**Software Design Document**

By Group

zhou\_zhang

Group members: Xin Zhang

Xin Zhou

**Table of Contents**

**1.** **INTRODUCTION 3**

1.1. Purpose ………………………. 3

1.2. Scope………………. 3

1.3. overview…. 3

**2.** **SYSTEM OVERVIEW 4**

**3.** **SYSTEM ARCHITECTURE 5**

3.1. Architectural Design

3.2. Decomposition Description

3.3. Design Rationale 5

**4.** **DATA DESIGN 6**

4.1. Data Description 6

4.2. Data Dictionary 6

**5.** **COMPONENT DESIGN 7**

**6.** **USER INTERFACE DESIGN 8**

6.1. Overview of User Interface 8

6.2. Screen Images 8

6.3. Screen Objects and Actions 8

**7.** **REQUIREMENTS MATRIX 9**

**8.** **APPENDICES 0**

# Introduction

## Purpose

This software design document describes the architecture and system design of a currency broker company. This SDD’s intended audience is the system administrator from the currency broker company.

## Scope

This document describes the implementation details of the broker company software. The software will consist of a two major functions. First to inquire the currency pair rate, and the second to notify user if the currency pair rate reaches the target rate.

## Overview

This document contains the following information:

* System architecture design
* Data design
* Component design
* User interface design

# SYSTEM OVERVIEW

pairList function: The system displays all of the currency pair list.

pairinput function: The user input a currency pair that he wants to monitor.

parseXML function: The system parses a XML file from the website.

checkPair function: The system checks the currency pair whether it is valid.

outputCurrencyPairInformation function: The system displays the relative information about the currency pair.

targetValueInput function: The user input a target rate of the currency pair.

checkTargetRate function: The system will check the currency pair rate in real time. If it has reached the target rate, then the system will notify the user.

# SYSTEM ARCHITECTURE

## Architectural Design

Error Messages

Web site

(XML file)

Formatted data

Error Messages

**Retrieve HTML files:**

We’ll first retrieve data from website (http://rates.fxcm.com/RatesXML) if the data retrieved OK, then use this data as the input of the parse XML file.

**Parse HTML files:**

We’ll go through each line of the HTML file, check if the line contains useful data; if so, parse the line and get the data.

**Format outputs:**

Put parsed data in the format that we’ll discuss in the interface section. Then check if we get the correct data. The reason that we wait until this part to check the data instead of doing that right after we get the data is efficiency. We don’t want spending too much time checking data. If the data is correct, then write it to file. Otherwise, log errors.

## Decomposition Description

## Design Rationale

This architectural design is simple and easy to maintain and add new features. Since we only implemented core function right now, so we may add more features and functions later if we need to.

# DATA DESIGN

## Data Description

All of data comes from the website <http://rates.fxcm.com/RatesXML>, the system directly parses this XML file, and the system won’t store these data in a local computer.

## Data Dictionary

1. pair

Type: string

Description: input of the currency pair name by the user

1. targetRate

Type: double

Description: the target rate by the user

1. currentRate

Type: double

Description: the system parses a current rate from the website.

# COMPONENT DESIGN

This system has two major components: one is client-side website and application and another one is server-side php application and mysql database.

The client-side is separated into two parts:

1. the functional component(Java). It receives user inputs and performs all of the calculation required, detail information required, and check target rate.
2. the graphical component(XML). PHP manages incoming and outgoing messages and Mysql Database stores the data such as login history, price history, user operation history, and so on.

# User INTERFACE DESIGN

## Overview of User Interface

In this section, the user interface will be explained by the description since it is still in developing.

## Screen Images

On the way.

## Screen Objects and Actions

1. **Login to System**

Login screen will be used for all users. Users can enter their Email as account ID and password to log into their own account. This will provide the security of the system.

The account will be locked for one hour after three unsuccessful attempts in unknown devices. Then users will reach the information page through this page.

1. **Information Page**

This screen will show the basic information of all currency pairs in real time. There will be an option to choose a pair that user wants to see details.

1. **Details Page**

This screen will show the details of the pair. There will be an option to let user set a target rate, then user will be notified when the target rate is reached. User can choose either going-down target rate or going-up target rate;

1. **Account Page**

This screen will show the information of the user.

This page can be accessed by all pages except Login Page.

User can change account information such as password, clean history, and etc.

# REQUIREMENTS MATRIX

This shows the system requirements satisfy each functional requirements.

|  |  |
| --- | --- |
| Requirements | System Components |
| User requirements | View currency pair in real time, set target value. |
| User interface requirements | Buttons, browsers, and tables. |
| System and integration requirements | Updates. |

# APPENDICES

A list of referenced and/or related publications.

http://rates.fxcm.com/RatesXML