**Software Implementation Report**

**For**

**Armadillo 4000**

**Version 1.1 approved**

**Prepared by Juergen Kriz, Ryan Mock, Sean Dunbar, Sunveer Sandhu, Zach Cockle**

**March 01, 2018**

Table of Contents

[1 Introduction 3](#_Toc507677521)

[1.1 Purpose 3](#_Toc507677522)

[1.2 System Overview 3](#_Toc507677523)

[1.2.1 System Description 3](#_Toc507677524)

[1.2.2 Assumptions and Constraints 3](#_Toc507677525)

[1.2.3 System Organization 4](#_Toc507677526)

[1.3 Glossary 4](#_Toc507677527)

[2 Management Overview 4](#_Toc507677528)

[2.1 Description of Implementation 4](#_Toc507677529)

[2.3 Major Tasks 4](#_Toc507677530)

[2.4 Implementation Schedule 5](#_Toc507677531)

[2.5 Security and Privacy 5](#_Toc507677532)

[2.5.1 System Security Features 5](#_Toc507677533)

[2.5.2 Security Set Up During Implementation 5](#_Toc507677534)

[3 Implementation Support 5](#_Toc507677535)

[3.1 Hardware, Software, Facilities, and Materials 6](#_Toc507677536)

[3.1.1 Hardware 6](#_Toc507677537)

[3.1.2 Software 6](#_Toc507677538)

[3.1.3 Facilities 6](#_Toc507677539)

[3.1.4 Materials 6](#_Toc507677540)

[3.2 Documentation 7](#_Toc507677541)

[3.4 Outstanding Issues 7](#_Toc507677542)

[3.5 Implementation Impact 7](#_Toc507677543)

[3.6 Performance Monitoring 7](#_Toc507677544)

[3.7 Configuration Management Interface 7](#_Toc507677545)

[4 Implementation Requirements by Site 7](#_Toc507677546)

[4.2 Acceptance Criteria 7](#_Toc507677547)

# 1 Introduction

## 1.1 Purpose

The intended purpose of this Software Implementation Report is to outline the goals and objectives necessary to complete the Armadillo 4000. It will also lay out the schedule on which we plan to complete them.

## 1.2 System Overview

The system in question is the Armadillo 4000, a program that can be used to monitor stocks and cryptocurrencies, and alert the user based on values that they set. This is a standalone program, as it is not part of a larger project.

## 1.2.1 System Description

The processes the Armadillo 4000 will support include monitoring financial markets (stocks & crypto-currencies), as well as displaying alerts when user-specified conditions are met. The program is referred to as “Armadillo 4000 v1.1”.

## 1.2.2 Assumptions and Constraints

Assumptions:

* There will be no outside help required to complete the project.
* There are no additional costs to produce the project.
  + Software being used is free.

Constraints:

* The project is required to be complete in 30 days.
* Project is limited to command line as project members have a lack of experience with GUI’s.
* Queries are limited to one per second, due to the API.

## 1.2.3 System Organization

Program API Stock Server

Results

## 1.3 Glossary

|  |  |
| --- | --- |
| Term | Definition |
| GUI | Graphical User Interface |
| API | Application Programming Interface |
| RSI | Relative Strength Index |
| SMA | Simple Moving Average |

# 2 Management Overview

## 2.1 Description of Implementation

Armadillo 4000 will be deployed to the user(s) via an “instant-on” approach. The program will not be installed, but instead run like an executable (.exe) file, as the user will be able to simply double click the icon and have the program launch.

## 2.3 Major Tasks

Implementation

The program will be implemented through a one-time conversion, as it’s a single piece of software that does not change the overall infrastructure of any existing systems. As it is a consumer product, the user would be able to simply download the program from our website and run the program as is.

# 3 Implementation Support

## 3.1 Hardware, Software, Facilities, and Materials

## 3.1.1 Hardware

Hardware required:

* Keyboard
* Mouse
* Monitor
* Computer
* Valid network connection, with access to the internet

## 3.1.2 Software

Software required:

* Functional Windows OS (XP or later)
* The program will be proprietary, as it has been developed for a single company.
* A Java Runtime Environment of 1.4.0

## 3.7 Configuration Management Interface

* Versions will be distributed as updates are made, available for users to download from our website.

# 4 Implementation Requirements by Site

## 4.2 Acceptance Criteria

**Selecting Stocks:**

The user will be able to choose which stocks they wish to set rules for. This will be done by typing the stock symbol of the desired stock. The program will then verify that the stock exists, and a message will be displayed that the stock was successfully selected.

**Input Trading Rules:**

After successfully selecting a stock for the program to monitor, the user will be asked by the system to input values for when the system should alert the user. These are referred to as the trading rules. The value of each stock will be compared to the trading rule(s) chosen. For percentage, they will be set as an increase and decrease between the first value of the stock retrieved compared to the most recent value. RSI will look for the stocks Relative Strength Index to go above 70 or below 30, and SMA will look for the long-range and short-term values to cross.

**Retrieve Data in Real-Time:**

Once the user has determined the stocks they wish to monitor and the trading rules by which the system will alert them, the program will begin pulling information from the stock market using an API.

**Compare Data to Trade Rules:**

As the program receives new data, it will compare it to the original value it retrieved when it was started. The value gained from the comparison will be checked against the user’s trade rules for the stock in question.

**Display alerts:**

Once a user’s trade rule has been met, it will provide the user with a notification. The user must then decide whether to sell or buy the stock, however the actual trade will take place outside the system; this app will not support stock trading. This notification must occur within a maximum of ten seconds from the time that the system receives the stock valuation data that triggers the notification.

**Non-functional acceptance criteria**

* Ability of user to operate program, without assistance, within 5 minutes. (Easy to use).
* Number of terminals to be supported: 1.
* Number of simultaneous users to be supported: 1.
* Type of info to be handled: User’s custom values. 3 trading rules.

Type: Numeric (double). Stock symbols (Strings).