# GOM TRADE

## MAIN PROJECT REPORT

*Submitted by*

**ASHA MARIA ARICATT**

**HARIKRISHNAN T**

**KRIPA JOHN**

**VINEETH JOHN**

*in partial fulfillment for the award of the degree*
*of*

**BACHELOR OF TECHNOLOGY (B.TECH)**

in

**COMPUTER SCIENCE & ENGINEERING**

of

**UNIVERSITY OF CALICUT**

Under the guidance of

**Ms SEENIA FRANCIS**

JUNE & 2011

**Department of Computer Science & Engineering**
**JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY**
THRISSUR 679 531

**Department of Computer Science & Engineering**
**JYOTHI ENGINEERING COLLEGE, CHERUTHURUTHY**
**THRISSUR 679 531**

JUNE & 2011

## BONAFIDE CERTIFICATE

Certified that this project report **“ . . . GOM TRADE . . . ”** being submitted in partial fulfillment of the requirements for the award of degree of **Bachelor of Technology** of **University of Calicut** is the bonafide work of **“ . . . ASHA MARIA ARICATT, HARIKRISHNAN T, KRIPA JOHN, VINEETH JOHN . . . ”**, who carried out the project work under our supervision.

Prof. Muralee Krishnan C
**HOD**
Dept. of CSE

Ms.Seenia Francis
**PROJECT GUIDE**
senior Lecturer
Dept. of CSE

# CONTENTS

# ACKNOWLEDGEMENT

We take this opportunity to express our heartfelt gratitude to all respected personalities who had guided, inspired and helped us in the successful completion of this project.

First and foremost, we express our thanks to **The Lord Almighty** for guiding us in this endeavour and making it a success.

We are thankful to our Principal **Dr. U Lazar John** and the Management for providing us with excellent lab and infrastructure facilities.

Our sincere thanks to the Head of the Department of Computer Science & Engineering, Prof. **Muralee Krishnan C** for his valuable guidance and suggestions.

We would like to express our deepest gratitude to **Ms.Seenia Francis** for her valuable contributions and guidance.

Last but not least, we thank all our teaching and non teaching staffs of Department of Computer Science & Engineering, and also our friends for their immense support and help in all the stages for the development of the project.

# ABSTRACT

Over the last few decades, the average person's interest in the stock market has grown exponentially. This demand coupled with advances in trading technology has opened up the markets so that nowadays nearly anybody can own stocks. Our project GLOBAL ONLINE MOBILE SHARE TRADING SYSTEM (GOM Trade) addresses the following problem: no platform exists allowing investors the ability to trade on international markets from a single location. The main objective of this project is to develop the initial design, simulate, and conduct analysis of an online trading system that provides global market access to any potential investor with an internet connection.

# CHAPTER 1

# Introduction

Wouldn't you love to be a business owner without ever having to show up at work? Imagine if you could sit back, watch your company grow, and collect the dividend checks as the money rolls in! This situation might sound like a pipe dream, but it's closer to reality than you might think. As you've probably guessed, we're talking about owning stocks. This fabulous category of financial instruments is, without a doubt, one of the greatest tools ever invented for building wealth. When you start on your road to financial freedom, you need to have a solid understanding of stocks and how they trade on the stock market. Over the last few decades, the average person's interest in the stock market has grown exponentially. This demand coupled with advances in trading technology has opened up the markets so that nowadays nearly anybody can own stocks.

Online share trading, one of the most promising inventions of the modern world which bring the vast world of share market to the common people and people began to make investments using this facility. Usually the online share trading system will be installed in a pc or laptop. We know that India is the second largest handset market in the world and within 4 years the mobile consumers in India will be around 62 crores. So there is a huge scope for doing online share trading using mobiles. Indian bankers use mobile-based trading service at the Bombay Stock Exchange in Mumbai on September 21, 2010. India has launched stock trading on mobile phones, hoping to capitalize on the country's position as the world's fastest-growing handset market by catering to tech-savvy investors.

The main objective of this project is to implement Global online mobile share trading system (GOM Trade). No platform exists allowing investors the ability to trade on international markets from a single location, so the main goal of this project is to develop the initial design, simulate, and conduct analysis of an online mobile share trading system that provides global market access to any potential investor with an internet connection. To keep our project manageable we limit our access to Indian market and NASDAQ exchanges.

## 1.1 Overview

Asia's oldest stock market, the Bombay Stock Exchange (BSE), started trading in shares on mobile phones on 21 September 2010. One in 10 of the world's mobile phones are sold in India, according to technology research specialists Gartner, and the country adds 15-17 million new mobile subscribers every month. So the mobile trading has great importance and a great future scope. Traditional share trading system concentrates only in a particular market. So the investors do not have the ability to trade on international markets from a single location. Similarly the various global market trends, sector wise analysis etc will not be available directly in the existing system. So the proposed system GOM Trade has the following advantages when compared with the existing system. They are

1) No platform exists allowing investors the ability to trade on international markets from a single location. 2) This online trading system that provides global market access to any potential investor with an internet connection. 3) We can any time detect and realize the variations of the Indian market, Asian market and world market. Usually existing systems does not support this type of applications, they only show the variations of Indian market only. When a user wants to know the current trends of the market they usually check other websites such as www.moneycontrol.com or watch the market television such as cnbc TV 18 or something like that. 4) The proposed system consists of a provision for News Headlines tab, i.e. the various information regarding to the stock market will be displayed on a single click. 5) This application includes a particular provision for offline mode. That means even though the net facility is not available, the user can view his home page and the transaction details. This is not included in existing systems. 6) We can view the performance of various sectors such as banking sector, power sector, infra structure etc on a single click. This feature is not included in currently available technology. The user has to browse other web pages for that information.

Since share market is highly sensitive and confidential, the real time buying and selling operations can only be performed by authorized stock brokers. So the real time buying and selling of shares is not at all practical in this scenario. So what we are doing is we can take the real market fluctuations into the mobile but we cant trade the shares in real time. So we are creating a simulation just for buying and selling of the shares. The remaining part except this will be done in real time.

## 1.2 Progress of project

For our GOM TRADE, ie Global mobile online share trading system, we have completed the mobile application user interface and we have also completed the web server module about 80. The ways to get the quotes about different shares in the market, News about the share market are added in the web server. The database needed to keep the account information about the users is made in the server. The provisons for doing different sharing trading activities like, add script,remove script,buy script etc are established in the server.

## 1.3 Member roles and responsibilities

The organizing function of management deals with devising roles for individuals and assigning responsibility for accomplishing project goals. Organization is basically motivated by the need for cooperation when the goals are not achievable by a single individual in a reasonable amount of time. We follow a decentralized-control team organization, i.e. all work is considered group work. The following table shows the team roles and responsibilities.

**1.1: Team Organization**

| **Name** | **Role/Responsibility** |
|---|---|
| Vineeth John | Leader |
| Harikrishnan T | Designer |
| Asha Maria Aricatt | Debugger |
| Kripa John | Programmer |

Vineeth John: Coordinates and controls all sorts of development of the project, making appropriate planning to meet the deadline and also coordinating the group members.

Harikrishnan T: He designing the GOM TRADE.

Asha Maria Aricatt: She is responsible for debugging the program and also for preparing the project report and presenting the GOM TRADE for final presentation.

Kripa John: She is responsible for developing code.

# CHAPTER 2
# Literature Survey

## 2.1 Documentation

The core idea of our project is based on mainly 2 IEEE papers 1) Design and Implementation of Online Stock Trading System by Huazhu Song, Mingzhi Zhang, Zhuang XuSchool of Computer Science and Technology, Wuhan University of Technology, Wuhan, Hubei, 430070.

2) Global Online Trading System by Gunnar Cuevas, Matthew Dale, Asmaa Idrisu, Jonathan Lee, Students, George Mason University. George Donohue, Andy Loerch, Faculty Advisors. Michael Parodi, Sponsor (Edgewood Asset Management, LLC.)

These two papers mainly based on share trading system using computers. But we are implementing this project in mobile systems.

The advantages of our system when compared with the existing system is that

1) No platform exists allowing investors the ability to trade on international markets from a single location. 2) This online trading system that provides global market access to any potential investor with an internet connection. 3) We can any time detect and realize the variations of the Indian market, Asian market and world market. Usually existing systems does not support this type of applications, they only show the variations of Indian market only. When a user wants to know the current trends of the market they usually check other websites such as www.moneycontrol.com or watch the market television such as cnbc TV 18 or something like that. 4) The proposed system consists of a provision for News Headlines tab, i.e. the various information regarding to the stock market will be displayed on a single click.

5) This application includes a particular provision for offline mode. That means even though the net facility is not available, the user can view his home page and the transaction details. This is not included in existing systems. 6) We can view the performance of various sectors such as banking sector, power sector, infra structure etc on a single click. This feature is not included in currently available technology. The user has to browse other web pages for that information.

# CHAPTER 3

# Proposed System

## 3.1 Development Plan

Here we use the evolutionary or incremental model or approach in our project. This is because there may be errors in earlier stages which we modify and only the final stage is delivered. This also helps in eliminating the flaws in the requirements analysis. Our project has a stepwise development.

The first thing is to build the online share trading system as a mobile application. This application should have the following requirements.The particular application must be included in the main menu of the mobile phone. Menu- applications-online share trading.

When we select on that particular application a login page must be displayed. In which there should be mainly 9 attributes 1) Username 2) Membership password 3) Trading password 4) Login 5) Cancel 6) Forgot password 7) Change password 8) Offline mode 9) New user

Various details are collected from popular share trading sites.We focused mainly on www.sharekhan.com

The Offline mode indicates that suppose if the net facility is not available then the user should get his last trading details on the screen. If he wants to make some modifications then the user name and password should be asked for ensuring the proper security. Similarly when the username and password is wrong then the system must display a warning message indicating that the username and password is incorrect. If the user failed in typing the username and password more than 2 times then a warning message should appear and it should tell him that Sorry please contact the system administrator

If the person click on the link Forgot password then a dialogue box must be appear and it should tell him that Contact the System Administrator Similarly if a person want to Change password then another box should be appeared and it should tell him that enter the current trading password, enter the new trading password, and confirm the new trading password. There should be a save box for making the changes. If the current password is wrong or the newly entered password is not same then warning message should be appeared and it should tell the user that The password you entered is not correct, please try again note that the membership password and username is unique and the user cannot make any modifications to it once it is allotted.When a person click on the box cancel then the entire application should be closed, before closing another dialogue box must appear and it should contain a message Do you really want to exit?With 2 options YES and NO. If a person click on YES then the application should be closed, otherwise the login page should be displayed.

If a person is logging into the page for the first time then the entire column entries should be empty and we must be able to add the various scrip names into the page. When a person adds various scrip names into the page it should be automatically saved by the system.

The following requirements are considered for the proposed system. 1)When we click on a particular tab market action then the various trends of global market should be displayed. It contains the attributes Indian market, sectors, gainers/losers; world market etc. when we click on the Indian market the current value of nifty and sensex should be notified. When we click on the sectors the various sector performances such as banking, infra structure etc should be displayed. When we click on the gainers and losers the top gainer of the day and the top loser of the day should be displayed. When we click on the world market various world market trends should be informed such as shanghai, NASDAQ, dowjones, Nikkei etc. all these details can be obtained from moneycontrol.com just upload it to our particular apps. 2) There should be a stock predictor included (such as stock neuro master or something like that and we have to predict the future scope of the share) trend identifier. 3) Important headlines related to share market should be included when we click on a particular tab called The headlines 4) There should time and date display indicating how much time is to close market etc.

# CHAPTER 4

# System Requirements Specification

## 4.1 Software Requirements

Our project being related to many areas of engineering, has many hardware and software requirements.

1. PLATEFORM: Antroid
2. LANGUAGE: Java
3. MOBILE APPLICATION MODULE: Eclipse IDE
4. WEB APPLICATION MODULE: Net Beans IDE

## 4.2 Hardware Requirements

1. MOBILE PHONE:Samsung Gallexy or HTC
2. PROCESSOR: Pentium IV,1.6 GHz
3. RAM: 1GB
4. MONITOR: 15 colour monitor
5. KEYBOARD:104 keys standard keyboard
6. MOUSE: Standard 3 button

# CHAPTER 5
# Design & Analysis

## 5.1 System Analysis

The entire system is divided into 3 modules. The details of those modules and the responsibilities of each member of the group are described in the later sections.

### 5.1.1 Module breakup

**5.1: Module Description**

| Module | Description |
|---|---|
| Mobile Application | A module that contains the user interface for Gom Trade |
| Web Application | A module that contains the web server for share trading |
| Website | A module from which the details for the web server is taken |

### 5.1.2 Module description

#### Mobile application module

Mobile application side mainly includes a GPRS activated mobile phone in which our application is installed. When a user wants to do some trading operations, then he will login into his system by using the username and password. When he logged in the system will automatically connect to the host, which is a java based website, which is hired for this particular application. We are using the eclipse IDE for mobile application development and net beans IDE for web application development.

Android OS: Android is a mobile operating system initially developed by Android Inc., a firm purchased by Google in 2005. Android is based upon a modified version of the Linux kernel. Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero. The feature was originally disabled at the kernel level (possibly to avoid infringing Apple's patents on touch-screen technology). Google has since released an update for the Nexus One and the Motorola Droid which enables multi-touch natively. Support for A2DP and AVRCP were added in version 1.5, sending files (OPP) and accessing the phone book (PBAP) were added in version 2.0, and voice dialing and sending contacts between phones were added in version 2.2. The web browser available in Android is based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine. The browser scores a 93/100 on the Acid3 Test.

GPRS: General packet radio service (GPRS) is a packet oriented mobile data service on the 2G and 3G cellular communication systems global system for mobile communications (GSM). The service is available to users in over 200 countries worldwide. GPRS was originally standardized by European Telecommunications Standards Institute (ETSI) in response to the earlier CDPD and i-mode packet switched cellular technologies. It is now maintained by the 3rd Generation Partnership Project (3GPP). It is a best-effort service, as opposed to circuit switching, where a certain quality of service (QoS) is guaranteed during the connection. In 2G systems, GPRS provides data rates of 56-114 kbit/second. 2G cellular technology combined with GPRS is sometimes described as 2.5G, that is, a technology between the second (2G) and third (3G) generations of mobile telephony. It provides moderate-speed data transfer, by using unused time division multiple access (TDMA) channels in, for example, the GSM system. GPRS is integrated into GSM Release 97 and newer releases. GPRS usage charging is based on volume of data, either as part of a bundle or on a pay as you use basis. An example of a bundle is up to 5 GB per month for a fixed fee. Usage above the bundle cap is either charged for per megabyte or disallowed. The pay as you use charging is typically per megabyte of traffic. This contrasts with circuit switching data, which is typically billed per minute of connection time, regardless of whether or not the user transfers data during that period.

Mobile Applications: Mobile application development is the process by which applications are developed for small low-power handheld devices such as personal digital assistants, enterprise digital assistants or mobile phones. These applications are either pre-installed on phones during manufacture, or downloaded by customers from app stores and other mobile software distribution platforms.

Java Mobile Applications: Java ME technologies cover a wide range of extremely tiny commodities, and enable security, connectivity, and useful utility programs inside smart cards, pagers, set-top boxes, and other small appliances. Java ME technologies are only one part of the Java software product family. Related Java platforms include the Java Platform, Standard Edition (Java SE, formerly J2SE platform), and the Java Platform, Enterprise Edition (Java

EE, formerly J2EE platform). Java technology also provides ways to create Web services, XML information transfers, numerous networking protocols, toolkits, and the Java Web Start application.

### Web application module

Why we are using a web application module in between the mobile application and website is that we cant access the real time data directly from the concerned website to the mobile. What we want is an interface or a bridge between the mobile and website and thats the reason why a web application module is included. Also this web application module can be otherwise treated as a server which contains all the necessary details. (It should be asked). In order to get the real time fluctuations in the mobile phones, the mobile application will request the web application module in a timely manner (say for every one minute) when the host receives a request, it will poll the website that contains the necessary details.( for example moneycontrol.com). After the polling finished the web application will receive the newly updated data and it will be driven to the mobile phone. Since we are dealing with mobile phone this take a much more time when compared with other systems because the data rate between a mobile system and a host website is low. But when recent technologies like 3g is coming, the system will make a much better performance.

Web Application: A web application is an application that is accessed over a network such as the Internet or an intranet. The term may also mean a computer software application that is hosted in a browser-controlled environment (e.g. a Java applet) or coded in a browser-supported language (such as JavaScript, combined with a browser-rendered markup language like HTML) and reliant on a common web browser to render the application executable. Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity, as is the inherent support for cross-platform compatibility. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions.

Servlets: A Servlet is a Java class in Java EE that conforms to the Java Servlet API, a protocol by which a Java class may respond to HTTP requests. They are not tied to a specific client-server protocol, but are most often used with this protocol. The word ”Servlet” is often used in the meaning of ”HTTP Servlet”.[1] Thus, a software developer may use a servlet to add dynamic content to a Web server using the Java platform. The generated content is commonly

HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as CGI and ASP.NET. Servlets can maintain state in session variables across many server transactions by using HTTP cookies, or URL rewriting. The servlet API, contained in the Java package hierarchy javax.servlet, defines the expected interactions of a Web container and a servlet.[1] A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights. A Servlet is an object that receives a request and generates a response based on that request. The basic servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the servlet's configuration parameters and execution environment. The package javax.servlet.http defines HTTP-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application. Servlets can be generated automatically from JavaServer Pages (JSP) by the JavaServer Pages compiler. The difference between Servlets and JSP is that Servlets typically embed HTML inside Java code, while JSPs embed Java code in HTML. While the direct usage of Servlets to generate HTML (as shown in the example below) is relatively rare nowadays, the higher level MVC web framework in Java EE (JSF) still explicitly uses the Servlet technology for the low level request/response handling via the FacesServlet. A somewhat older usage is to use servlets in conjunction with JSPs in a pattern called ”Model 2”, which is a flavour of the model-view-controller pattern.

Browsers: A web browser or Internet browser is a software application for retrieving, presenting, and traversing information resources on the World Wide Web. An information resource is identified by a Uniform Resource Identifier (URI) and may be a web page, image, video, or other piece of content.[1] Hyperlinks present in resources enable users to easily navigate their browsers to related resources. Although browsers are primarily intended to access the World Wide Web, they can also be used to access information provided by Web servers in private networks or files in file systems. Some browsers can also be used to save information resources to file systems.

## 5.2 System Design

For this project we are making use of an architecture called Blackboard architecture. Blackboard architecture: sometimes it is necessary for subsystems to be able to communicate with more than just their neighbour subsystem. If many subsystems need to communicate with each other, then blackboard architecture may be appropriate. In this architecture one subsystem is designated as the blackboard and serves as the communication medium among the other subsystems. Essentially the blackboard is an interface for writing information and receiving queries. So our system architecture is Blackboard architecture in which the web application module(web server module) that is www.gomtrade.info works as the blackboard

### 5.2.1 Use Case Models / Flow Diagrams

Every non-trivial engineering system must be specified. Data flow diagrams(DFDs) are a well-known and widely used notation for specifying the functions of an information system and how data flow from functions that manipulate data.Data can be organized in several ways. They can be stored in data repositories,they can flow in data flows,and they can be transferred to or from the external environment.

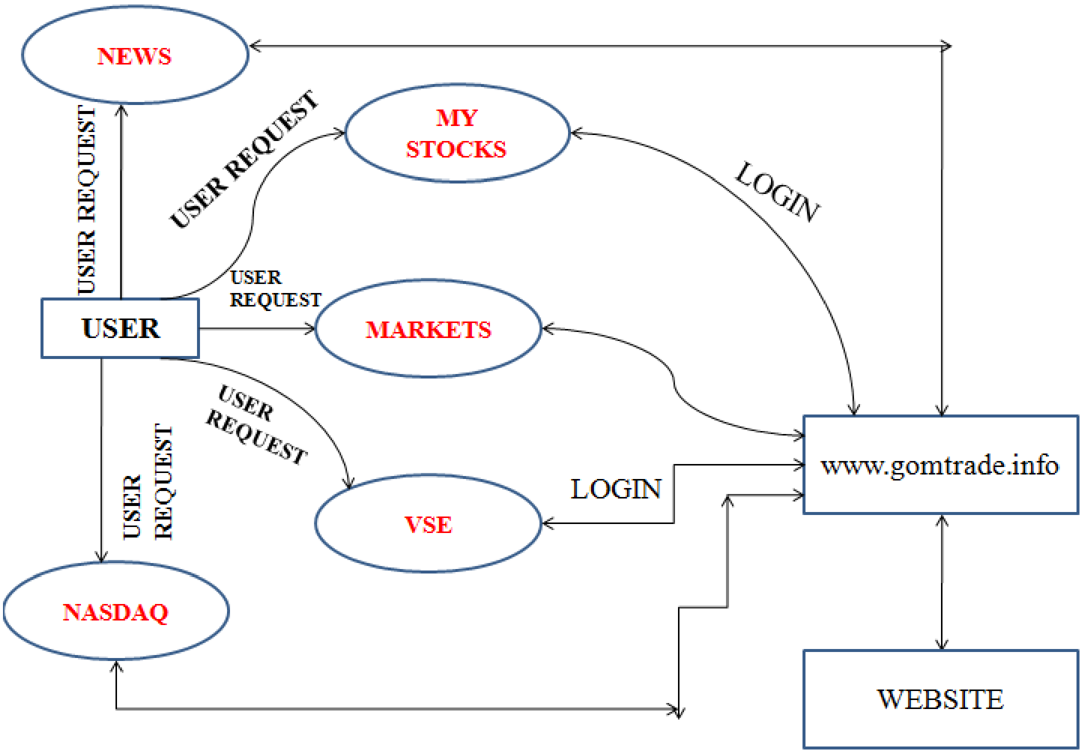


**Fig 5.2.1 DFD GOM Trade system**

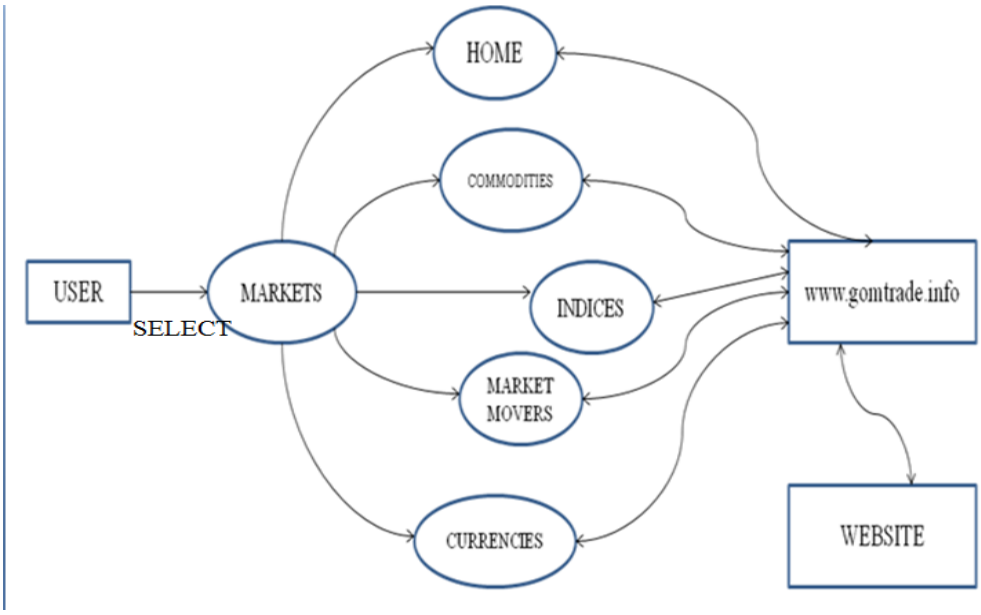


**Fig 5.2.2 Level 1 DFD Market system**

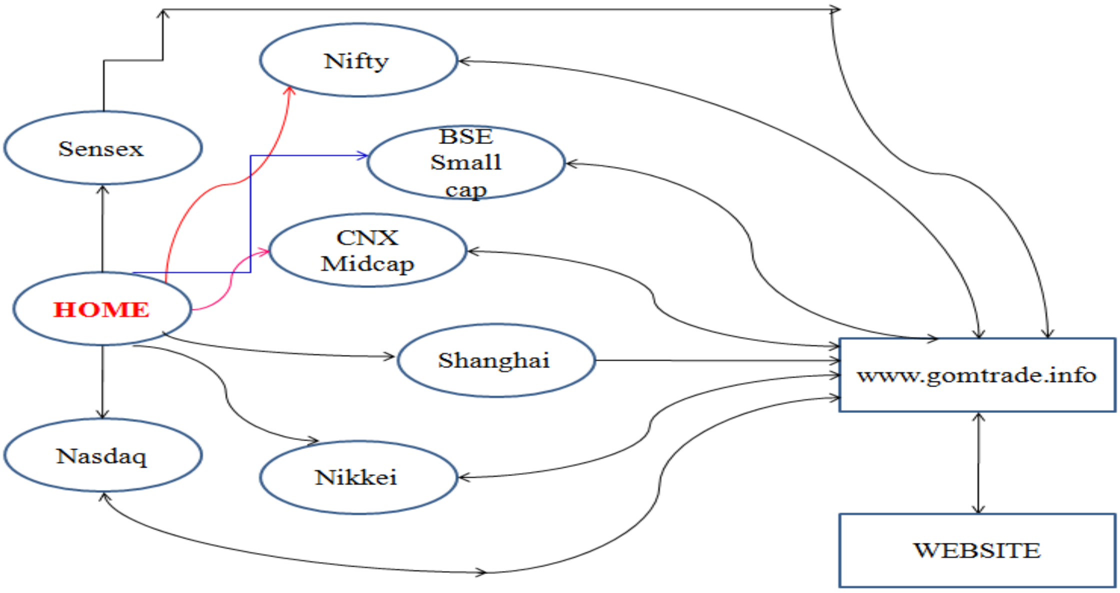


**Fig 5.2.3 Level 1.1 DFD Market home**

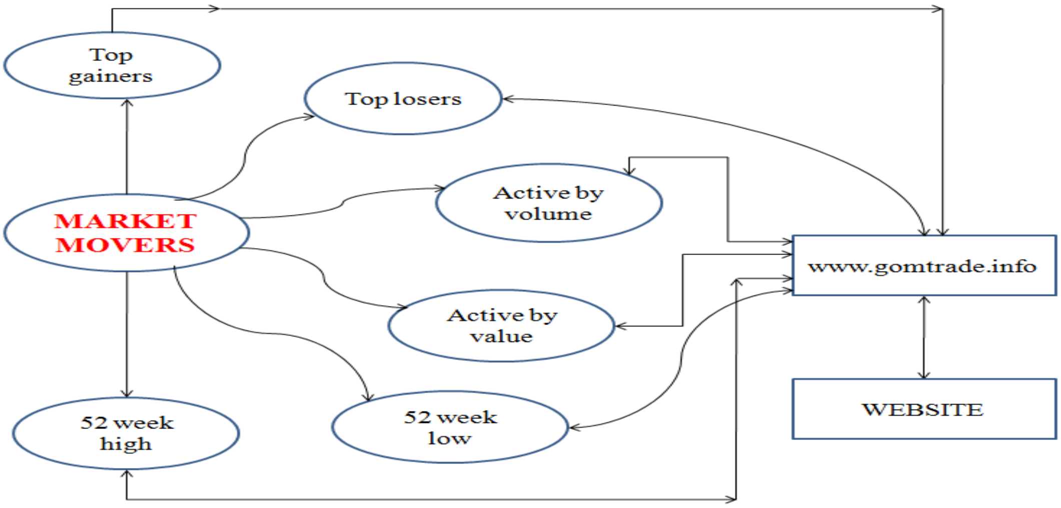


**Fig 5.2.4 Level 1.2 DFD Market-market movers**

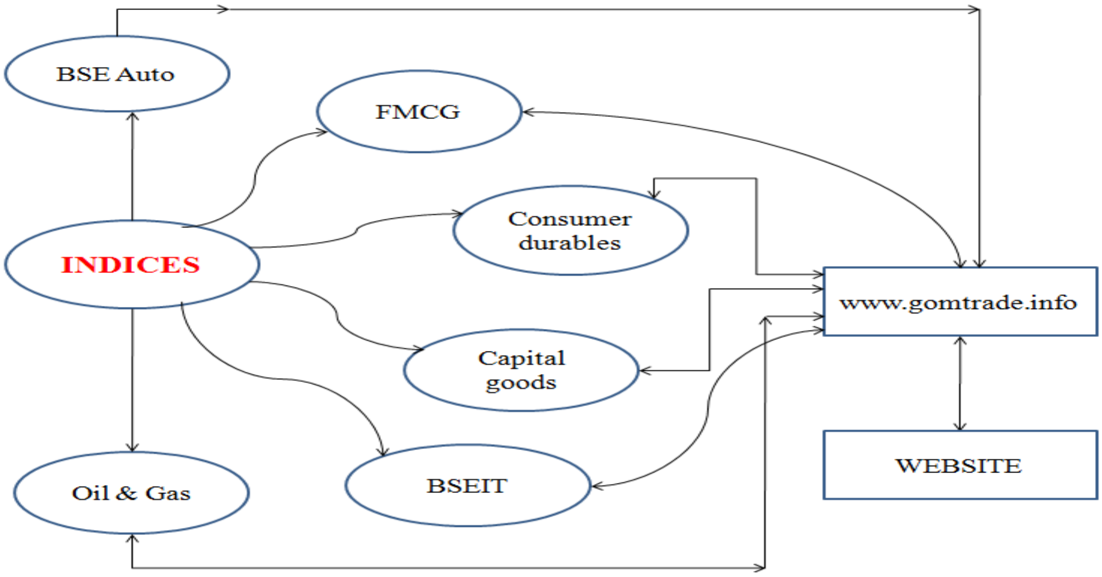


**Fig 5.2.5 Level 1.3 DFD Market Indices**

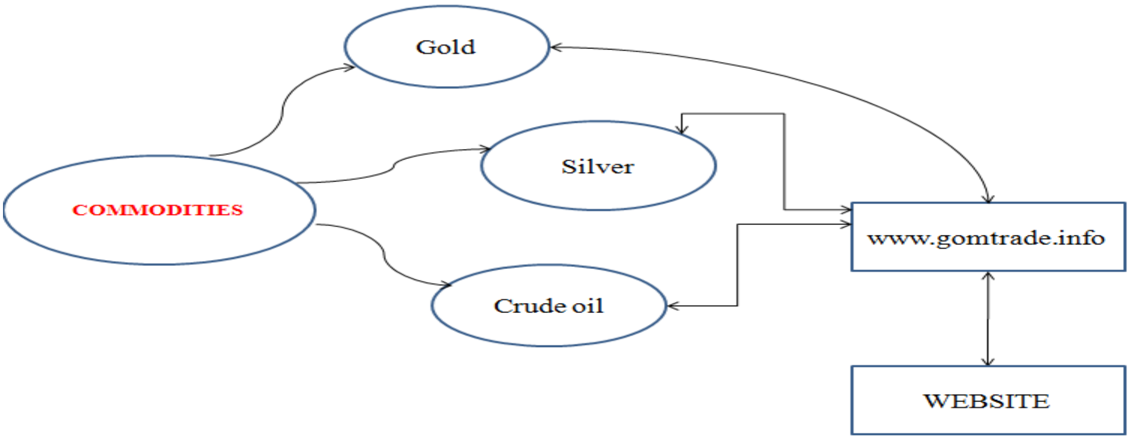


**Fig 5.2.6 Level 1.4 Market Commodities**

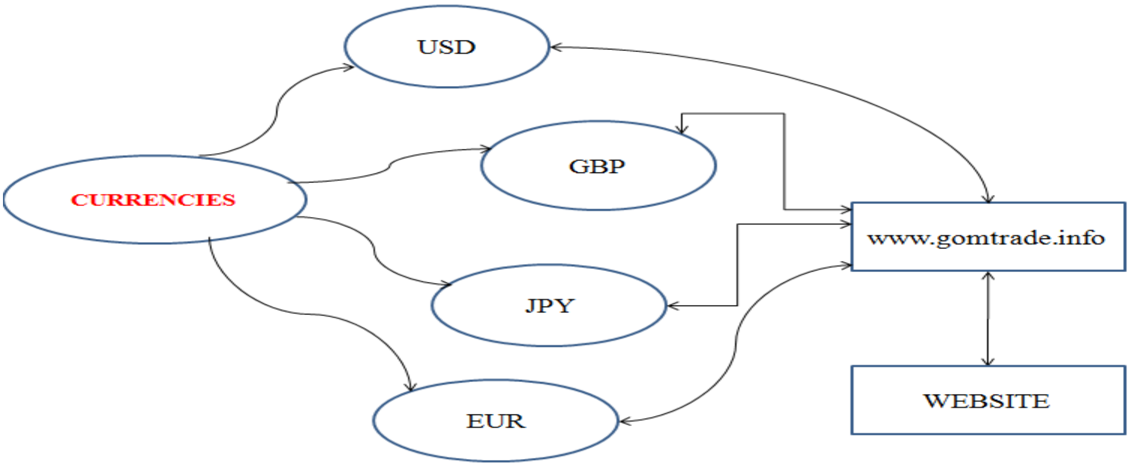


**Fig 5.2.7 Level 1.5 Market Currencies**

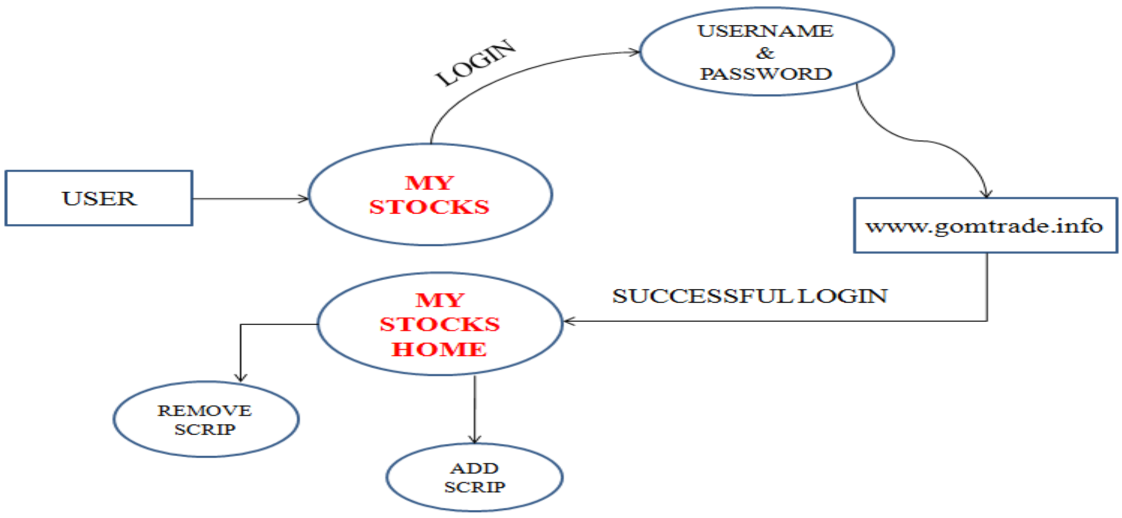


Fig 5.2.8 Level 2 My Stocks

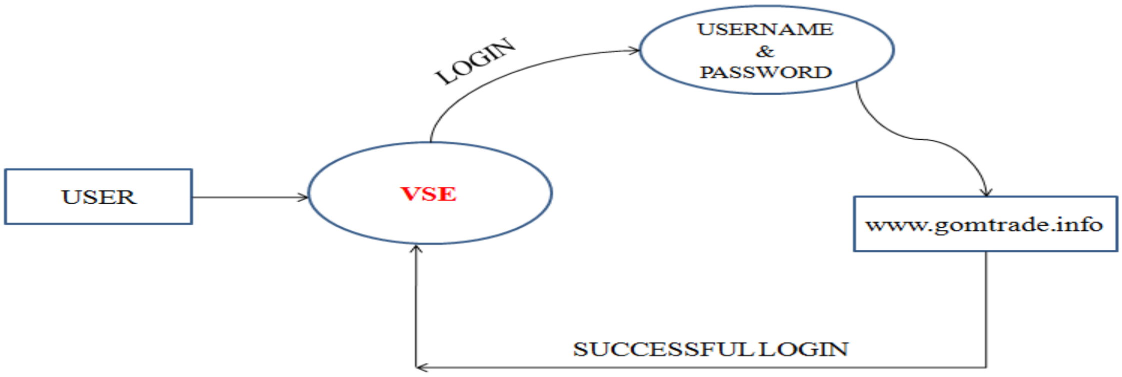


Fig 5.2.9 Level 3 virtual Stock exchange

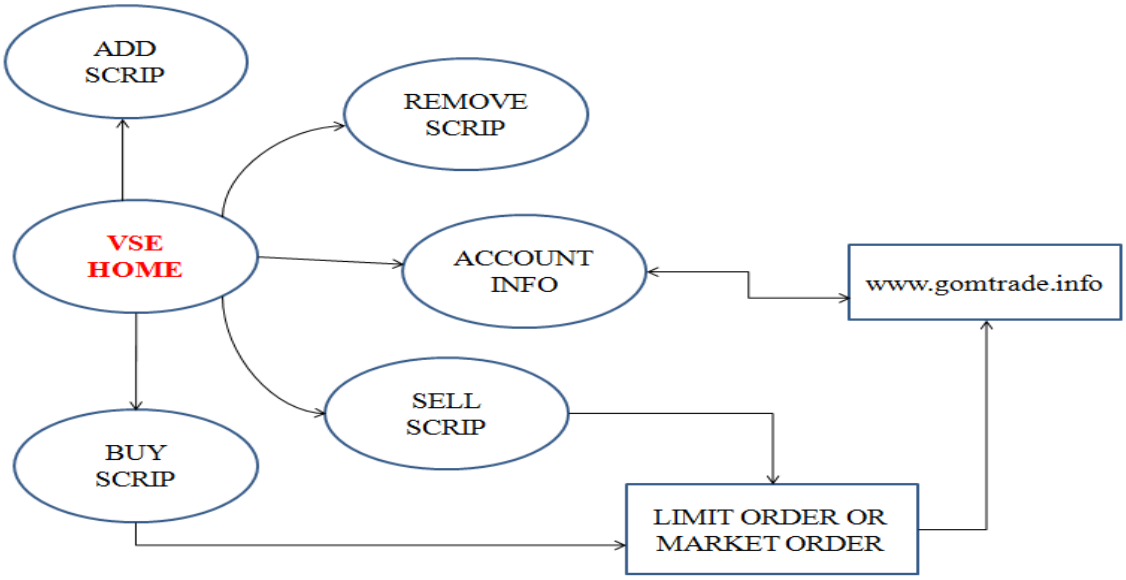


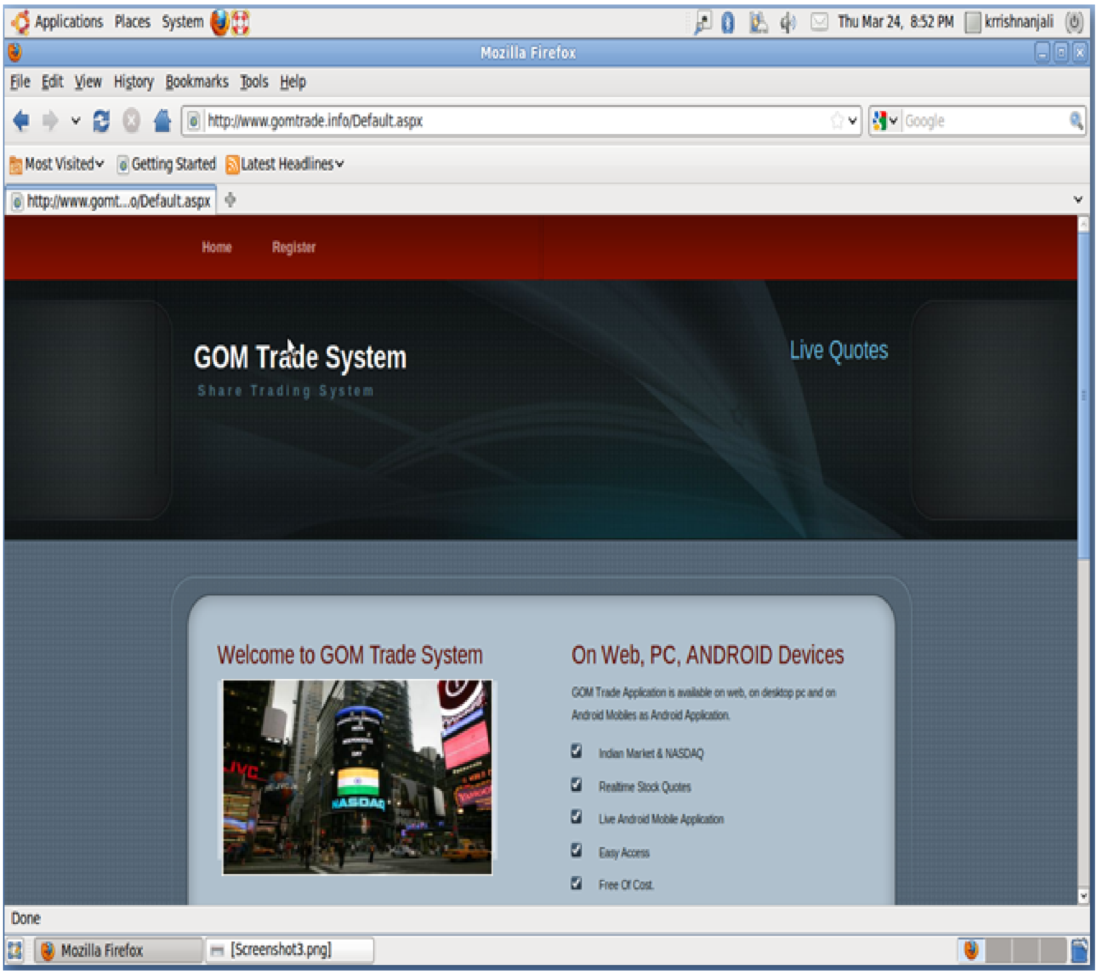
**Fig 5.2.10 Level 3.1 VSE home**

# CHAPTER 6

# Implementation

## 6.1 Introduction

We are presenting the prototype with the web server. The screen shorts will show the various functions and services provied by the web server.#. The screen shots of the web server is given below.



GOM Trade System
Share Trading System

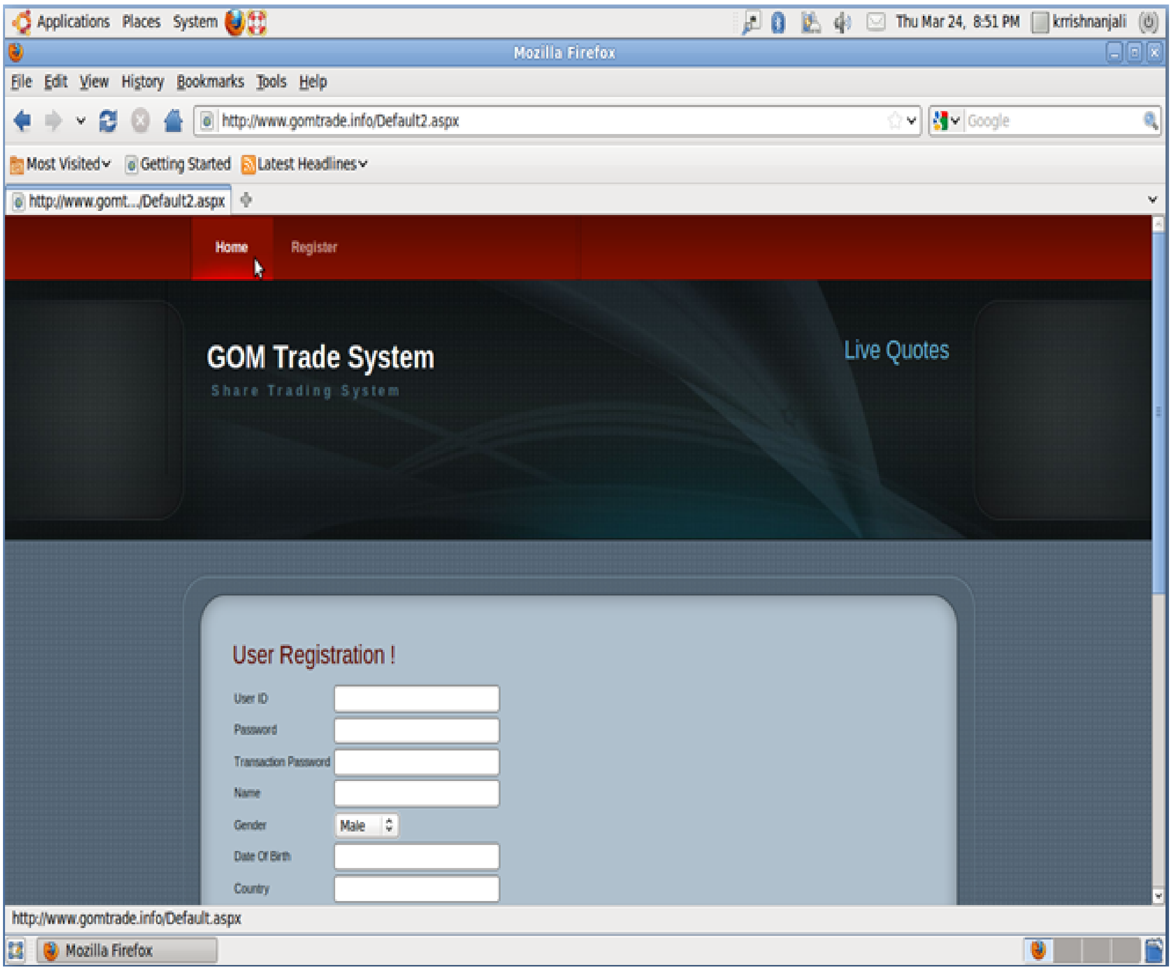
Welcome to GOM Trade System

On Web, PC, ANDROID Devices

GOM Trade Application is available on web, on desktop pc and on Android Mobiles as Android Application.

☑ Indian Market & NASDAQ
☑ Realtime Stock Quotes
☑ Live Android Mobile Application
☑ Easy Access
☑ Free Of Cost.

**Fig 8.1 Screen shot 1**


Applications Places System
Thu Mar 24, 8:51 PM
krrishnanjali
Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.gomtrade.info/Default2.aspx
Google
Most Visited
Getting Started
Latest Headlines
http://www.gomt.../Default2.aspx
Home
Register
GOM Trade System
Share Trading System
Live Quotes
http://www.gomtrade.info/Default.aspx
Mozilla Firefox

User Registration !

User ID

Password

Transaction Password

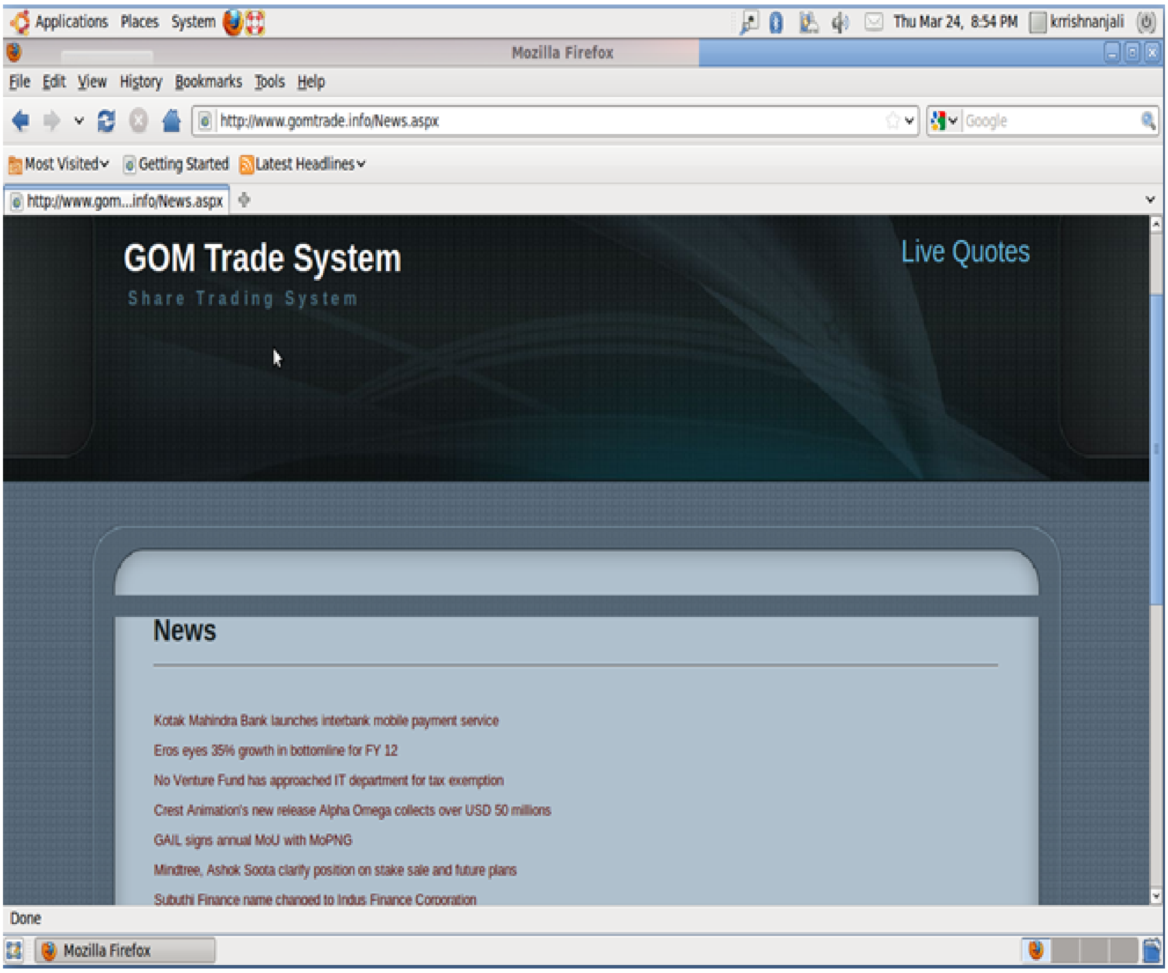
Name

Gender Male

Date Of Birth

Country

**Fig 8.2 Screen shot 2**



GOM Trade System

Share Trading System

Live Quotes

News

Kotak Mahindra Bank launches interbank mobile payment service

Eros eyes 35% growth in bottomline for FY 12

No Venture Fund has approached IT department for tax exemption

Crest Animation's new release Alpha Omega collects over USD 50 millions

GAIL signs annual MoU with MoPNG

Mindtree, Ashok Soota clarify position on stake sale and future plans

Subuthi Finance name changed to Indus Finance Corporation

**Fig 8.3 Screen shot 3**

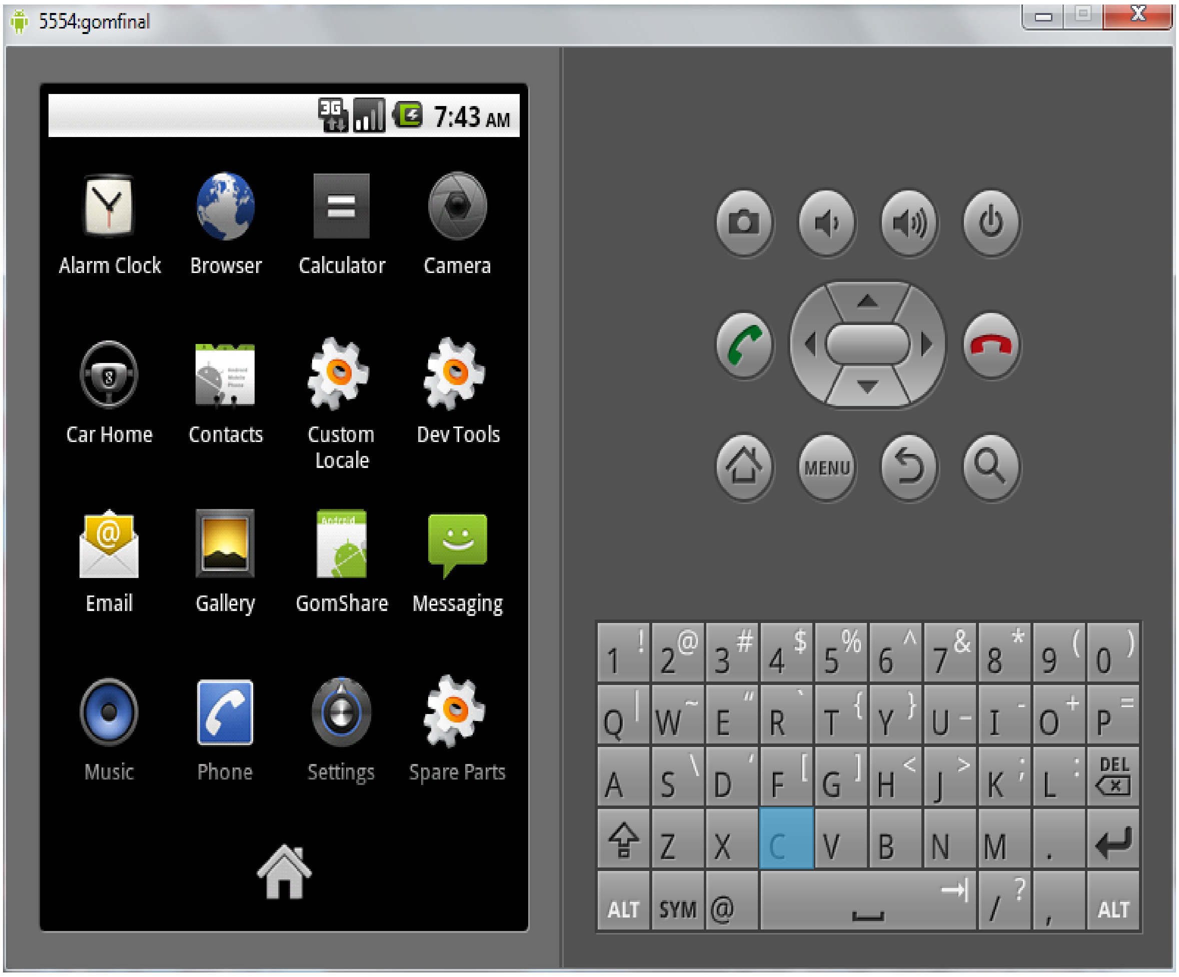
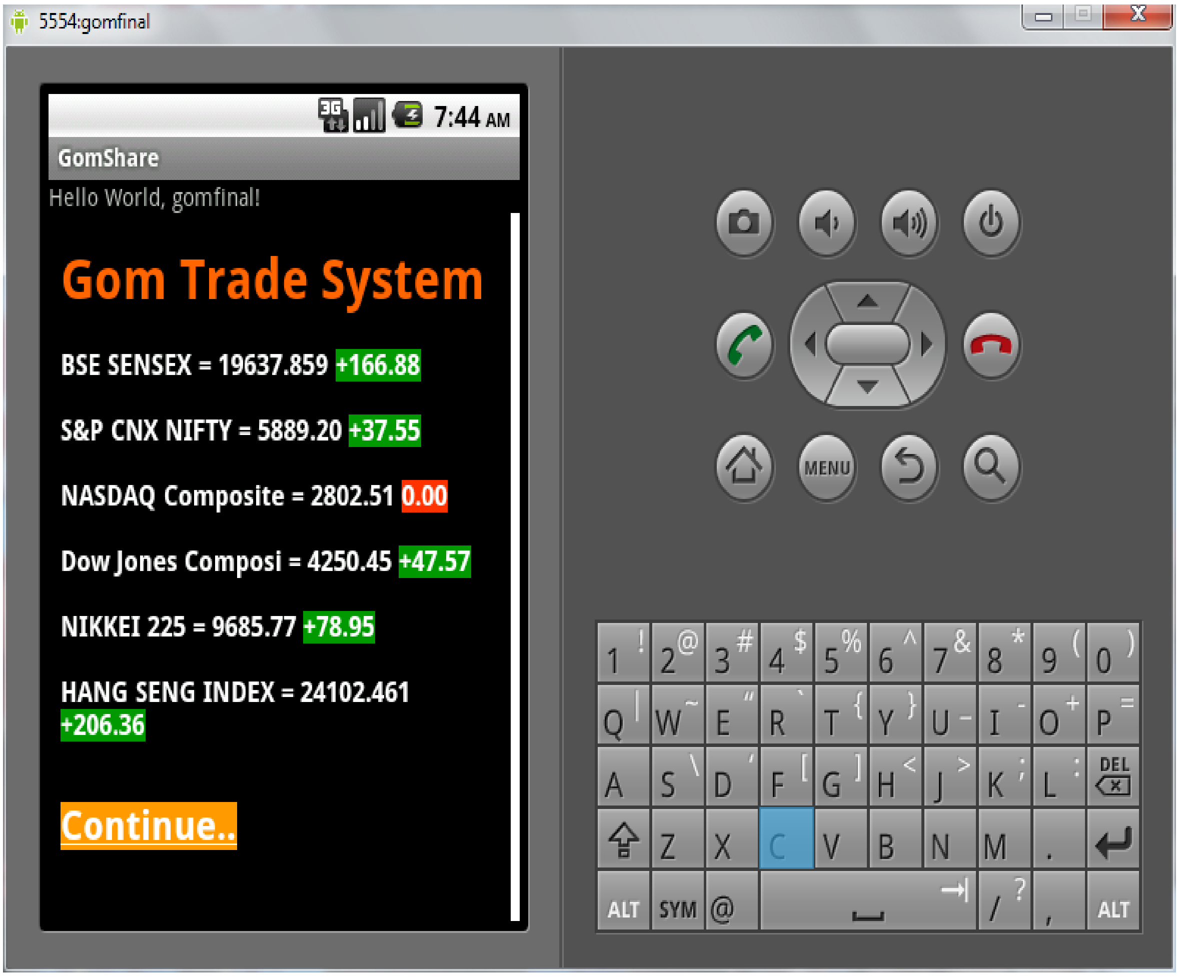


Fig 8.3 Screen shot 4



GomShare

Hello World, gomfinal!

Gom Trade System

BSE SENSEX = 19637.859 +166.88

S&P CNX NIFTY = 5889.20 +37.55

NASDAQ Composite = 2802.51 0.00

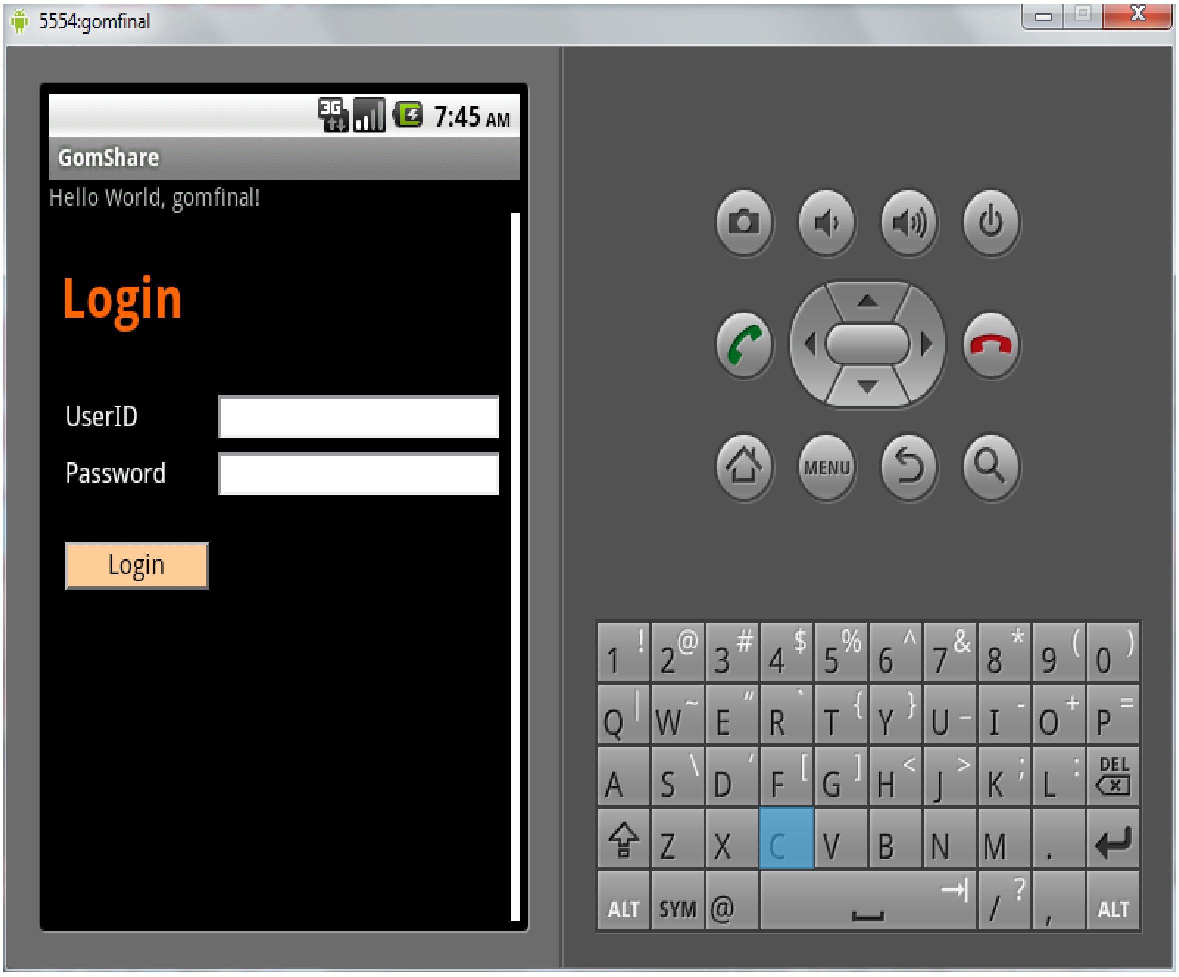
Dow Jones Composi = 4250.45 +47.57

NIKKEI 225 = 9685.77 +78.95

HANG SENG INDEX = 24102.461 +206.36

Continue..

**Fig 8.3 Screen shot 5**



5554:gomfinal

GomShare

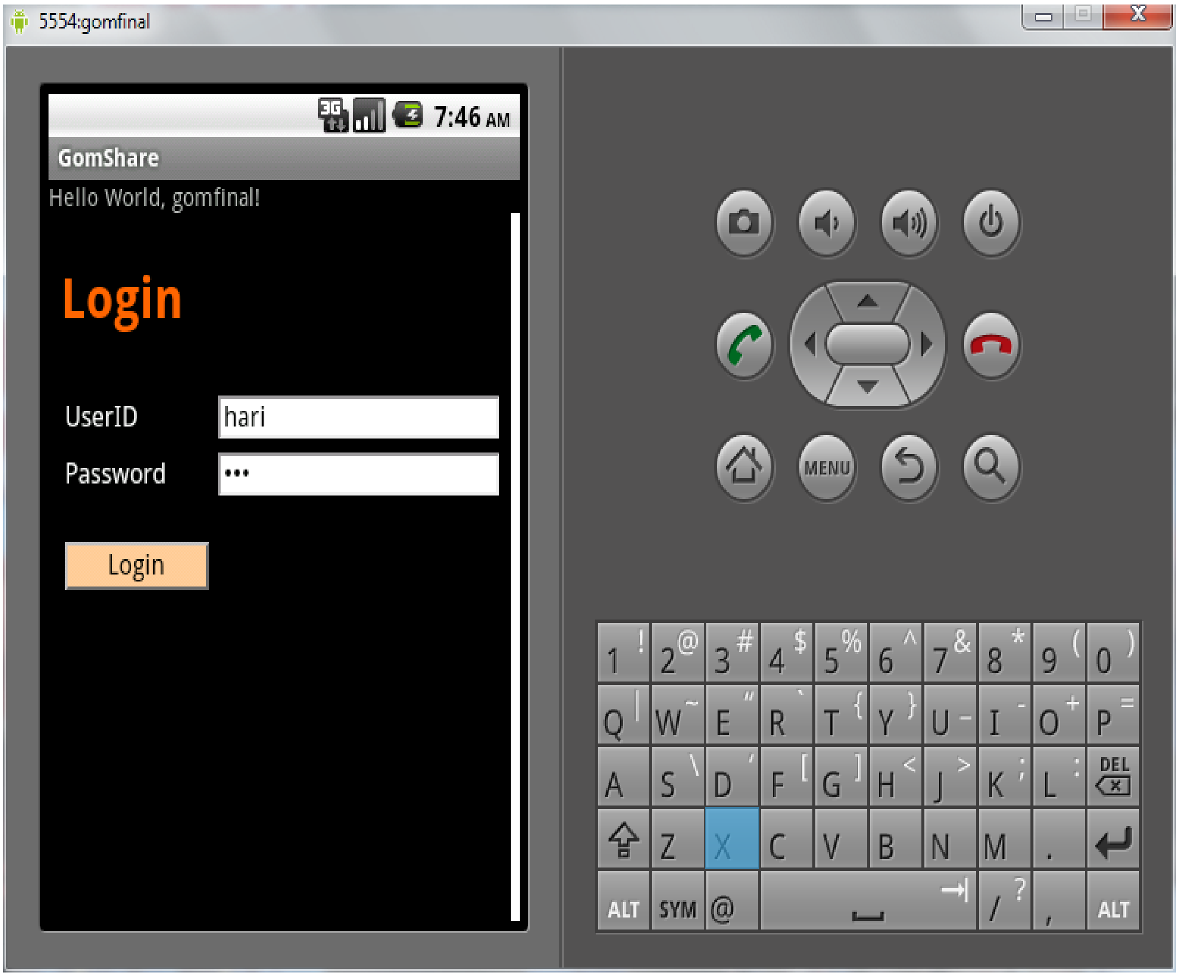
Hello World, gomfinal!

Login

UserID

Password

**Fig 8.3 Screen shot 6**



5554:gomfinal

7:46 AM

GomShare

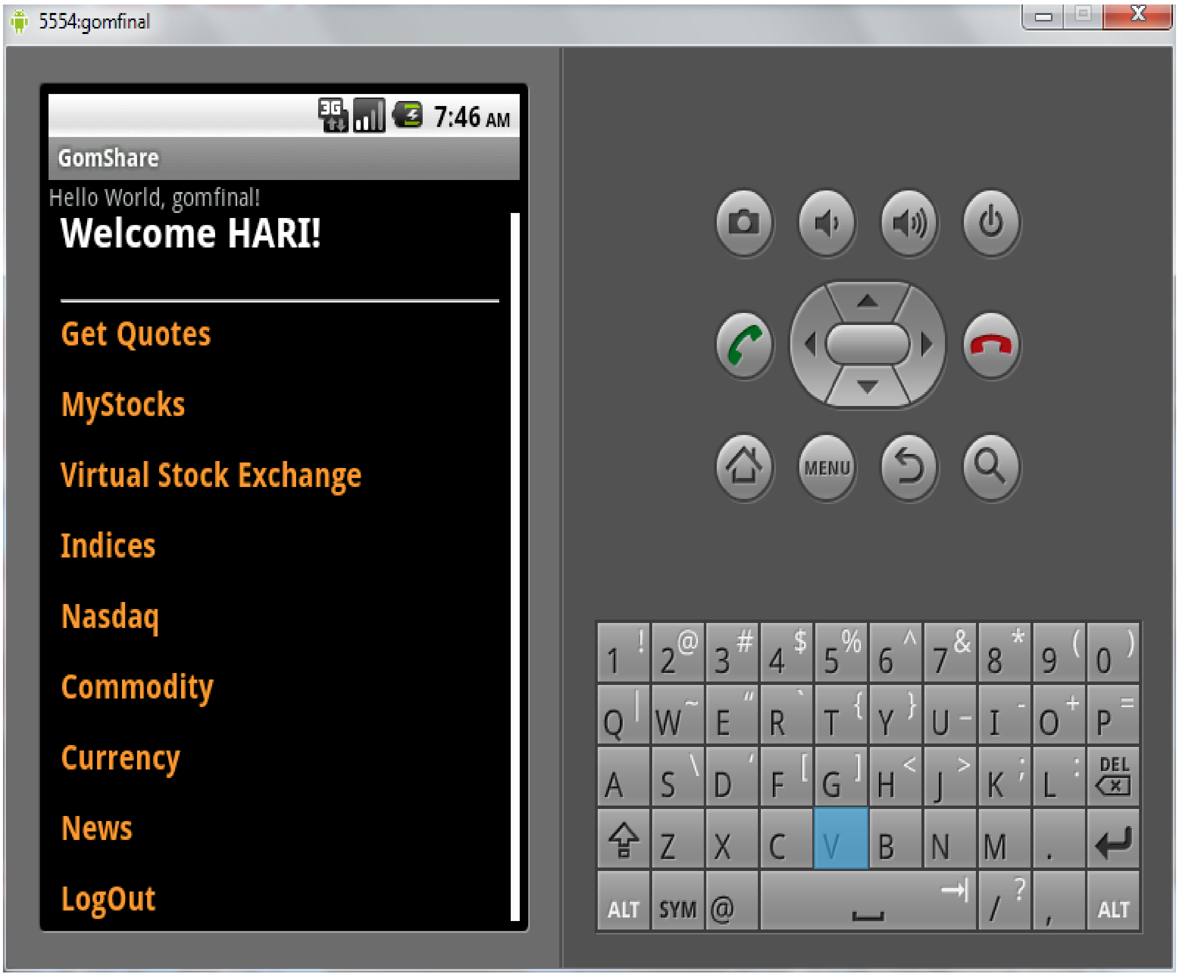
Hello World, gomfinal!

## Login

UserID: hari

Password: •••

Fig 8.3 Screen shot 7



5554:gomfinal

7:46 AM

GomShare

Hello World, gomfinal!

Welcome HARI!

Get Quotes

MyStocks

Virtual Stock Exchange

Indices

Nasdaq

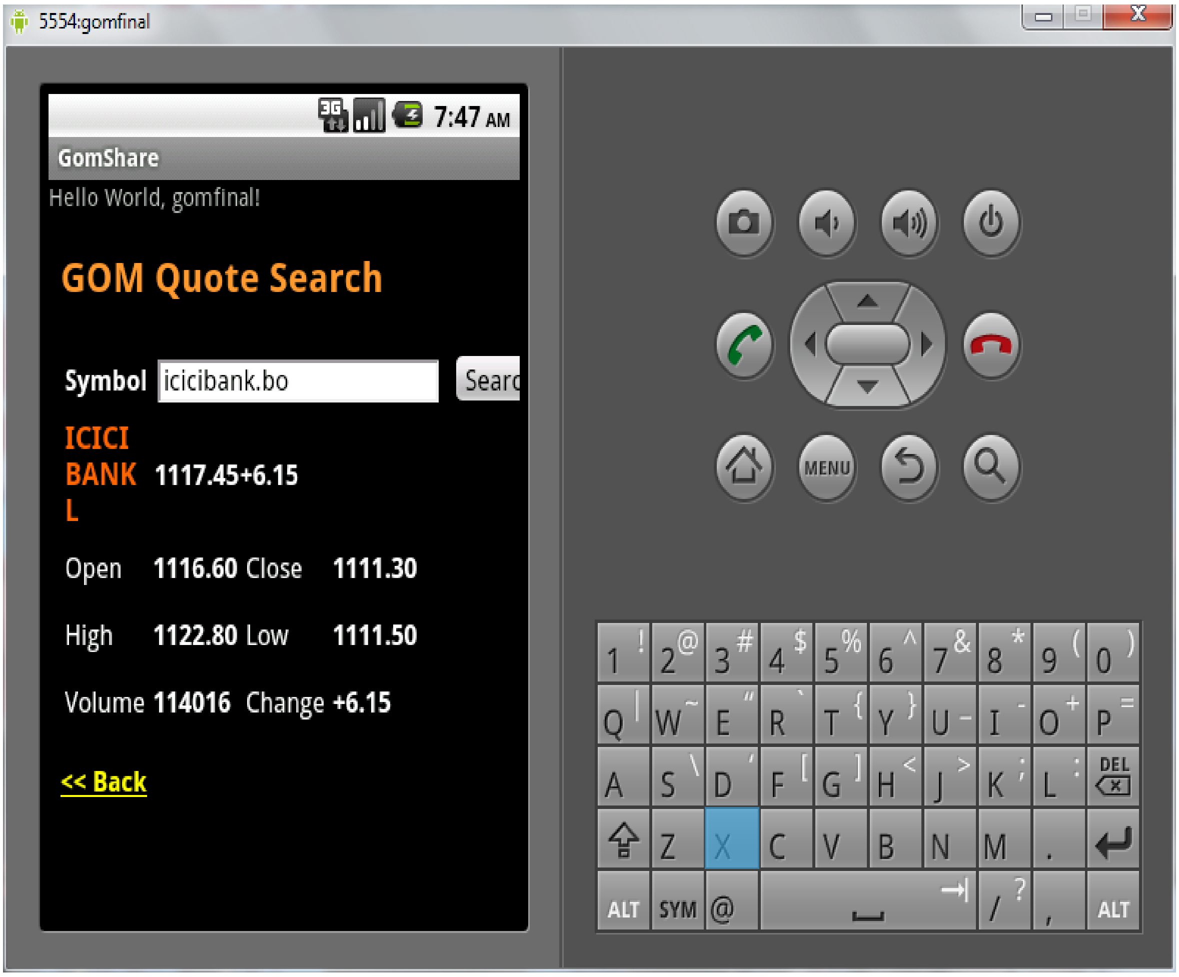
Commodity

Currency

News

LogOut

**Fig 8.3 Screen shot 8**



Hello World, gomfinal!

GOM Quote Search

Symbol icicibank.bo Searc

ICICI BANK L 1117.45+6.15

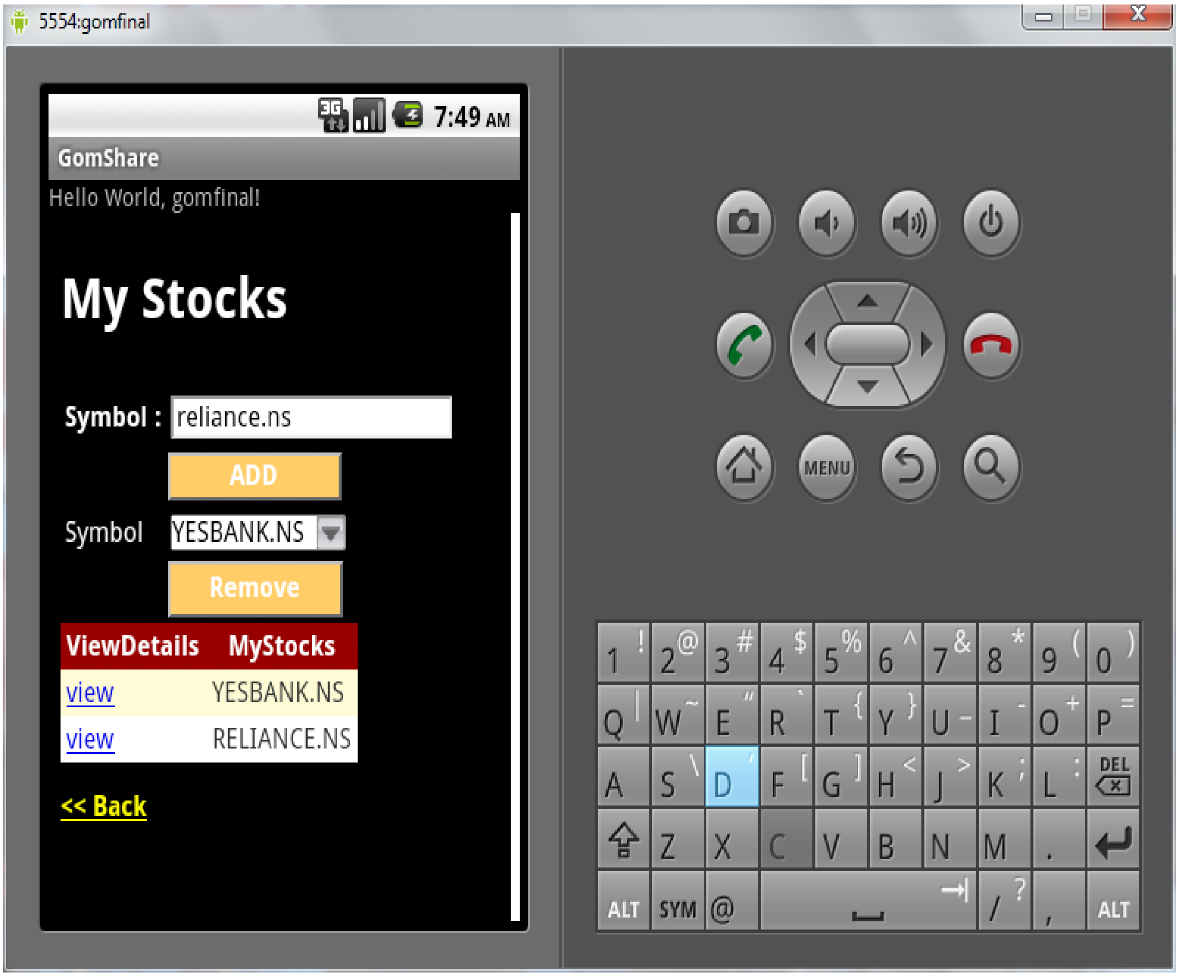
Open 1116.60 Close 1111.30

High 1122.80 Low 1111.50

Volume 114016 Change +6.15

<< Back

**Fig 8.3 Screen shot 9**



5554:gomfinal

7:49 AM

GomShare

Hello World, gomfinal!

My Stocks

Symbol : reliance.ns

ADD

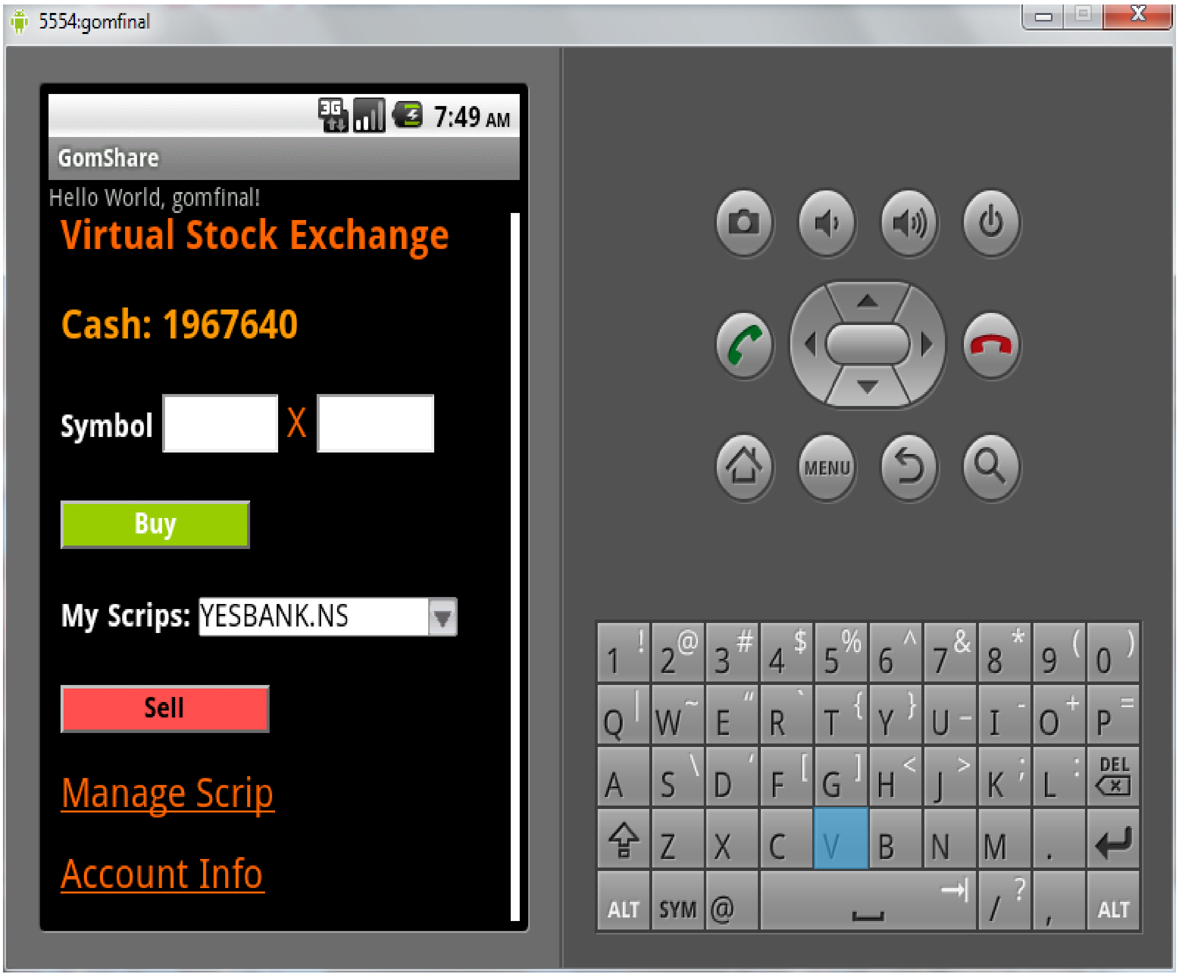
Symbol YESBANK.NS

Remove

| ViewDetails | MyStocks |
|---|---|
| view | YESBANK.NS |
| view | RELIANCE.NS |

<< Back

**Fig 8.3 Screen shot 10**



5554:gomfinal

GomShare

Hello World, gomfinal!

Virtual Stock Exchange

Cash: 1967640

Symbol X

Buy

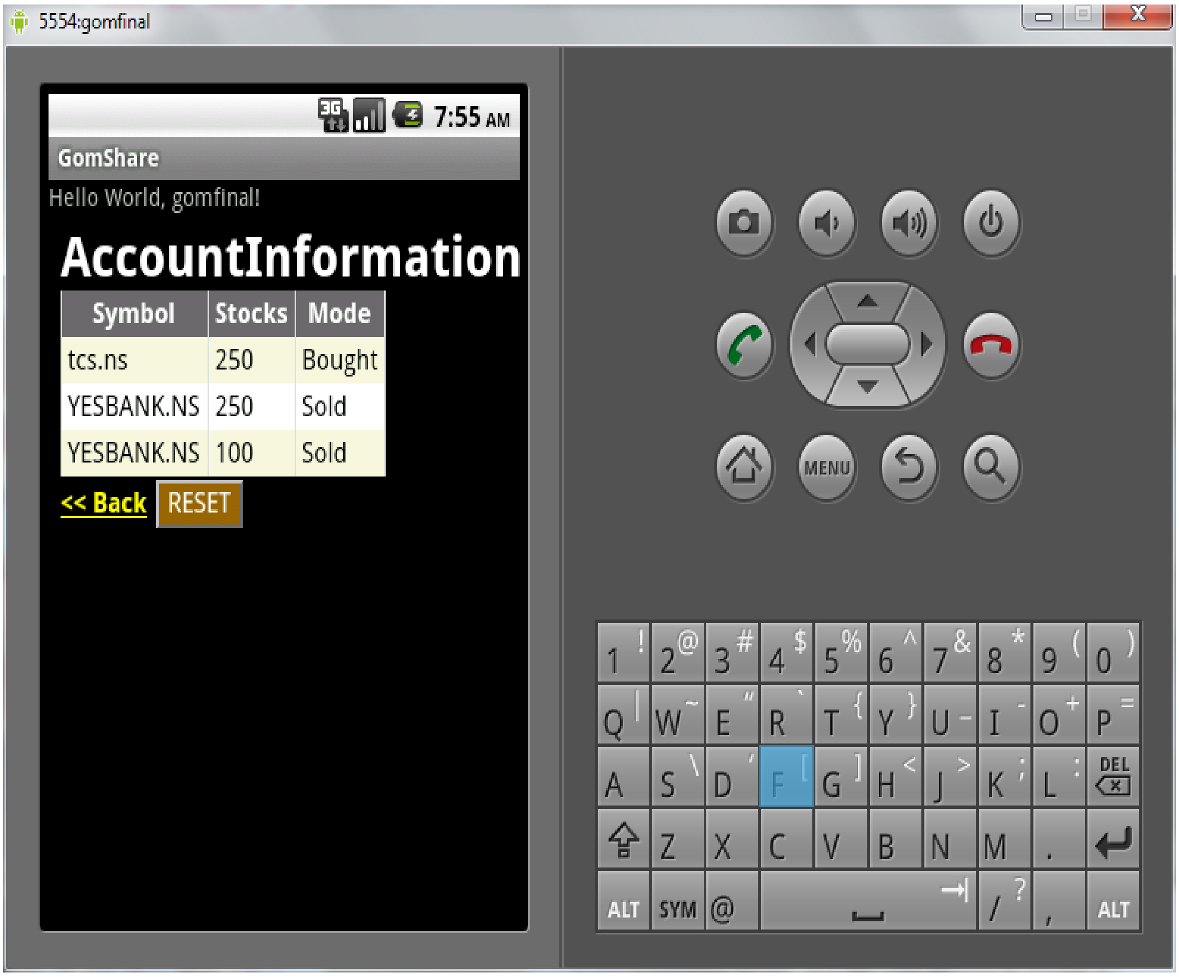
My Scrips: YESBANK.NS

Sell

Manage Scrip

Account Info

**Fig 8.3 Screen shot 11**



5554:gomfinal

7:55 AM

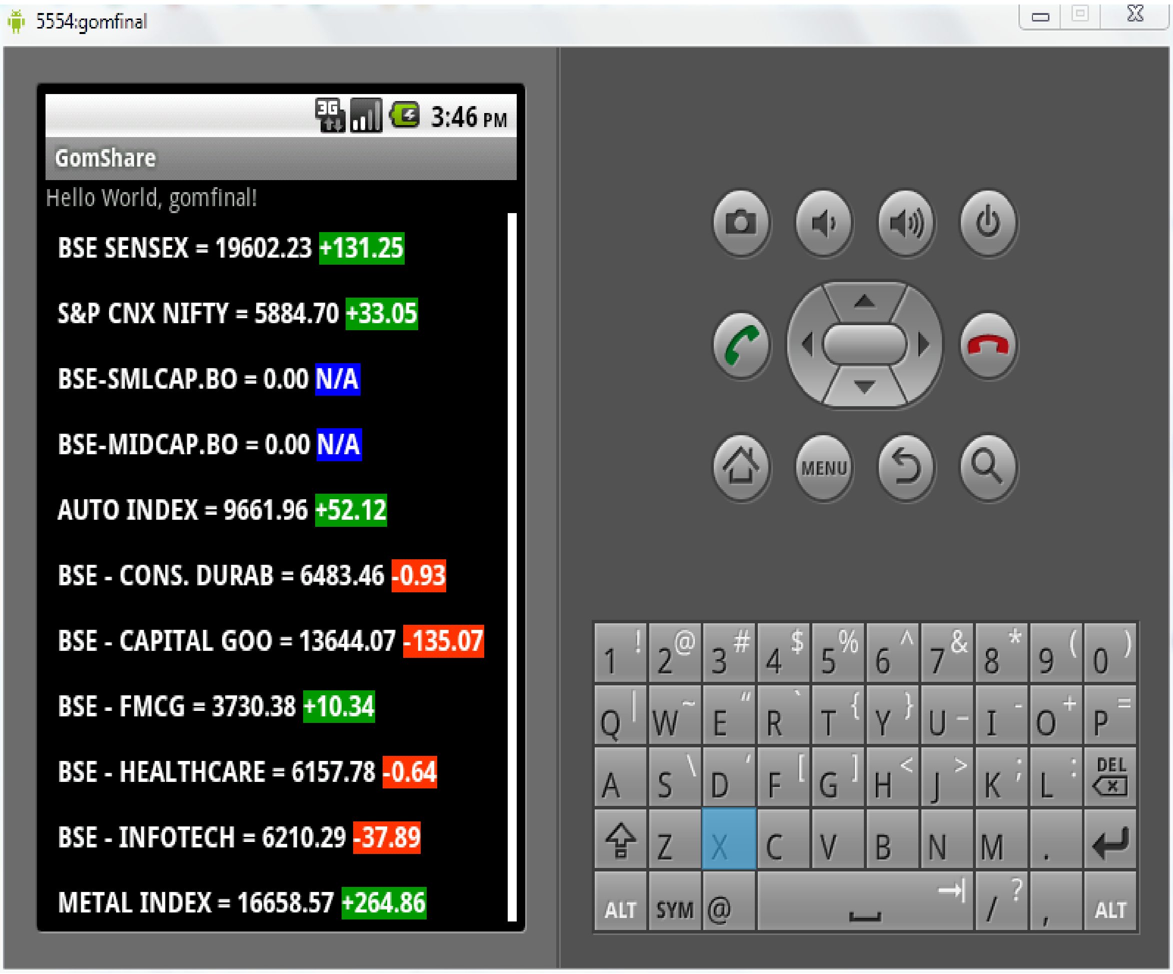
GomShare

Hello World, gomfinal!

AccountInformation

| Symbol | Stocks | Mode |
|---|---|---|
| tcs.ns | 250 | Bought |
| YESBANK.NS | 250 | Sold |
| YESBANK.NS | 100 | Sold |

**Fig 8.3 Screen shot 12**



5554:gomfinal

3:46 PM

GomShare

Hello World, gomfinal!

BSE SENSEX = 19602.23 +131.25

S&P CNX NIFTY = 5884.70 +33.05

BSE-SMLCAP.BO = 0.00 N/A

BSE-MIDCAP.BO = 0.00 N/A

AUTO INDEX = 9661.96 +52.12

BSE - CONS. DURAB = 6483.46 -0.93

BSE - CAPITAL GOO = 13644.07 -135.07

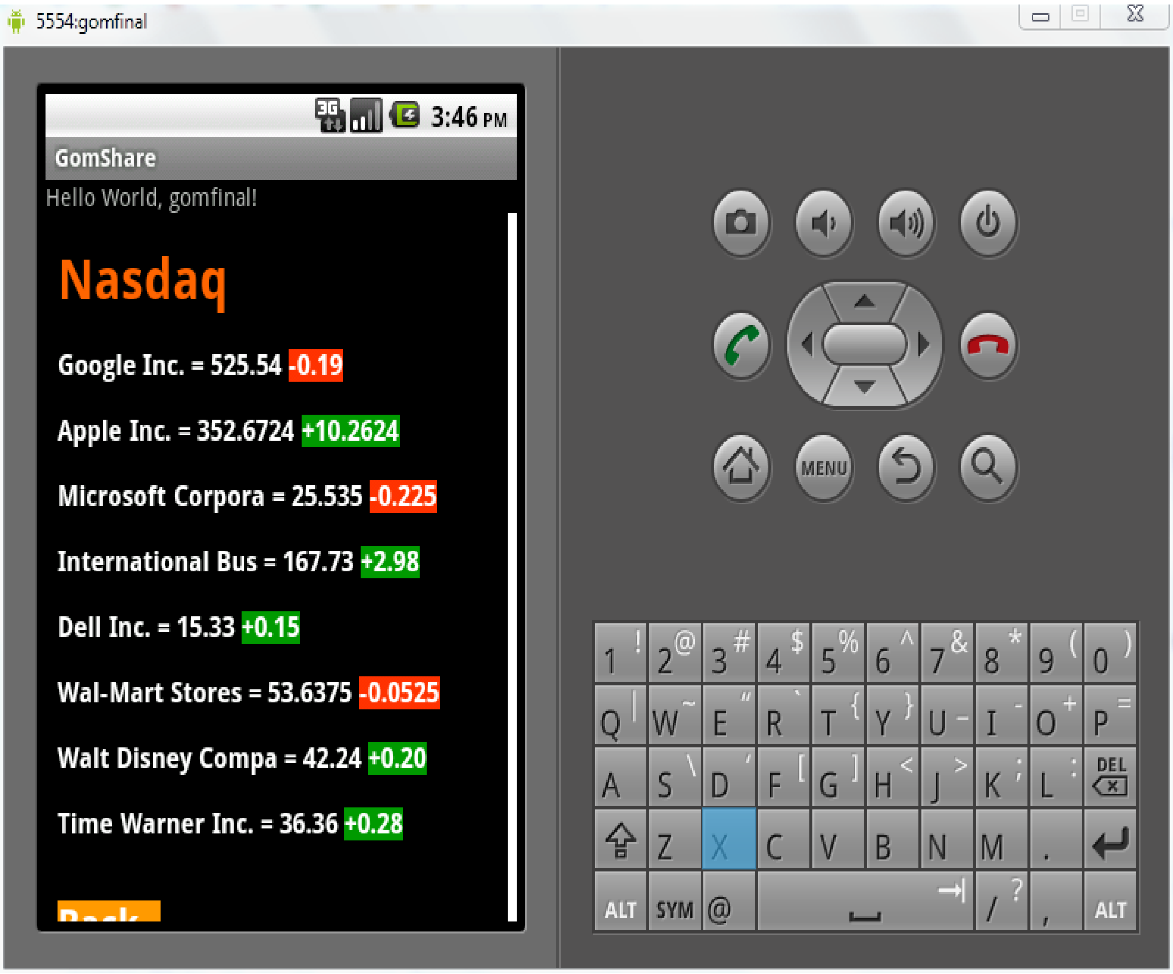
BSE - FMCG = 3730.38 +10.34

BSE - HEALTHCARE = 6157.78 -0.64

BSE - INFOTECH = 6210.29 -37.89

METAL INDEX = 16658.57 +264.86

Fig 8.3 Screen shot 13



5554:gomfinal

3:46 PM

GomShare

Hello World, gomfinal!

Nasdaq

Google Inc. = 525.54 -0.19

Apple Inc. = 352.6724 +10.2624

Microsoft Corpora = 25.535 -0.225

International Bus = 167.73 +2.98

Dell Inc. = 15.33 +0.15

Wal-Mart Stores = 53.6375 -0.0525

Walt Disney Compa = 42.24 +0.20

Time Warner Inc. = 36.36 +0.28

Back

**Fig 8.3 Screen shot 14**

## 6.2 Algorithm / Pseudo codes

**User Interface**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;
public class AndroidAccess:System.Web.Services.WebService
{
```

```
public AndroidAccess()
{
}

[WebMethod]
public string HelloWorld()
{
    return "Hello␣World";
}
[WebMethod]
public string CheckUser(string usrid,string mpass,string tpass)
{

        if (ct > 0)
        {
            foreach (var user in Data)
            {
                s = user.Userid;
            }
        }
        else
        {
            s = "User␣Not␣Exists";
        }
        return s;
    }
    catch (Exception ex)
    {
        return "Failed";
    }
}
{
    try
    {
     UserAccessDataContext Datas=new UserAccessDataContext();
        var m = (from k in Datas.Members select k);
```

```
            m.pass = pass;
            m.tpass = tpass;
            Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
      {
        try
        {
        UserAccessDataContext Datas=new UserAccessDataContext();
            Member m = new Member();
            m.Address = address;
            m.Birthday = birthday;
            m.Country = country;
            m.Email = email;
            m.gender = gender;
            m.Mobile = mobile;
            m.name = name;
            m.pass = pass;
            m.tpass = tpass;
            m.Userid = userid;
            Datas.Members.InsertOnSubmit(m);
            Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
    {
      try
       {
```

```
   UserAccessDataContext Datas=new UserAccessDataContext();
        var m = (from k in Datas.Members select k);
        m.Address = address;
        m.Birthday = birthday;
        m.Country = country;
        m.Email = email;
        m.gender = gender;
        m.Mobile = mobile;
        m.name = name;
        m.pass = pass;
        m.tpass = tpass;


        Datas.SubmitChanges();
        return true;
    }
    catch (Exception ex)
    {
        return false;
    }
}

[WebMethod]
public bool DeleteUser(string userid)
{
  try
  {
  UserAccessDataContext Datas=new UserAccessDataContext();
        Member m = new Member();

        m.Userid = userid;
        Datas.Members.DeleteOnSubmit(m);
        Datas.SubmitChanges();
        return true;
    }
    catch (Exception ex)
    {
        return false;
    }
}
{
    try
    {
    StockDetailsDataContext Datas=new StockDetailsDataContext();
        StockDetail Data = new StockDetail();
        Data.AnnRange = annrange;
        Data.Change = change;
        Data.CloseV = closev;
        Data.High = high;
        Data.IndexVal = indexval;
        Data.Low = low;
        Data.MktCap = mktcap;
        Data.OnDate = ondate;
        Data.OpenV = openv;
        Data.PE = PE;
        Data.PrevClose = prevclose;
        Data.StockId = Stockid;
        Data.StockName = stockname;
        Data.Symbol = symbol;
        Data.TradeTime = tradetime;
        Data.Volume = volume;
        Datas.StockDetails.InsertOnSubmit(Data);
        Datas.SubmitChanges();
        return true;
    }
    catch (Exception ex)
    {
        return false;
    }
}
{
    try
    {
    StockDetailsDataContext Datas = new StockDetailsDataContext()
        Data.AnnRange = annrange;
        Data.Change = change;
        Data.CloseV = closev;
        Data.High = high;
        Data.IndexVal = indexval;
        Data.Low = low;
        Data.MktCap = mktcap;
        Data.OnDate = ondate;
        Data.OpenV = openv;
        Data.PE = PE;
        Data.PrevClose = prevclose;

        Data.StockName = stockname;
        Data.Symbol = symbol;
        Data.TradeTime = tradetime;
        Data.Volume = volume;

        Datas.SubmitChanges();
        return true;
    }
    catch (Exception ex)
    {
        return false;
    }
}

[WebMethod]
public bool DeleteStock(string Stockid)
{
 try
 {
  StockDetailsDataContext Datas=new StockDetailsDataContext();
    StockDetail Data = new StockDetail();

        Data.StockId = Stockid;

        Datas.StockDetails.DeleteOnSubmit(Data);
```

```
            Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
    {
        try
        {
            InterestedStock Data = new InterestedStock();
            Data.InterestId = Interestid;
            Data.MemberID = memberid;
            Data.StockSymbol = stocksymbol;
            Datas.InterestedStocks.InsertOnSubmit(Data);
            Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
    {
        try
        {
         Data.MemberID = memberid;
         Data.StockSymbol = stocksymbol;
                 Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
    [WebMethod]
    public bool DeleteInterestStock(string Interestid)
    {
        try
        {
            InterestedStock Data = new InterestedStock();
            Data.InterestId = Interestid;

            Datas.InterestedStocks.DeleteOnSubmit(Data);
            Datas.SubmitChanges();
            return true;
        }
        catch (Exception ex)
        {
            return false;
        }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Xml.Linq;


using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.Linq;
using System.Data.Linq.Mapping;
using System.Linq;
using System.Linq.Expressions;
using System.Reflection;
```

```
[global::System.Data.Linq.Mapping.DatabaseAttribute(Name=”gomtrdb”)]
public partial class InterestedStockDataContext : System.Data.Linq.D
{

        private static System.Data.Linq.Mapping.MappingSource mapping

  #region Extensibility Method Definitions
  partial void OnCreated();
  partial void InsertInterestedStockDB(InterestedStockDB instance);
  partial void UpdateInterestedStockDB(InterestedStockDB instance);
  partial void DeleteInterestedStockDB(InterestedStockDB instance);
  #endregion

        public InterestedStockDataContext() :
                        base(global::System.Configuration.Configurati
        {
                OnCreated();
        }

        public InterestedStockDataContext(string connection) :
                        base(connection, mappingSource)
        {
                OnCreated();
        }

        public InterestedStockDataContext(System.Data.IDbConnection c
                        base(connection, mappingSource)
        {
                OnCreated();
        }

        public InterestedStockDataContext(string connection, System.
                        base(connection, mappingSource)
        {
                OnCreated();
        }

        public InterestedStockDataContext(System.Data.IDbConnection c
                        base(connection, mappingSource)
        {
                OnCreated();
        }

        public System.Data.Linq.Table<InterestedStockDB> InterestedSto
        {
                get
                {
                        return this.GetTable<InterestedStockDB>();
                }
        }
}

{

        private static PropertyChangingEventArgs emptyChangingEventAr

        private string _InterestId;

        private string _MemberID;

        private string _StockSymbol;

    #region Extensibility Method Definitions
    partial void OnLoaded();
    partial void OnValidate(System.Data.Linq.ChangeAction action);
    partial void OnCreated();
    partial void OnInterestIdChanging(string value);
    partial void OnInterestIdChanged();
    partial void OnMemberIDChanging(string value);
    partial void OnMemberIDChanged();
    partial void OnStockSymbolChanging(string value);
    partial void OnStockSymbolChanged();
    #endregion

        public InterestedStockDB()
        {
                OnCreated();
        }

        public string InterestId
        {
                get
                {
                        return this._InterestId;
                }
                set
                {
                        if ((this._InterestId != value))
                        {
                                this.OnInterestIdChanging(value);
                                this.SendPropertyChanging();
                                this._InterestId = value;
                                this.SendPropertyChanged(”InterestId”
                                this.OnInterestIdChanged();
                        }
                }
        }

        [global::System.Data.Linq.Mapping.ColumnAttribute(Storage=”_M
        public string MemberID
        {
                get
                {
                return this._MemberID;
                }
                set
                {
                        if ((this._MemberID != value))
```

```
                {
                this.OnMemberIDChanging(value);
                this.SendPropertyChanging();
                this._MemberID = value;
                this.SendPropertyChanged(”MemberID”);
                this.OnMemberIDChanged();
                }
        }
    }

    [global::System.Data.Linq.Mapping.ColumnAttribute(Storage=”_S
    public string StockSymbol
    {
        get
        {
        return this._StockSymbol;
        }
        set
        {
                if ((this._StockSymbol != value))
                {
                this.OnStockSymbolChanging(value);
                this.SendPropertyChanging();
                this._StockSymbol = value;
                this.SendPropertyChanged(”StockSymbol”);
                this.OnStockSymbolChanged();
                }
        }
    }

    public event PropertyChangingEventHandler PropertyChanging;

    public event PropertyChangedEventHandler PropertyChanged;

    protected virtual void SendPropertyChanging()
    {
        if ((this.PropertyChanging != null))
```

```
                {
                this.PropertyChanging(this, emptyChangingEventArgs);
                }
        }

        protected virtual void SendPropertyChanged(String propertyNam
        {
                if ((this.PropertyChanged != null))
                {
                this.PropertyChanged(this, new PropertyChangedEventAr
                }
        }
}
```

# CHAPTER 7

# Conclusion

The proposed system concentrates on doing online share trading globally using a single independent platform. The number of people investing in the share market increasing day by day. This demand coupled with advances in trading technology has opened up the markets so that nowadays nearly anybody can own stocks. Since no one has implemented this type of global marketing system, so this proposed project has a great future scope. We are planning to implement this particular system as an application in Android OS, which is the emerging OS in recent years. Also we are planning to implement this in java supporting phones also.

# REFERENCES

1. www.moneycontrol.com

2.www.ieeexplore.ieee/trading/market

3.Design and Implementation of Online Stock Trading System by Huazhu Song, Mingzhi Zhang, Zhuang XuSchool of Computer Science and Technology, Wuhan University of Technology, Wuhan, Hubei, 430070.

4.Global Online Trading System by Gunnar Cuevas, Matthew Dale, Asmaa Idrisu, Jonathan Lee, Students, George Mason University. George Donohue, Andy Loerch, Faculty Advisors. Michael Parodi, Sponsor (Edgewood Asset Management, LLC.)