute query and store the result in resultSet

        WHILE resultSet not null
                Assign result to count

        Close the statement
    }   Finally

        Close the Connection

        Return count
}
```

```
Public void insertEmailId (String from) throws ClassNotFoundException,
SQLException {
Try {

        Establishing connection with database

        Call create statement method

        Execute update query

   } Finally
                Close the connection
}


Public void insertMobileNumber (String mobileNumber) throws
ClassNotFoundException, SQLException {
Try {

        Establishing connection with database

        Call create statement method

        Execute update query

    } Finally
                Close the connection
}
```

6. TESTING

6.1. TEST CASE REPORTS

The objective of testing is a process of executing a program with the intent of finding the errors. Testing is an individualistic process and the number of different development approaches.

There are three major types of testing they are,

- ➢ Unit testing
- ➢ Integration testing
- ➢ Validation Testing

6.1.1. UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software design that is the module. The unit testing is also called white box testing. The module interface is tested to ensure that the information properly flows into and out of the program unit test.

Test Execution Procedure

Send Email

1. Program Name : Send Email
2. Functionality Tested : Sending email to recipient.
3. Test Execution Procedure :

    - Test cases are prepared according to design / specifications.
    - Each Test case and the items under test are documented in detail in a set format.
    - The input and the initial values are then filled in the Unit Test Plan (UTP) as per each test case & theoretical values evaluated.
    - The UTP is then reviewed and executed.

4. Test Results Checking Method : Not Applicable.

5. Test Data :

1	Email_From_Input_Empty	No input is given.	Email Id should be existing	Display as “From Email Id should not be empty”.	fnSendEmail ()	Passed
2	Email_From_Valid_ Input	Give some valid input.	Email Id should be existing	It should send an email to the recipient.	fnSendEmail ()	Passed
3	Email_From_Invalid _Input	Give some invalid input.	Email Id should be existing	Display as “Enter a valid Email Id”.	fnSendEmail ()	Passed
4	Email_Id_ Input_ Empty	No input is given.	Email Id should be existing	Display as “Enter at least one email Id”	fnSendEmail ()	Passed
5	Email_To_ Valid_ Input	Give some valid input.	Email Id should be existing	It should send an email to the recipient.	fnSendEmail ()	Passed
6	Email_To_ Invalid _Input	Give some invalid input.	Email Id should be existing	Display as “Enter a valid Email Id”.	fnSendEmail ()	Passed
7	Email_Cc_ Valid _Input	Give some valid input.	Email Id should be existing	It should send an email to the recipient.	fnSendEmail ()	Passed
8	Email_Cc_ Invalid _Input	Give some invalid input.	Email Id should be existing	Display as “Enter a valid Email Id”	fnSendEmail()	Passed
9	Email_Bcc_ Valid _Input	Give some valid input.	Email Id should be existing	It should send an email to the recipient.	fnSendEmail ()	Passed
0	Email_Bcc_ Invalid _Input	Give some invalid input.	Email Id should be existing	Display as “Enter a valid Email Id”	fnSendEmail ()	Passed
1	Email_Message _ Empty_Input	No input is given.	None	Display as “Message should not be empty”.	fnSendEmail ()	Passed
2	Email_Message _ Valid_Input	Give some valid input.	None	It should send an email to the recipient.	fnSendEmail ()	Passed

Table 6.1

Send SMS

1. Program Name : Send SMS
2. Functionality Tested : Sending SMS to Recipient.
3. Test Execution Procedure :
    - Test cases are prepared according to design / specifications.
    - Each Test case and the items under test are documented in detail in a set format.
    - The input and the initial values are then filled in the Unit Test Plan (UTP) as per each test case & theoretical values evaluated.
    - The UTP is then reviewed and executed.
4. Test Results Checking Method : Not Applicable.
5. Test Data :

[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
1	SMS_MobileNo_ Empty_Input	No input is given.	Mobile number should be existing	Display as "Mobile number should not be empty".	fnSendSms ()	Passed
2	SMS_MobileNo_ Valid_Input	Give some valid input.	Mobile number should be existing	Should send an SMS to the recipient.	fnSendSms ()	Passed
3	SMS_MobileNo_ Invalid_Input	Give some invalid input.	Mobile number should be existing	Display as "Enter a valid Mobile number".	fnSendSms ()	Passed
4	SMS_Message_ Empty_Input	No input is given.	None	Display as "Message should not be empty".	fnSendSms ()	Passed
5	SMS_Message_ Valid_ Input	Give some valid input.	None	Should send an SMS to the recipient.	fnSendSms ()	Passed

Table 6.2

6.1.2. INTEGRATION TESTING

The Integration Test Plan entails the development of a document which instructs a tester what tests to perform in order to integrate and test already existing individual code modules.

The objectives of the Integration Test Plan activity are:

- To develop a plan which specifies the necessary steps needed to integrate individual modules and test the integration.
- Integration testing involves test cases designed to validate that end-to-end processes are properly working together. It's a logical extension of unit testing and is performed to establish whether the components interact with each other according to the specification.
- Integration testing is accomplished through the execution of predefined business flows, or scenarios, that emulate how the system will run the business.

Purpose and Scope of Integration Test:

Integration is a systematic approach to build the complete software structure specified in the design from unit-tested modules. While doing integration, tests are conducted to find defects associated with interfacing. Integration testing validates the interfaces within the application i.e., among different modules being integrated. The external interfaces of the modules with other systems in the business environment are tested.

The integration plan describes the sequence of integration, overhead software test environment and resources required. The integration environment include: Physical characteristics of the hardware, Communications software, System software, Mode of usage (e.g., stand-alone) and any other software or supplies to support integration. If necessary, the level of security that must be provided for the test facilities, system software, and proprietary components such as software, data, and hardware are also to be specified. If there are common procedures to be adopted across test cases to check results (for instance, comparing two output files), they have to be identified.

Integration Test Cases

Test Scenario 1 :

SCENARIO	1
DESCRIPTION ·	Test for Login
EXPECTED RESULTS	Should Display options
TEST DATA REFERENCE	None
STATUS	Pass
DATE	22-Apr-2008
TESTED BY	Karthiga Chinnachamy

Steps :

#	BUSINESS PROCESS STEPS	INPUT DATA	INTERMEDIATE OUTPUT	EXPECTED OUTPUT	STATUS (PASS/FAIL)	COMMENTS
1.	Enter username in the username field	400562	Accept Input	Display options	Pass	Valid username
2.	Enter username in the username field	ads	Accept Input	Display Error Message	Pass	Display Message
3.	Enter password in password field	“Se12!#”	Accept Input	Display options	Pass	Valid Password
4.	Enter password in password field	“123”	Accept Input	Display Error Message	Pass	Display Message
5.	Enter Null value	null	Accept Input	Display Error Message	Pass	Display Message

Table 6.3

Test Scenario 2 :

SCENARIO	2
DESCRIPTION	Test for sending Email
EXPECTED RESULTS	Should send Email
TEST DATA REFERENCE	None
STATUS	Pass
DATE	22-Apr-2008
TESTED BY	Karthiga Chinnachamy

Steps :

#	BUSINESS PROCESS STEPS	INPUT DATA	INTERMEDIATE OUTPUT	EXPECTED OUTPUT	STATUS (PASS/FAIL)	COMMENTS
1.	Enter email-id in from field	From an Xml file	Accept Input	Email sent	Pass	Valid Email-id
2.	Enter email-id in to field	400562@infosys.com	Accept Input	Email sent	Pass	Valid Email-id
3.	Enter email-id in cc field	abcd@yahoo.com	Accept Input	Email sent	Pass	Valid Insertion
4.	Enter email-id in bcc field	karthiga@infosys.com	Accept Input	Email sent	Pass	Valid Insertion
5.	Enter email-id in to field	123	Accept Input	Display Error Message	Pass	Display Message
6.	Enter email-id in to field	abcd@yahoo	Accept Input	Display Error Message	Pass	Display Message
7.	Enter email-id in to field	abcd$@yahoo.com	Accept Input	Display Error Message	Pass	Display Message
8.	Enter a message in message field	"Hi, Hello?"	Accept Input	Email sent	Pass	Valid Message
9.	Enter null value in message field	Null	Accept Input	Display Error Message	Pass	Display Message
10.	Enter Null value	Null	Accept Input	Display Error Message	Pass	Display Message

Table 6.4

Test Scenario 3 :

SCENARIO	3
DESCRIPTION	Test for Sending SMS
EXPECTED RESULTS	Should send SMS
TEST DATA REFERENCE	None
STATUS	Pass
DATE	22-Apr-2008
TESTED BY	Karthiga Chinnachamy

Steps :

#	BUSINESS PROCESS STEPS	INPUT DATA	INTERMEDIATE OUTPUT	EXPECTED OUTPUT	STATUS (PASS/FAIL)	COMMENTS
1.	Enter Mobile number in field	9986543210	Accept Input	Message Sent	Pass	Valid Mobile number.
2.	Enter Mobile number in field	'ABC'	Accept Input	Display Error Message	Pass	Display Message.
3.	Enter Mobile number in field	123$%^&)	Accept Input	Display Error Message	Pass	Display Message.
4.	Enter a message in message field	"Hi, Hello?"	Accept Input	Message Sent	Pass	Valid Message.
5.	Enter null value in message field	Null	Accept Input	Display Error Message	Pass	Display Message.
6.	Enter Null value	Null	Accept Input	Display Error Message	Pass	Display Message.

Table 6.5

Test Scenario 4 :

SCENARIO	4
DESCRIPTION	Test for view Report
EXPECTED RESULTS	Should display number of records
TEST DATA REFERENCE	None
STATUS	Pass
DATE	28-Apr-2008
TESTED BY	Karthiga Chinnachamy

Steps :

#	BUSINESS PROCESS STEPS	INPUT DATA	INTERMEDIATE OUTPUT	EXPECTED OUTPUT	STATUS (PASS/FAIL)	COMMENTS
1.	Select email check box	From and to date	Accept Input	Display number of emails sent	Pass	Display number of emails sent
2.	Select SMS check box	From and to date	Accept Input	Display number of SMS sent	Pass	Display number of SMS sent
3.	Select email, SMS check box	From and to date	Accept Input	Display number of SMS, emails sent	Pass	Display number of emails sent
4.	Select none	Null	Accept Input	Display Error Message	Pass	Display Message.
5.	Select email or SMS check box	Select invalid date	Accept Input	Display Error Message	Pass	Display Message.

Table 6.6

6.1.3. VALIDATION TESTING

At the end of the integration testing, software is completely assembled as a package, interfacing errors have been uncovered and correction testing begins. Validation refers to a different set of activities to ensure that software that has been built is traceable to the customer requirements.

Validation testing is the process of answering the question "**Are we building the right product**".

7. PERFORMANCE AND LIMITATIONS

7.1. MERITS OF THE SYSTEM

- The system should be ready for use at any specified time.
- The system or component should perform its required functions under stated conditions for a specified period of time.

7.2. LIMITATIONS OF THE SYSTEM

- Minimum configuration will be needed when integrating with other system.

7.3. FUTURE ENHANCEMENTS

This application has opportunities for enhancement in following different options.

Employee database of the organization can be incorporated with the application and the authentication module can refer the organization's employee data base which is 'source of the truth'. This would help to avoid the redundancy of the employee data and accuracy in sign-on (login) process.

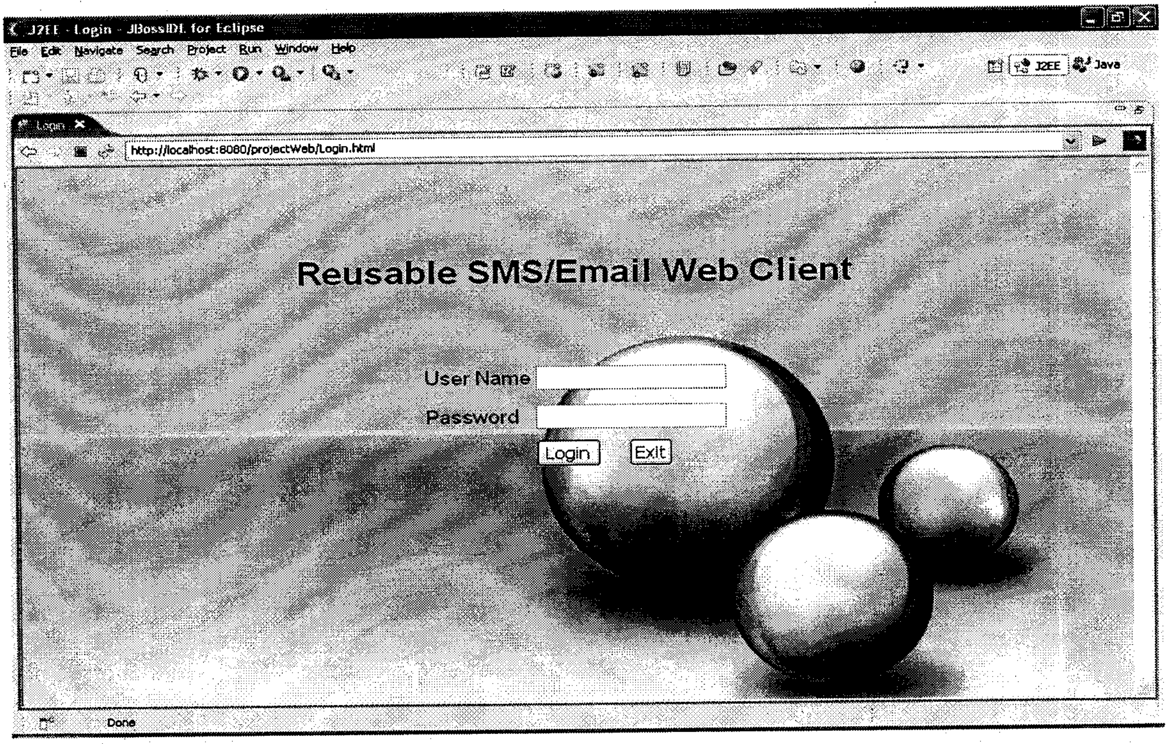
In the report generation, in addition to the existing reports, E-Mail ID, Mobile numbers, Subject, Date and Time fields can be added for easy references for the management. This would give entire E-Mail and SMS transaction.

A new screen can be developed for exploring the data residing in the E-mail Report Table and SMS Report Table. Using this new screen, separate reports can be generated for E-Mail and SMS.

8. APPENDICES

8.1. SAMPLE SCREENS

Login Screen: (Fig 8.1)

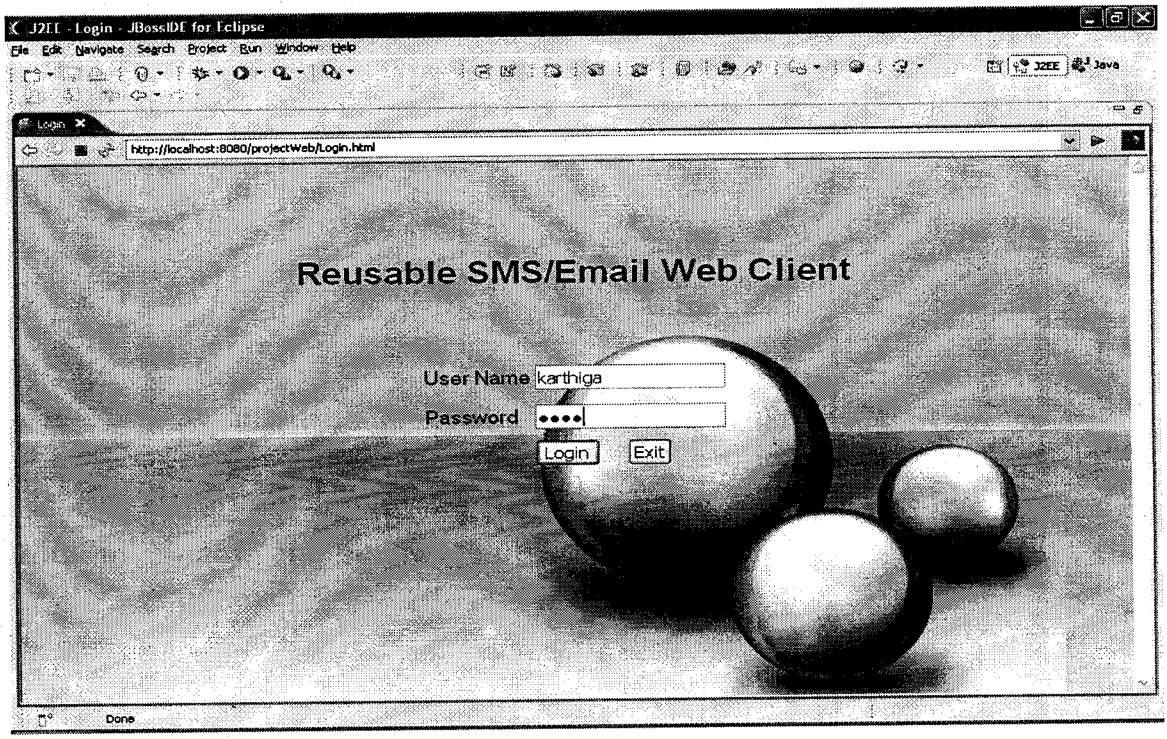

J2EE - Login - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Login
http://localhost:8080/projectWeb/Login.html
Login
Exit
Done

Reusable SMS/Email Web Client

User Name

Password

Enter user name and password on Login Screen: (Fig 8.2)

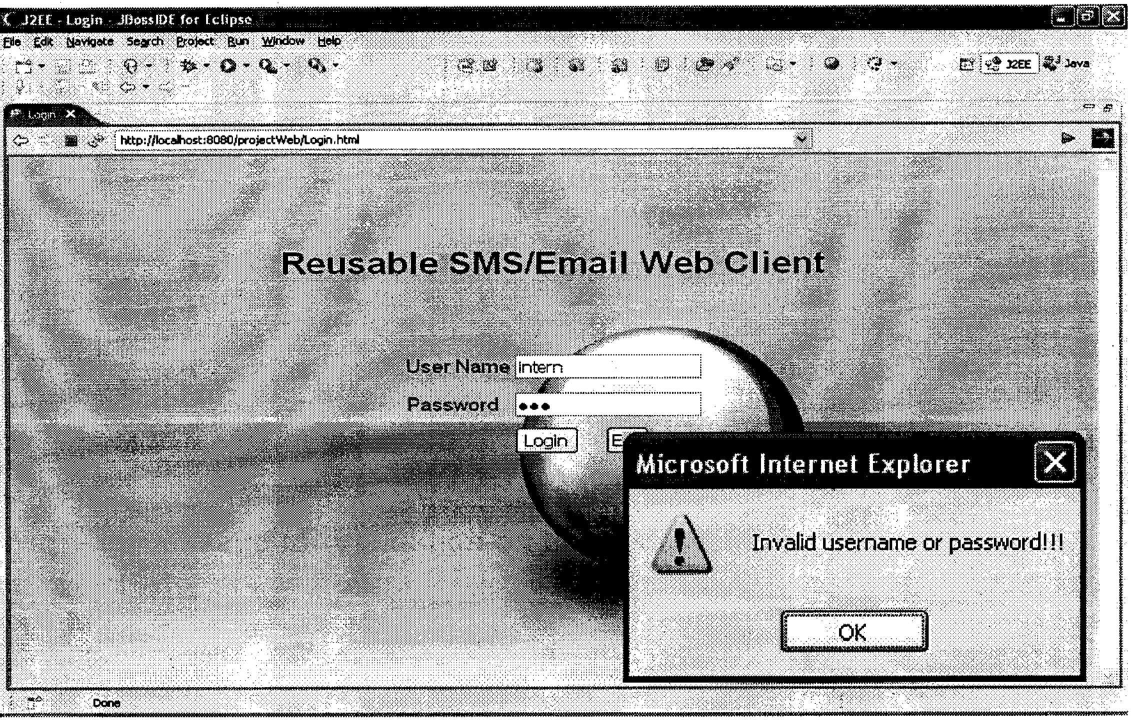

J2EE - Login - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Login
http://localhost:8080/projectWeb/Login.html
Login
Exit
Done

Reusable SMS/Email Web Client

User Name karthiga

Password ••••

Warning Message for Validation on Login Screen: (Fig 8.3)



J2EE - Login - JBossIDE for Eclipse

http://localhost:8080/projectWeb/Login.html

Reusable SMS/Email Web Client

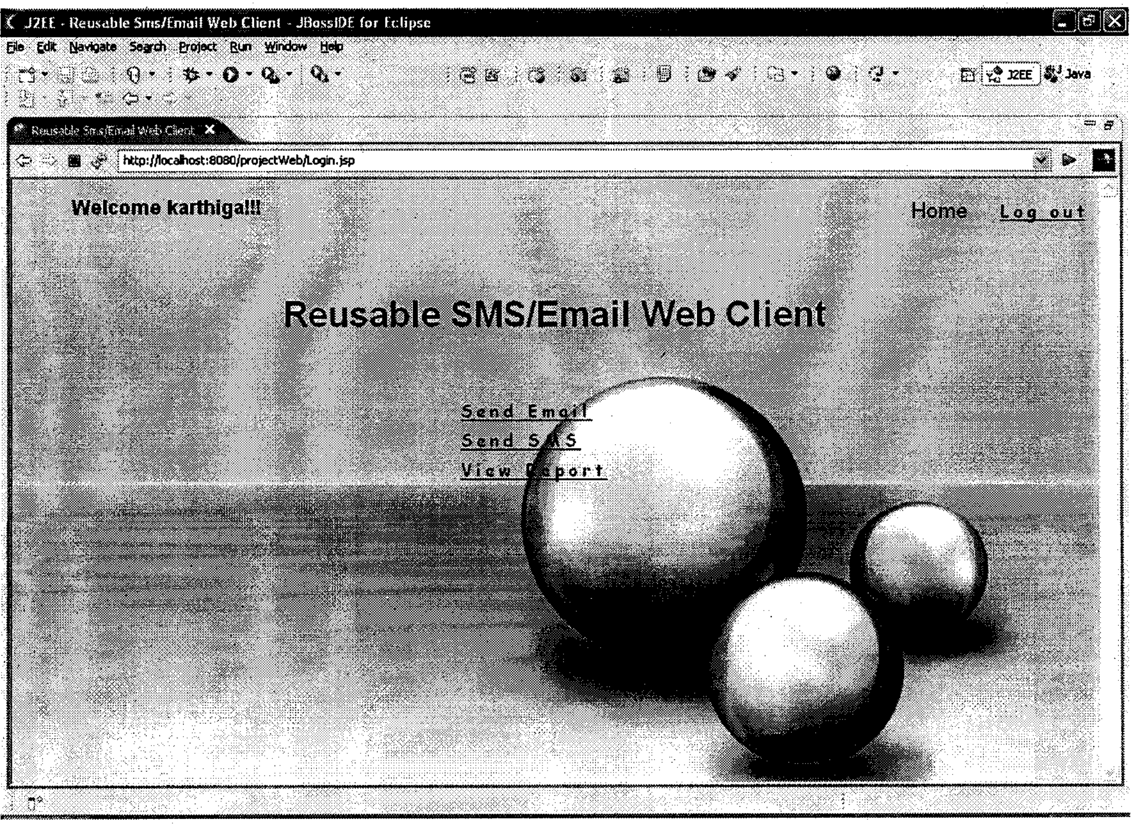
User Name: intern

Password: •••

Microsoft Internet Explorer

Invalid username or password!!!

Home Page screen after successful login: (Fig 8.4)



J2EE - Reusable Sms/Email Web Client - JBossIDE for Eclipse

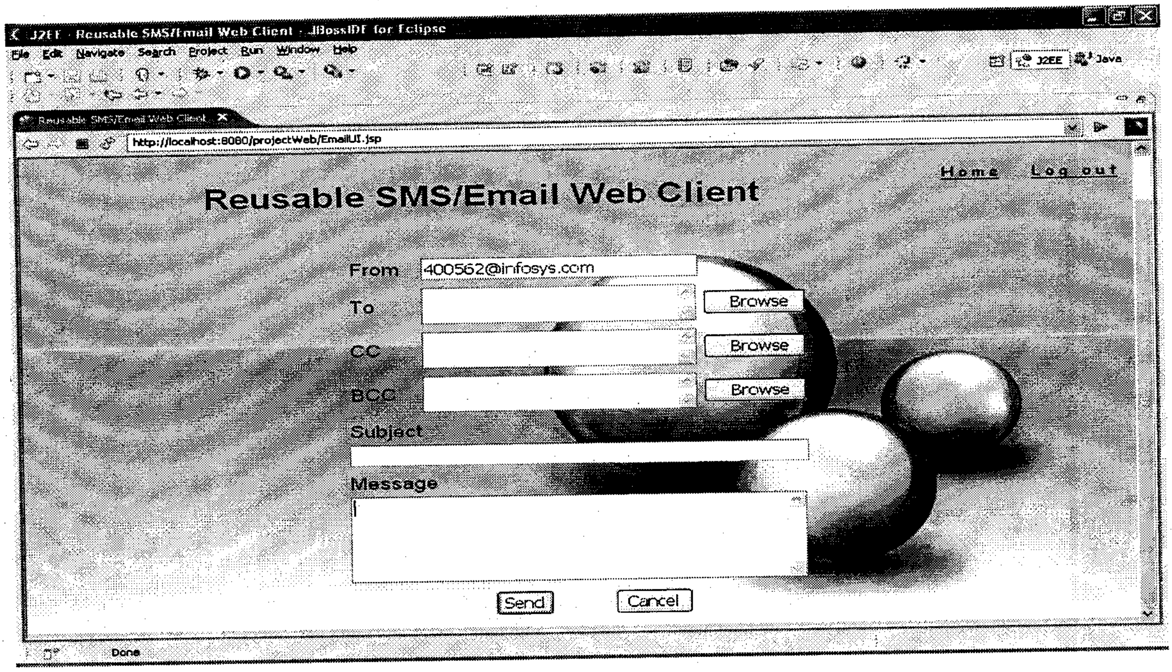
http://localhost:8080/projectWeb/Login.jsp

Welcome karthiga!!!

Reusable SMS/Email Web Client

File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable Sms/Email Web Client
Home
Log out
Send Email
Send SMS
View Report

E-Mail User Interface Page: (Fig 8.5)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse

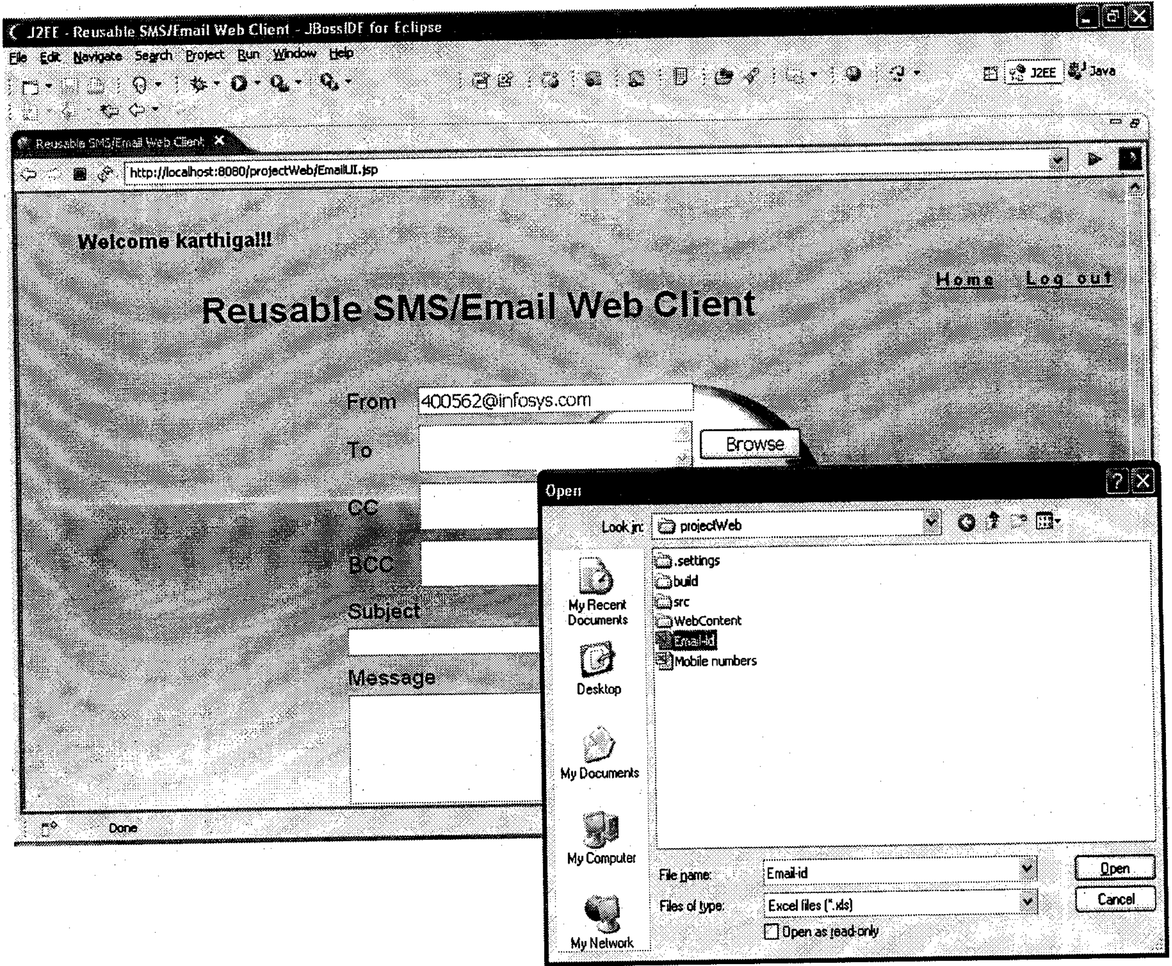
http://localhost:8080/projectWeb/EmailUI.jsp

Reusable SMS/Email Web Client

From: 400562@infosys.com

File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
Home
Log out
To
Browse
CC
Browse
BCC
Browse
Subject
Message
Send
Cancel
Done

Select spread sheet with E-Mail IDs to load: (Fig 8.6)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse

http://localhost:8080/projectWeb/EmailUI.jsp

Welcome karthigal!!

Reusable SMS/Email Web Client

From: 400562@infosys.com

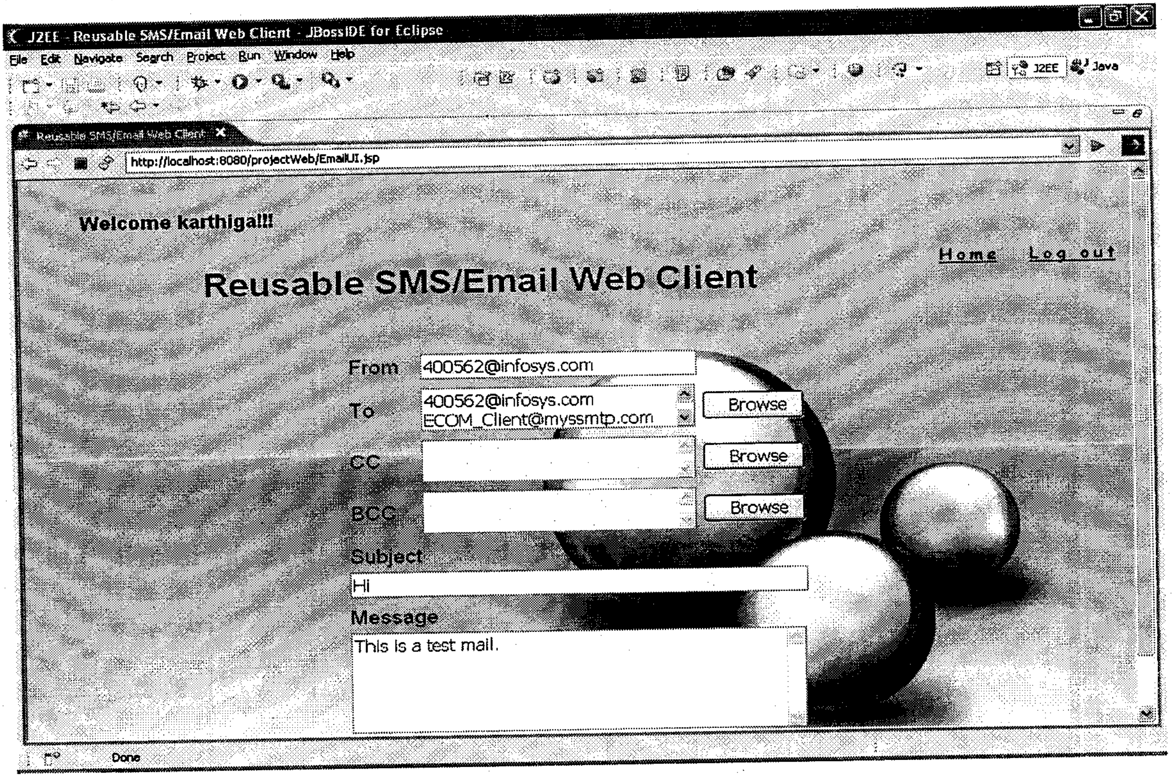
Open

Look in: projectWeb

File name: Email-id

Files of type: Excel files (*.xls)

E-Mail IDs are loaded on 'TO' field from Spread Sheet: (Fig 8.7)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse

http://localhost:8080/projectWeb/EmailUI.jsp

Welcome karthiga!!!

Reusable SMS/Email Web Client

From 400562@infosys.com

To 400562@infosys.com
ECOM_Client@myssmtp.com

CC

BCC

Subject

Hi

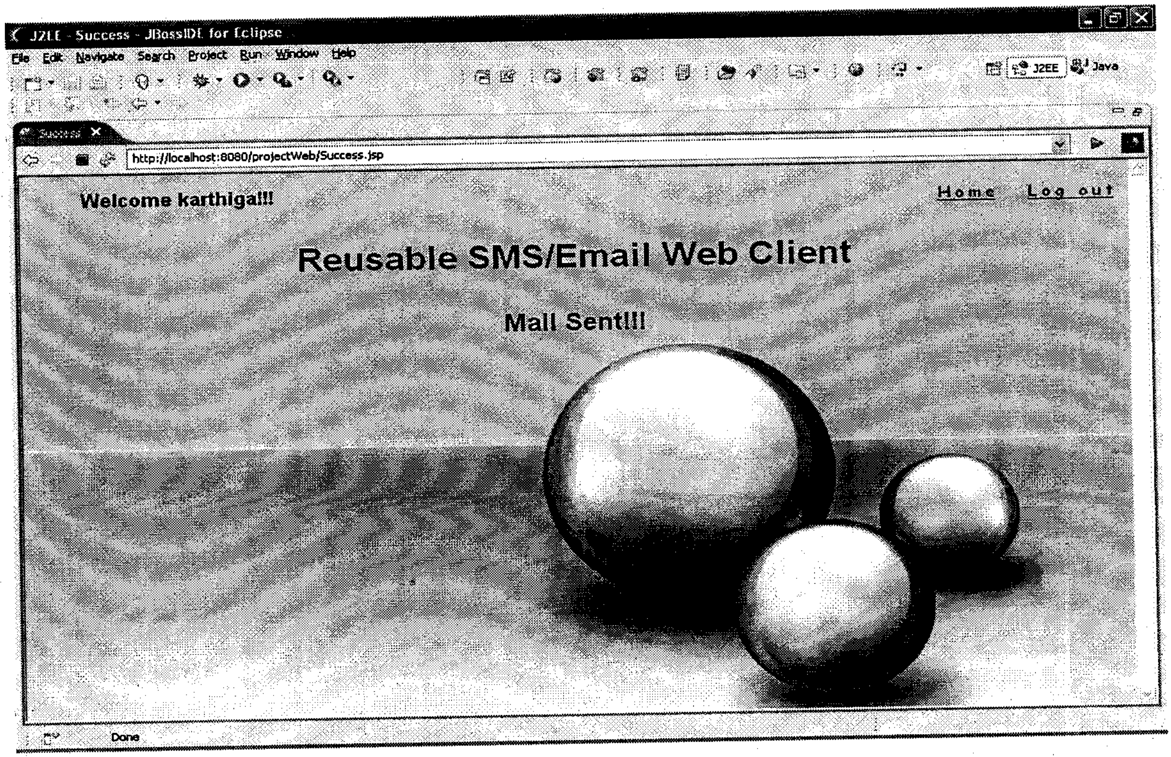
Message

This is a test mail.

Done

File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
Home
Log out
Browse
Browse
Browse

'Mail Sent' information message: (Fig 8.8)



J2EE - Success - JBossIDE for Eclipse

http://localhost:8080/projectWeb/Success.jsp

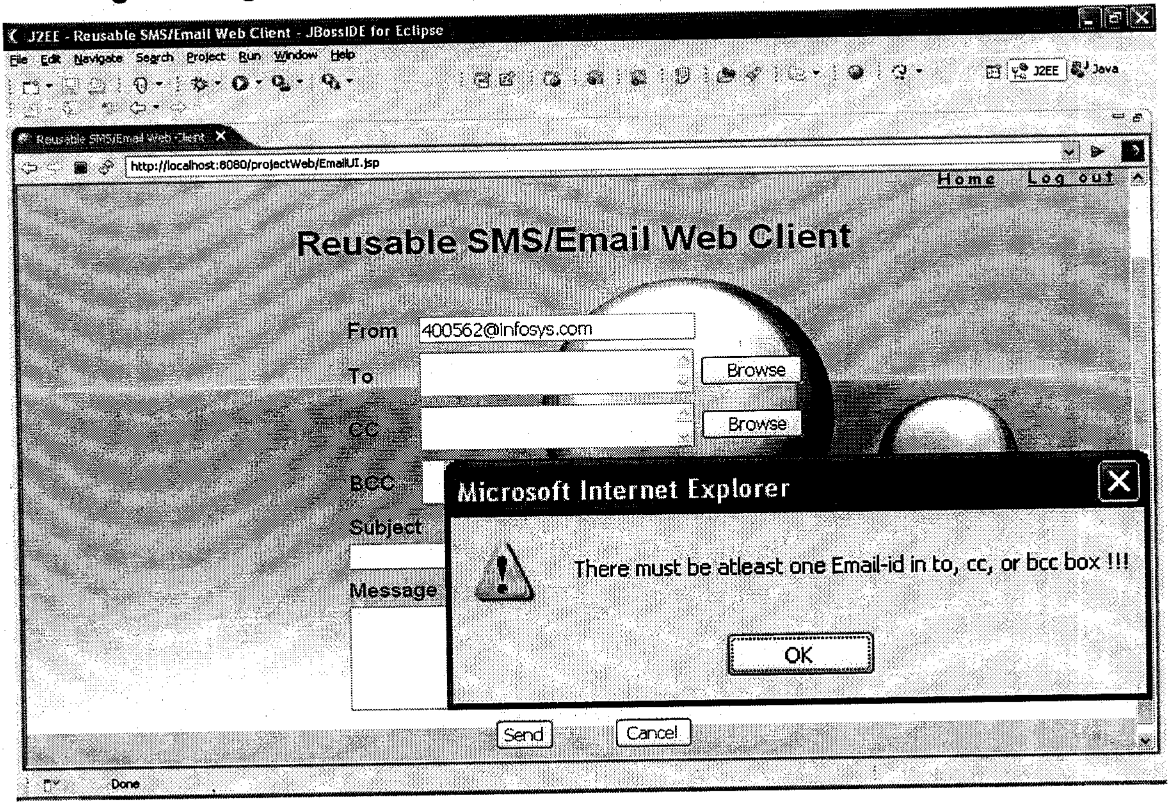
Welcome karthiga!!!

Reusable SMS/Email Web Client

Mail Sent!!!

Done

Warning messages for E-Mail ID Validation 1: (Fig 8.9)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
http://localhost:8080/projectWeb/EmailUI.jsp
Home
Log out

Reusable SMS/Email Web Client

From 400562@infosys.com

To

CC

BCC

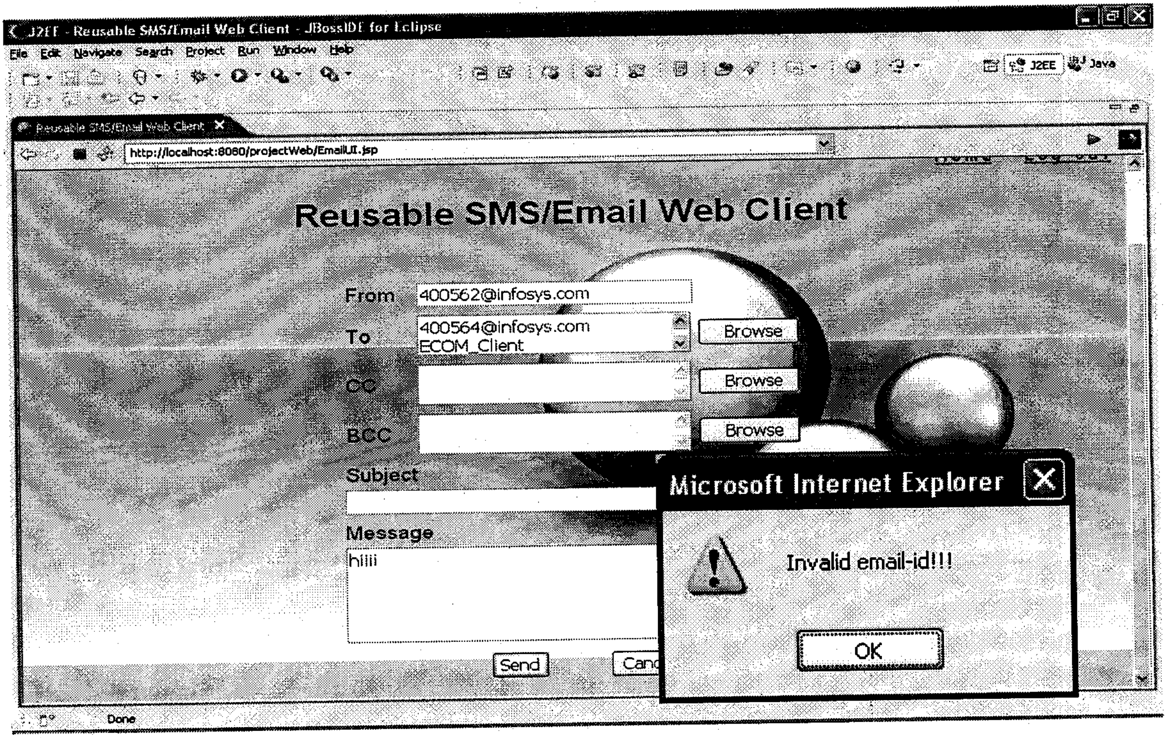
Subject

Message

Browse
Browse
Send
Cancel
Microsoft Internet Explorer

There must be atleast one Email-id in to, cc, or bcc box !!!

Warning messages for E-Mail ID Validation 2: (Fig 8.10)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
http://localhost:8080/projectWeb/EmailUI.jsp

Reusable SMS/Email Web Client

From 400562@infosys.com

To 400564@infosys.com ECOM_Client

CC

BCC

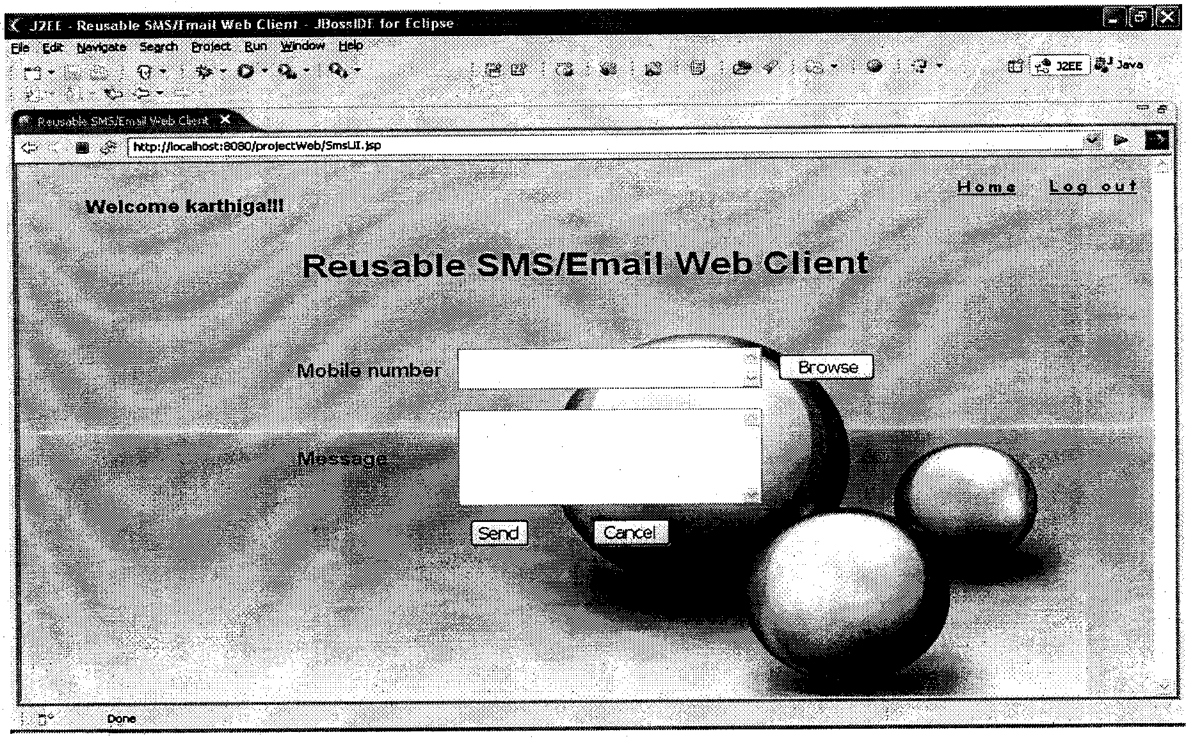
Subject

Message hiiii

Browse
Browse
Browse
Send
Canc
Microsoft Internet Explorer

Invalid email-id!!!

SMS User Interface Page: (Fig 8.11)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse

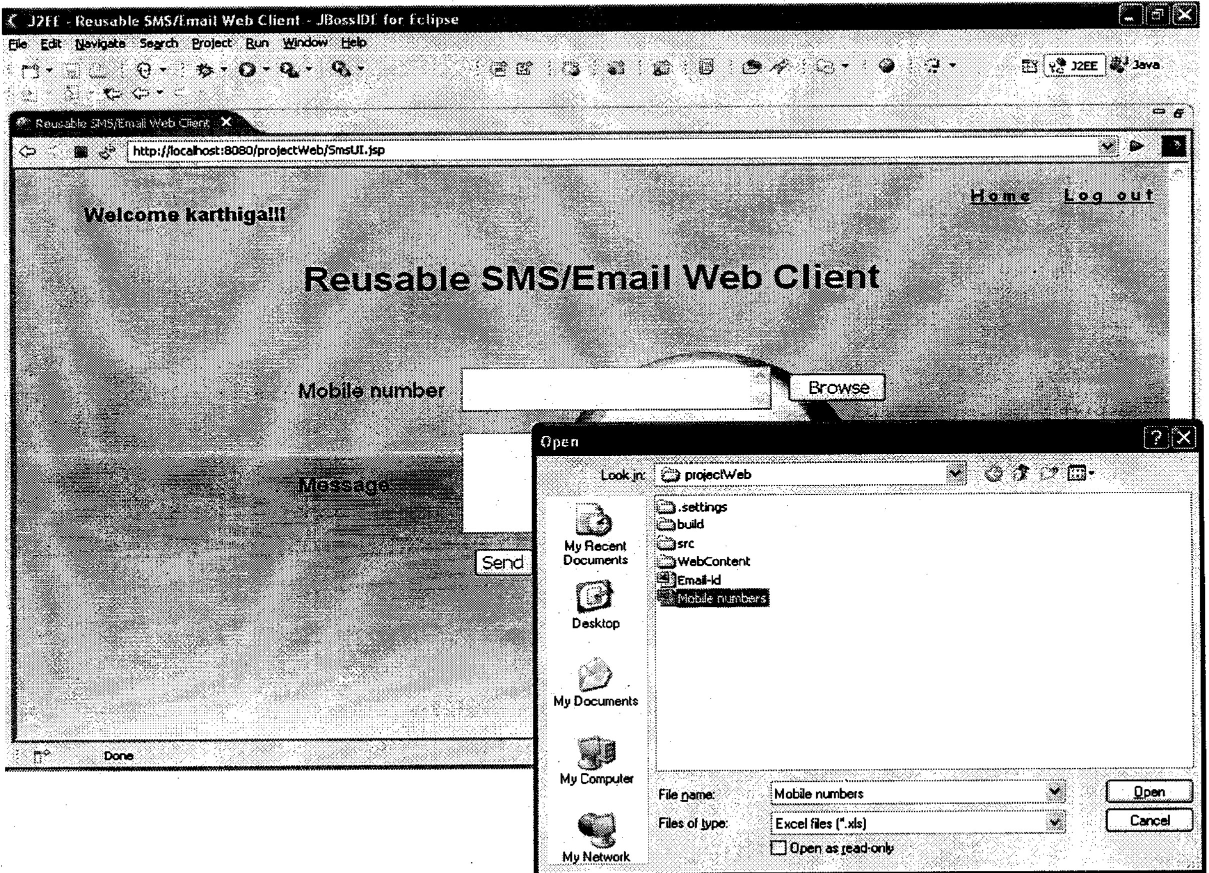
http://localhost:8080/projectWeb/SmsUI.jsp

Welcome karthiga!!!

Reusable SMS/Email Web Client

File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
Home
Log out
Mobile number
Browse
Message
Send
Cancel
Done

Select spread sheet with Mobile Numbers to load: (Fig 8.12)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse

http://localhost:8080/projectWeb/SmsUI.jsp

Welcome karthiga!!!

Reusable SMS/Email Web Client

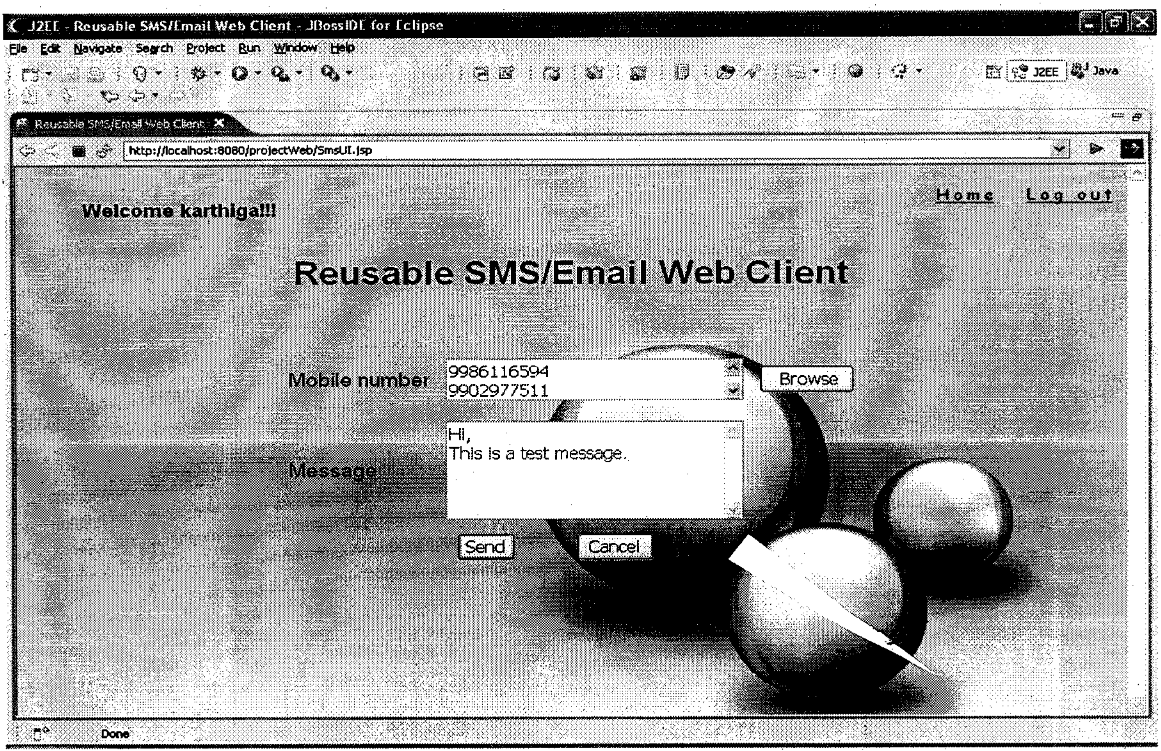
Look in: projectWeb

File name: Mobile numbers

Files of type: Excel files (*.xls)

□ Open as read-only

Mobile Numbers are loaded on 'TO' field from Spread Sheet: (Fig 8.13)


J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Reusable SMS/Email Web Client
Home
Log out
Browse
Send
Cancel
Done

http://localhost:8080/projectWeb/SmsUI.jsp

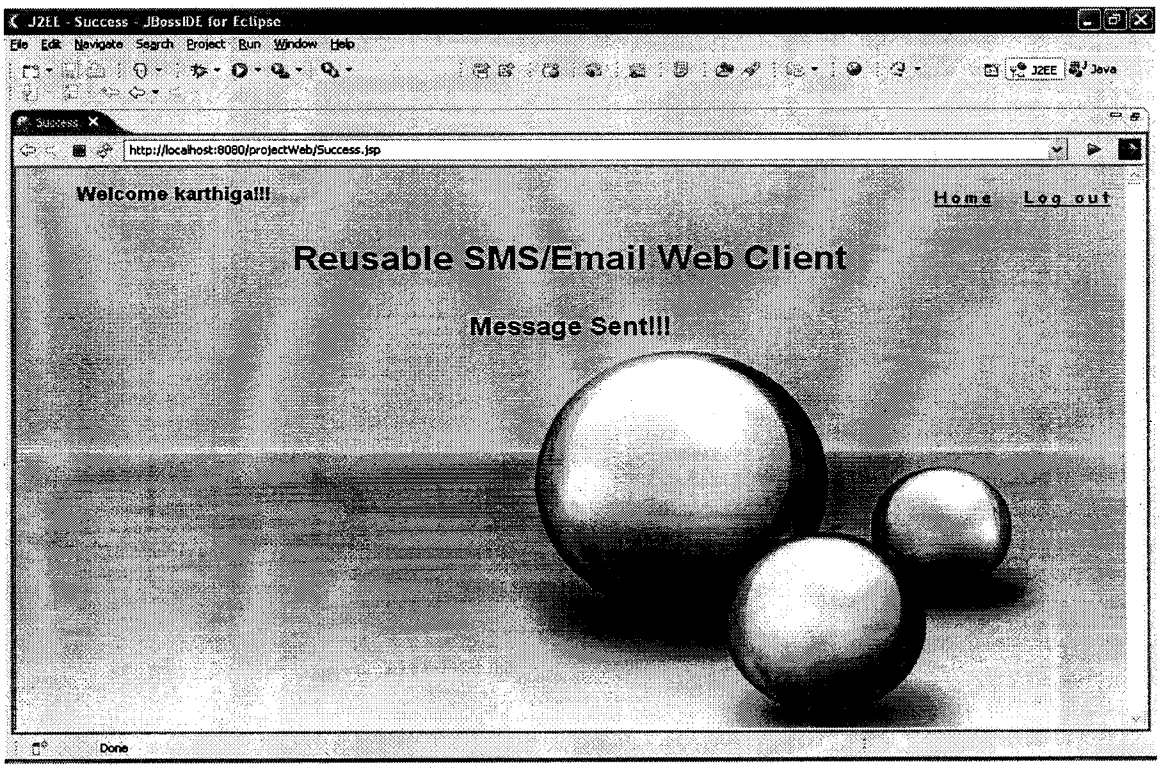
Welcome karthiga!!!

Reusable SMS/Email Web Client

Mobile number: 9986116594
9902977511

Message: Hi,
This is a test message.

'SMS Sent' information message: (Fig 8.14)


J2EE - Success - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Success
Home
Log out
Done

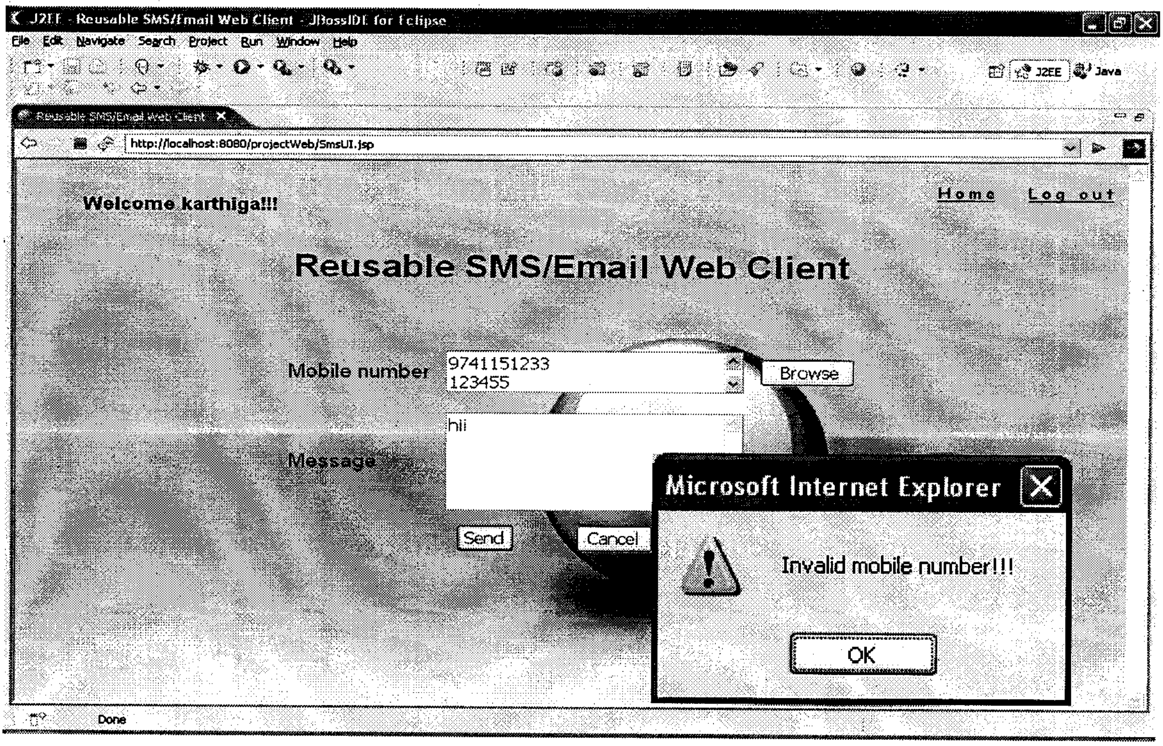
http://localhost:8080/projectWeb/Success.jsp

Welcome karthiga!!!

Reusable SMS/Email Web Client

Message Sent!!!

Warning messages for Mobile Number Validation: (Fig 8.15)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Reusable SMS/Email Web Client
http://localhost:8080/projectWeb/SmsUI.jsp
Home
Log out

Welcome karthiga!!!

Reusable SMS/Email Web Client

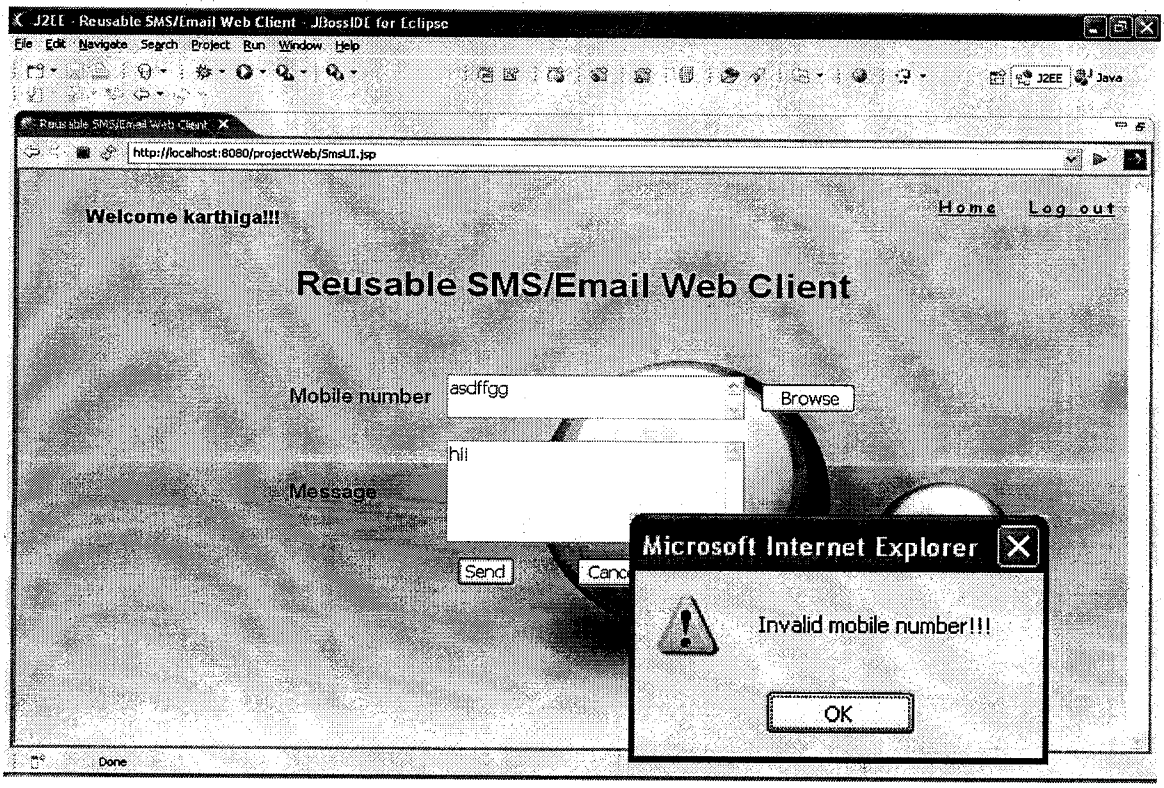
Mobile number: 9741151233
123455

Message: hii

Microsoft Internet Explorer

Invalid mobile number!!!

Warning messages for Mobile Number Validation: (Fig 8.16)



J2EE - Reusable SMS/Email Web Client - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Reusable SMS/Email Web Client
http://localhost:8080/projectWeb/SmsUI.jsp
Home
Log out

Welcome karthiga!!!

Reusable SMS/Email Web Client

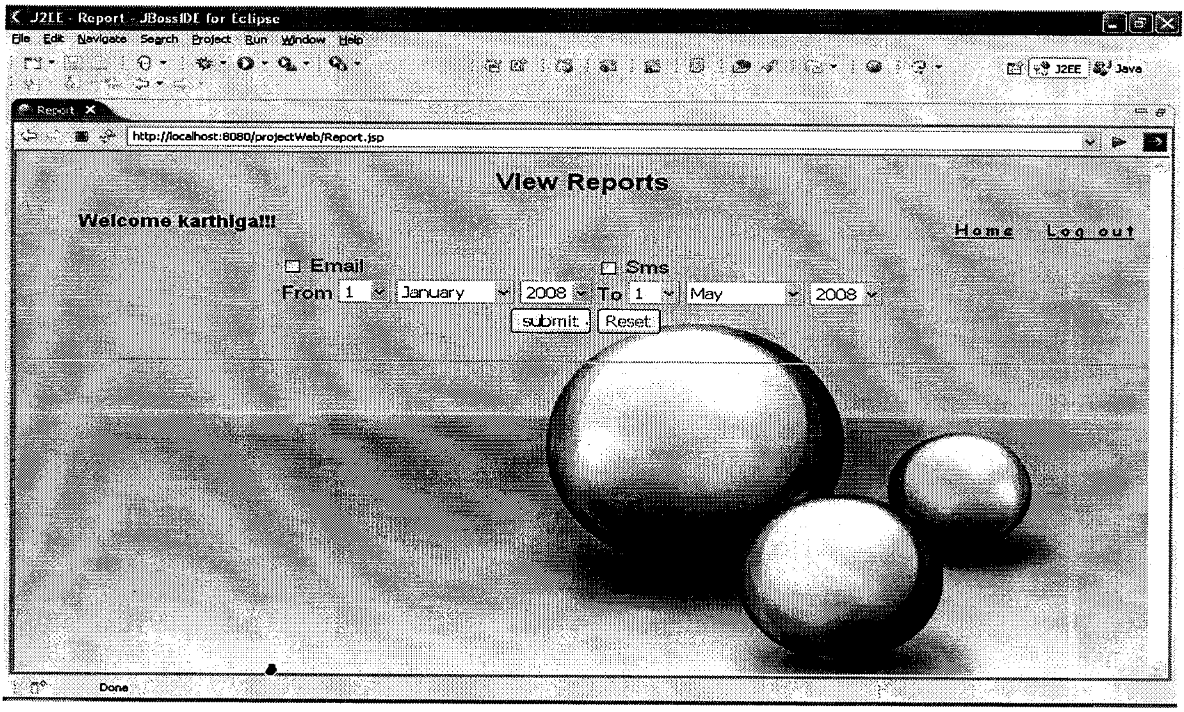
Mobile number: asdffgg

Message: hii

Microsoft Internet Explorer

Invalid mobile number!!!

Report enquiry screen: (Fig 8.17)


J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Report
Done

http://localhost:8080/projectWeb/Report.jsp

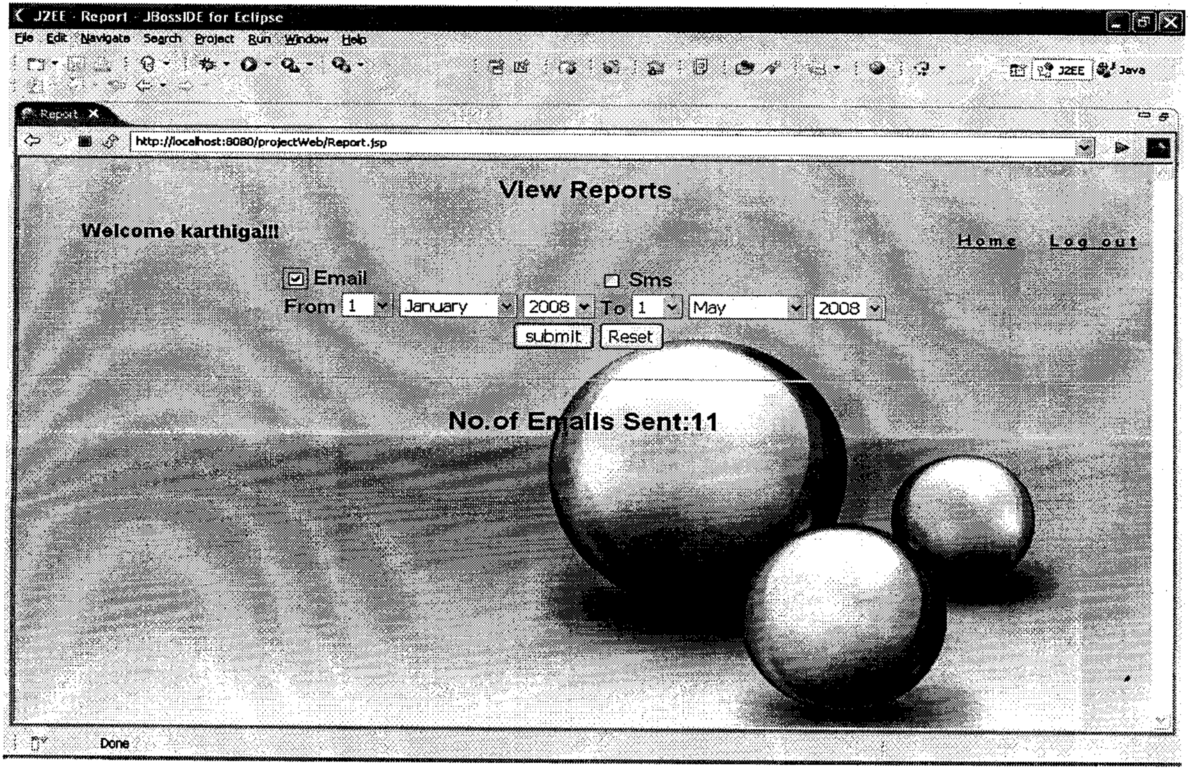
View Reports

Welcome karthiga!!!

□ Email □ Sms

From 1 January 2008 To 1 May 2008

Report Result Screen for E-Mail: (Fig 8.18)


J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Report
Done

http://localhost:8080/projectWeb/Report.jsp

View Reports

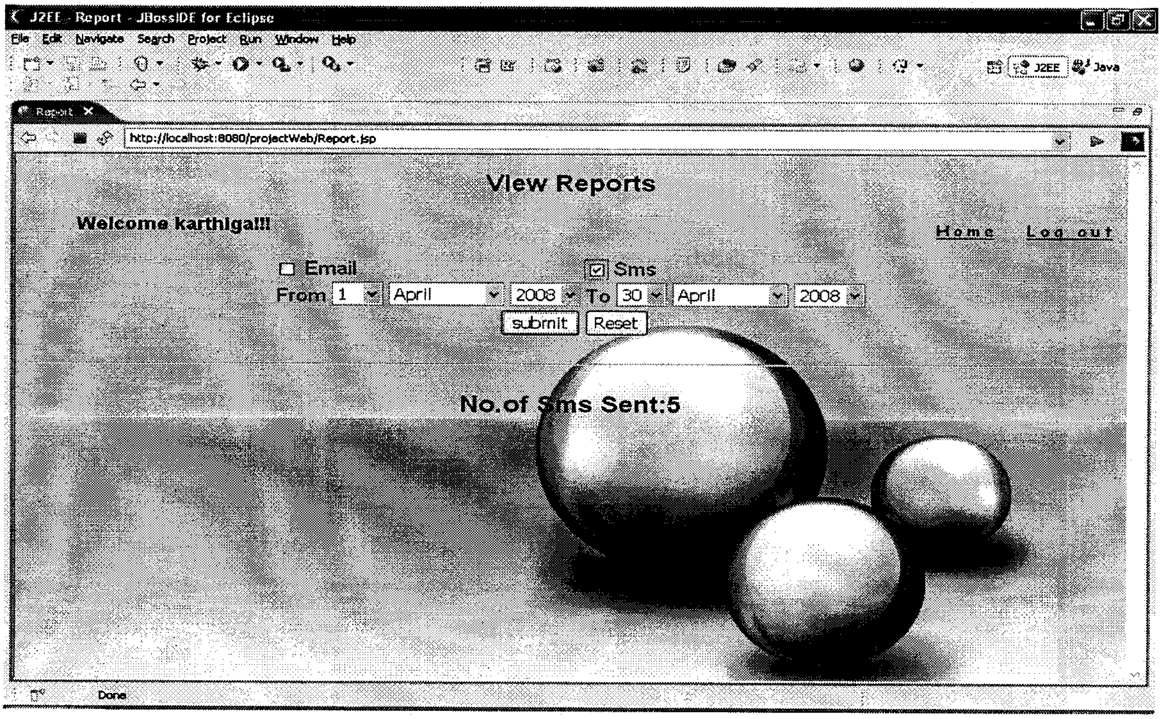
Welcome karthiga!!!

☑ Email □ Sms

From 1 January 2008 To 1 May 2008

No.of Emails Sent:11

Report Result Screen for SMS: (Fig 8.19)



J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Report
Done

http://localhost:8080/projectWeb/Report.jsp

View Reports

Welcome karthiga!!!

Home Log out

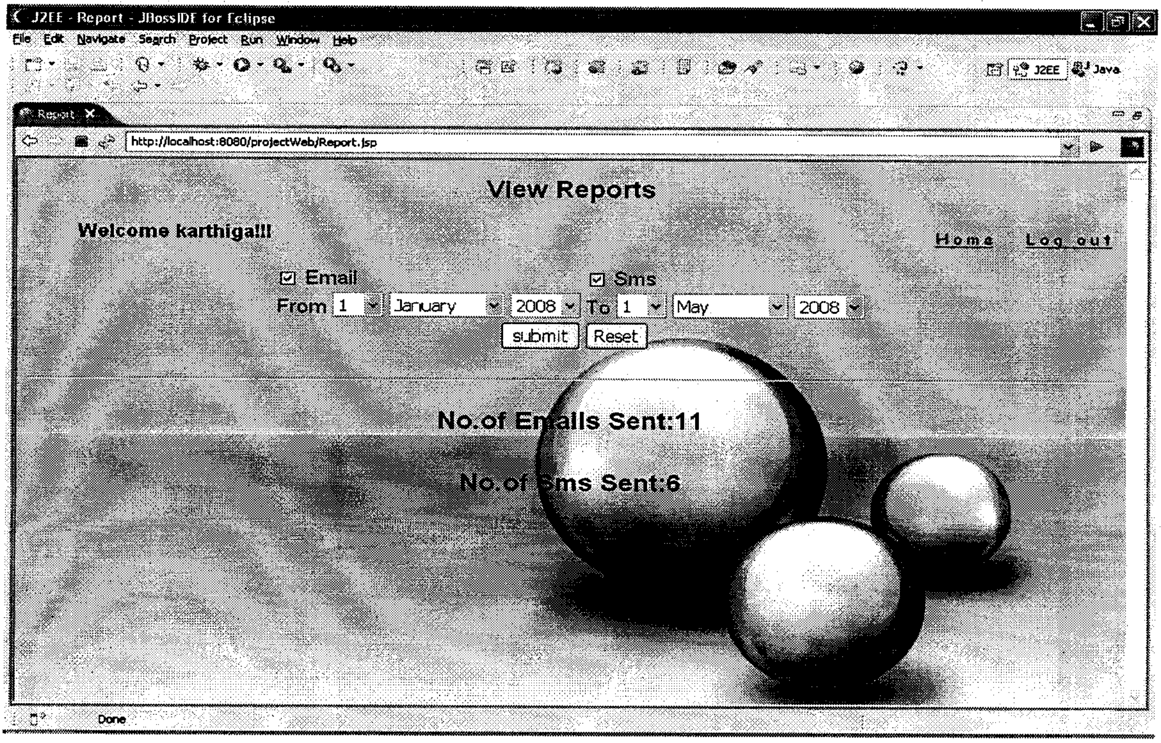
□ Email ☑ Sms

From 1 April 2008 To 30 April 2008

submit Reset

No.of Sms Sent:5

Report Result Screen for both E-Mail and SMS: (Fig 8.20)



J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Report
Done

http://localhost:8080/projectWeb/Report.jsp

View Reports

Welcome karthiga!!!

Home Log out

☑ Email ☑ Sms

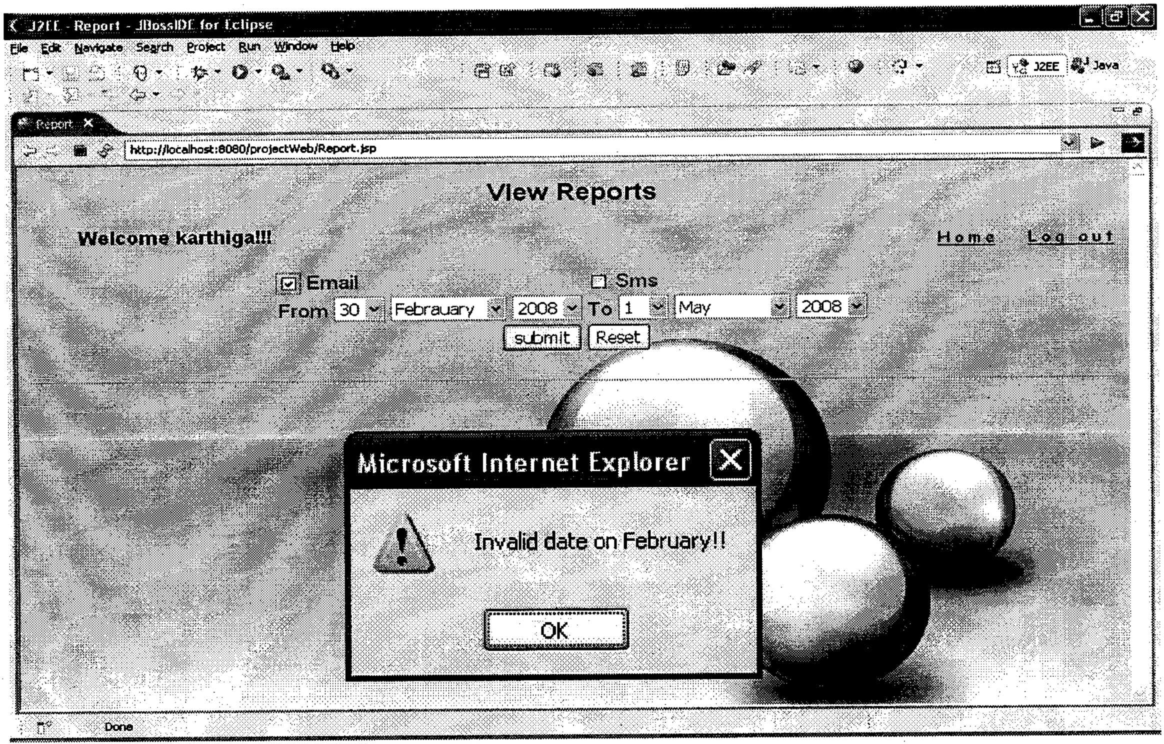
From 1 January 2008 To 1 May 2008

submit Reset

No.of Emails Sent:11

No.of Sms Sent:6

Warning messages for Date Validation in Report Screen: (Fig 8.21)


J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Report

http://localhost:8080/projectWeb/Report.jsp

View Reports

Welcome karthiga!!!

Home Log out

☑ Email □ Sms

From 30 Febrauary 2008 To 1 May 2008

submit Reset

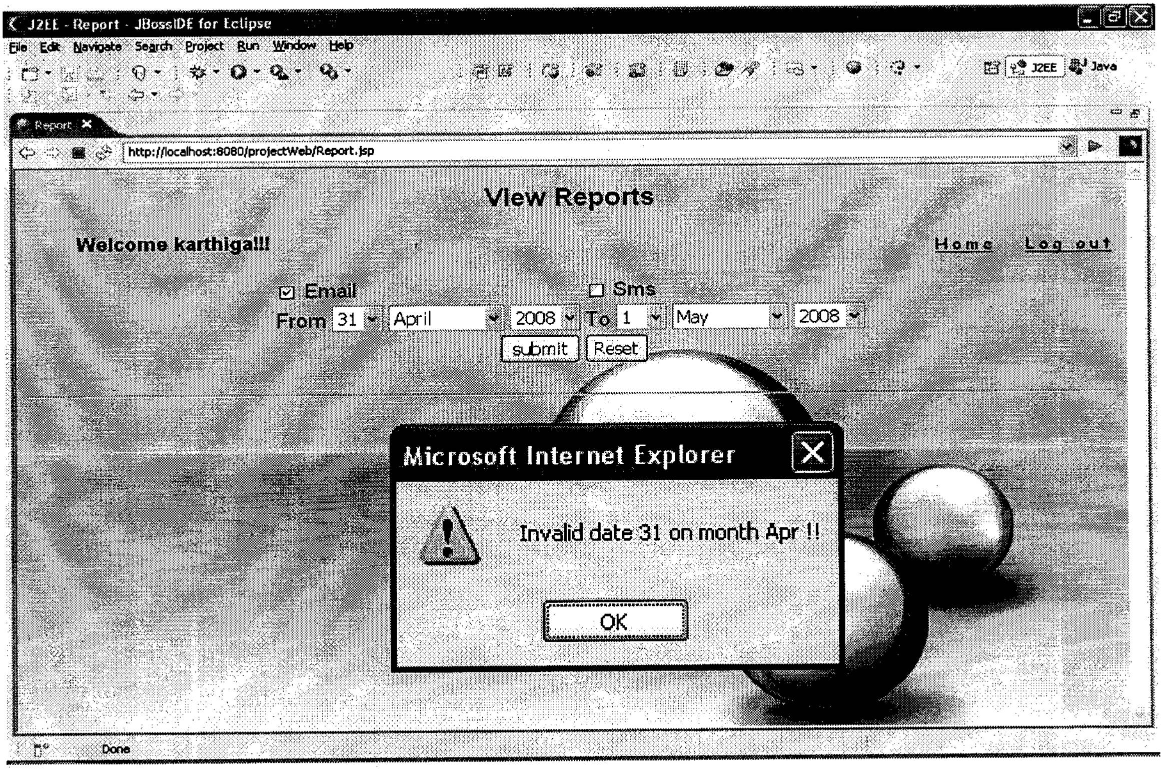
Microsoft Internet Explorer

Invalid date on February!!

OK

Done

Warning messages for Date Validation in Report Screen: (Fig 8.22)


J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE
Java
Report

http://localhost:8080/projectWeb/Report.jsp

View Reports

Welcome karthiga!!!

Home Log out

☑ Email □ Sms

From 31 April 2008 To 1 May 2008

submit Reset

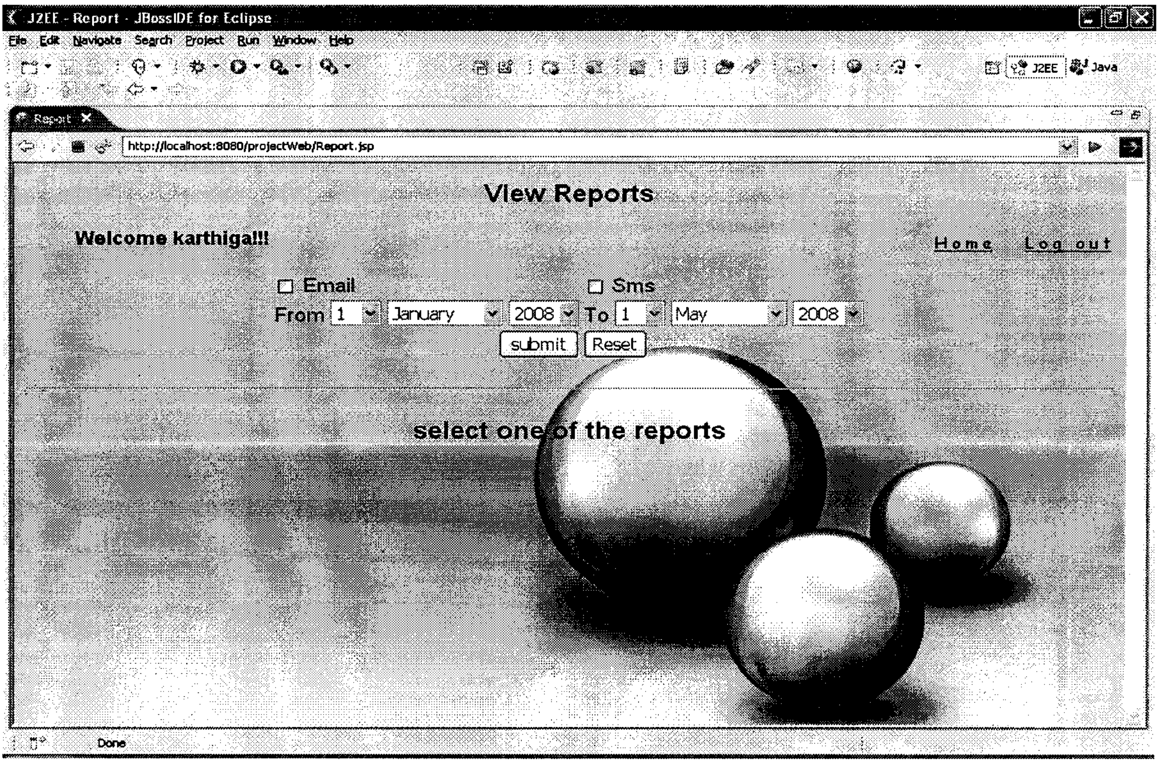
Microsoft Internet Explorer

Invalid date 31 on month Apr !!

OK

Done

Warning messages for selection in Report Screen: (Fig 8.23)


J2EE - Report - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Report
http://localhost:8080/projectWeb/Report.jsp
Done

View Reports

Welcome karthiga!!!

Home Log out

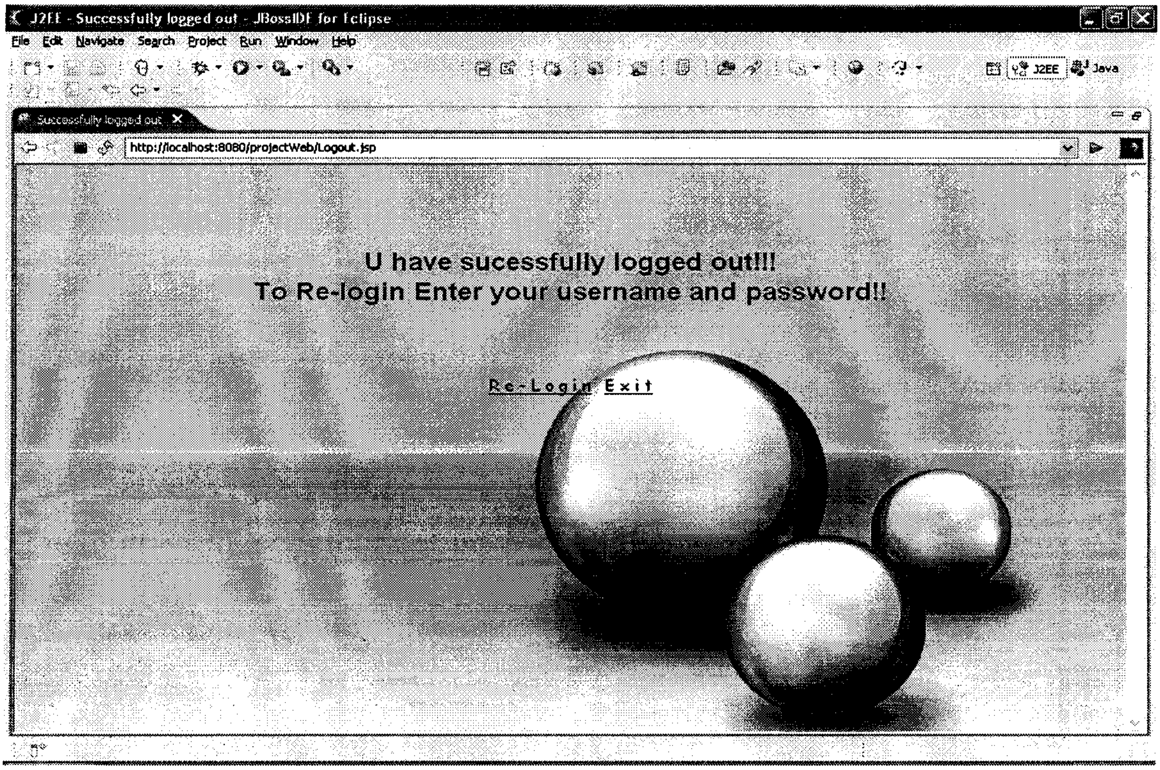
□ Email □ Sms

From 1 January 2008 To 1 May 2008

submit Reset

select one of the reports

Screen after successful Logout: (Fig 8.24)


J2EE - Successfully logged out - JBossIDE for Eclipse
File Edit Navigate Search Project Run Window Help
J2EE Java
Successfully logged out
http://localhost:8080/projectWeb/Logout.jsp

U have sucessfully logged out!!!
To Re-login Enter your username and password!!

Re-Login Exit

9. REFERENCES

REFERENCES:

1. James Edward Keogh (2002), **J2EE: The complete Reference**, McGraw-Hill Osborne Media; 1st Edition.

2. Robert McGovern and Stuart Charlton (2002), **Code Notes for J2EE: EJB, JDBC, JSP and SERVLETS**, Edited by Gregory Brill, Random House.

3. Marty Hall, Larry Brown (2005), **Core SERVLETS and JAVA SERVER PAGES**, Volume 1: Core Technologies, 2nd Edition, Sun Micro Systems Press.

WEBSITE REFERRED:

1. ***http://java.sun.com/developer/Books/javaserverpages/cservletsjsp/index.html***

2. ***http://java.sun.com/products/jsp/docs.html***

3. ***http://www.javaworld.com/javaworld/jw-12-1999/jw-12-ssj-jspmvc.html***

4. ***http://www.webdevelopersjournal.com/articles/jsp_build.html***