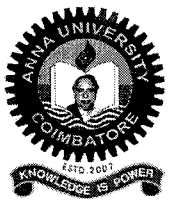
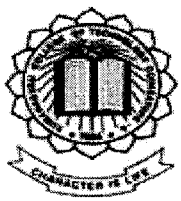
P-3253





SMS WEB SERVICES

PROJECT REPORT

Submitted By

R.REKHA

Register No.: 0720300036

in partial fulfillment for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

in

COMPUTER APPLICATIONS

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

May, 2010

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

COIMBATORE – 641 006.

Department of Computer Applications

PROJECT WORK

MAY 2010

This is to certify that the project entitled

SMS WEB SERVICES

is the bonafide record of project work done by

R.REKHA

Register No: 0720300036

of MCA (Computer Applications) during the year 2009-2010.

Project Guide

Head of the Department

Submitted for the Project Viva-Voce examination held on 17.05.10

Internal Examiner

External Examiner

DECLARATION

I affirm that the project work titled **SMS WEB SERVICES** being submitted in partial fulfilment for the award of **MASTER OF COMPUTER APPLICATIONS** is the original work carried out by me. It has not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other University.

R.REKHA

Register No. : 0720300036

I certify that the declaration made above by the candidate is true.

Signature of the Guide

Cognizant Technology Solutions India Private Ltd.
Mountain Mist Campus, Unit -1, STPI - IT Park,
No.363, KGISL Campus, Thudiyalur Road,
Coimbatore - 641035
Phone : (+91-0422) 3923000
Fax : (+91-0422) 3923060

www.cognizant.com

Cognizant

Dear Sir/Madam

This is to certify that **Rekha R**, a **MCA** student of **Kumaraguru College of Technology** has done project work in the company on SMS-Web Services under the guidance of Aravindan Balakrishnan, as part of the college requirement, between the period January 2010 and April 2010.

Yours sincerely,

for **Cognizant Technology Solutions India Pvt. Ltd.**,

Sriram Iyer
Manager – Human Resources

I accept the terms and conditions of the offer as mentioned above.

Signature: R. Rekha
Date: 07-05-2010

ABSTRACT

This project **"SMS Web Services"** deals with providing services to the customer who seek services through SMS **(Short Message Service)**. The Services available are stored into the server. The users of the system are Administrator, Agents, Members and Customers. The system identifies the members and customers by based on the keyword of the message.

It provides two services RENT and BE. The purpose of RENT service is register and searching the house for rent. And BE service is for student can search Engineering College based on the cutoff mark.

Administrator maintains the overall system. Agents are allocated to each and every area. They will verify the registered members as faith and collect the registration fees from corresponding member.

The member, who can afford some services, can register and provide the details through SMS. If a customer is in need of some services, he/she can request through SMS regarding the service. The message sent to and received from the server must be of some predefined format.

In this website "Supersms.com" the administrator and every agent can access this site by using valid username and password.

This System is developed in PHP with HTML and Flex 3, MySQL and WampServer2.0i. Hence it can be accessed by all Members and Customers irrespective of the location.

ACKNOWLEDGEMENT

I wish to express my thanks to **Dr. S. Ramachandran, Principal,** Kumaraguru College of Technology, Coimbatore, for permitting me to undertake this project.

I wish to express earnest thanks to **Dr. A. Muthukumar, Course Co-ordinator,** Master of Computer Applications, Kumaraguru College of Technology, Coimbatore, for his support and encouragement.

My deepest acknowledgement to the project co-ordinator **Mr. S. Hameed Ibrahim, Senior Lecturer**, Master of Computer Applications, Kumaraguru College of Technology, Coimbatore, for his timely remarks and comments about the project, which helped me to emend my project and knowledge.

I am very much indebted to my guide **Mr. M. Manikantan, Senior Lecturer**, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore, for his support and encouragement in making this project a success. His excellent suggestions and timely encouragement helped me complete the project and the project report.

I am very much indebted to **Mr. S. Ganesh Babu, Senior Lecturer**, Department of Computer Applications, Kumaraguru College of Technology, Coimbatore for his complete assistance, guidance and support given to me throughout my project.

I wish to express my gratitude to my CTS Project guide, **Mr.Aravindan Balakrishnan,** Cognizant Technology Solutions, Coimbatore, for his useful suggestions and support given throughout the project.

Finally, I owe a lot to my beloved parents and family members and to my friends and to my department staffs for their help and co-operation to complete this project successfully.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER-1
INTRODUCTION

1.1 ORGANIZATION PROFILE

Cognizant (NASDAQ: CTSH) is a global IT services and business process outsourcing solutions provider. By leveraging highly flexible business processes, a seamless global delivery network and deep domain expertise, Cognizant delivers a better "return on outsourcing."

Cognizant is a different kind of offshore outsourcing company. They bring you the best of both worlds: the savings and superior quality of offshore development, and the immediacy and trust of a local company.

They deliver the full range of application outsourcing, business process, and consulting & system integration services that you expect. And, because of our background savings the Global 2000, they are experts at managing the highly complex, long-term engagements that our clients require. Unlike our rivals, they work, not from contract, but in long-term partnerships with our customers.

They are financially strong with $150m in cash, and profit margins. In October 2003 we topped Forbes list of "hotshots" as the number one company. They are audited and certified to BS 7799, the set of international regulations governing physical and network security, and are well-experienced in working with clients that implement their own U.S. standards.

They are routinely audited for security by our own clients, and routinely enforce their U.S.-level standards wherever the work in performed. They can help you gain the benefits of a world-class staff, whose quality and process disciplines have achieved ISO 9001 and SEI CMM Level 5 certifications. However, they go much further than achieving the certifications that are expected of offshore service providers.

1.2 PROJECT OVERVIEW

"SMS Web Services" system deals with providing services to the customer who seek services through SMS (Short Message Service). The Services available are stored into the server. Members and Customers are the two types of users. The system identifies the users based on the keyword of the message.

It provides two services RENT and BE. The purpose of RENT service is register and searching the house for rent. And BE service is for student can search Engineering College based on the cutoff mark.

Administrator maintains the overall system. Administrator can add, delete and edit the advertisement, cutoff and pin code details and view the services, report details through the username and password. And also specify the message format

Agents are allocated to each and every area. They will verify the registered members as faith and collect the registration fees from corresponding customer. Agent can view the all services details, customer details and reports using their username and password.

The member, who can afford some services, can register and provide the details to the server through SMS. And the member's service details are verified through the agent.

If a customer is in need of some services, he/she can request through SMS regarding the service. The requesting message will be directed to the server. Then with the help of key words, server is searched for the service. If the service is available, the reply message will be sent to the customer.

The message sent to and received from the server must be of some predefined format.

CHAPTER-2

SYSTEM STUDY AND ANALYSIS

System analysis is the process of understanding a problem domain and the user requirements for the purpose of developing an application to serve the users. This chapter deals with the drawbacks and disadvantages of existing system and advantages of the proposed system.

2.1 EXISTING SYSTEM

The various aspects of the existing system are thoroughly analyzed and the need for the proposed system is taken into account. In general existing system follows the manual approach. So the user has to search from various places. The students have to visit the colleges to know about the counseling details.

2.1.1 Drawbacks of the existing system

- ✓ Time Consuming is very large to get the details.
- ✓ User has to search the services by moving from place to place. So it takes more time and human resource also increased.
- ✓ Students can know about the counseling details by visiting the college.

2.2 PROPOSED SYSTEM

The main objective of the proposed system is to reduce human resource and time consumption. When the user sent a message using predefined format and the server will sent a quick response through SMS.

The system includes facilities like sending advertisement details to the user while search. The Student can easily search the college which matches their cutoff marks.

2.2.1 Benefits of the Proposed System

- ✓ Saves a lot of time.
- ✓ The user can request services at any time and any where through the SMS.
- ✓ Efficient retrieval of information.
- ✓ Helps in easier broadcasting of existing services such as house for rent.
- ✓ Easy to search Engineering College Details.
- ✓ Helps in publishing Advertisement through SMS when response to the user.

CHAPTER-3

DEVELOPMENT ENVIRONMENT

This chapter development environment is a technical specification of requirement for the software product. It explains about the front end and back end tools used for this system.

3.1 HARDWARE REQUIREMENTS

The hardware requirements are,

Processor	:	Intel Pentium IV
Speed	:	3.1 GHZ
Memory	:	1 GB RAM
Hard Disk Capacity	:	80 GB
Monitor	:	15" inch SVGA
Mouse	:	HP Optical Scroll Mouse
Keyboard	:	HP 106 Keys

3.2 SOFTWARE REQUIREMENTS

The software needed for the development of the system is given below,

Operating System	:	Windows XP
Designing Tool	:	HTML, Zend Studio 6.1.2, Flex Builder 3
Server-side script	:	PHP 5.3.0
Client-side script	:	JavaScript
Web Server	:	WampServer2.0i
Back-end	:	MY SQL 5.1

3.3 SOFTWARE OVERVIEW

WINDOWS-XP:

Windows-XP is a powerful system. It includes many useful programs and accessories. It provides a wealth of features and it keeps the complex features relatively simply to use. It tries to create a work environment for the computers that model the actual working environment on our desk.

Windows–XP has many new and enhanced features.

- ✓ It has the facility for multitasking.
- ✓ It is user friendly.
- ✓ Every application has its own online help facility.

PHP:

PHP is a programming language devised by Rasmus Lerdorf in 1994 for building dynamic, interactive Web sites. PHP is a server-side scripting language that allows your Web site to be truly dynamic. PHP stands for *PHP: Hypertext Preprocessor*. Its flexibility and relatively small learning curve (especially for programmers who have a background in C, Java, or Pearl) make it one of the most popular scripting languages around. PHP's popularity continues to increases as businesses and individuals everywhere embrace it as an alternative to Microsoft's ASP language and realize that PHP's benefits most certainly outweigh the costs. According to Zend Technologies, Ltd., the central source of PHP improvements and designers of the Zend Engines, which supports PHP applications, PHP code can now be found in approximately 9 million Web sites.

PHP's main use is as a cross-platform, HTML-embedded server-side Web scripting language. Let's take a moment to examine these terms:

- **Cross-platform:** Most PHP code can be processed without alteration on computers running many different operating systems. For example, a PHP script that runs on Linux generally also runs well on Windows.
- **HTML-embedded:** PHP code can be written in files containing a mixture of PHP instructions and HTML code.
- **Server-side:** The PHP programs are run on a server – specifically a Web server.
- **Web-scripting language:** PHP programs run via a Web browser.

JAVASCRIPT:

In early December 1995, Netscape and Sun Microsystems jointly announced a scripting language known as JavaScript. Microsoft acknowledged the potential power and popularity of the language by implementing it in IE3.

Newer browser makers automatically provided support for JavaScript. When employed on the client computer, the language can help turn a static page of content into an engaging, interactive, and intelligent experience.

JavaScript provides following kinds of solutions:

- Getting your Web page to respond or react directly to user interaction with form elements and hypertext links.
- Distributing small collections of database-like information and providing a friendly interface to that data.
- Controlling multiple-frame navigation, plug-ins, or Java applets based on user choices in the HTML document.
- Preprocessing data on the client before submission to a server.
- Changing content ant styles in modern browsers dynamically and instantly in response to user interaction.

MySQL:

MySQL is a Structured Query Language server designed for heavy loads and processing of complex queries. As s relational database system, MySQL allows many different tables to be joined together for maximum efficiency and speed.

The most popular features of this program are as follows:

- Multiple CPUs usable through kernel threads.
- Multi-platform operation.
- Numerous column types cover virtually every type of data.
- Group functions for mathematical calculations and sorting.
- Commands that allow information about the databases to be easily and succinctly shown to the administrator.
- Function names that do not affect table or column names.
- A password and user verification system for added security.
- Up to 32 indices per table permitted; this feature has been successfully implemented at levels of 60,000 tables and 5,000,000,000 rows.

- International error reporting usable in many different countries.

FLEX:

Flex is a highly productive, free, open source framework for building expressive web applications that deploy consistently across browsers, desktops, and operating systems by leveraging the Adobe® Flash® Player and Adobe AIR® runtimes. While Flex applications can be built using only the Flex framework,

Adobe Flash Builder™ (formerly Adobe Flex® Builder™) software can accelerate development through features like intelligent coding, interactive step-through debugging, and visual design of the user interface layout.

Flex is the way to make rich Internet applications (RIAs) quickly and easily. At its basic level, it's a framework for creating RIAs based on Flash Player. Along with being a framework, Flex is also a new language. At its heart is MXML, a markup language based on Extensible Markup Language (XML) that makes it really easy and efficient to create applications.

Unlike developing for some desktop platforms requiring a proprietary binary file format, MXML is just text, so it's easy to read and modify using just a text editor. Therefore, sharing code is as easy as sharing a simple text file.

MXML is an XML language that you use to lay out user interface components for Adobe® Flex® applications. You also use MXML to declaratively define nonvisual aspects of an application, such as access to server-side data sources and data bindings between user interface components and server-side data sources.

MXML development is based on the same iterative process used for other types of web application files such as HTML, Java Server Pages (JSP), Active Server Pages (ASP), and ColdFusion Markup Language (CFML). Developing a useful Flex application is as easy as opening your favorite text editor, typing some XML tags, saving the file, requesting the file's URL in a web browser, and then repeating the same process.

In the Flex model-view design pattern, user interface components represent the view. The MXML language supports two types of user interface components: controls and containers. Controls are form elements, such as buttons, text fields, and list boxes. Containers are rectangular regions of the screen that contain controls and other containers.

Remote-procedure-call (RPC) services let your application interact with remote servers to provide data to your applications, or for your application to send data to a server. Flex is designed to interact with several types of RPC services that provide access to local and remote server-side logic.

The MXML components that provide data access are called RPC components. MXML includes the following types of RPC components:

- WebService provides access to SOAP-based web services.
- HTTPService provides access to HTTP URLs that return data.
- RemoteObject provides access to Java objects using the AMF protocol (Adobe Lifecycle Data Services ES only).

WAMP SERVER:

WAMP5 (WAMP means Windows Apache MySQL PHP) is a platform of Web development under Windows. WAMP is a form of mini-server that can run on almost any Windows Operating System. WAMP includes Apache 2, PHP 5 (SMTP ports are disabled), and MySQL (phpMyAdmin and SQLitemanager are installed to manage your databases) preinstalled.

WampServer is a Windows web development environment. It allows you to create web applications with Apache, PHP and the MySQL database. It also comes with PHPMyAdmin and SQLiteManager to easily manage your databases.

An icon on the taskbar tray displays the status of WAMP, letting you know if; a) WAMP is running but no services are opened (the icon will appear red), b) WAMP is running and one service is opened (the icon will appear yellow) or c) WAMP is running with all services opened (the icon will appear white). Apache and MySQL are considered to be services (they can be disabled by left-clicking on the taskbar icon, guiding your cursor over the service you wish to disable and selecting "Stop Service").

The files/web pages that are hosted on your WAMP server can be accessed by typing *http://localhost/* or *http://127.0.0.1/* in the address bar of your web browser. WAMP must be running in order to access either of the above addresses.

SMSC and XML API:

An SMS center (SMSC) is responsible for handling the SMS operations of a wireless network. When an SMS message is sent from a mobile phone, it will reach an SMS center first.

The SMS center then forwards the SMS message towards the destination. An SMS message may need to pass through more than one network entity (e.g. SMSC and SMS gateway) before reaching the destination.

The main duty of an SMSC is to route SMS messages and regulate the process. If the recipient is unavailable (for example, when the mobile phone is switched off), the SMSC will store the SMS message. It will forward the SMS message when the recipient is available.

Long-Code services: ValueFirst

- ValueFirst Provides unique long code/ virtual number service that lets two-way global communications through SMS. With long-code services, enterprises are empowered with two-way messaging /SMS capabilities using the familiar 10 digit mobile number.

- ValueFirst Pace is a store and forward mechanism, using a middleware, deployed on Internet for sending and receiving SMS through the API endpoint(s) to the clients.

- ValueFirst provides end-to-end mobile data services from back-end integration to application hosting to dedicated connectivity with the SMSC's of multiple GSM/CDMA operators in host of countries

CDMA – CODE DIVISION MULTIPLE ACCESS

CDMA is most broadly used by the protocol IS-95, also known by its "brand" name cdmaOne, as a cellular phone technology. CDMA is a form of multiplexing, allowing numerous signals to occupy a single transmission channel. It uses a spread spectrum approach for digitally transmitting data or voice over radio frequencies. Sounds are digitized and then the information is broken into data packets that are encoded with a unique identification code. All of the packets are sent over a spread range of radio frequencies.

The cell phone or data device on the other end receives all of the data packets, reassembles those packets with the correct code for your conversation or connection, and then converts the broken bits of data into useful sound and information.

This allows innumerable amounts of calls to occur simultaneously and a bigger amount of traffic for a finite number of available frequencies than analog or the other digital standards. Not only does CDMA's spread spectrum technology ensure that there will be no interference, it also makes the signals difficult to detect and jam. For this reason the military uses CDMA transmission for secure phone calls.

CHAPTER-4
SYSTEM DESIGN

System design is the most creative and challenging phase in the life cycle of system development. The requirements of the proposed are detailed by use of use cases and data flow diagrams.

4.1 DIAGRAMS

The sections below describes are the diagrams used in this system.

4.1.1 OVERALL ARCHITECTURE:

The user can enable to search and register their details through the mobile. Then the details will send and receive to the SMS System via SMS Provider for XML API. The reply will be send to the corresponding users mobile via the SMS Provider.

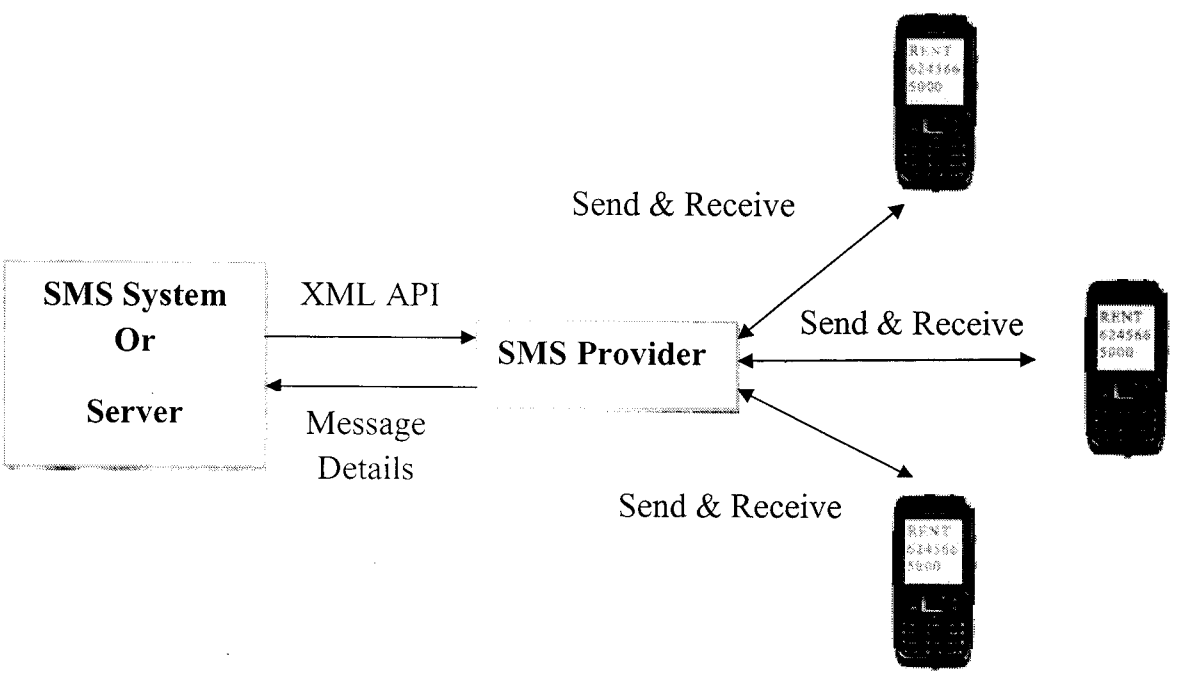


Figure 4.1.1.1: Overall Architecture

4.1.2 USE-CASE DIAGRAM

The **use-case diagram** is used to present a graphical overview of the functionality provided by a system in terms of actors, their goals, represented as use cases and any dependencies between those use cases.

Actors

This system will be used by the following actors:

1. Admin
2. Agent
3. Member
4. Customer

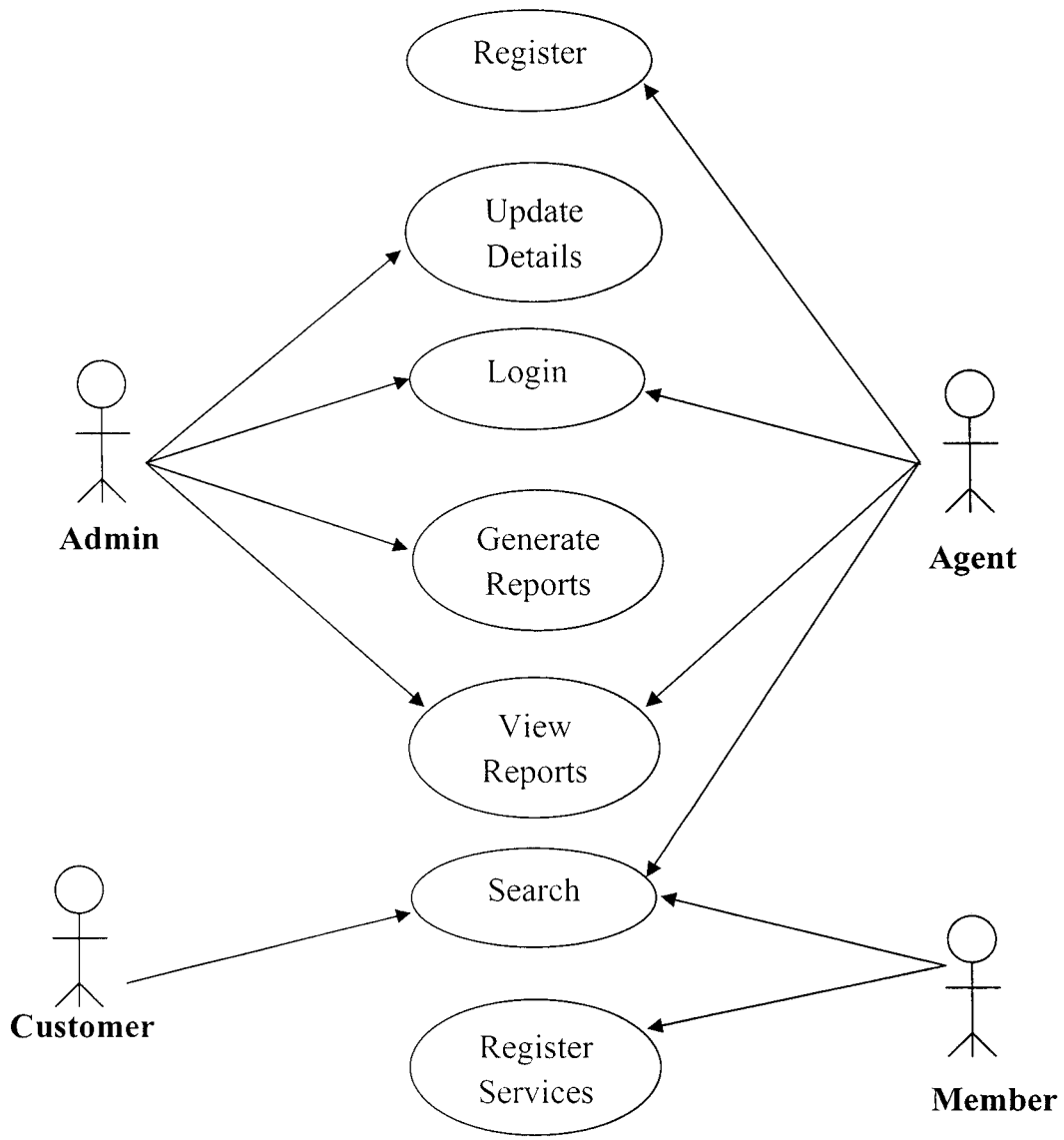


Figure 4.1.1.2: Use-Case Diagram

4.1.3 DATA FLOW DIAGRAM

The Data Flow Diagram (DFD) is a tool used for structured design. DFD shows the flow of data from external entities into the system, how the data move from one process to another as well as its logical storage.

A Data Flow Diagram is a process-oriented graphical representation of an application system. It is a picture of the movement of the data between external entities and the processes and data stores within a system.

Benefits of Data Flow Diagram:

- ✓ Provides a pictorial, non-technical representation.
- ✓ Easy to understand.
- ✓ Uses limited number of symbols with specific meanings.
- ✓ Uses a simple top down expansion.

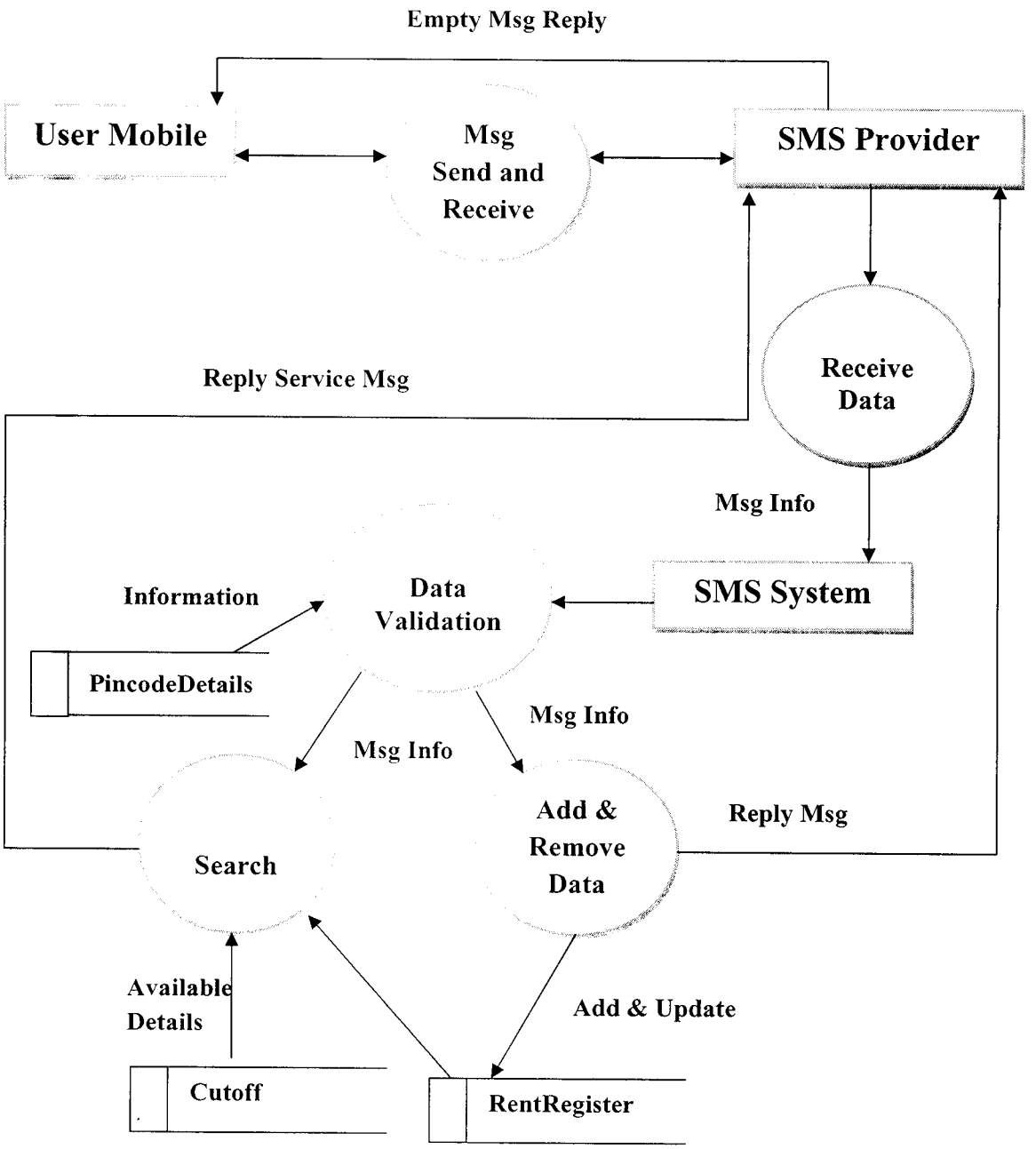


Figure 4.1.1.3: Data Flow Diagram

4.2 ELEMENTS OF DESIGN

System Design is the most creative and challenging phase in the development of a software system. Design implies to a description of the final system and the process by which it is developed. The first step is to determine what input data is needed for the system and then to design a database that will meet the requirements of the proposed system. The next step is to determine what outputs are needed from the system and the format of the output to be produced.

During the design of the proposed system some areas where attention is required are:

- What are the inputs required and the outputs produced?
- How should the data be organized?
- What will be the processes involved in the system?
- How should the screen look?

The steps carried out in the design phase are as follows:

- ✓ Input Design
- ✓ Output Design
- ✓ Database Design

4.2.1 INPUT DESIGN

Input design is a process of converting the user-originated inputs into the computer-based format. The goal of designing input data is to make the automation as easy and free from errors as possible. The requirements of input design such as user friendliness, consistent format and interactive dialogues for giving the right message and help for the user at right time are also considered for the development of the project.

The following are the features of the input design:

- Input design mainly includes options and menus. It helps the user to choose the option.
- The same format is used with related screens; users can identify easily where the selections are made.
- The consistent terminology is used which coordinate forms and screen designs.
- The screen is not over crowded. Often too neat, and pleasing. It facilitates the user to identify the labels easily and enter the data.
- Hence the input design will be easy to follow and does not induce errors. The screen designs are viewed in appendices.

4.2.2 OUTPUT DESIGN

The Output is mainly used to communicate with the user, processing the input data given by the user. It is documented in each stage of the project to ensure error free output. The output screens are designed in a very simple and easy to understand format. All user options are presented in well-formatted forms. The user friendliness of the output screen makes it very easy to use. The quality refers to the way by which the output is presented to the user.

The output design of the report is the screens that display the skill gap based on the house registration and search details. The purpose of these reports is admin can easily know about the rent registration for particular date, area code & amount, advertisement and cutoff details. And agent also can view these reports.

4.2.3 DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. During database design analyst studies the data to determine which data will require store on external storage device. Here, the analyst is responsible for designing a control system to prevent tampering with the data stored by the computer.

The details about the relevant data for the system are first identified. According to their relationship, tables are designed through following method.

- The data type for each data item is decided.
- The tables are then normalized.

TABLE STRUCTURE:

4.2.3.1 Table Name: Pincode

Description: This table contains details about pin code.

Fields	Data type	Size	Description	Constraint
Id	Int	30	Pin code ID	Primary Key
Place	Varchar	50	Place Name	Not null
Pincode	Int	20	Pin code	Not null
District	Varchar	50	District	Not null
State	Varchar	50	State	Not null

4.2.3.2 Table Name: Advertisement

Description: This table contains details about the advertisement.

Fields	Data type	Size	Description	Constraint
Aid	Int	20	Advertisement ID	Primary Key
Aname	Varchar	30	Advertisement Name	Not null
Amsg	Varchar	100	Advertisement Message	Not null
Akeyword	Varchar	20	Keyword	Not null
Acount	Int	20	Count	Not null

4.2.3.3 Table Name: Agent_regis

Description: This table has details about the agent.

Fields	Data type	Size	Description	Constraint
Agent_id	Varchar	30	Agent ID	Primary key
Agent_name	Varchar	50	Agent Name	Not null
Agent_dob	Date		Agent Date of Birth	Not null
Agent_sex	Varchar	30	Agent Sex	Not null
Agent_qual	Varchar	50	Agent Qualification	Not null
Agent_occup	Varchar	50	Agent Occupation	Not null
Agent_mstatus	Varchar	30	Marital Status	Not null
Agent_address	Varchar	100	Agent Address	Not null
Agent_mobileno	Varchar	12	Mobile Number	Not null
Agent_mailid	Varchar	50	Agent Mail ID	Not null
Agent_password	Varchar	20	Agent Password	Not null

4.2.3.4 Table Name: Rentenq

Description: This table contains the details about house and customers request and response.

Fields	Data type	Size	Description	Constraint
ID	Int	20	User ID	Primary key
Mobileno	Varchar	12	Mobile Number	Not null
Pincode	Varchar	20	Pin code	Foreign key referenced from Pincode.
You	Varchar	50	User Type	Not null
Place	Varchar	50	Place	Not null
Amt	Int	50	Amount	Not null
Datesent	DateTime		Date of request	Not null
Givendate	DateTime		Date of reply	Not null
Status	Int	20	Status	Not null
Area	Varchar	200	Area	
Details	Varchar	500	Detailed Message	
Count	Int	11	Count	Not null
Qtn	Int	11	Question Number	Not null
Changeval	Int	11	Change Value	Not null

4.2.3.5 Table Name: Cutoff

Description: This table contains the cutoff details.

Fields	Data type	Size	Description	Constraint
Cutid	Int	20	Cutoff ID	Primary key
District	Varchar	50	District of the College	Not null
College	Varchar	50	College Name	Not null
Course	Varchar	50	Course	Not null
Ocmincut	Double		OC Minimum Cutoff	Not null
Ocmaxrank	Double		OC Maximum Rank	Not null
Bcmmincut	Double		BCM Minimum Cutoff	Not null
Bcmmaxrank	Double		BCM Maximum Rank	Not null
Bcmincut	Double		BC Minimum Cutoff	Not null
Bcmaxrank	Double		BC Maximum Rank	Not null
Mbcmincut	Double		MBC Minimum Cutoff	Not null
Mbcmaxrank	Double		MBC Maximum Rank	Not null
Scmincut	Double		SC Minimum Cutoff	Not null
Scmaxrank	Double		SC Maximum Rank	Not null
Scamincut	Double		SCA Minimum Cutoff	Not null
Scamaxrank	Double		SCA Maximum Rank	Not null
Stmincut	Double		ST Minimum Cutoff	Not null
Stmaxrank	Double		ST Maximum Rank	Not null

4.2.4 MODULAR DESIGN

Modular design — or "modularity in design" — is an approach that subdivides a system into smaller parts (modules) that can be independently created and then used in different systems to drive multiple functionalities. Besides reduction in cost (due to lesser customization, and less learning time), and flexibility in design, modularity offers other benefits such as augmentation (adding new solution by merely plugging in a new module), and exclusion.

This system is also modularized to reduce the complexity of the system. This contains various modules.

A) User Modules:

User modules are modeled based on the users of the system. This module contains three users:

i) Admin:

- administers the entire system.
- can view the details of all advertisement, pin code, cutoff and rent enquiry details.
- add, delete and update the advertisement, pin code and cutoff details.

ii) Agent:

- can view the details of all advertisement, pin code, cutoff and rent enquiry details.
- can collect the amount from customer after the house confirmation.

iii) Member:

- can register their services via SMS.
- No service charges for member registration.

iv) Customer:

- can search the house for rent.
- can search the engineering college based on the cutoff details.

B) Reports Module

Reports forms are the main output of any system. This system too has many reports which will help admin and agent. Reports based on the advertisement name, pin code, cutoff id, dates, etc.

The RENT service contains the following reports for both customer and member,

-Status report

- date wise report
- pin code wise report
- Amount wise report
- pin code with amount wise report and etc.,

The BE service contains the following reports,

- College wise report
- Course wise report
- Cutoff and rank wise report
- District wise report and etc.,

CHAPTER-5
SYSTEM IMPLEMENTATION

Software implementation and testing is the process done in the development stage of the software.

5.1 IMPLEMENTATION

The system is implemented using Zend Studio- Flex builder, PHP and MYSQL.

5.1.1 Implementation of Business Logic

The business logic is implemented using Flex Builder. Flex builder contains many controls which make the website very user friendly. The data are entered using the controls and the output is displayed very effectively. This flex builder is very helpful in the way that it doesn't need to navigate through various pages. The loading of different pages for every function call is omitted which helps in improving performance. This IDE uses only one HTML page.

The controls are made visible and invisible at needed times. The communication between the IDE and Database is through the Scripting language named PHP. The communication is in the form of XML data. This IDE uses an HTTP Object to send the data to the Scripting language. After processing of PHP file, the results are returned to IDE in the XML format which can be displayed using various controls and containers.

5.1.2 Implementation of Database Communication

For database communication, MYSQL is used along with PHP. MYSQL is the most popular Open Source SQL database management system. MYSQL has many inbuilt functions to carry out the operations with database. These functions are used as a part of PHP file. Using these functions the queries are executed and the operations on database are carried out.

The inputs are received from action script file of IDE. The operations are performed according to the command received from IDE. Those operations are performed and the results are returned to the IDE using XML files.

CHAPTER-6
SYSTEM TESTING

6.1 SYSTEM VERIFICATION

System Verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. Verification is ensuring that the product has been built according to the requirements and design specifications- i.e., you built it right.

Verification is the assurance that the products of a particular development phase are consistent with the requirements of that phase and preceding phase(s). In data access, it verifies whether the right data is being accessed in terms of the right place and in the right way.

6.2 SYSTEM VALIDATION

System Validation is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. Validation checks that the product design satisfies or fits the intended usage (high-level checking) — i.e., you built the right product. This is done through dynamic testing and other forms of review. Validation ensures that the product actually meets the user's needs, and that the specifications were correct in the first place.

In this project, validation checks whether the developer is moving towards the right product. Validation coding is written using pre-defined valuators available in Flex 3. Each field in Agent registration form is validated such that the right username, password and mobile no. etc., is added. Any wrong entry display error messages or warnings. The login form is validated such that the valid registered user only can login to new page. Fields such as e-mail id and website are checked for its format. Validation also determines if this project complies with the requirements and performs functions for which it is intended and meets the organization's goal and user needs.

6.3 TESTING

This is to verify that all the system elements have been properly integrated and performs the allocated function. Testing is executing a program to test the logical changes needed in it and with intention of finding errors. Tests are conducted to find discrepancies between the new system and its original objective, current specifications and documents.

6.3.1 UNIT TESTING

In unit testing, we have to test the program making up the system. Unit testing focuses first on the modules independently of one another, to locate errors. This enables within that module alone.

In this System each Components are tested individually to verify that the detailed design for unit has been correctly implemented. Initially the flow of controls and data through that page is checked. In a page, each control is further tested in unit testing. The process is done in all the forms of the system.

6.3.2 INTEGRATION TESTING

Data can be lost across any interfaces. Once module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integrations testing for conducting tests to uncover errors associated within the interface. The objective is to take unit-tested module and build a program structure. The entire module are combined and tested as whole.

Many forms in the system have communication between each other. This helps in testing integration testing. For ex: if the agent logins, first system should check whether he/she is valid user and then it should view the report page. Here more than a single process is involved and so it needs integration testing.

CHAPTER-7
CONCLUSION AND FUTURE ENHANCEMENT

7.1 CONCLUSION

SMS Web Services is developed to help the customers who seek services through SMS. The application developed and designed in such a way that any further enhancements can be done with ease. The system has the capability for easy integration with other systems.

The system has been found to work effectively and will be implemented in short period of time. Proper documentation is provided. The end user can easily understand how the whole system is implemented by going through the documentation. The module developed would be friendly and does demand a little computer knowledge to work.

It is designed to be highly flexible, so that any future modification and requirements can easily be incorporated.

7.2 FUTURE ENHANCEMENT

The system has been designed in such a way that it can be modified with very little effort when such a need arises in the future.

The system is very flexible and changes can be made without difficulty. Further extensions in the system can be made to submit more reports to the management.

"SMS Web services" will be enhanced further with the following features:

- By developing automated multimedia web service users can send an image of the house to the system.
- More reports can be generated, if the need arises.
- Members can pay the registration fees through the SMS.
- Add more services like Train and Library Details.

CHAPTER-8
APPENDIX

8.1 SAMPLE CODING

MXML File:

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute"
backgroundGradientAlphas="[1.0, 1.0]" backgroundGradientColors="[#FBC197, #43FC03]"
width="1305" height="640">
<mx:Style source="style.css"/>
<mx:Script source="agent_registration.as"/>
<mx:Script source="login.as"/>
<mx:Script source="admin_home.as"/>
<mx:Script source="agent_home.as"/>
<mx:Script source="rent.as"/>
<mx:Script source="be.as"/>
<mx:VBox x="88" y="10" width="1107" height="2000" id="vbox">
<mx:Image width="1157" height="106" source="5.jpg"/>

<mx:ApplicationControlBar width="1102" height="37" id="acb">
<mx:LinkBar width="1079" horizontalAlign="center" height="24"
dataProvider="viewstack_home" id="lbar">
</mx:LinkBar>
</mx:ApplicationControlBar>

<mx:ViewStack id="viewstack_home" x="0" y="0" width="1105" height="100%">

<mx:Canvas label="Home" width="100%" height="100%" id="canvas_home">
<mx:Panel x="10" y="0" width="304" height="182" layout="absolute" id="panel_login"
title="User Login">
<mx:Label x="42" y="40" text="Username :" fontSize="16" fontFamily="Times New
Roman" color="#036215"/>
```

```
<mx:Label x="42" y="86" text="Password :" fontSize="16" fontFamily="Times New
Roman" color="#036215"/>
<mx:TextInput x="136" y="39" id="txt_uname" width="146" color="#036215"
fontSize="14" fontFamily="Times New Roman" height="26" fontWeight="bold"/>
<mx:TextInput x="134" y="89" width="148" height="26" fontSize="14" fontFamily="Times
New Roman" color="#036215" fontWeight="bold" id="txt_pwd" displayAsPassword="true"
keyDown="k(event)"/>
<mx:Button x="55" y="137" label="Login" id="but_login" click="log()"/>
<mx:Text x="163" y="142" text="Agent Registration" color="#FFFFFF"
fontFamily="Times New Roman" fontSize="14" click="registration()"/>
</mx:Panel>
<mx:Panel x="10" y="190" width="304" height="236" layout="absolute" title="Current
Update">
<mx:Text width="281" height="161" fontSize="14" fontFamily="Georgia"
color="#FFFFFF" y="40" x="10" fontStyle="italic" fontWeight="normal">
<mx:htmlText>
<![CDATA[
Easy to get Engg. College details based on the Cutoff Marks.<br>
Easy to get Train Details.
]]>
</mx:htmlText>
</mx:Text>
</mx:Panel>
<mx:Panel x="322" y="0" width="773" height="426" layout="absolute"
title="Superstarsms.com Advantages" fontSize="16" fontFamily="Arial">
<mx:Text width="640" height="305" fontSize="14" fontFamily="Verdana"
color="#EFF5EE" y="54" x="47" fontWeight="bold" fontStyle="normal">
<mx:htmlText>
<![CDATA[
Saves a lot of time.
The user can request services at any time and any where through the SMS.
Efficient retrieval of information.
Helps in easier broadcasting of existing services such as house for rent.
Helps in publishing Advertisement through SMS when response to the user.
```

Easy to get Engg. College details based on the Cutoff Marks.

Easy to get Train Details.

]]>

</mx:htmlText>

</mx:Text>

</mx:Panel>

<mx:Panel x="10" y="434" width="1085" height="21" layout="absolute">

<mx:Text x="333" y="0" text="All rights reserved by superstarsms.com@2010" width="370" color="#FFFFFF" fontWeight="bold" fontSize="12" fontFamily="Verdana" fontStyle="italic" height="30"/>

</mx:Panel>

</mx:Canvas>

<mx:Canvas label="Services" width="100%" height="100%">

<mx:Panel x="10" y="0" width="1085" height="435" layout="absolute" title="Service Details" fontSize="14">

<mx:Text x="408" y="28" text="Services are RENT and BE" fontSize="20" color="#0DBADF"/>

<mx:Text x="469" y="94" text="Message Format" width="147" height="27" fontWeight="bold" fontStyle="italic" color="#079528"/>

<mx:Text x="310" y="129" width="700" height="279" color="#F7E8F4">

<mx:htmlText>

<![CDATA[

You want House send "REND" to 9282121250

House Owners send "TOLET" to 9282121250

Students send "BE", Want to know top 10 Colleges send "TOP" to 9282121250

Want to become agent send "AGEND" 9282121250

]]>

</mx:htmlText>

</mx:Text>

<mx:Image x="425" y="94" source="phone.gif"/>

</mx:Panel>

</mx:Canvas>

```
<mx:Canvas label="Help" width="100%" height="100%">
<mx:Panel x="10" y="0" width="1085" height="435" layout="absolute" title="Help">
<mx:Text x="496.5" y="36" text="Help" color="#0DBADF" fontSize="20"
fontWeight="bold"/>
<mx:Text x="351.5" y="75" width="382" height="279" color="#F7E8F4" fontSize="14"
fontFamily="Georgia" fontWeight="bold">
<mx:htmlText>
<![CDATA[
Send Message <i>"HELP"</i> to 9282121250
Send Message <i>"HELP RENT"</i> to 9282121250
Send Message <i>"HELP BE"</i> to 9282121250
Send Message <i>"HELP TOLET"</i> to 9282121250
]]>
</mx:htmlText>
</mx:Text>
</mx:Panel>
</mx:Canvas>
</mx:ViewStack>
</mx:VBox>
</mx:Application>
```

ActionScript file:

```
import flash.events.KeyboardEvent;
import mx.controls.Alert;
import mx.rpc.events.ResultEvent;
import mx.rpc.http.HTTPService;

public function k(eee:KeyboardEvent):void
{
        if(eee.keyCode==13)
        log();

}
```

```
public function log():void
{
        ardg.visible=false;
        ardg.includeInLayout=false;
        var login:HTTPService=new HTTPService();
        login.method="POST";
        login.url="http://localhost/WS/login.php";
        login.useProxy=false;
        login.resultFormat="e4x";
        login.addEventListener("result",loginresult);
        var validentry1:Boolean=!(txt_uname.text=="");
        var validentry2:Boolean=!(txt_pwd.text=="");
        if(validentry1 && validentry2)
                {
                        var objsend:Object=new Object;
                        objsend.txt_uname=txt_uname.text;
                        objsend.txt_pwd=txt_pwd.text;
                        login.send(objsend);
                }
                else if(validentry1)
                {
                        Alert.show("Password is also required");
                        txt_uname.text="";
                }
                else if(validentry2)
                {
                        Alert.show("Login ID is must");
                        txt_pwd.text="";
                }
                else
                        Alert.show("Both Fields are required");
}
```

```
public function loginresult(e2:ResultEvent):void
{
        var s1:String=new String();
        s1=String(e2.result);
        if (s1=="Admin")
        {
                adminhome();
        }
        else if(s1=="Agent")
        {
                ardg.visible=false;
                ardg.includeInLayout=false;
                agenthome();
        }
        else
        {
                Alert.show(s1);
                txt_uname.text="";
                txt_pwd.text="";
        }
}
```

PHP File:

```
<?php
$mysql_connection=mysql_connect("localhost","root","");
mysql_select_db("rent_db");

$uname=$_POST['txt_uname'];
$pwd=$_POST['txt_pwd'];

if($uname=="admin" && $pwd=="admin")
{
	echo "Admin";
}
else
{
	$query = "SELECT * FROM agent_regis WHERE agent_id='$uname' AND
	agent_password='$pwd'";
	$result = mysql_fetch_array(mysql_query($query));
	if($result)
	{
		echo "Agent";
	}
	else
	{
		echo "Enter Correct Username and password";
	}
}
?>
```

8.2 SCREEN SHOTS

Login Form:

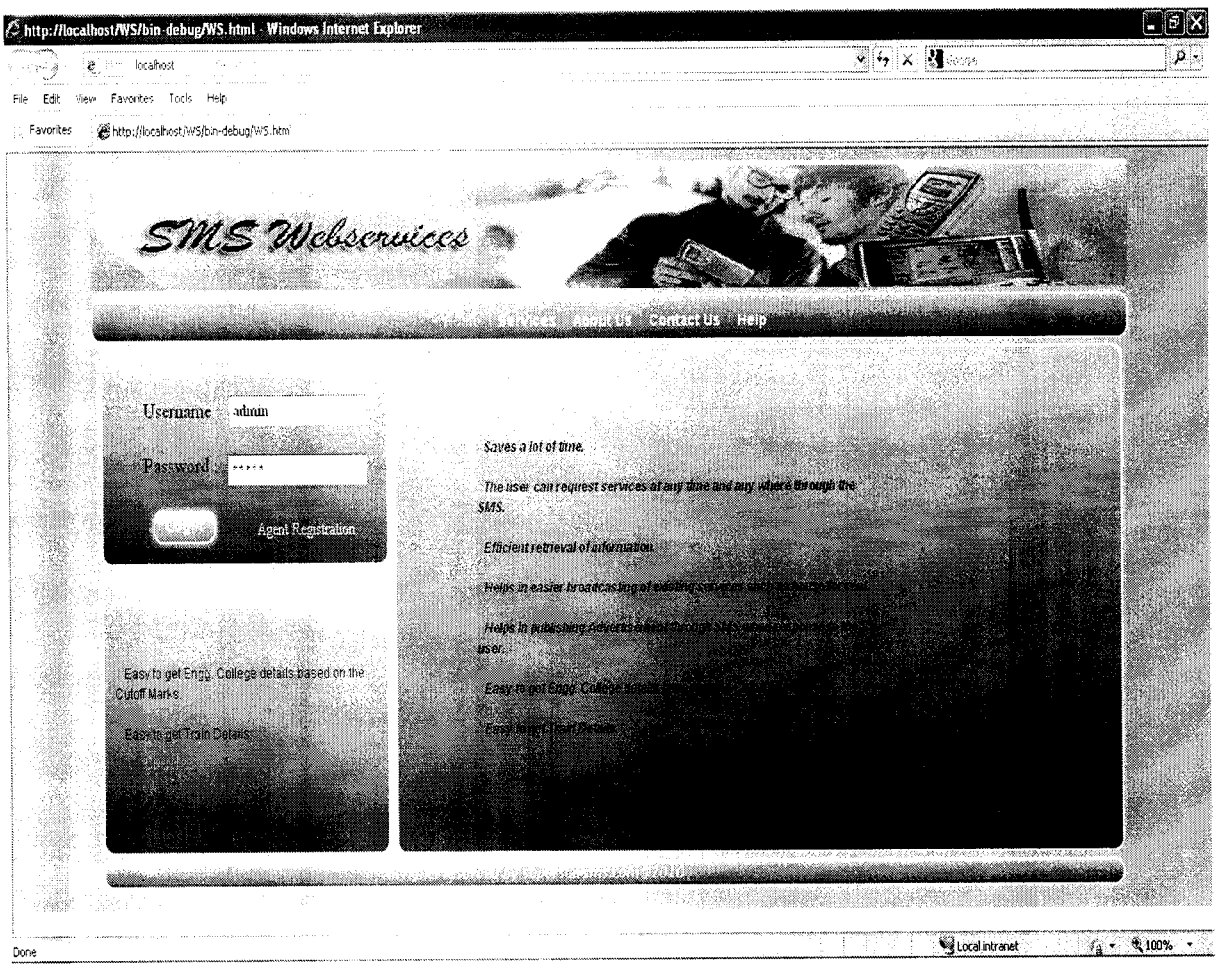


Figure 8.2.1

Agent Registration Form:

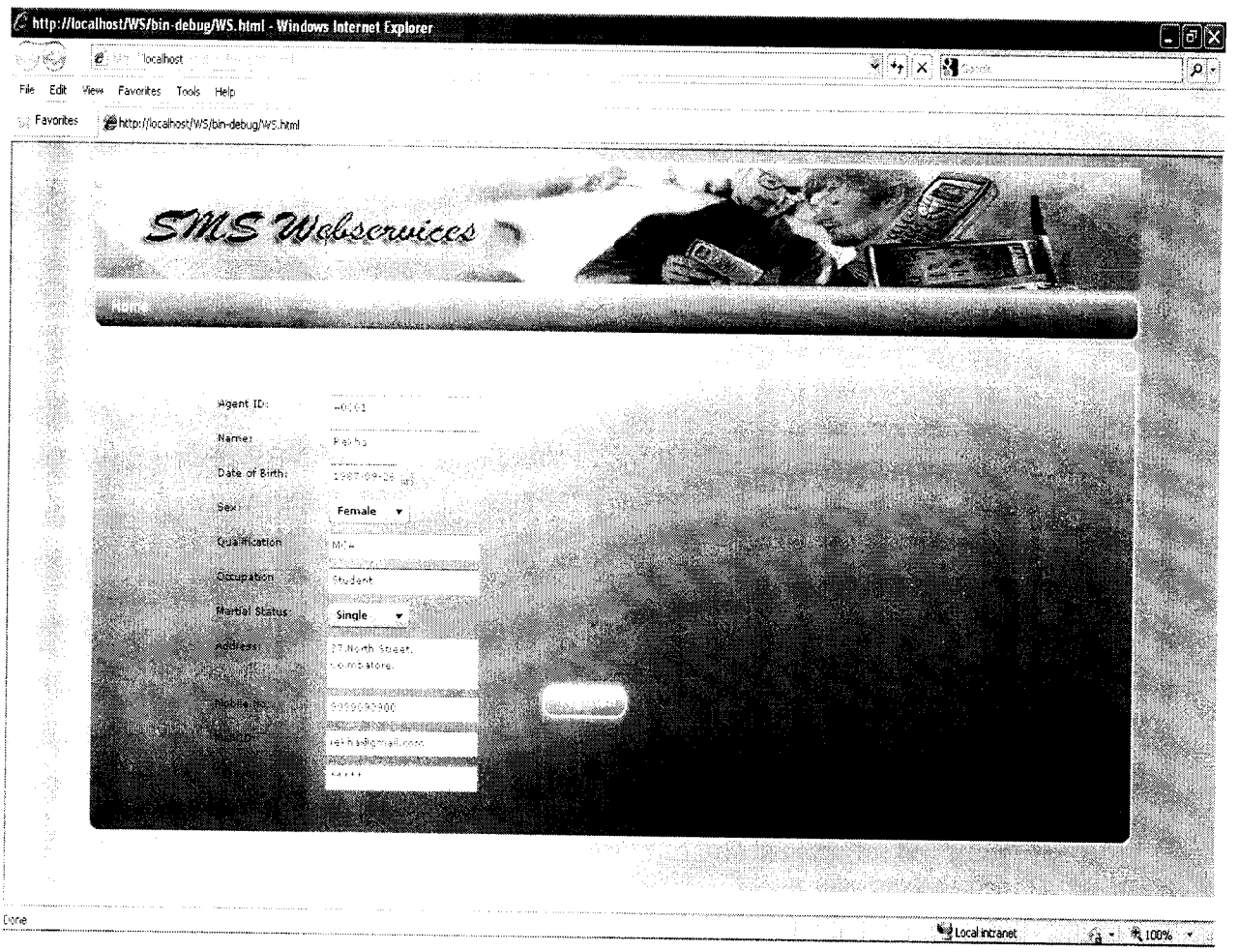


Figure 8.2.2

Agent Profile Page:

Figure 8.2.3

Agent Report:

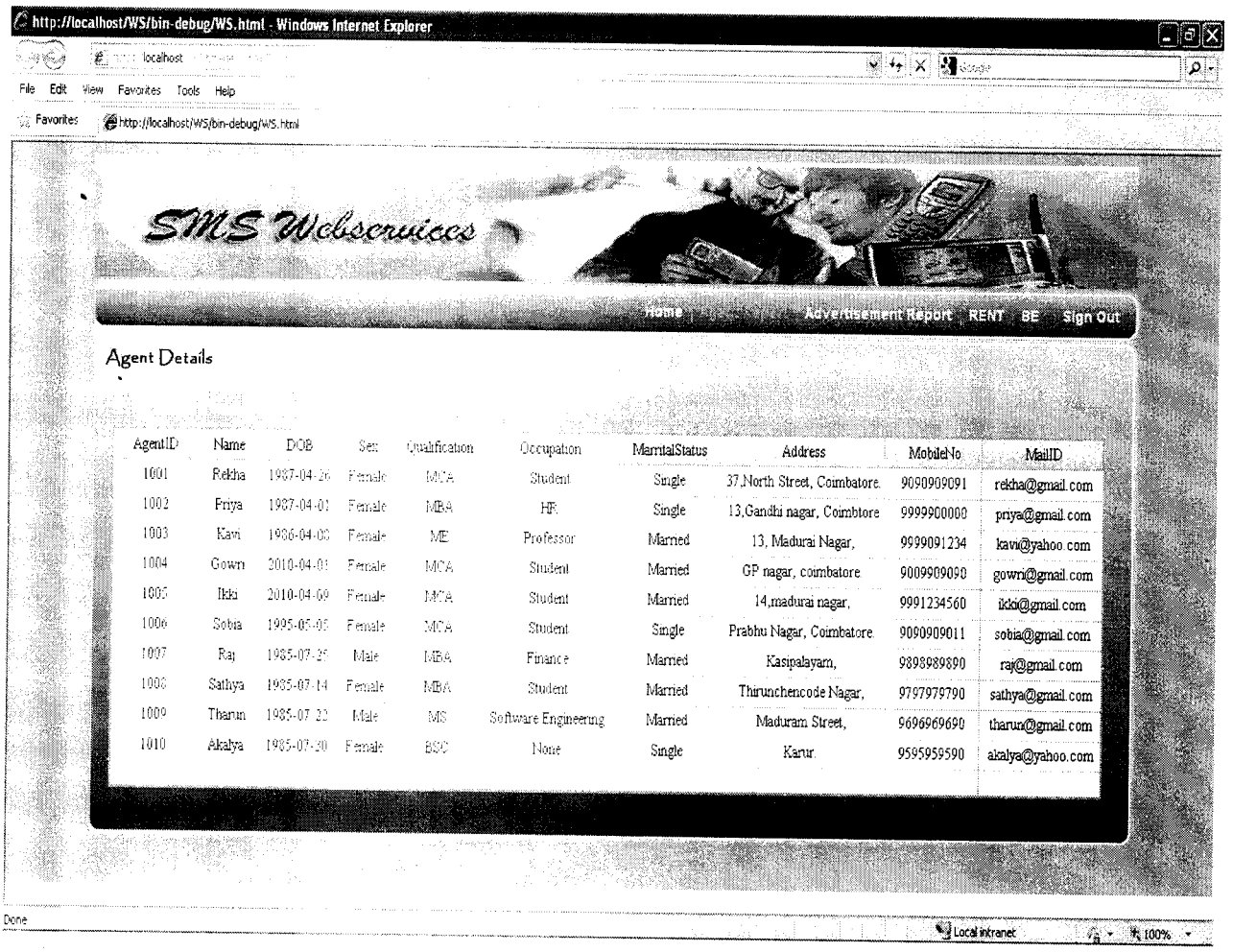


Figure 8.2.4

Advertisement Report:

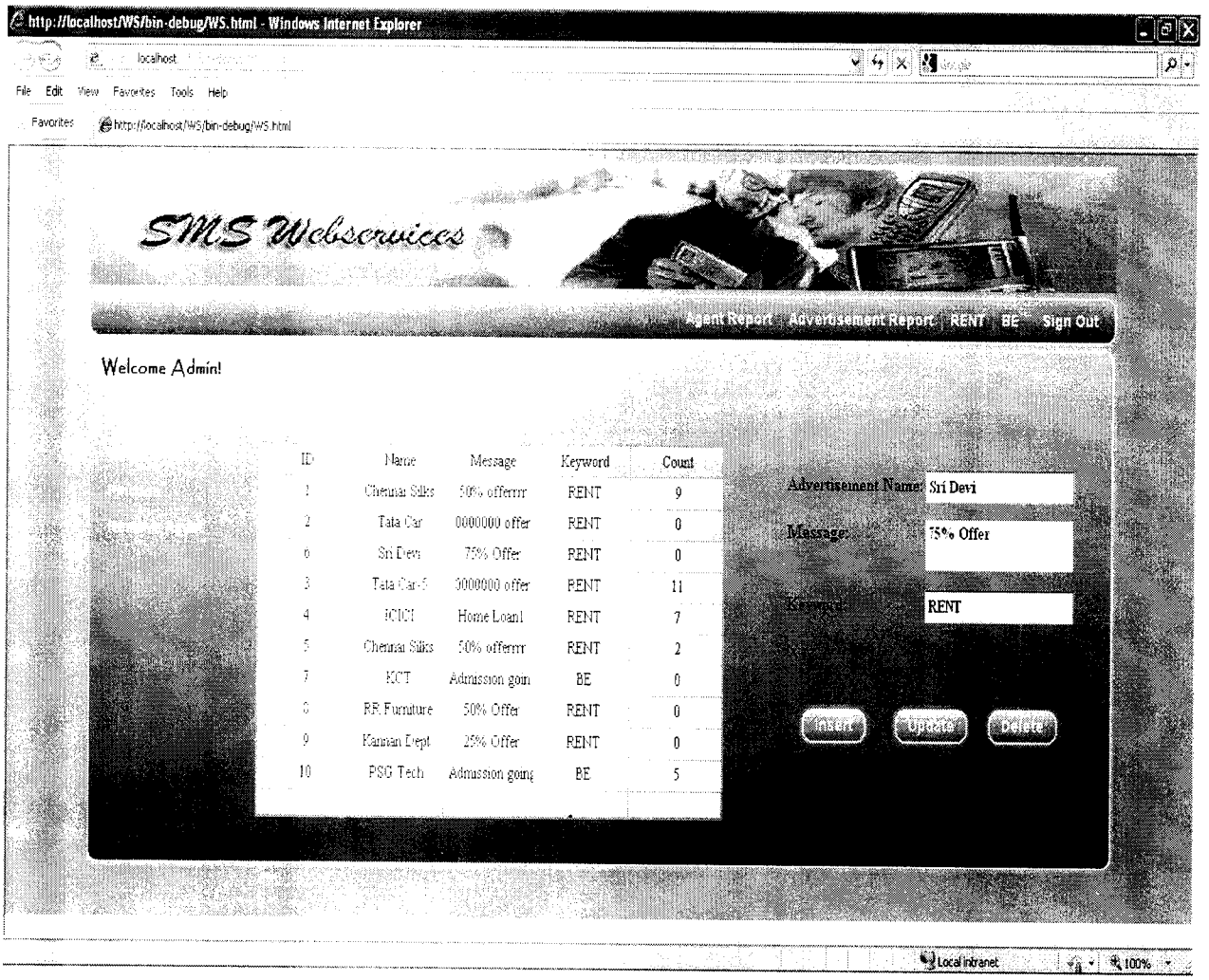


Figure 8.2.5

User Report:

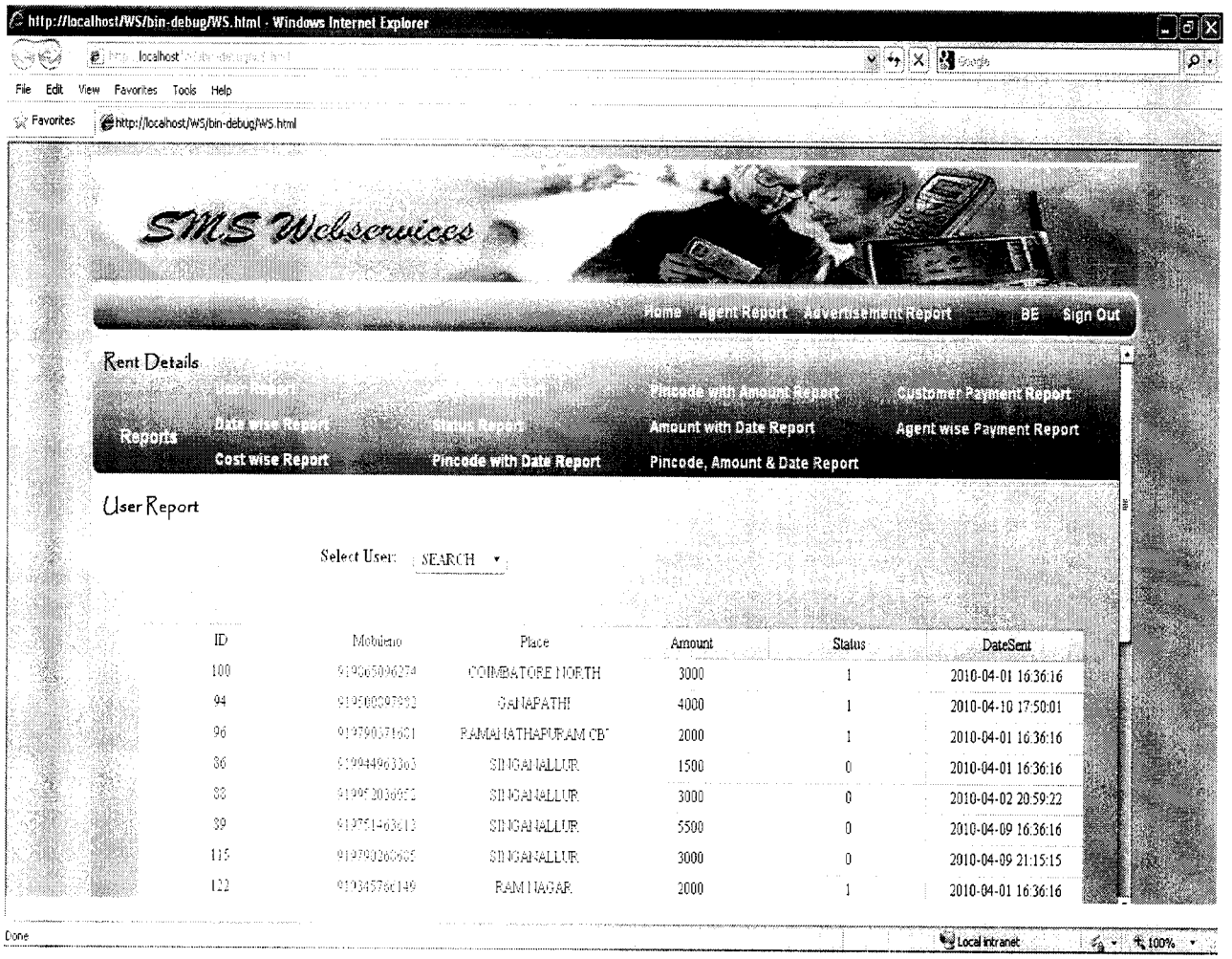


Figure 8.2.6

Date wise Report:

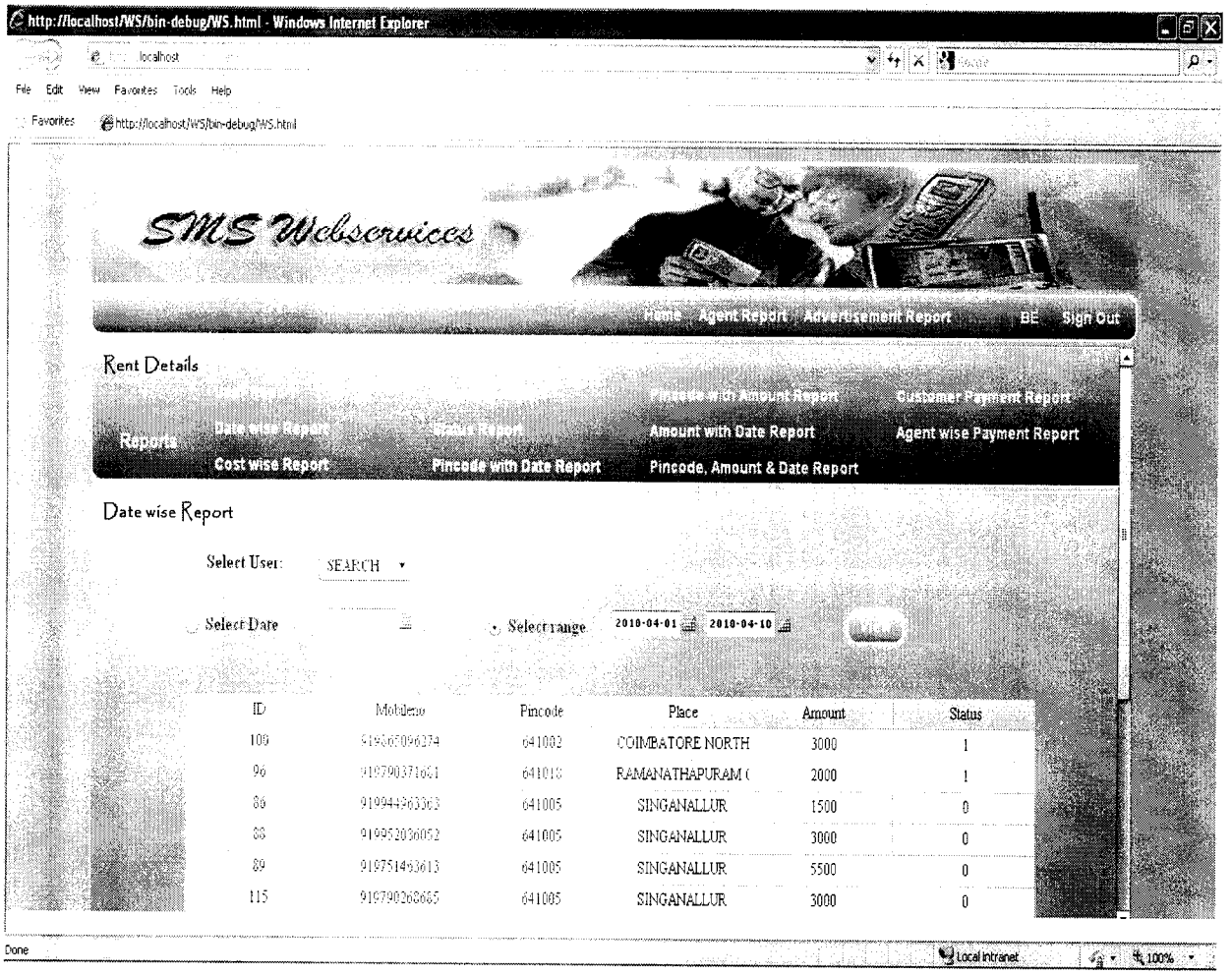


Figure 8.2.7

Payment Report:

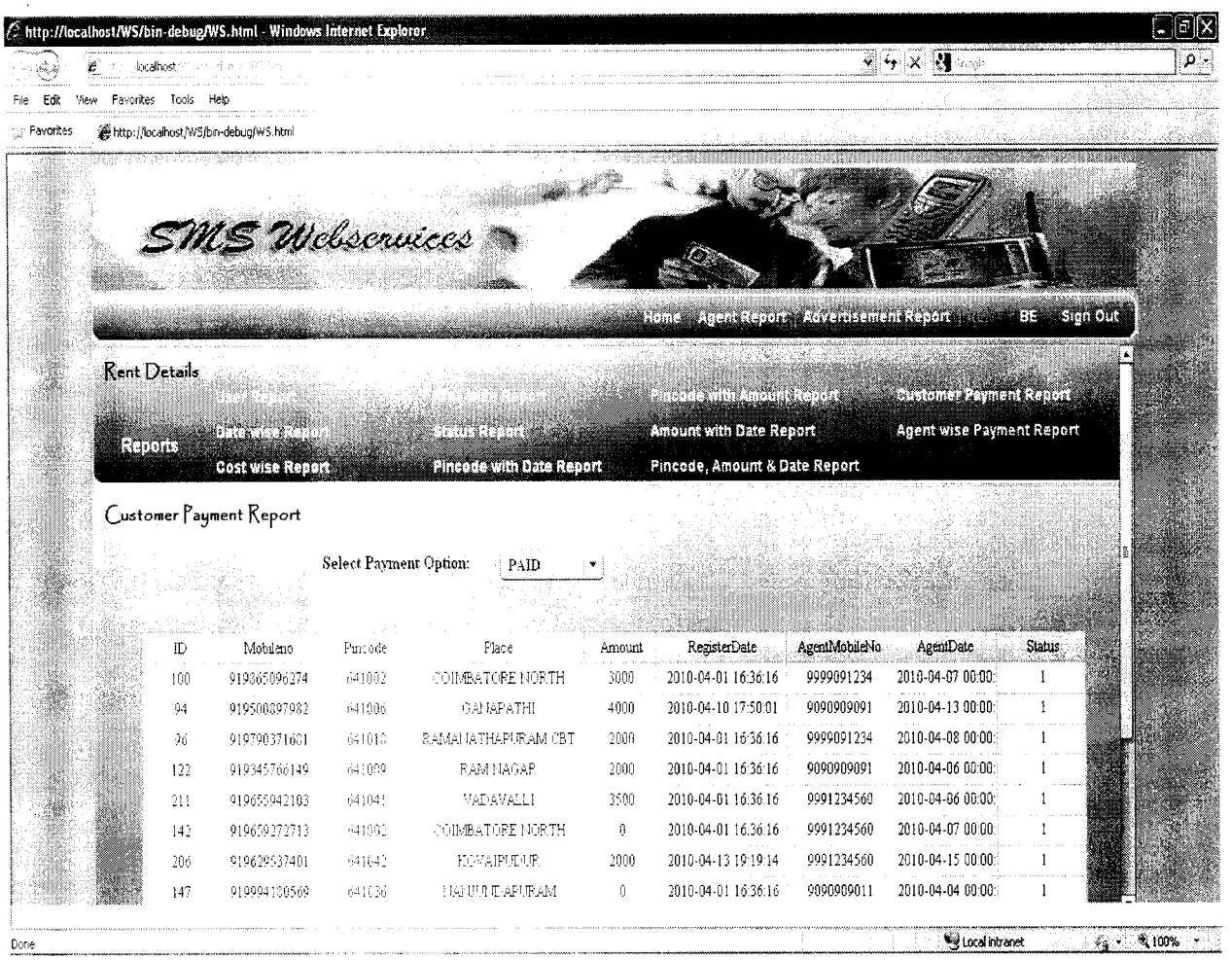


Figure 8.2.8

College Details:

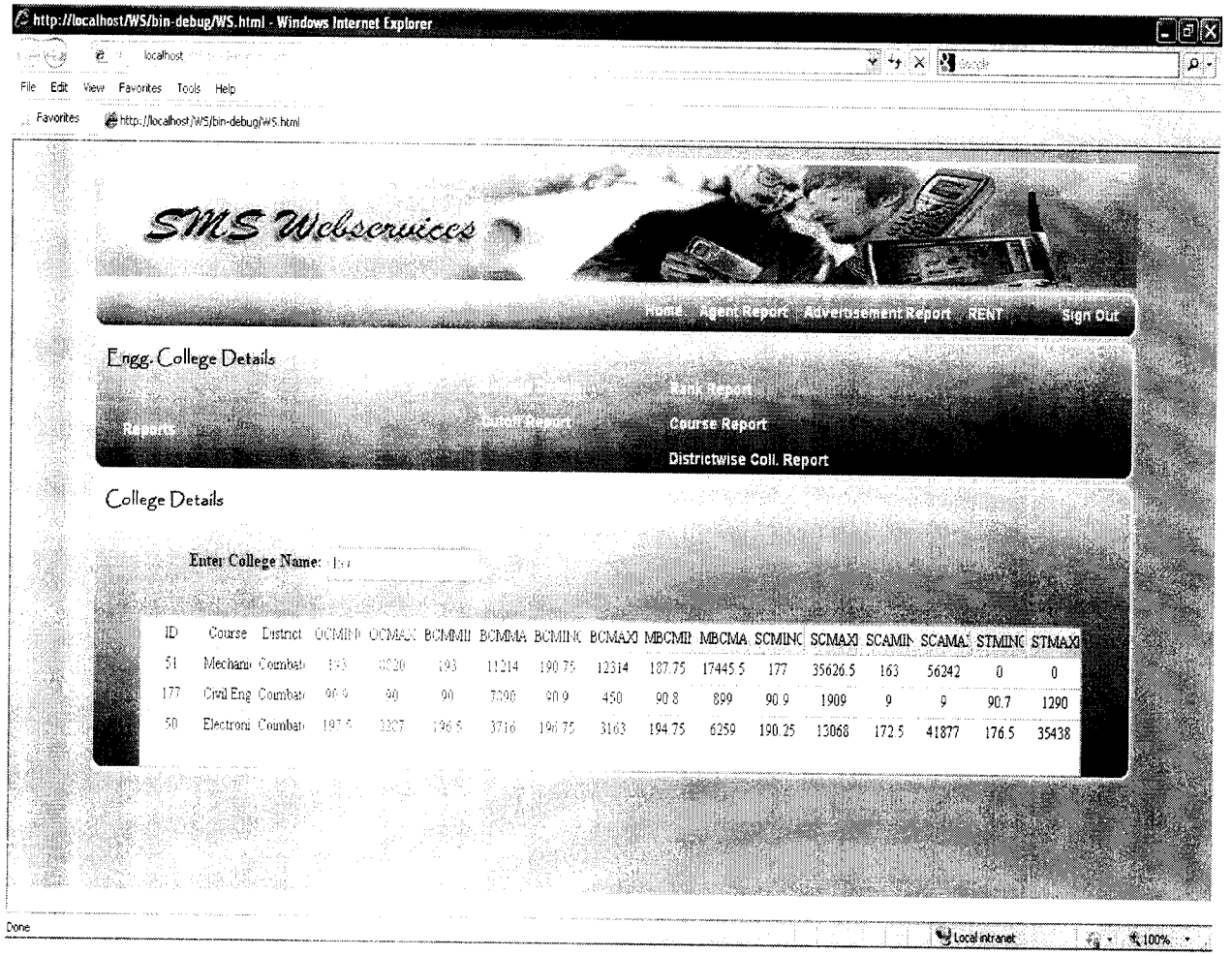


Figure 8.2.9

Course wise Report:

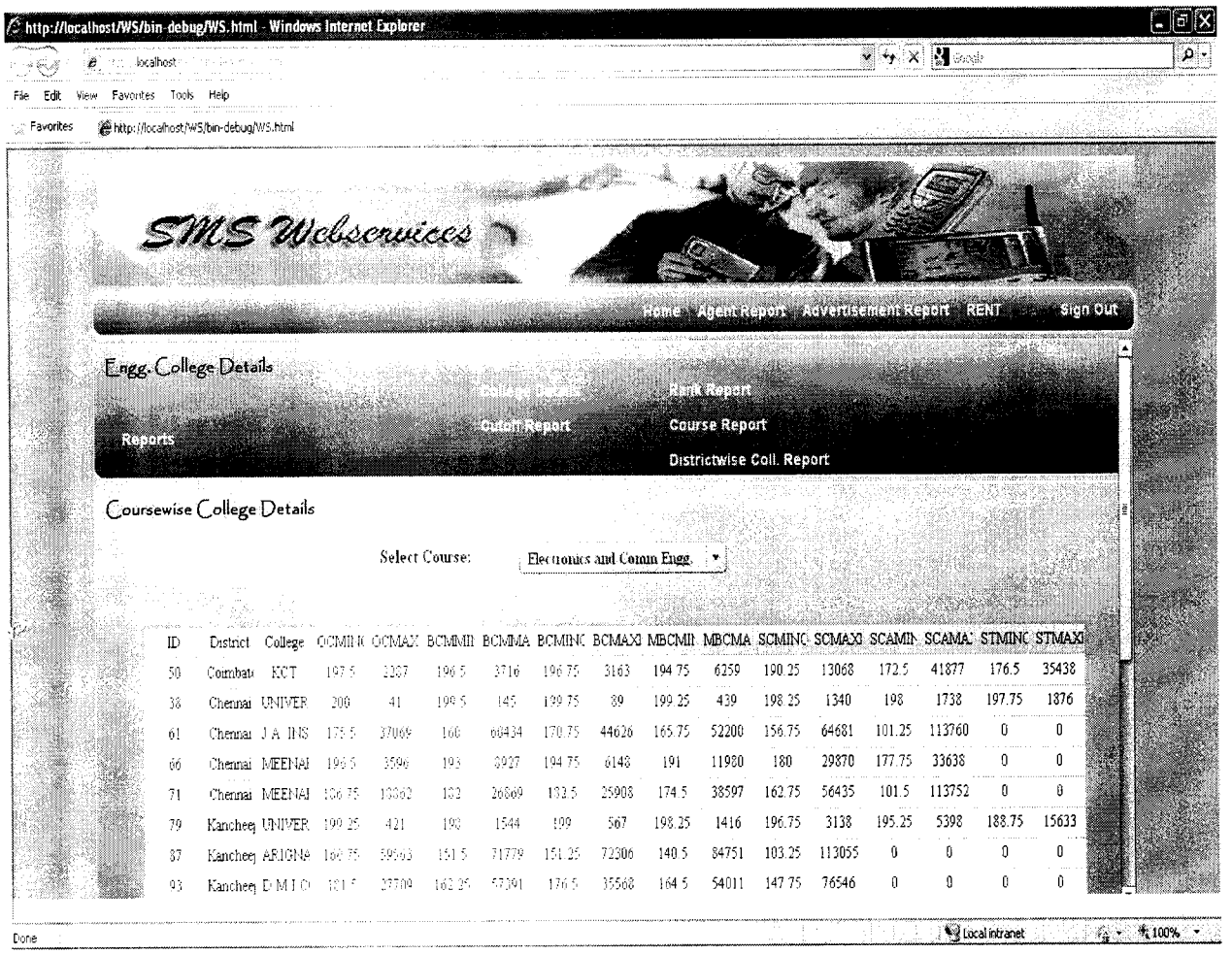


Figure 8.2.10

CHAPTER-9
REFERENCES

BOOK REFERENCES

1. **Learning Flex 3 – Alaric Cole -** O`Reilly Media, Inc., June 2008.

2. **Adobe® Flex® 3 Developer Guide** – Adobe Systems, 2008.

3. **PHP 6/MySQL® Programming for the Absolute Beginner – Andy Harris –** Course Technology, 2009.

WEB REFERENCES

1. **http://livedocs.adobe.com/flex/3**

2. **http://www.w3schools.com/PHP**

3. **http://devzone.zend.com**

4. **http://www.php.net/manual**

5. **http://www.actionscript.org**