



UNIVERSITY OF NAIROBI

SCHOOL OF COMPUTING AND INFORMATICS

MOBILE BASED STOCK MARKET AND FOREX EXCHANGE

NOTIFIER

BY

JAMES OGUYA

P15/35231/2010

SUPERVISOR: Mr. KAHONGE A.M

May 2012

Declaration

I hereby declare that this project is my own and has, to the best of my knowledge not been submitted in any other institution of higher learning.

Signed

Date

Oguya James Ogutu

This report has been submitted for examination with my approval as a university supervisor.

Signed

Date

Mr. Kahonge A.M

Acknowledgement

I would like to recognize the contributions and support given to me by my supervisor Mr. Kahonge, whose comments, suggestions and encouragement played a significant role in helping me achieve my project objectives.

Thanks to all the lecturers of SCI who have taught me various concepts of application and resourceful knowledge that facilitated my thinking through the project and conducive environment for undertaking the same.

My sincere regards to all SCI students for the support and cooperation that they offered during the period of this project development.

Special thanks to my family members for your love, prayers and considerations during this period.

Dedication

This project is dedicated to my parents for the support and care they've offered me.

Abstract

Mobile Based Stock market and forex exchange rates Notifier is an SMS based system that provides the general public and businessmen with access to crucial information about stock prices and forex rates. With the positive increase of mobile phone and internet usage, infrastructures have been established to better network and connect the users. Consequently, the services offered in mobile devices otherwise known as M-services, are thriving with plausible applications.

This project brings forward the development of Mobile Based Stock market and forex exchange rates Notifier for alerting users about the current stock prices and forex rates in order to inform shareholders and forex traders about the current rates in the market.

Table of Contents

LIST OF ABBREVIATIONS	VII
CHAPTER 1	8
1.0 BACKGROUND INFORMATION	8
1.2 PROBLEM DEFINITION	8
1.3 PROJECT GOAL	9
1.4 PROJECT OBJECTIVES	9
1.5 PROJECT JUSTIFICATION.....	9
 CHAPTER 2.....	 11
2.0 LITERATURE REVIEW	11
2.1 MONEY MARKETS.....	12
 CHAPTER 3.....	 14
3.0 METHODOLOGY.....	14
3.1 STAGES OF SSADM.....	14
 CHAPTER 4.....	 16
4.0 SYSTEM ANALYSIS.....	16
4.1 INTRODUCTION	16
4.2 FEASIBILITY STUDY	16
4.3 REQUIREMENTS DEFINITION	17
4.4 REQUIREMENTS AND CONSTRAINTS	17
4.5 SYSTEM REQUIREMENTS	18
4.6 DATA SOURCES.....	18
4.7 USERS	18
4.8 RESEARCH METHODS	18
4.9 ANALYSIS OF RESULTS.....	19
 CHAPTER 5.....	 20
5.0 SYSTEM DESIGN	20
5.1 LOGICAL STRUCTURE	20
5.2 PROCESS MODELLING	20
5.3 DATA FLOW DIAGRAMS	21
5.4 SYSTEM DATA FLOW DIAGRAMS	21
5.5 CONCEPTUAL DATA MODELING	23
5.6 DATABASE DESIGN	24
5.7 NORMALIZATION	24
5.8 USER INTERFACE DESIGN	28

CHAPTER 6.....	32
6.0 SYSTEM IMPLEMENTATION	32
6.1 INTRODUCTION	32
6.2 TESTING.....	33
6.3 TEST RESULTS	33
 CHAPTER 7.....	 35
7.0 CONCLUSION.....	35
7.1 ACHIEVEMENTS.....	35
7.2 CONSTRAINTS	35
7.3 RECOMMENDATION	35
7.4 CONCLUSION.....	35
 APPENDIX A	 36
APPENDIX B	37
APPENDIX C	45

List of Abbreviations

SMS	Short Messaging Service
USSD	Unstructured Supplementary Service Data
GSM	Global System for Mobile Communication
MSC	Mobile Switching Center
SSADM	Structured Systems Analysis and Design Methodology
PHP	Hypertext preprocessor
CSS	Cascading Style Sheet
HTML	Hypertext Markup Language

Chapter 1

1.1 Background Information

The forex exchange and stock markets are the largest financial market in the world, with a daily average turnover of well over US\$1 trillion - 30 times larger than the combined volume of all U.S equity markets. Unlike other financial markets, the forex market has no physical location or central exchange.

In Kenya, several companies sell their shares to the public or other companies i.e. subsidiaries. The public invest in these companies by buying their shares and becoming shareholders of these companies. The public are then informed whether or not their shares are making profits or losses by reading the stock market analysis of their respective companies i.e. losers and gainers analysis.

To keep track of these stock market rates, stakeholders use central securities exchange points, media or stock market websites as a source of stock market analysis information in order to speculate whether to buy or sell their current shares and to avoid losses by selling all their shares whenever there is a financial scoop.

Several mobile (Android and IOS) apps have been developed to assist stakeholders and businessmen to keep track of the latest forex and stock market rates. These apps provide the current up-to-date information about the financial markets including news, daily fix logs, and markets trends analysis. The only disadvantage is you have to own or acquire an android or IOS phone in order to use these apps.

1.2 Problem Definition

In Kenya, there are several hundreds of stakeholders. These include the general public, companies, government, banks etc who invest in securities such as shares/stocks, debentures/government bonds, mortgage finance etc. Several investors may not be properly informed about their current investment schemes and therefore may miss making huge profits by selling whenever there is a financial scoop due to lack of proper information symmetry. These investors need to be informed about their current investments by the media, stock securities websites or mobile apps. This information is crucial as it determines whether or not you as an investor/shareholder are making a loss or profit. People invest money based on the information they hold. This problem can be solved either by subscribing to SMS alerts or to download stock market mobile apps that inform you about the current stock market and forex exchange rates.

Available systems are not multiplatform. Current systems are smartphone based thus not suitable for people who don't have either an android or IOS phone. The general public who may want to know the current forex rates have to visit forex websites, banks, forex de bureau. This may be a limiting factor because if you are unable to visit the banks or the forex centers then you have to get the current forex rates online. Fortunately, this problem can be solved by USSD (Unstructured Supplementary Service Data) or SMS alerts.

1.3 Project Goal

The main project goal is to develop a mobile based stock market & forex exchange rates notifier which will be SMS based and enables the general public and investors to check the current forex exchange rates and stock market analysis summaries (losers and gainers).

The system will include an additional SMS alerts module that will be used to notify investors and subscribers about the current forex exchange rates and advise them to sell in the event of a financial scoop.

The new system should be readily available and cheaper than existing systems.

1.4 Project Objectives

The main objective of this project is to build a mobile based stock market and forex exchange rates notifier which is SMS based. The project objectives are as follows:

- Notify users about the current stock market trends and summaries.
- Enable users to subscribe for daily/weekly SMS alerts containing market securities and trends summaries.
- Provide users with the latest forex exchange rates.
- Fetch the current stock prices from Nairobi Securities Exchange website and avail it to the system users.

1.5 Justification

Penetration of mobile phones in Kenya (25.27 million subscribers) has favored the growth of mobile banking and mobile based applications in the country. Several mobile based applications have been developed to ease the day-today activities human beings do.

Several android and IOS financial apps have been developed with one purpose; informing investors about the current financial market rates and summaries. Some of these apps even allow the investors to trade directly from their phones. The only problem is you as an investor or a member of the general public have to own a smartphone in order to use these apps.

The new system will be SMS based, it will offer information about the current stock market prices and forex exchange rates to both smartphone and non-smartphone users. It also enables investors and the general public to subscribe for SMS alerts containing such crucial information. This information includes:

- Technical Stock Data: Refers to data relating to stock only. It includes:
 - The price at the end of the day
 - the volume of the shares traded per day
 - The highest and lowest price of a trading day

- **Forex Data:** It basically includes the current forex rate for a given currency against another currency.

Chapter 2

2.0 Literature Review

Delivering public services through the Short Messaging Service (SMS) channel is becoming popular in developed and developing countries. In Kenya, several banks such as KCB, Barclays Bank of Kenya, SCB etc offer m-banking services which are based on SMS and USSD.

In December 2008, 54 national government agencies of the Philippines were providing SMS services to augment traditional public services; since 2006 Singapore's citizens have been able to access 150 public services through a single SMS number. In Australia, SMS is used for bushfire alerts in Victoria and notification for public transport timetables in Adelaide.

In Ghana, the Philippines and Indonesia, most local authorities provide SMS-based services for listening to people's opinions. In Oman people can apply for jobs via SMS and currently the Bahrain government and the Chichester Council in the UK are developing integrated SMS systems (SMSeGov.info, 2009).

SMS-based services have a strategic role both in developed and developing countries. It has been reported that providing public services through the SMS channel has significantly reduced time and cost; introduced a cheaper, easier and faster information-accessing channel; improved transparency, accountability, communication, and relationship between SMS service providers and their customers.

For developing countries, SMS-based services allow more people to access and to use m-banking services and other services. In the Philippines, for example, people prefer to contact their government using the SMS-based channel (87%) rather than Internet (11%). People prefer a technology channel that is more familiar, simple and easy to use, supports their native language, uses a readily available device and infrastructure and is low cost. Therefore, in order to engage more people, it is argued that SMS-based services should become a front-line system for delivering services in developing countries.

For people with smart phones, SMS based forex and stock notifier can be a popular complementary channel of existing internet based smartphone applications. The advantages of SMS are: it is simple, easy to use, extensive in coverage, reliable in delivering the message, low in cost, and can reach citizens anywhere anytime including areas with no Internet access.

In developed countries SMS-based services are provided to deliver information about emergency situations, reminders, or any other business activity which needs a prompt action by the clients such as a reminder of a tax due date or warnings of extreme weather. Some of the developed countries have also provided SMS-based transaction services such as purchasing a bus ticket or a parking ticket, paying tax, and voting via SMS.

Types of sms-based services include:-

- Information Push

- Information Pull
- State update
- Mobile based systems
- Redirects
- Downloads

Information Push

The main purpose of Information push is to broadcasting information to a group of people. Such information may include bible verses, jokes, pickup lines etc. Additionally, it can also be used for alerts and notifications. For example; weather warnings, coupons, email notification, stock notification, credit card alerts. Users can subscribe and unsubscribe to information push sms services via SMS, web portal or email.

Information Pull (Query)

Is a two way interaction whereby user sends formatted SMS requesting information. For instance a user can send a SMS with the text 'INFO KES' in order to view the current forex rates for Kenyan shillings. Most Information pull services require its users to subscribe or be charged per message they receive (MT).

State Updates

A user sends a message to update the state of a system. An acknowledgement may be returned regarding the status of the update sent. Examples include;

- Voting systems e.g. idols East Africa
- update the state of an website e.g. update Facebook/Twitter status, update personal blog, micro-publishing
- Disaster information e.g. Update disaster locations to inform others via SMS
- SMS forms e.g. medical records, field surveys, business agents

Mobile Based Systems

In such systems users interact with the full system via SMS and have continued presence with the system. Mobile based systems are menu driven with session support and database backed. Examples of Mobile based systems are; Independent MPesa, IM networks, classified ads.

2.1 Money Markets

Stock Markets allow suppliers of funds to get equity or funds from the public. Both common stock and preferred stock offers holders capital gains if stock appreciates in value. They provide a periodic dividend payment; preferred stock is at a fixed rate while common stock varies with time & thus they are uncertain.

In primary stock markets, companies raise funds for the first time and shares are normally sold by underwriters through investment banks. These first time issues are also referred to as Initial Public Offerings (IPOs).

Forex Exchange markets allow currency to be exchanged in order to facilitate international trade. Forex market is a global market and is conducted over the counter through the use electronic platform or by telephone. Unlike stock markets which have tens of thousands of stock to choose from, the Forex market revolves around more or less eight major currencies. Forex trading is always done in pairs, since any trade involves the simultaneous buying of a currency and selling of another currency. The trading revolves around 14 main currency pairs.

These pairs are:

GBP/USD	EUR/JPY
USD/JPY	EUR/GBP
USD/CHF	EUR/CHF
USD/CAD	GBP/JPY
AUD/USD	GBP/CHF
CHF/JPY	EUR/CAD

Where:

USD: United States Dollar

GBP: Great Britain Pound

JPY: Japanese Yen

EUR: Euro

CHF: Swiss Franc

CAD: Canadian Dollar

AUD: Australian Dollar

NZD: New Zealand Dollar

Chapter 3

3.0 Methodology

The methodology to be employed in developing the system is Structured System Analysis and Design Method (SSADM). The system development methodology was suitable due to its structured approach to systems development.

The processes of the system are decomposed by using the DFD technique. SSADM places emphasis upon data, which is more stable than processes, thus enabling more subsequent flexibility. Review is done with the users of the products of each stage.

This methodology is appropriate for the analysis and designs of systems which exist in well structured environment, i.e. objectives are realistic, requirements are consistent and useful, communication is effective and reliable and attitudes are consistent and useful.

The techniques used in SSADM include:

- Logical data modeling
- Data flow modeling
- Entity event modeling

3.1 Stages of SSADM

- **Feasibility study:** Feasibility study enables a project to be developed within tight budgetary and time constraints. It involves a high level analysis of a business area with the aim of determining whether a system can cost-effectively support the business requirements.
- **Requirements analysis:** This stage is broken down further into two modules.
 - *Investigation of the current Environment-* involves learning the terminology of the business, what the users do and how they do it and construction of a data model.
 - *Business System options-* Involves collection of ideas to form a set of two or three options which are presented to the user. Considerations made are: the degree of automation, cost/benefit and the impact of the new system.
- **Requirements Specifications:** The selected way to go is developed and refined. The specifications must be free from error, ambiguity and inconsistency. This also involves building of the logical models for both the data flows and the entity relationship models.
- **Logical System Specification:** This stage is further broken down into two:
 - *Technical System Options-* At this stage a large number of options for the implementation of the new system are generated. The considerations are: the hardware architectures, the software to use, cost implication and the staffing required.
 - *Logical Design-* This level specifies the main methods of interaction in terms of menu structures and command structures. The main interface with the users will

interact with the system. The products of this stage are: data catalog, data structure and a logical process model.

- **Physical Design:** At this stage all the logical specifications of the system are converted to descriptions of the system in terms of real hardware and software. The product is complete. Physical design which could tell software engineers how to build the system specification details of the hardware and software and the appropriate standards.
- **Testing-** The system is tested using test data e.g. entering data into the system and checking whether it performs as expected with the data. After the implementation period, the system is released.

Chapter 4

4.0 System Analysis

4.1 Introduction

System analysis is the study of a system with the aim of getting facts about business activity. The main purpose of system analysis is to get information and determine requirements.

This was the first step of understanding the current available systems and the need for system changes and what the users would like to have in the new system.

Requirements determination involves getting to know what the system should do and generating system functionalities. A feasibility study was conducted by involving users of existing systems. Once this was done, requirements analysis was done to elicit and define user and system requirements. Requirements specification then generated users and system requirements by focusing on their functionalities.

4.2 Feasibility Analysis.

All projects are feasible given unlimited resources and infinite time. Feasibility study enables a project to be developed within tight budgetary and time constraints. This activity was carried out in the following categories:

1. *Economic Feasibility*: This is a cost benefit analysis that identifies the financial benefits & costs associated with the development of the project. This was assessed. Since the project development mainly used open source tools and software, it passed the feasibility test.
2. *Operational feasibility*: This is as an assessment of how a proposed system solves business problems or takes advantage of opportunities. There was sufficient support for the project from users. Users preferred SMS systems to smartphone application systems therefore the system would solve their problems.
3. *Technical Feasibility*: is a measure of the practicality of a specific technical solution and the availability of technical resources and expertise. Evaluating technical feasibility was the challenging part of feasibility study. This was due to fact that the proposed system was to be based on USSD but it had to be changed to SMS because of the practical drawbacks of implementing USSD and its costs.
Therefore SMS was the only option which had less implementation issues and costs.
4. *Schedule Feasibility*: This was an assessment of timeframe & project completion dates with respect to the required functionalities. It was estimated and concluded the project functionalities would be completed before the deadline.

4.3 Requirements Definition

The purpose of requirements definition was to identify and extract system problems and solution requirements from the user community. Problem analysis was carried out to identify the problem and understand any constraints that may have limited the solution. Preliminary analysis on both functional and non-functional analyses was carried out with a view to getting an overall understanding of the problem domain.

4.4 Requirements and Constraints

Functional Requirements

- Provide the current forex rates to mobile users.
- Allow users to subscribe to daily SMS alerts containing the forex rates.
- Allow users to unsubscribe from daily SMS alerts containing the forex rates.
- Provide help and error handling mechanisms that guide the user in the event of an error occurrence.
- Provide the latest stock prices to mobile users.
- Allow users to subscribe to daily SMS alerts containing the stock market trends.
- Provide a summary of all stock prices from NSE.
- Allow users to unsubscribe from daily SMS alerts containing the stock market trends.
- Provide help and error handling mechanisms that guide the user in the event of an error occurrence.

Nonfunctional Requirements

- The system should be able to provide real time responses to user requests without failure.
- Error messages sent back to users should be user friendly and provide better explanation for the error that occurred.

4.5 System Requirements

The system requirements was divided into modules each described below.

1. *Forex Module:* The forex module was required to provide the following functionalities.
 - Provide the current forex rates to mobile users.
 - Allow users to subscribe to daily SMS alerts containing the forex rates.
 - Allow users to unsubscribe from daily SMS alerts containing the forex rates.
 - Provide help and error handling mechanisms that guide the user in the event of an error occurrence.
2. *Stock Module:* The specifications for stock module were as follows:
 - Provide the latest stock prices to mobile users.
 - Allow users to subscribe to daily SMS alerts containing the stock market trends i.e. the losers and gainers.
 - Provide a summary of all stock prices from NSE.
 - Allow users to unsubscribe from daily SMS alerts containing the stock market trends.

- Provide help and error handling mechanisms that guide the user in the event of an error occurrence.
3. *Web Simulator:* The SMS simulator module involved integrating all modules of the system and coordinating them to work together. This module allows a web user to simulate the functionality of the system without sending event an SMS. It provides a form where by a user inputs a mobile number for the simulator and the SMS to send to the system. The system in turn displays the expected results according to the previous SMS sent by the user.

These modules had to be integrated so as to meet all the objectives of the system. This was done so as to enable the simulator users to access all the functionalities of the system from a single interface.

4.6 Data Sources

Data is the main backbone for this system; without it the system would not function hence rendered useless. The main data source for this system is forex RSS feeds from themoneyconverter.com and stocks data from rich.co.ke and nse.co.ke

The forex data in the form of RSS is parsed using php script and stored in a database locally so is the stock data.

4.7 Users

The users for the system include:-

- forex traders
- stock traders
- web users/simulator users

4.8 Research Methods

Different research techniques were used in investigating and research for system requirements gathering.

- *Interviews:* Interviews are a fact-finding technique whereby the systems analysts collect information from individuals through face-to-face interactions. Interviews were held with several businesspersons at random locations to determine their understanding on both stock and forex markets and how they handle and react to money market information in their day-to-day activities. Structured interviews were used. Specific set of questions were used during the interview. Different opinions were obtained from shareholders and forex traders.
- *Questionnaires:* These are special-purpose documents that allow the analyst to collect information and opinions from respondents. Questionnaires were administered to assist in gathering information from business students. Although response was not good as in the interviews, a lot was learned.
- *Observations:* Observation is a fact-finding technique wherein the systems analyst either participates in or watches a person perform activities to learn about the system. Users were directly observed as they conducted their normal business.

4.9 Analysis of Results

Results gathered were analyzed and helped greatly in determining system requirements. Feedback from questionnaires was not as great but the little gathered was relevant. This helped greatly in determining the expectations from the new system.

It was necessary to obtain the necessary system requirements at an early stage to prevent inefficiencies in system requirements structuring and several other factors which are a result of incorrect requirements. These drawbacks aided in understanding the expectations of the proposed system.

Chapter 5

5.0 System Design

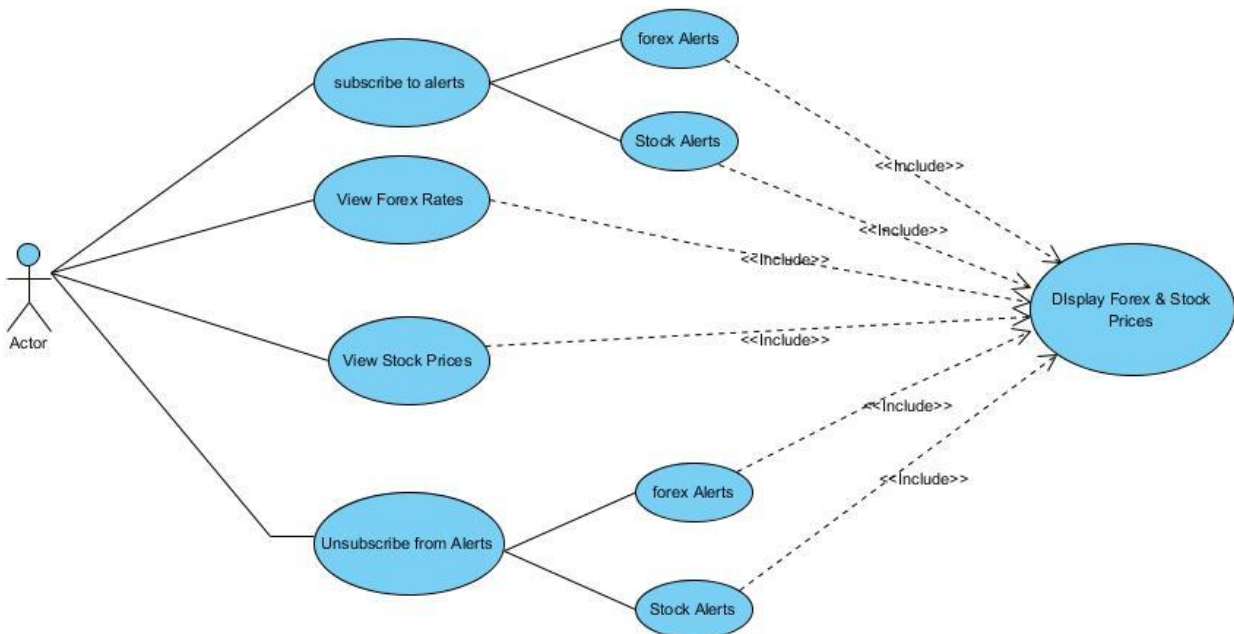
System design focuses on representing the information gathered as part of requirements determination. Designs made included:

- Logical structure
- Process models

5.1 Logical Structure

This structure defines the relationships among the semantic elements of the system. The logical structure of the system is modeled using a use model. Use case diagram shows how the different actors of the system interact with the system.

The use case model for the entire system is showed below:



5.2 Process Modeling

Process modeling involves graphically representing the functions, or processes which capture, manipulate, store, and distribute data between a system and its environment and between components within a system.

The following aspect was considered in process modeling of the system functionalities.

- Data Flow Diagrams(DFD)

5.3 Data Flow Diagrams (DFD)

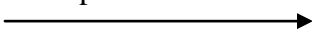
A data flow diagram is graphical tool used to describe and analyze the movement of data through a system. These are the central tool and the basis from which the other components are developed. DFDs are flexible diagramming tools and they capture the process model using only four symbols.

Definition and Symbols

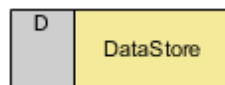
DFDs encompasses only four symbols in representations

- Data flows
- Data stores
- Processes

Data Flow: - Represents data in motion, from one place of the system to another.

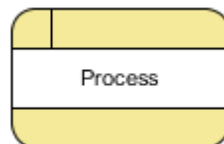
Symbol: 

Data Store: - Represents stored data e.g. table in a database



symbol:

Process: - Represents the action being performed on data so that they can be stored, transformed, or distributed.

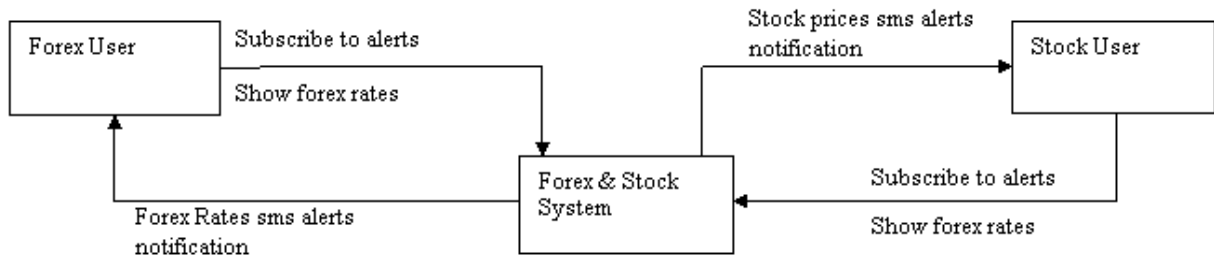


Symbol

5.4 System Data Flow Diagrams

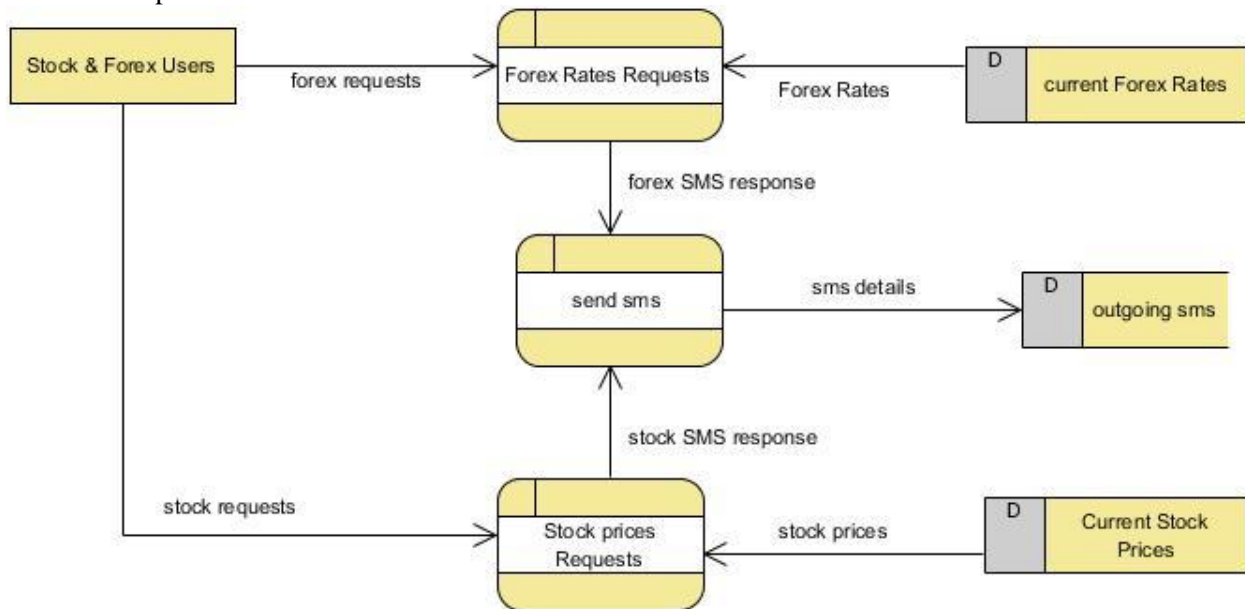
Context Diagram

The module has external entities forex and stock users. These entities interact with the system as shown:



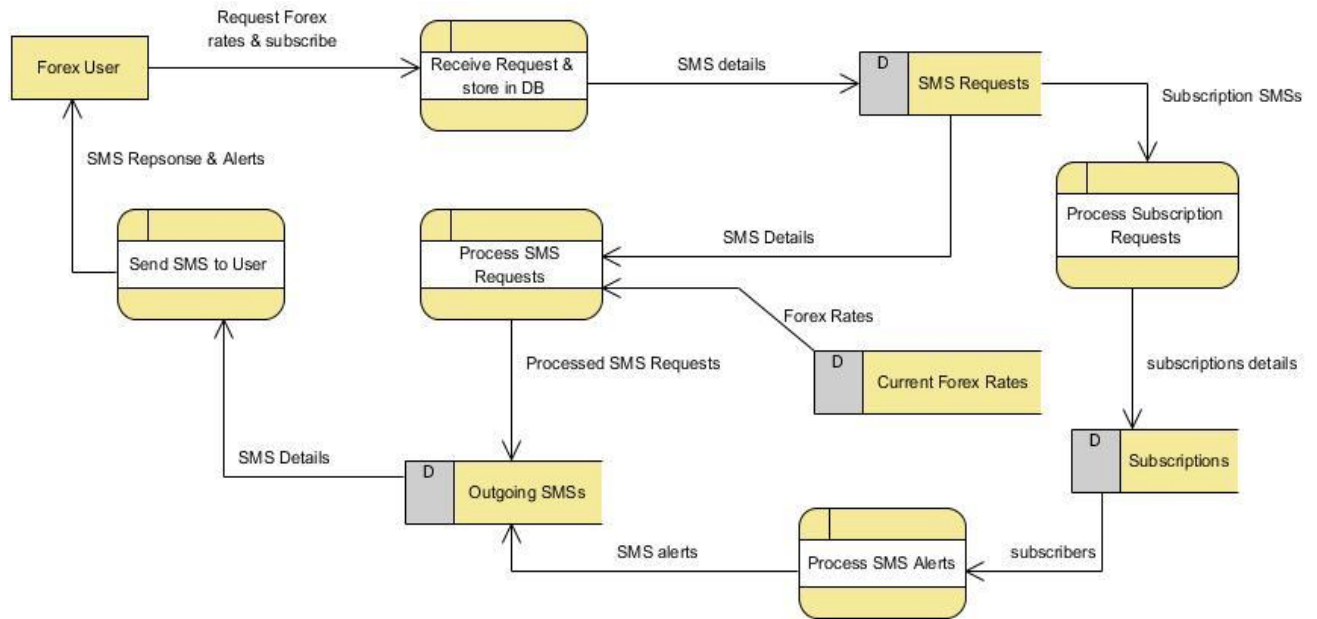
Level 0 DFD

Below is a representation of level 0 DFD

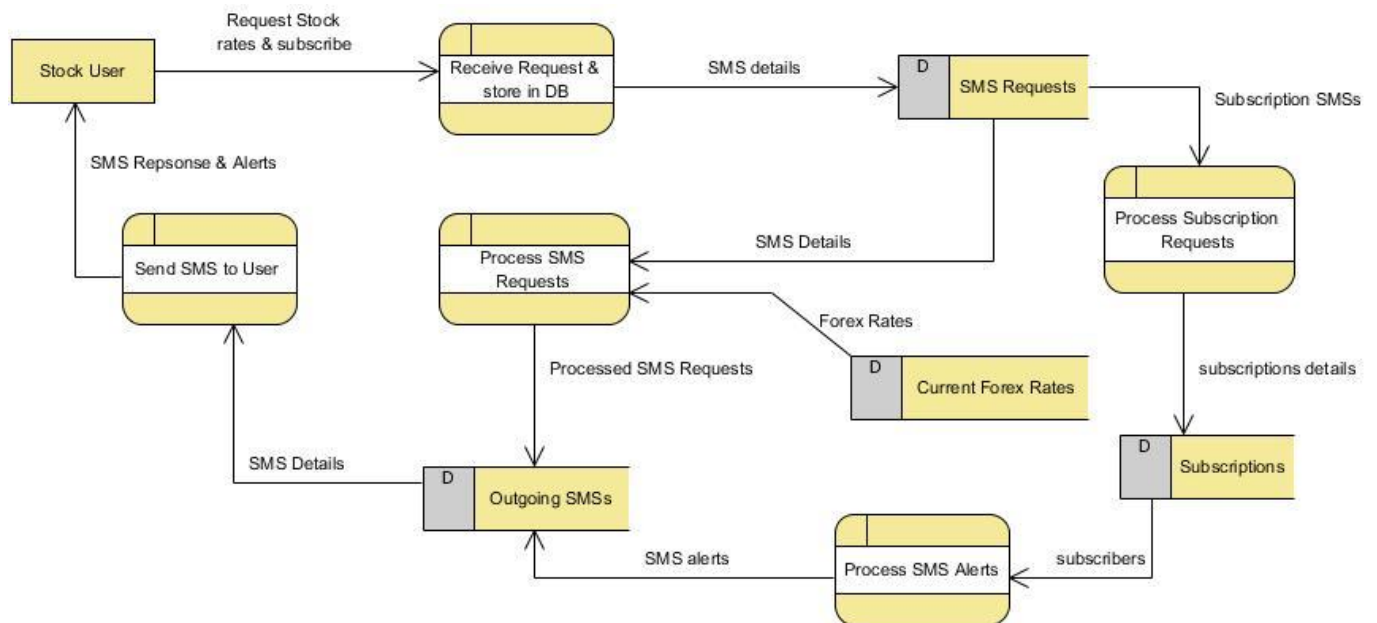


Level 1 DFD

Below is a level 1 DFD for forex and stock module



Forex Module Level 1 DFD



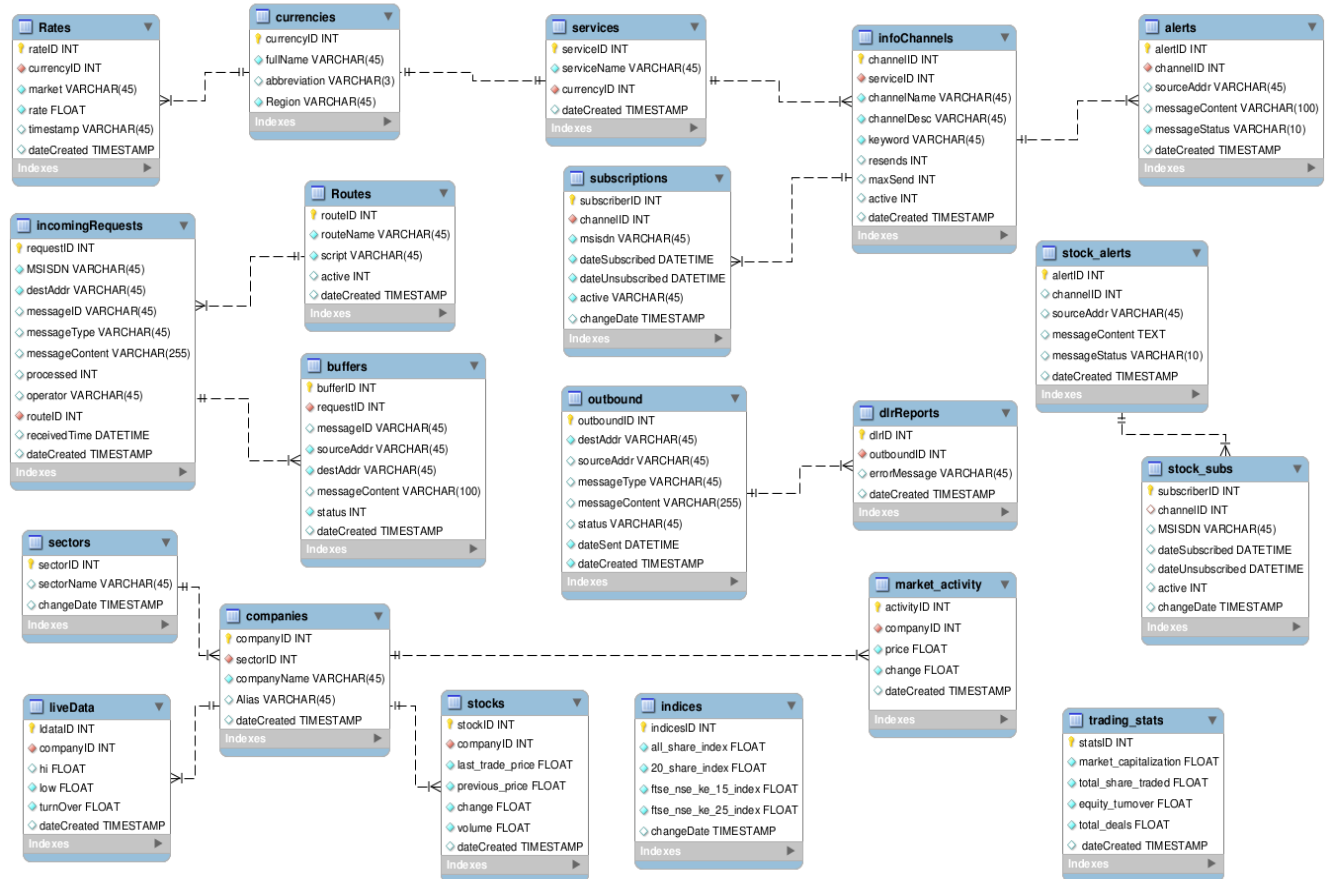
Stock Module Level 1 DFD

5.5 Conceptual Data Modeling

Introduction

Conceptual data modeling concentrates on data supporting data flows and processing captured in DFDs. A conceptual data model is a representation of organizational data. It shows rule about the meaning and interrelationships among data.

Entity Relation Diagram (ERD)



5.6 Database Design

The relational database model was used as the style for logical database model. Normalization was carried out to the Third Normal Form.

5.7 Normalization

Is a process of converting table entities to a standard form. The process is used to handle the problems that can arise due to data redundancy. Relations were decomposed by splitting relations into multiple relations to eliminate anomalies and maintain data integrity.

Descriptions of the normalized tables:

Database Model

Rates table

Field	Type	Null	Key	Default	Extra
rateID	int(11)	NO	PRI	NULL	auto_increment
currencyID	int(11)	NO	MUL	NULL	
market	varchar(45)	NO		NULL	
rate	float	NO		NULL	
timestamp	varchar(45)	YES		NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

currencies table

Field	Type	Null	Key	Default	Extra
currencyID	int(11)	NO	PRI	NULL	auto_increment
fullName	varchar(45)	NO		NULL	
abbreviation	varchar(3)	YES	UNI	NULL	
Region	varchar(45)	NO		NULL	

services table

Field	Type	Null	Key	Default	Extra
serviceID	int(11)	NO	PRI	NULL	auto_increment
serviceName	varchar(45)	NO		NULL	
currencyID	int(11)	NO	MUL	NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

alerts table

Field	Type	Null	Key	Default	Extra
alertID	int(11)	NO	PRI	NULL	auto_increment
channelID	int(11)	NO	MUL	NULL	
sourceAddr	varchar(45)	YES		254714044696	
messageContent	varchar(255)	YES		NULL	
messageStatus	varchar(10)	NO		send	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

incomingRequests table

Field	Type	Null	Key	Default	Extra
requestID	int(11)	NO	PRI	NULL	auto_increment
MSISDN	varchar(45)	NO		254710900900	
destAddr	varchar(45)	NO		254714044696	
messageID	varchar(45)	YES		NULL	
messageType	varchar(45)	YES		SMS: TEXT	
messageContent	varchar(255)	YES		NULL	
processed	int(11)	YES		5	
operator	varchar(45)	YES		Safaricom	
routeID	int(11)	NO	MUL	NULL	
receivedTime	datetime	YES		NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	

companies table

Field	Type	Null	Key	Default	Extra
companyID	int(11)	NO	PRI	NULL	auto_increment
sectorID	int(11)	NO	MUL	NULL	
companyName	varchar(45)	NO		NULL	
Alias	varchar(45)	YES	UNI	NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

stocks

Field	Type	Null	Key	Default	Extra
stockID	int(11)	NO	PRI	NULL	auto_increment
companyID	int(11)	NO	MUL	NULL	
last_trade_price	float	NO		0	
previous_price	float	NO		0	
change	float	NO		0	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

liveData table

Field	Type	Null	Key	Default	Extra
ldataID	int(11)	NO	PRI	NULL	auto_increment
companyID	int(11)	NO	MUL	NULL	
hi	float	YES		0	
low	float	NO		0	
turnOver	varchar(50)	YES		NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

alerts table

Field	Type	Null	Key	Default	Extra
alertID	int(11)	NO	PRI	NULL	auto_increment
channelID	int(11)	NO	MUL	NULL	
sourceAddr	varchar(45)	YES		254714044696	
messageContent	varchar(255)	YES		NULL	
messageStatus	varchar(10)	NO		send	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

incomingRequests table

Field	Type	Null	Key	Default	Extra
requestID	int(11)	NO	PRI	NULL	auto_increment
MSISDN	varchar(45)	NO		254710900900	
destAddr	varchar(45)	NO		254714044696	
messageID	varchar(45)	YES		NULL	
messageType	varchar(45)	YES		SMS: TEXT	
messageContent	varchar(255)	YES		NULL	
processed	int(11)	YES		5	
operator	varchar(45)	YES		Safaricom	
routeID	int(11)	NO	MUL	NULL	
receivedTime	datetime	YES		NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	

indices table

Field	Type	Null	Key	Default	Extra
indicesID	int(11)	NO	PRI	NULL	auto_increment
all_share_index	float	NO		0	
20_share_index	float	NO		0	
ftse_nse_ke_15_index	float	NO		0	
ftse_nse_ke_25_index	float	NO		0	
changeDate	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

trading stats

Field	Type	Null	Key	Default	Extra
statsID	int(11)	NO	PRI	NULL	auto_increment
market_capitalization	float	NO		NULL	
total_share_traded	float	NO		NULL	
equity_turnover	float	NO		NULL	
total_deals	float	NO		NULL	
dateCreated	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

outbound table

Field	Type	Null	Key	Default	Extra
outboundID	int(11)	NO	PRI	NULL	auto_increment
destAddr	varchar(45)	NO		NULL	
sourceAddr	varchar(45)	YES		254714044696	
messageType	varchar(45)	YES		SMS:TEXT	
messageContent	text	YES		NULL	
status	varchar(45)	YES		send	
dateSent	datetime	NO		NULL	
dateCreated	timestamp	NO		CURRENT_TIMESTAMP	

dLrReports table

Field	Type	Null	Key	Default	Extra
dLrID	int(11)	NO	PRI	NULL	auto_increment
outboundID	int(11)	NO	MUL	NULL	
errorMessage	varchar(45)	YES		NULL	
dateCreated	timestamp	YES	UNI	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

sectors table

Field	Type	Null	Key	Default	Extra
sectorID	int(11)	NO	PRI	NULL	auto_increment
sectorName	varchar(45)	YES		NULL	
changeDate	timestamp	YES		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

Table Name	Description
Rates	Stores forex rates for each currency.
currencies	Stores details about all the currencies in forex markets.
Services	Contains a list of services that can be subscribed to.
Alerts	contains a template of the subscription sms to be sent out
incomingRequests	Stores all the incoming sms requests to be processed.
Buffers	Temporarily holds messages being processed.
Outbound	stores messages to be sent out or that have been sent out
dlrReports	stores delivery reports of sent messages and their respective error messages
Sectors	Contains a list of sectors in NSE eg Agricultural
companies	contains a list of companies that trade at NSE
Stocks	Contains stock prices for various companies and the End of day
liveData	Contains live Data feed from NSE
Indices	Contains NSE indices..such as FTSE 100 index
Trading_stats	contains other trading statistics e.g. equity turnover
Market activity	stores losers and gainers of the day

5.8 User Interface Design

The interface for the web SMS simulator module was developed using HTML & CSS. Sample of the main interfaces are shown below;

SMS Simulator

Phone Number:

Message:

SMS Simulator

Sender Number:

Receiver Number:

Message:

You have successfully
subscribed to stock Alerts

Subscribing to stock alerts

SMS Simulator

Phone Number:

Message:

STOCK STOP

SMS Simulator

Sender Number:

Receiver Number:

Message:

You have successfully
unsubscribed from stock Alerts

Unsubscribing from stock alerts

SMS Simulator

Phone Number:

Message:

SMS Simulator

Sender Number:

Receiver Number:

Message:

Subscribing to Kenyan Shillings forex rates alerts

SMS Simulator

Phone Number:

Message:

SMS Simulator

Sender Number: 254714044696

Receiver Number: +254738530962

Message: You have successfully unsubscribed from KES alerts

New SMS Back

Unsubscribing from Kenyan Shillings forex rates alerts

SMS Simulator

Phone Number: +254738530962

Message: STOCK SCOM

Send SMS Cancel

SMS Simulator

Sender Number: 254714044696

Receiver Number: +254738530962

Message: Company: Safaricom Ltd Hi: 3.5
Low: 3.5 TurnOver: 59,150
Timestamp: 2012-05-16 09:58:38

New SMS Back

Querying the current stock prices for Safaricom shares

Chapter 6

6.0 System Implementation

6.1 Introduction

This is the process where the physical design specifications were turned into a working computer code.

Software Platform used

- Linux OS
- Mysql database management system
- Python
- php 5.3 and above
- Apache
- Ozeki NG SMS gateway
- Eclipse IDE
- Open Office

Hardware Platform used

- Intel core 2 Duo
- 2 GB RAM
- 3G Modem

Choice of System Development Tools

PHP: It is a dynamic Web-development and scripting language that is easy to learn and use. It is compatible with Mysql database and gives user easy interaction with the internal components of the system.

Python: The Python programming language manages to reconcile many apparent contradictions: it's elegant and pragmatic, simple and powerful, a high-level language that doesn't get in your way when you want to fiddle with bits and bytes, suitable for programming novices and great for experts too.

Mysql: The Mysql database management system has become quite popular in recent years. This has been true especially in the Linux and open source communities, but Mysql presence in the commercial sector now is increasing as well. It is well liked for several reasons: Mysql is fast, and it's easy to set up, use, and administrate. Mysql runs under many varieties of UNIX and Windows, and Mysql-based programs can be written in many languages. Mysql is especially heavily used in combination with a web server for constructing database-backed web sites that involve dynamic content generation.

6.2 System Testing

Testing is aimed at detecting and correcting system errors. Most of the testing was integrated with coding and happened in parallel.

The major function of testing is to confirm whether the system satisfies requirements.

- Functional Testing: - Each module was independently tested to see if it performed as per the specifications.
- Integrated Testing: This was done systematically by testing each module after coding.
- System Testing: This involved testing the entire system. The system was tested as a whole.

6.3 Test Results

The following tests were carried out:

Test Case ID	Scenario	Forex currency keyword	Stock keyword	Expected Result
Scenario 1	View forex rate	V	N/A	Display forex rates for the specified currency
Scenario 2	Invalid /unsupported forex currency	I	N/A	Error Message: sent to user.
Scenario 3	View stock prices	N/A	V	Display current stock prices for the specified company
Scenario 4	Invalid/unsupported company	N/A	I	Error Message: sent to user.
Scenario 5	Subscribe to forex rates SMS alerts	V	N/A	Successful subscription message sent to user.
Scenario 6	Invalid/unsupported forex rate keyword.	I	N/A	Error Message sent to user.
Scenario 7	Subscribe to stock prices SMS alerts	N/A	V	Successful subscription message sent to user.
Scenario 8	Unsubscribe from stock prices SMS alerts	N/A	V	Successful unsubscription message sent to user.
Scenario 9	Unsubscribe from forex rate SMS alerts	V	N/A	Successful unsubscription message sent to user.

Scenario 10	Invalid/unsupported forex currency	I	N/A	Basic flow
----------------	---------------------------------------	---	-----	------------

Chapter 7

7.0 Conclusion

7.1 Achievements

- The main achievement of this project was the development of a mobile based stock market and forex exchange rates notifier which is an SMS based system. The system is flexible enough to act as a backend or a data source for forex or stock smartphone applications.
- Developing web based sms simulator that enable web users to simulate the operation of the system.
- Acquired recommendable knowledge in SMS system development and design.
- Implementation of my idea that can solve a real life problem in the business sector.
- Appreciation of available open source technology to develop systems for SME(Small and Medium Enterprises).

7.2 Constraints

- High cost of obtaining a USSD short code was a major constraint. The proposed system was supposed to be USSD based but due to limited USSD simulation tools and its high costs, I had to change the system to SMS based. The system had to be adapted to SMS based in order to achieve the objectives and produce a working prototype within the limited timeline.
- Getting a reliable real-time stock data was also an issue. NSE real-time stock data is not available to the public, so the data is only available on websites of data venders authorized by NSE.

7.3 Recommendation

- An additional android application can be included in the system for smartphone users. This can be possible since the system can act as a backend for the smartphone application.

7.4 Conclusion

- The mobile based stock and forex notifier developed in this system consists of the services needed by forex and stock traders. However a few functionalities need to be implemented to solve some aspects in the real world in regards to stocks trading but have been left out due to the scope and time constraints.

APPENDIX A

Reference and Bibliography

Christopher A. Jones and Fred L. Drake, Jr. Python & XML(1st ed.). O'Reilly.

Guide to Online Forex Trading. (n.d.). Forex Trading. Retrieved February 24, 2012, from <http://www.gcitrading.com/gci-forex-ebook.pdf>

Prospectus for the 2nd and 15 year bond. (n.d.). <http://www.nse.co.ke/media-center/press-release.html> . Retrieved February 24, 2012, from <http://www.nse.co.ke/media-center/press-release.html>

Ozeki NG SMS gateway www.ozeki.hu

APPENDIX B

User and Technical Manual

Web server installation & configuration

For the web server to run, it requires: Apache Web Servers Software, MySQL database server and PHP Engine. This is provided in one package known as XAMPP 5.2.1 for Microsoft Windows which was used in this case.

SQL to SMS Gateway configuration - MySQL overview

This shows how to configure the SMS gateway to be able to send SMS messages using a Mysql database server. Installation steps are provided, the configuration details such as the database connection string, database connection to Mysql. To send and receive SMS from a Mysql database you need to install the MyODBC driver to your computer. This driver was downloaded from the MySQL website. In this case MyODBC v5.1 was used, because it supports unicode characters. After the MyODBC driver was installed on the computer, the database was configured as database user in the SMS Gateway.

During the configuration the database connection type, you should choose is:
Odbc

The database connection string used is:

Driver={MySQL ODBC 5.1 Driver};Server=localhost;Database=hafs;User=root;Password=;Option=4;

Database connection | SQL for sending | SQL for receiving | Logging | Advanced

Connection

Please specify the database connection string. You can use both FileDB and ODBC connection strings. Connection string information for various database servers is available at <http://www.ozekisms.com/index.php?owpn=171>

Connection string type: Odbc

Connection string:

Driver={MySQL ODBC 5.1 Driver};Server=127.0.0.1;Database=ozekisms;User=ozeki;Password=abc123;Option=4;

Date format

Please specify the date format used to create the date value for the SQL statements. Formatting information is available at <http://www.ozekisms.com/index.php?owpn=227>

Date format string: yyyy-MM-dd HH:mm:ss

☒ Enable on startup.

OK Cancel

Figure 1 - Specifying the connection string for the Mysql database

Database configuration form is as below.

```
CREATE TABLE `outbound` (
  `outboundID` int(11) NOT NULL AUTO_INCREMENT,
  `destAddr` varchar(45) NOT NULL COMMENT 'receiver of the text',
  `sourceAddr` varchar(45) DEFAULT '254714044696',
  `messageType` varchar(45) DEFAULT 'SMS:TEXT',
  `messageContent` text,
  `status` varchar(45) DEFAULT 'send' COMMENT 'send; sending; sent; notsent; delivered;',
  `dateSent` datetime NOT NULL,
  `dateCreated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`outboundID`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1
COMMENT='stores messages to be sent out or that have been sent out'
```

Download and Install Ozeki NG SMS Gateway from ozeki [website](http://www.ozekisms.com) .

Logging In

After you have downloaded and installed Ozeki NG - SMS Gateway, the first thing you will do is login to the user interface (the management console). You can login using Internet Explorer or Mozilla Firefox, by opening the login window. The login window can be accessed at the <http://127.0.0.1:9501> URL, but in most cases it will show up right after the installation of the program, along with the Installation Complete dialog box. You can also open the login window by clicking the Manage Ozeki NG program.

After you have downloaded and installed Ozeki NG - SMS Gateway, the first thing you will do is login to the user interface (the management console). You can login using Internet Explorer or Mozilla Firefox, by opening the login window. The login window can be accessed at the <http://127.0.0.1:9501> URL, but in most cases it will show up right after the installation of the program, along with the Installation Complete dialog box. You can also open the login window by clicking the Manage Ozeki NG program icon in Start / All Programs / Ozeki NG



Figure 1 - Opening the program

If the login window does not show up, you can try to open it by entering its URL in the browser: <http://127.0.0.1:9501/>. If it still won't show up, you should consult the Login problems page. You are very likely to find some advice there to solve this problem. In the login window, enter your username and password in the respective fields. The default username is admin, and the default password is abc123. You can change the password in the Authentication tab of a Standard user.

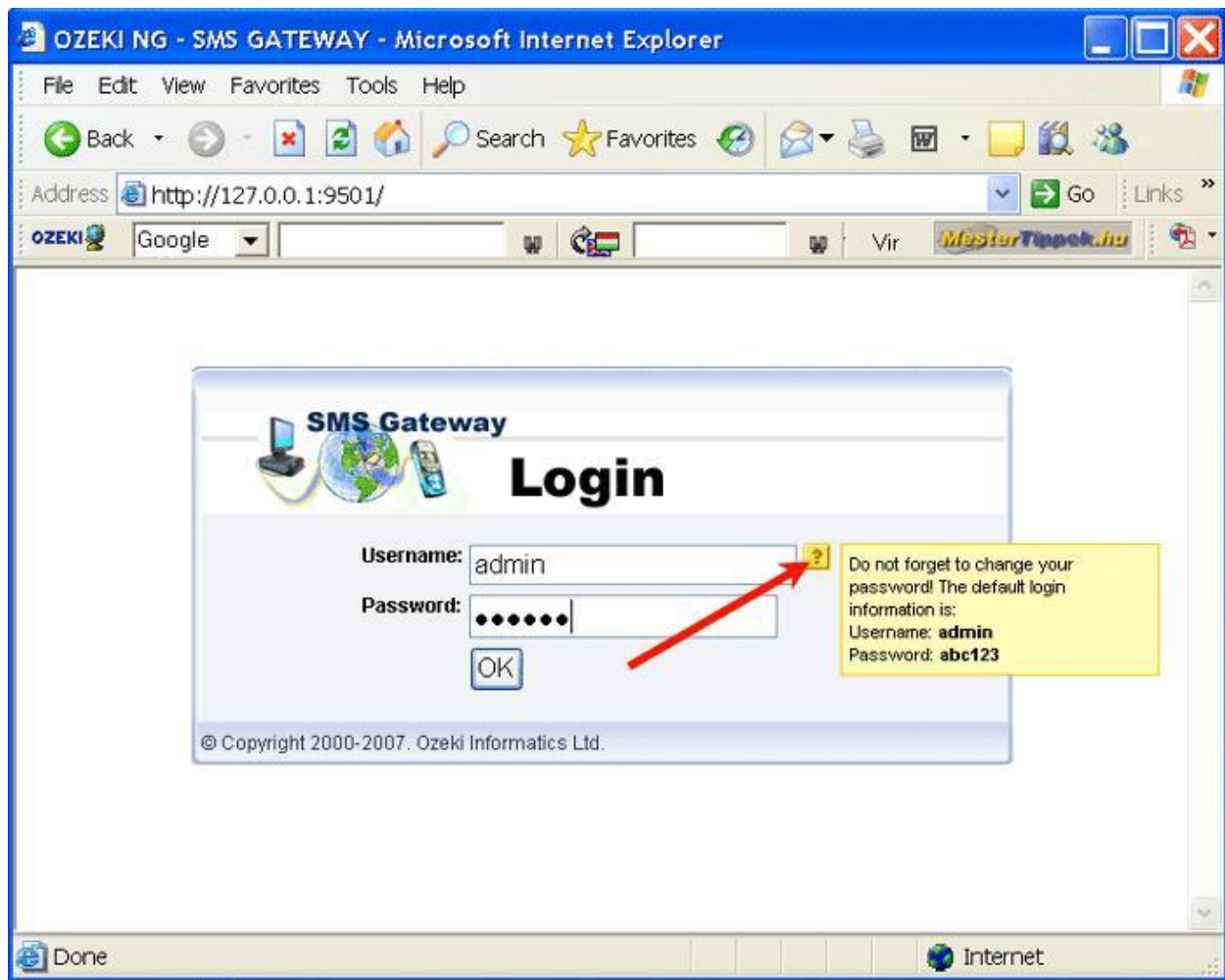


Figure 2 - Reminder to change the password

If you misspell your password, the login window will show up with a blank username and password field again.

Basic Configuration Steps

Once you have passed the login screen, you will see an interface called Management Console. One of the components of this interface is the Management Console tab. If you click it, you will get back to this starting screen. On the right of the title bar, you can see the name of the logged-in user and a link to log out (Figure 3).

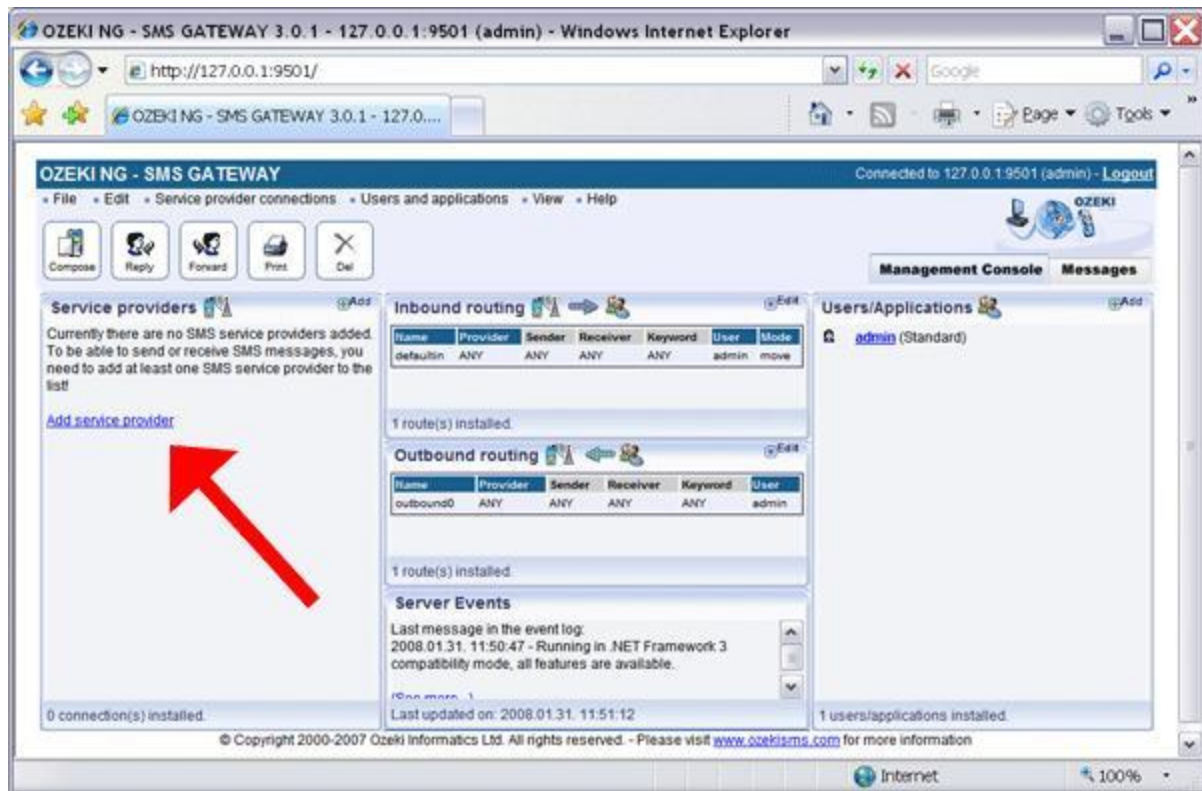


Figure 3 - Management Console

To be able to send SMS messages, you need to install and configure a service provider connection. You can do it by clicking the Add service provider link or the Add button in the top right-hand corner of the Service provider's panel on the left of the interface. You can also add a service provider connection by clicking the Add service provider item in the Service providers menu. After you have clicked one of these, the Add service provider panel will show up on the right of the interface. The panel contains a list of service provider connection options you can install and use for communication with the SMS network.

To send and receive SMS messages using a GSM modem, you need to attach a GSM Modem for SMS messaging to your computer with a data cable or Bluetooth connectivity. In this case I used a GSM phone. To install a GSM phone, click the Install link in the respective entry in the list (Figure 4).

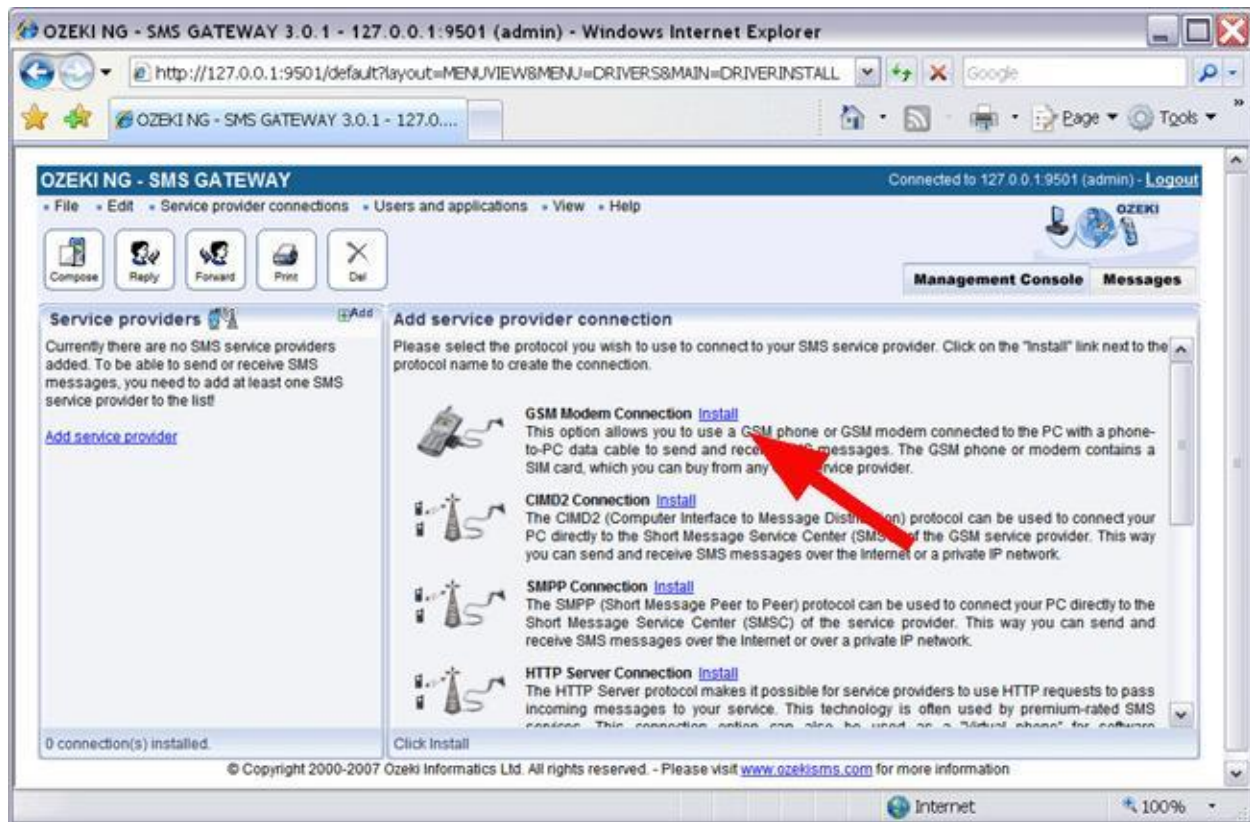


Figure 4 - Installing a service provider connection

To send and receive SMS messages using a GSM device (phone), you need to select and install the GSM phone in the list e.g. Nokia N70 in this case. When you have installed the GSM phone connection, a configuration panel will show up on the right of the interface. On this form you need to select the port your device is connected to and you need to click Auto detect. The Auto detect feature is going to recognize the GSM phone you have attached to your computer.

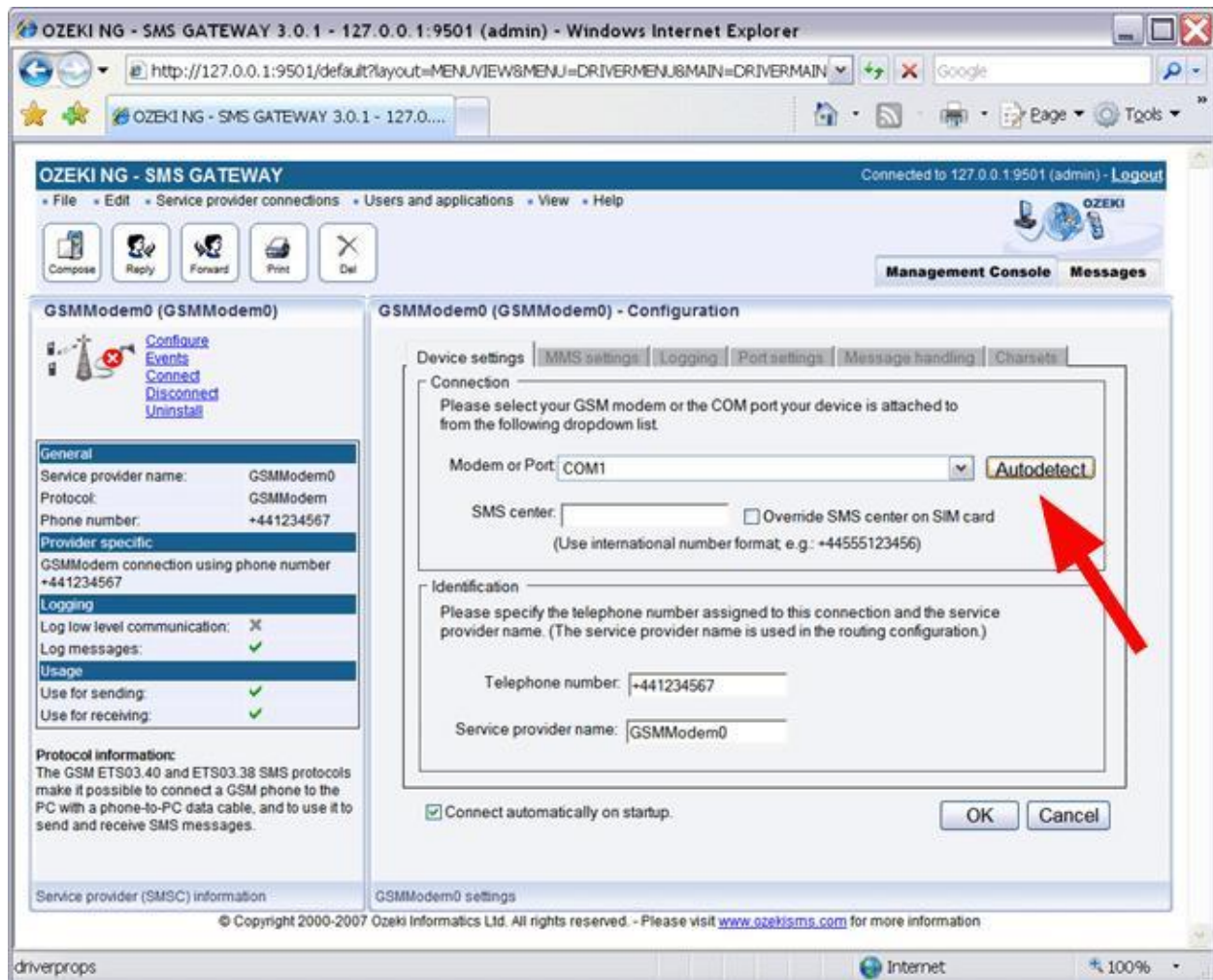


Figure 5 - Click Auto detect to configure your modem.

After the phone has been detected, specify the telephone number assigned to your SIM Card and the service provider connection name. (You can pick any name you want to identify the connection)

Enter the information in the respective edit boxes. Remember to enter the phone number in international format: starting with "+". Note that the phone number to be specified in the Telephone number edit box is a mandatory value. If you leave the edit box blank, and if the outgoing SMS message does not carry a sender phone number either, the sending will be unsuccessful. Once you have configured the service provider connection, you can get back to the Management Console interface by clicking the Management Console tab in the top right hand corner. After your connection is configured, you can send messages.

Configuring php scripts

Copy the forex_stock folder to the webroot. In windows the webroot is C:\wamp\www\

In Unix system, run the command

```
cp -r forex_stock /var/www/html/
```

Change the permissions to 755 by executing the following command:
`chmod 0755 /var/www/html/forex_stock/`

APPENDIX C

SAMPLE PROGRAMS

Php code to connect to database

```
<?php
###logging to file
include 'global_functions.php';
###connects to db
$host="localhost";
$username="root";
$password="";
$db="sentinel";
$connect=mysql_pconnect($host,$username,$password);
if(!$connect){

    $logs="DB error: ".mysql_error()."\n";
    flog($logs,$_SERVER["SCRIPT_FILENAME"]); //log to file
    die("DB error: ".mysql_error()."\n");

}
if(!mysql_select_db($db)){
    $logs="DB error: ".mysql_error()."\n";
    flog($logs,$_SERVER["SCRIPT_FILENAME"]); //log to file
    die("DB error: ".mysql_error()."\n");
}
?>
```

Python code to connect to database and store stock prices data

```
#!/usr/bin/python
import sys
import MySQLdb
from filter_stock_data import sqlData
#connect to db
try:
    db = MySQLdb.connect(host="localhost",user="root",password="",db="sentinel")
    cursor = db.cursor()
except MySQLdb.Error, e:
    print "%d: %s" % (e.args[0],e.args[1])
    sys.exit(1)
x=0;
try:
    status = "new"
    for x in range(0,len(sqlData)):
        #print"Details:",sqlData[x]
        sql="insert into tempData(company,hiData,loData,turnOver,timestamp,status) values
('"+sqlData[x][0].strip()+"','"+sqlData[x][1].strip()+"','"+sqlData[x][2].strip()+"','"+sqlData[x][3].strip()+"'
','"+sqlData[x][4].strip()+"','"+status+"')"
        print "SQL:",sql
```

```

        cursor.execute(sql)
    cursor.close()
except MySQLdb.Error, e:
    print "%d: %s" % (e.args[0],e.args[1])
    sys.exit(1)

```

Python code to parse html pages and fetch data

```

from HTMLParser import HTMLParser
from htmlentitydefs import name2codepoint
import urllib
import cgi
def recordData(data):
    outputFile=open('liveData.log','a')
    outputFile.write(''.join('%s' % x for x in data))
    outputFile.close()

class StockHTMLParser(HTMLParser):
    def handle_starttag(self,tag,attrs):
        print "Start Tag: ",tag
        liveData = "Start Tag: ",tag
        for attr in attrs:
            print "attr:",attr
            liveData += "attr:",attr

    #recordData(liveData)
    def handle_endtag(self,tag):
        print "End Tag: ",tag
        liveData = "End Tag: ",tag

    #recordData(liveData)
    def handle_data(self,data):
        print "Data: ",data
        if data:
            liveData = "\nData: ",data
            recordData(liveData)
    def handle_comment(self,data):
        print "Comment: ",data
        liveData = "Comment: ",data

    #recordData(liveData)
    def handle_entityref(self,name):
        if name == "G":
            #name = cgi.escape("&")
            name = "nG"
        else:
            c = unichr(name2codepoint[name])
            print "Named ent:",c
    def handle_charref(self, name):
        if name.startswith('x'):
            c = unichr(int(name[1:],16))

```

```

else:

    c = unichr(int(name))
    print "Num ent: ",c
    liveData = "Num ent: ",c
    #recordData(liveData)

def handle_decl(self,data):
    print "Decl: ",data
    liveData = "Decl: ",data

    #recordData(liveData)
def emptyFile(self):
    File=open('liveData.log','w')
    File.close()
parser = StockHTMLParser()
#empty file
parser.emptyFile()
#proxy settings
proxies = {'http': 'http://P15%2F35231%2F2010%40students:rzd%40uon@proxy.uonbi.ac.ke:80/'}
urlopener = urllib.FancyURLopener(proxies)
#open html file
stocks_html = urlopener.open("http://live.rich.co.ke/tdy_turnover2.php").read( )
parser.feed(stocks_html)

```