02/09/14

"Stock Management System"

Project Proposal and Plan

1. Introduction

There are already several applications out there that help you find to manage stocks. There are not however, any System that not only helps you find stocks across the globe, but this finds you stock from different location and helps to manage them.

This will be the purpose of the "Managing Stocks" belonging to different locations across the globe. **It will be a system that helps you buy by location, sell by location, find by location, Indicate stock by location, monitor favorites by any location, represents stock data's graphical, Stock alerter , instant updates of stock.**

"Stock Management System" will leverage existing technologies in order to add the new dimension of Stocking. The creation of "Stock Management System" will involve several developmental tiers: UI, User Management System, Database, and Web Screens.

2. Project Scope and Objectives

Stock Management System" will function first by customer's creating an account in the application. This account creation will involve communicating with a web server that houses a database that maintains the customer's information. Once the customer has an account, several inputs relating to stocks are obtained. Further his personal details are recorded in the database and can be used for stocking in future. Stocks will be queried through the Oracle and Java technologies and represented to the User with best GUI's. From there a customer can select a stock and choose the further business actions he wants to execute. Each execution can be pulled from the stock tables. Stocks are instantly updated by build in Oracle jobs which synchronizes the stock activity in the market there on providing the most updated stock details. "

3. Project Details:

1. Website:
    - The home page of the website displays all the updated stock values of different companies.
    - The page includes a graphical representation of the variations in the stock market of every company.
    - To make a transaction, the webpage directs the user to a login page or to register for a new account.
    - The website shows the stocks activity every time the page gets refreshed.

2. User account:
    - Every new user has to register in order to make any transaction.
    - The database will fetch an updated balance of the amount invested by the account holder in different stocks.
    - The account holder will have the details of all the stocks purchased earlier, with updated share value.
    - The account shows the invoice history of the transactions made by the user till date.
    - The account holder can add a set of desired companies into his wish list and set up a stock indicator to a value for monitoring the share values from his wish list.

3. Making a transaction:
    - The account holder can buy or sell stocks.

- Initially, the account holder must maintain a minimum balance in his account to make a transaction.
- While making a transaction, if the available balance is not sufficient, the account holder should provide the details of his credit/debit card for the number of shares that he wants to buy/sell.
- The database updates this record into the account with the new balance.

4. Database:
    - The database holds a table of various companies and their stock values which changes dynamically.
    - The stock history shows the opening and closing price of the share and its peak value for the day.
    - The database holds the details of all the registered users.
    - The database maintains records of the purchases, balance, and credit history of different account holders.

4. Project Resources

RESOURCE	**MODE OF WORK**	**ROLE**
Mohamed Zakriea Niyaz	Our project involves building a web base database which acts as a platform to help people to manage their Stock. Niyaz tasks will involve in setting the Oracle 11g Database on Windows server and mount a developer portability tool (either Toad or Pl-sql developer). Will also involve in creation of database schema with abstract privileges along with packages, procedures, triggers ,jobs , DBMS_refresh if needed. He will also take the responsibility of reviewing the codes and evaluating the keys to make sure the Database remains in BCNF.	Database Admin, Pl-sql developer.
Jude Nishanth Wilfred	The Project is likely to build with the strong analytics functions as a part of web pages which provides all available determination of a stock. This elite functionality on stock are brought to life through Spring frame works which serves one among the few prominent frame works in today's Java. Jude tasks involve in setting up the web server which also include creation of web pages with all the functionalities expressed vividly through JSP, HTML, CSS. Jude Leads the team in assisting the implementation of Javascript and Jquery for the UI pages and will also involve setting up an API which will be responsible for culminating the data's into the system	Web Admin, Web Designer

Divya Arveni	Stocks refresh periodically where price, volume and respective related characteristics changes instantly . The instant changes are analyzed , validated and executed rigorously through built in Core Java packages. Divya task involves in implementation of instant changes of stock characteristics. This involves building of java files which builds stock information instantly every time when the system get refreshed and also responsible for all the stock characteristics math	Java Developer, Java Admin
Haritha	A Stock web should have the best GUI only through which the analytics can be understood by the end user. In order to bring best GUI we adapt HTML pages merged with jquery , Haritha task involves is providing the right GUI which makes customer to ease the application without any difficulties. The one of the best GUI practices in JAVA and HTML are brought in the web pages by involving graphs and statics and these are extensively carried out through html and java plug-ins which promote high level simplicity in the application	HTML Admin, HTML designer

5. Software

1. **Java** - Java is an object-oriented programming language developed by Sun Microsystems. Java is a platform-independent, multi-threaded programming environment designed for creating programs and applications for the Internet and Intranets.

2. **JavaScript** - JavaScript is a scripting language developed by Netscape Communications designed for developing client and server Internet applications. Netscape Navigator is designed to interpret JavaScript embedded into Web pages. JavaScript is independent of Sun Microsystem's Java language.

3. **Spring** - Spring is the most popular application development framework for enterprise Java. It is used to create high performing, easily testable, reusable code without any lock-in

4. **Oracle** - The Oracle Database (commonly referred to as Oracle RDBMS or simply as Oracle) is an object-relational database management system (ORDBMS). It stores data logically in the form of tablespaces and physically in the form of data files ("datafiles").

5. **Pl-sql –** This is a Oracle commands which will be used to provide a flexible database activity in order to promote best achievements in obtaining a elegant working system.

6. Special Resources

- Wensheng Wu – Feedback, Consultant
- Tingting Zhong – Feedback,Consultant

7. Relational Schema

1. ***CUSTOMER***(**CUSTOMER_ID**, PASSWORD ,FIRST NAME,LAST NAME,ADDRESS,EMAIL)
2. ***ACCOUNT***(**ACCOUNT ID**,CUSTOMER_ID,BALANCE,ACCOUNT DESCRIPTION,TRANSACTION TYPE,DATE)
3. ***ORDER***(**ORDER_ID**,CUSTOMER_ID,STOCK_ID,STOCK VOLUME,ORDER DESCRIPTION,ORDER TYPE,ORDER STATUS,DATE)
4. ***STOCK***(**STOCK_ID**,STOCK NAME,STOCK_PRICE,STOCK VOLUME,LOCATION,TIMESTAMP)
5. ***STOCK_HISTORY***(**STOCK_ID**, **DATE,** STOCK NAME,OPEN PRICE,CLOSE PRICE,PEAK PRICE, STOCK VOLUME,LOCATION)
6. ***PURCHASE_HISTORY***(**STOCK_ID** ,**CUSTOMER_ID**, **DATE,** VOLUME BROUGHT,VOLUME SOLD,NET VOLUME,)
7. ***STOCK FAVORITES***(**CUSTOMER_ID** ,**STOCK_ID**,STOCK_NAME,STOCK_PRICE,STOCK VOLUME)
8. ***STOCK INVENTORY***(**STOCK_ID**,STOCK NAME,STOCK_PRICE,LOCATION, WEB_SOURCE,STOCK VOLUME)

8. Functional Dependencies

1. **CUSTOMER**

   CUSTOMER_ID → PASSWORD, FIRST NAME, LAST NAME, ADDRESS, EMAIL
2. **ACCOUNT**

   ACCOUNT_ID→CUSTOMER_ID, BALANCE, ACCOUNT DESCRIPTION, TRANSACTION TYPE, DATE
3. **ORDER**

   ORDER_ID → CUSTOMER_ID, STOCK_ID, STOCK VOLUME, ORDER DESCRIPTION, ORDER TYPE, ORDER STATUS, DATE
4. **STOCK**

   STOCK_ID→STOCK NAME, STOCK_PRICE, STOCK VOLUME, LOCATION, TIMESTAMP
5. **STOCK_HISTORY**

   STOCK_ID, **DATE**→STOCK NAME, OPEN PRICE, CLOSE PRICE, PEAK PRICE, STOCK VOLUME, LOCATION

6. **PURCHASE _HISTORY**

   STOCK_ID, CUSTOMER_ID, DATE → VOLUME BROUGHT, VOLUME SOLD, NET VOLUME

7. **STOCK FAVOURITES**

   CUSTOMER_ID, STOCK_ID →STOCK_NAME, STOCK_PRICE, STOCK VOLUME

8. **STOCK INVENTORY**

   STOCK_ID → STOCK NAME, STOCK_PRICE, LOCATION, WEB_SOURCE, STOCK VOLUME

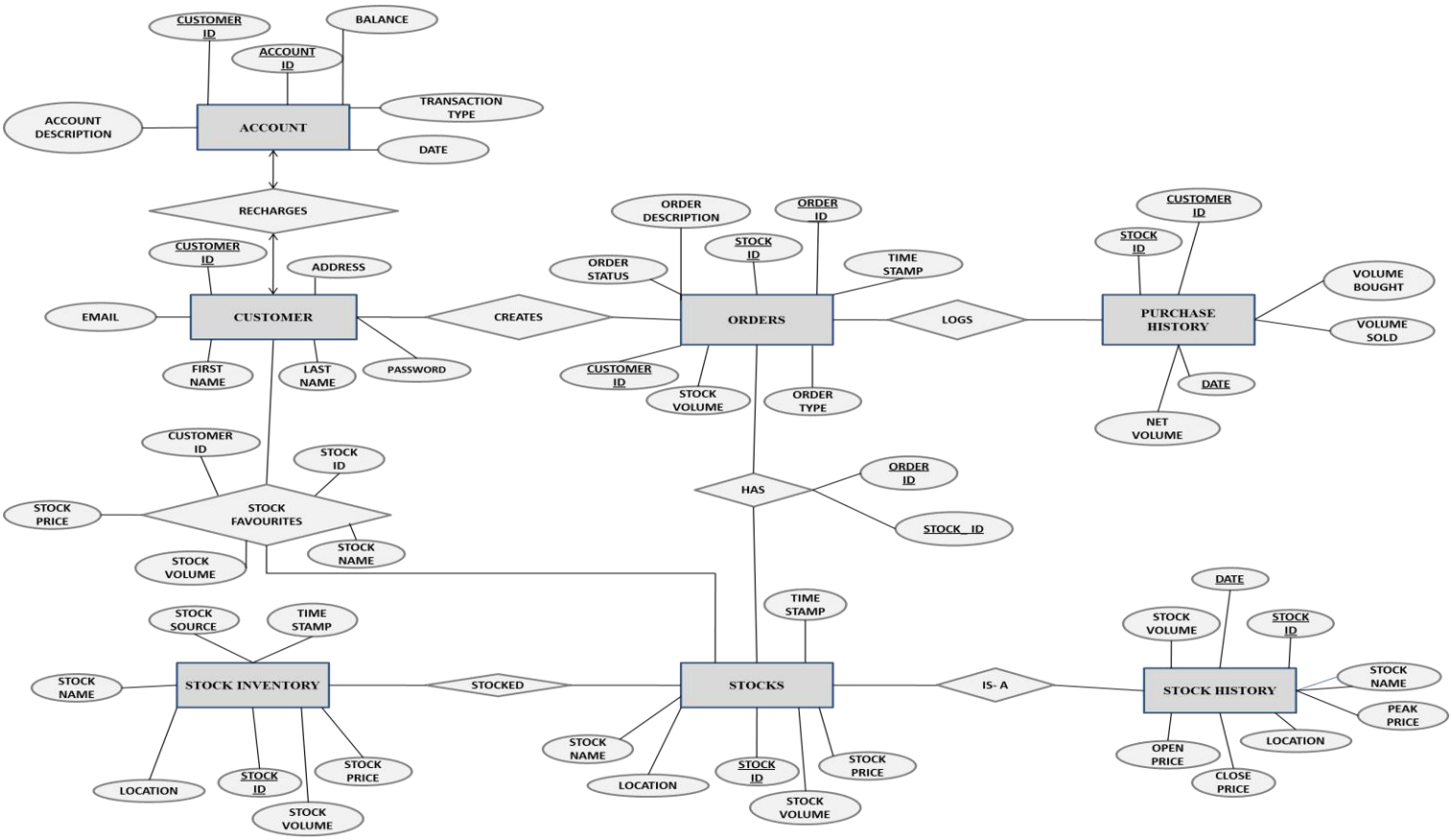
9. Data Extraction

The data are gathered from numerous web providers which provide data's in CSV format. The data's are extracted from the providers periodically using built packages and placed on the server and triggering the oracle stored procedures or packages. The Oracle packages are responsible for inserting the CSV data's from server into respective audit tables ,once the data's are validate through built in oracle functionality then the data's are moved to the main tables likely the data marts. The frequency of the data extraction and loading is yet to be determined at the initial phase and will be likely determining on or before next deliverables.

Possible Source extractions

a. www.chartoasis.com
b. www.finanace,yahoo.com
c. www.biz.swcp.com
d. ww.analyxit.com

10. Entity Relationship Diagram



PS :ER Diagram Revised