Architecture Requirements Document

Project:
Financial markets simulation with multiple competing algorithmic trading entities.

Client: Cortical Systems

Group:

Members:

Daniel Makgonta 12147100
Moeletji Semenya 12349136
Madimetja Shika 12127877

Publication Date: 20 May 2014

Version: 0.0

Change Log

Date	Name	Reason	Version
14/05/2014	Madimetja	Creation	0.0
20/05/2014	Daniel	Editing and formatting	0.0
16/05/2014	Madimetja	Updated Requirements	0.1
17/05/2014	Moeletji	Editing and formatting	0.1

Contents

This document discusses the requirements around the software infrastructure within which the application functionality is to be developed.

1 Access channel requirements

The system will be accessible by human users through the following front-end channels:

- From a stand-alone Java application interface. The system will be accessible from any desktop or laptop once the appication is installed on the device.

2 Quality Requirements

2.1 Security

- Entity integrity should be maintained. No entity should be able to see sensitive information, particularly strategies, from another entity.
- Only system administrators may modify the system behavior.

2.2 Auditability

- The system should provide mechanisms through which the activity of every entity/participant can be monitored. This information would include
  - the action taken by the participant,
  - the date and time the action was taken,
  - the immediate impact the action had on the market,
  - the state of the participant before and after the action.

2.3 Testability

- Every service offered by the system should be testable against its pre and post condition.

2.4 Usability

- The system should be usable, in that the system should provide a fully functional matching engine.
- Users in the financial/investment sector should be able to understand and use the system without any training or tutorial.

- Most of the users excluding those in the financial/investment sector should be able to understand the system through the affordance that the system provides through its interface.

2.5 Scalability

- If need be, the system should allow for independent entities operating from independent machines to participate in the market.
- The system should allow for concurrent trading.

2.6 Performance

- Algorithmic performance as measured by profit and loss should take preference over absolute system performance.

3 Integration Requirements

The system is self-contained and there is no need for integration with other external systems or components.

4 Architectural Constraints

- The system is to be built using the Java programming language and relevant Java concepts and libraries.
- In order to minimize cost and complexity, the system can be developed to run and operate on a single machine.

5 Glossary