

Portfolio Investment in Focus of Using Agent in Stock Trading Environment

By

Yusnizam Bin Che Din

Final Dissertation Report in partial fulfillment of
the requirement for the
Bachelor of Technology (Hons)
(Business Information System)

JULY 2005

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

t
HF
5548.32
.794
2005

1) Electronic commerce
2) Internet
3) IT/IS -- thesis

CERTIFICATION OF APPROVAL

Portfolio Investment in Focus of Using Agent in Stock Trading Environment

By

Yusnizam Bin Che Din

A project dissertation submitted to the
Information System Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION SYSTEM)

Approved by,

A handwritten signature in black ink, appearing to read 'H. Rais', is written over a horizontal line. The signature is stylized and somewhat cursive.

(Mr Helmi Bin Rais)

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

July 2005

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons



YUSNIZAM B. CHE DIN

ABSTRACT

Portfolio Investment in Focus of Using Agent in Stock Trading Environment is the collaboration between stock trading application and agent program. It can be regarded as the value added process to the stock trading system. The rationale to embed an agent program in the system is to increase the efficiency and effectiveness of the system. The agent program that exists in the application has been programmed to do certain tasks such as to buy and sell stocks on behalf of a trader. The agent performs their job based on the data or value that is defined by the user. The main reason to develop the application that uses an agent is because of realizing the fact that people in this modern society always concern about the standard of living. They want everything around them to be automated. In other words, we can say that people want computers to work for them. There are two objectives that have been set for this project. The first objective is performing a small scale of study regarding the agent and the second objective is to develop a simple web site regarding stock trading application using an agent. To ensure both of the objectives can be achieved, the author has set the scope of study at the planning phase of the project. Basically, there are three scopes of study that have been defined. The first one is to study about the agent such as its environment, functionalities and characteristics. The second scope is study about the concept of system remoteness. System remoteness means, users can access the system from their remote location via internet or World Wide Web (www) technology. The final scope of study is to study about current stock simulators such as Investopedia simulator. The reason is to get some idea for system design and implementation. For the methodology, the Rapid Application Development (RAD) has been employed. The methodology has been chosen because it is effective and suitable for short duration projects. It was designed for developers and users to join together and work intensively toward their goal. For Rapid Application Development methodology, the system is basically developed by using a prototype. Developing a system using a prototype can shorten the development time and the final product of the project is early visible. By using the RAD methodology, the stock trading system using an agent is able to be completed within the time allocated.

ACKNOWLEDGEMENT

First and foremost, Praise Be upon to Allah S.W.T for His Mercy has given me the strength to complete this project within the time given.

My gratitude goes to my supervisor, Mr. Helmi Rais for precious assistance and guidance throughout the project. Special thanks to Mr. Lai Fong Woon for his advises regarding the financial matter. Further thanks to all respondent that involved in doing the testing. Those responds and feedbacks are really valuable for this project. This appreciation also goes to all lecturers, final year students, technician for their assistance, ideas, and support throughout completing the project. I also would like to thank to my lovely family for their support for all these times.

Finally, thank you for all individuals that have contributed their ideas, knowledge, support, and assistance in this project.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL	ii
CERTIFICATION OF ORIGINALITY	iii
ABSTRACT	iv
ACKNOWLEDGEMENT	v
TABLE OF CONTENTS	vi
LIST OF FIGURES	ix
ABBREVIATIONS AND NOMENCLATURES	x
CHAPTER 1: INTRODUCTION	1
1.0 Background of project	1
1.1 Benefits of Agent in Stock Trading Application	2
1.2 Application: Portfolio Investment Using Agent	3
1.3 System Overview	3
1.4 Problem Statement	4
1.5 Significant of Project	4
1.6 Objectives and Scope of Study	5
1.6.1 Objectives	5
1.6.2 Scope of Study	5
CHAPTER 2: LITERATURE REVIEW AND THEORY.	6
2.0 The Agent	6
2.1 Adding Value with Intelligent Agent	7
2.2 Agent as Automated Personal Assistant	9
2.3 Agent as Cooperative Problem Solver	10
2.4 Agent in Electronic Commerce	10
2.5 Agency, Intelligent and Mobility	11
2.6 Agent Characteristics	12
2.6.1 Situatedness	12

	2.6.2	Autonomy	13
	2.6.3	Adaptivity	13
	2.7	Search Engine and Agent	14
	2.8	Market Analysis and Research	15
	2.9	The Important of Using Agent in the Stock Market	15
	2.10	Traditional Search Engine and Agent Search Engine	16
			
CHAPTER 3:		METHODOLOGY	19
	3.0	Introduction to Methodology Approach	19
	3.1	Potential Types of Methodology	20
	3.2	Factor for Choosing Methodology.	20
	3.3.1	Time Constraint	20
	3.3.2	Cost for System Development	21
	3.3.3	Degree of User Involvement	22
	3.3.4	Human Resource	23
	3.3	Result and Discussion	24
	3.4	Process Flow	25
	3.5	Deliverables for Each RAD Phase	27
CHAPTER 4:		RESULT AND DISCUSSION	29
	4.0	Result of the System	29
	4.1	System Architecture	29
	4.2	Agent Architecture and Functionality	37
	4.2.1	Buying Stock Using Agent	38
	4.2.2	Selling Stock Using Agent	39
	4.2.3	Take over Job from Agent	41
	4.2.4	Architecture of the System	42
	4.2.4	Architecture of the Agent	43
	4.3	Database Architecture	46
	4.4	Discussion of the System	47

	4.4.1 Result of System Testing	. . .	47
	4.4.1.1 Description of System Testing	. . .	48
	4.4.1.2 Data – First session Testing	. . .	48
	4.4.1.3 Data – Second session Testing	. . .	50
	4.4.1.2 Discussion on Testing Result	. . .	51
	4.4.2 Future Enhancement	53
CHAPTER 5:	CONCLUSION AND RECOMMENDATION	. . .	55
	5.0 Conclusion of the project	55
REFERENCES	58
APPENDICES	59
Appendix 1	Gantt chart		

LIST OF FIGURES

Figure 1	Main Page Interface
Figure 2	About Stock simulator Interface
Figure 3	Registration Interface
Figure 4	Login Interface
Figure 5	Transaction Page Interface
Figure 6	Stock Name Interface
Figure 7	Seventh Script Prompt – Buying Stock
Figure 8	Agent Help Interface
Figure 9	Selling and Buying Stock Using Agent Interface
Figure 10	Buying Stock Using Agent Interface
Figure 11	Agent Log Interface – Pending To Buy Stock
Figure 12	User Account Interface
Figure 13	Selling Stock Using Agent Interface
Figure 14	Agent Log Interface – Pending To Sell Stock
Figure 15	Script Prompt To Buy Stock
Figure 16	System Architecture
Figure 17	Agent Architecture
Figure 18	Database Table
Figure 19	Chart - Manual transactions for both sessions
Figure 20	Chart - Using Agent to do transaction for both sessions
Diagram 1	Context Diagram
Diagram 2	Data Flow Diagram
Table 1	First Session Testing – Manual Transaction
Table 2	First Session Testing – Agent Transaction
Table 3	Second Session Testing – Agent Transaction
Table 4	Second Session Testing – Manual Transaction

ABBREVIATIONS AND NOMENCLATURES

<i>UTP</i>	Universiti Teknologi PETRONAS
<i>WWW</i>	World Wide Web
<i>PHP</i>	Hypertext Preprocessor
<i>RAD</i>	Rapid Application Development
<i>KLSE</i>	Kuala Lumpur Stock Exchange
<i>IT</i>	Information Technology
<i>JAD</i>	Joint Application Design
<i>HTML</i>	Hypertext Markup Language
<i>CASE</i>	Computer Aided Software Engineering
<i>DFD</i>	Data Flow Diagram
<i>T3</i>	Transaction day 3
<i>PC</i>	Personal Computer

CHAPTER 1

INTRODUCTION

1.0 BACKGROUND OF PROJECT

The title that has been chosen for the final year project (FYP) is “**Portfolio Investment in Focus of Using Agent in Stock Trading Environment**”. The agent programming is the technology that created to help people to do their work more efficiently and organized. Before describing more details about the application that is to be developed, let understand the definition of “Portfolio Investment” and the definition of “Agent”. The term “Portfolio Investment” is refers to the type or category of investment and the term “Agent” refers to the computer program that is situated in environment and that is capable of performing actions in particular environment in order to meet its design objective. For this project, the author will focus on developing an application and embedded it with agent program for stock investment environment.

Basically there are two basic properties of software agents. The first one is they are *reliant*, means that the agent are dependent and make their job base on the instruction given by user. The second property is *situated*. This property does not constrain the notion of an agent very much since virtually all software can be considered to be situated in an environment. However, where agents differ is the type of environments. For instance, agent tends to be used where the environment is more challenging or more specific.

1.1 BENEFITS OF AGENT IN STOCK TRADING APPLICATION

The usage of agent will bring certain benefits that suited for certain types of applications. In this section, the author will describe the benefits that will be provided by the agent in the stock trading application. The benefits that will stated below is based from the original system blueprint that is prepared as a guide for system development.

- The agent will perform transaction on behalf the user when asked to do so. This is the key advantages of the system. From the research, some people that invest in the stock market having problem to monitor the trend of the shares that they purchased. The reason is they have other tasks to be performed at the same time.
- Reduce human error is other benefits that can be gain by using the agent technology. For instance, investor may rely or depending on broker agent to trade on their behalf. The broker agent that they choose may have others clients and need to do lot of transactions for them. Sometimes the broker agent may feel burden or pressure because has to do lot of transactions in the same time. As a result the broker agent can not concentrate on the tasks effectively and may have possibility to do wrong transaction for his clients.
- Reduce communication cost with broker is another advantage of using the application's agent. By using the agent that embedded in the stock trading application, the cost to communicate with broker via mobile phone or other devices can be cut or reduce.

1.2 APPLICATION: PORTFOLIO INVESTMENT IN FOCUS OF USING AGENT IN STOCK TRADING ENVIRONMENT.

Portfolio Investment in Focus of Using Agent in Stock Trading Environment is the name of the application that going to be developed. The title is choosing because of several reasons as described below:

- To help investor in stock market to trade effectively, they can use agent to trade on their behalf. The trading process will become easier.
- To encourage the use of agent technology in the stock trading environment. By using the agent, it will help to solve the problem regarding human error which has been discussed earlier.

1.3 SYSTEM OVERVIEW

Portfolio Investment in Focus of Using Agent in Stock Trading Environment will be developed to help stock trader to perform transaction effectively via web. In the current situation, the trader needs to perform the transaction by their own or delegate the work to the stock broker from various companies such as OSK Securities Berhad and other securities companies. If the traders choose to trade by themselves, they have to keep monitoring the trend of the stock for every minute because the stocks price are always keep changing. If they unable to do that, they might lost the opportunity to make profits or they might suffer from a huge loss. Some of investor might think to hand over the task to selected broker agent from certain companies. However, trader has to aware about the issue regarding human error. For instance, one broker agent may be assigned by many traders to purchase and sell their stocks. As a result, they can't do their jobs effectively.

To overcome this problem, the author has decided to develop the stock trader agent which has the capability to trade on behalf of user. The agent will purchase or sell the stock by themselves based on the value of transaction that stated by user.

1.4 PROBLEM STATEMENT

Stock trader always need a lot of time to monitoring the trend of stock. Beside that they also have to surf through internet to find information about certain stock that attracts their interest. However, not all people have enough time to monitoring the trend of the stock for the whole day. To understand this statement, let divide the trader into two categories. There are the full-time trader and part-time trader. The full-time trader may have the time to find all the information they need regarding the stock and do the transaction by themselves. But the thing could be different for the part-time investor. The part time investor usually has other job to be done. So they don't have sufficient time to monitor and trade stock by themselves. If they able to do the transaction, it will become less effective.

1.5 SIGNIFICANT OF THE PROJECT

The intention to work on this project is based on the reason to help investor to trade effectively and overcome some of the problems that arise in transaction process. There are several significant or impact that should happened when the project is completed.

- The agent program that live resides in the web application will perform selling and buying transaction on behalf user.
- The time that use by trader to do transaction will be shortens. They just need to allocate few minutes of their time to set up the type and conditions of transaction for agent to execute.

1.6 OBJECTIVES AND SCOPE OF STUDY

1.6.1 Objectives

- To perform small scale study regarding the agent.
- To develop simple web stock trading application that using an agent. This application will be used to trade on behalf of trader.

1.6.2 Scope of study

To ensure the system that will develop meet the requirement and functional as required, several scope of study have to define. The scope of study for the project is stated as below:

- To study about the agent such as it functionality. For the project, the author will study on how the agent will help the trader to perform stock transaction.
- To study about the system remoteness and apply the concept to the system that will be developed. This is to ensure the website can be access by using two or more workstation.
- To study about the current stock simulator. Basically, the author didn't find any stock simulator that locally developed. The study will be focusing on the systems that develop by foreign developer. Based on the system, the author will develop the application that using the agent which will benefits the user of the system.

CHAPTER 2

LITERATURE REVIEW

Software agent technology is a rapidly developing area of research. Application domains in which agent solutions are being applied or research into include workflow management, telecommunications network management, air traffic control, business process, education, e-mail filtering, digital libraries and many others areas. Guilefoyle (1995) notes:

“In 10 years time most IT development will be affected, and many consumer products will contain embedded agent-based systems.”

From the statement above, we can say that the agent technology is a technology that brings great impact to the information technology. Most of the system that will be developed will use this kind of technology in order to gain the advantages. For the Final year project, the author decides to develop a simple web application for stock trading environment. The web will have the agent program will help investor in making stock transaction such as to buy and sell stock. From author's points of view, by implementing this technology in the web application, it will benefit trader such as increase shorten their time to do the transaction by their own.

2.0 THE AGENT

The agent application is the technology that develops to help people in doing their task. In the modern era, people always concern about the standard of living. They want everything around them to be automated. In other words, people want computer to work

for them. For instance, few years ago we have to find information by ourselves such as gathering information through reading or asking someone who is expert in certain field.

However, in the technology age, we can use variety type of search engine to find information on their behalf. For example, we can use the search engine application such as google, yahoo and others. All of the application that author mention above can be considered as agent.

Major Jim Jansen in his journal – Using an Intelligent Agent to Enhance Search Engine Performance defines *“Agents are software robots. They can think and will act on behalf of a user to carry out tasks. Agents will help meet the growing need for more functional, flexible, and personal computing and telecommunications systems. Uses for intelligent agents include self-contained tasks, operating semi-autonomously, and communication between the user and systems resources”*

From the Major Jensen statement we can see that the agent can be design or create to have different functionalities in order to work on different tasks. The agent technology will help to satisfy the need of the people that always grow and change over the time.

2.1 ADDING VALUE WITH INTELLIGENT AGENTS IN FINANCIAL SERVICES.

There are many financial systems that available today is implementing the concept of agent. The reason is, system owner has realize that the technology bring the positive impact to the system. In other word, we can say that the agent technology is adding the value to this system by making it more flexible and working more independently. The word flexible means the traditional system has been adding with the agent technology which has many other functions and work more effectively. While the independent means the system that implementing the agent concept has the ability to work freely and producing their own decision.

According to D. Wenger – Swiss Bank Corporation and A.R. Probst – University of Lausanne -*“Just as the production of goods has largely been automated by artificial agents (robots, machines), so it is the objective of financial information engineering to increase the level of automation in financial services.”*

People are reluctant to invest in the stock market because of several reasons. The first reason is most of the people do not have much time to surf through internet or to read from newspaper about the financial news even they have interest to do so. The second reason is because people are lacking on the financial knowledge which is important for investor to do stock analysis and finally to make the best decision to invest.

It is true that investing in the stock market involving high risk. However, it also brings a good opportunity to investor to gain profit. By solving the limitation or problem that exists by using the agent technology, people will become more confident to investing their money in order to make profit.

Kees Jonkheer in his journal – Intelligent Agent, Market and Competition has stated that *“It is obvious that since intelligent agents operate with a great amount of autonomy, they can play a major role in the distribution of products. Depending on their functionality, agents can act as information brokers (search and compare) or even as mature market players (negotiating and making specific transactions). Intelligent agents, compared to traditional parties, can act without organizational implications on behalf of many users. For instance, if an intelligent agent is sent out to purchase the least expensive product, this information - once retrieved - is virtually available to all.”*

Kees Jonkheer has state in his journal that agent program has the ability to play in major role. It can replace human or act as their assistance in certain circumstances. For instance, the agent can do negotiating or make specific transactions when they ordered to do so. The way it works is depending on how it was programmed.

2.2 AGENT AS AUTOMATED PERSONAL ASSISTANTS

The agent itself is the automated program. Means that, it able to do it task independently. What the user can do is to define the constraint to the agent and let it work by following the guideline stated by user.

S.C. Laufmann – *“The notion of agents is often associated with that of automated personal assistants. As such, the agents perform some kind of task(s) for a person. This view emphasizes the type of application to which the agent is targeted, and it based on certain task knowledge and behaviors, or task-specific skills. Thus the necessary and sufficient conditions for inclusion with set of agents are based primarily on what the agent does, without substantial regard for how it is done.”*

As stated by laufmann, the agent can be assuming as personal assistance. In this modern society, people are valuing the concept “time value of money”. The statement means that time is regard as the valuable opportunity. By developing the agent as personal assistant, people can do many things at one time.

For instance, when the agent is build to aid trader to invest in the stock market, some of the task that trader actually do is delegate to an agent. Basically the trader will navigate through the internet to find information related to the stock activities. Then they will analyze the data in order to invest in the potential stock.

After recognize the potential stock through their analysis, they will do transaction to buy or to sell stock. From the statement, we can see that trader have to do so many things, starting from finding information to perform transaction. So, by taking agent as assistance, the trader can focus on certain part and let their assistant to work on other part. For example, trader can delegate the job to monitor and to do the transaction to agent while the trader can focus to find information regarding stock market.

2.3 AGENT AS COOPERATING PROBLEM SOLVER

S.C. Laufmann – *“Agents utilize advanced reasoning capabilities to perform cooperative work effectively, presumably to effect some result that none would be capable of achieving alone”*

The statement above show that the agent is very useful in solving problem that user cannot solve alone. Agent is embedded with advance reasoning capabilities which able it to find the relevance information that related to problem scenario then analyze the information and then come out with reasonable decision. The agent technology is using widely for decision making process especially in critical situation. For instance to perform analysis on stock based on past historical data.

2.4 AGENT IN THE ELECTRONIC COMMERCE

Jennings and Wooldridge – *“currently, commerce is almost entirely driven by human interaction; humans decide when to buy goods, how much they are willing to pay, and so on. But in principle, there is no reason why some commerce cannot be automated. By this, we mean that some commercial decision making can be placed in the hands of agents.”*

According to the statement, we know that the commerce is basically performed by the interaction of the parties that involve in the transaction. Human decide everything starting with their planning to buy a good until the end of the transaction process. We can conclude that the interaction of human in this transaction is not effective in term of time and cost. It is more appropriate if the transaction is automated. It means the agent technology can be applied in this field.

By applying the agent technology into the commerce field, the transaction would become more effective. The user can delegate their work to the agent and let the agent to decide everything for them.

2.5 AGENCY, INTELLIGENCE, AND MOBILITY

Joseph P. Bigus and Jennifer Bigus, the writer of the book *Constructing Intelligent Agent with Java* have stated that “ *Agency deals with the degree of autonomy the software agent has in representing the user to other agents, applications and computer system*”

The statement means the agent represents the user, help the user. Guides the user, and in some cases they will takes unilateral actions on user’s behalf.

When we talk about software agents, we have to focus on the three dimensions which we use to measure the capabilities which are the agency, intelligence and mobility. The agency means, the agent will representing the user in making their job such as finding the information that they need.

Intelligence means the ability of the agents to do the right things when they are order to do so. For instance, if the agent is order to find the information about certain stock and analyze it, the agent must follow the order by capturing the correct information, filter the data and analyze it accordingly as the order. If the agent fails to interpret the order, then it will result error result.

To support above explanation, the author has quoted the statement of Joseph P. Bigus and Jennifer Bigus “*Intelligent refers to ability of the agent to capture and apply application domain-specific knowledge and processing to solve problems. Thus our agent can be relatively dumb, using simple code logic, or they can be relatively sophisticated, using complex AI based methods such as inferencing and learning*”

The term mobility is referring to the characteristic of an agent to move inside the network. Joseph P. Bigus and Jennifer Bigus say that “*An agent is mobile if it can move between systems in a network. Mobility introduce additional complexity to an intelligent agent, because it raises concern about the security (the agent’s and target system’s)*”

The term “*inference*” is referring to the ability of the agent to interpret the problem that receives from user. The agent must be able to understand the problem in order to find

best solution. The term “*learning*” that used by Joseph P. Bigus and Jennifer Bigus is referring to the ability of the agent or program to learn new thing. If the program has this ability, it will be more useful and sophisticated.

2.6 AGENT CHARACTERISTIC

The characteristic of an agent make it unique if compared to the other technology that comparable to it. For an instance, we can compare the intelligent agent with search engine technology. The search engine technology has the limitation in representing the information to user. For example, the information provided by search engine is not specific and it searches in the passive way.

The intelligent agent is embedded with the new characteristics to enhance the performance that provided in the search engine technology. There are several characteristics that will be discussed in this section.

2.6.1 Situatedness

Matthias Rehm in his journal- Multimodal concept formation in situated agents said that *“Only if an agent is situated in an environment, it can - by an active exploration and interaction - form the concepts, that are relevant for its task. The established concepts are thus always referring to the concrete situations encountered during their formation process. The given sensoric equipment constitutes the basis for this concept formation. This is the idea denoted by embodiment. The agent can only work with the data received by its sensoric equipment and preprocessed by its perceptual system.”*

Situatedness means the agent receives some form of sensory input from its environment, and it performs some action according to it environment. When the environment change, then the agent must cope with that changes. For an example, the information about the stock market is always change due to the trader activities (buy and sell stock) in the stock market. Thus, the agent that works in this environment must have the capability to sense the changes that occur and able to work well in that situation.

2.6.2 Autonomy

Hanh Tran & Thaovy Tran said that *“All agents are autonomous, which means that an agent has control over its own actions. All agents are also goal-driven. Agents have a purpose and act accordance with that purpose”*

The agent can act without direct intervention by humans or other agents and that it has control over its own actions and internal state. An agent could also senses changes in its environment and responds to these changes. This characteristic of the agent is at the core of delegation and automation.

For an example, if the user tell the agent “if A happens, then do B” and the agent is always waiting the event to occurs in order to execute the action b. To ensure the agent can do the specific job, the developer has to equip with script in order to define the action of the agent which lead the agent to accomplish it mission.

2.6.3 Adaptivity

Bjom Hermans is his journal said that *“Users will not start to use agents because of their benevolence, proactivity or adaptivity, but because they like the way agents help and support them in doing all kinds of tasks; soon users will use all sorts of convenient (i.e. "intelligent") applications, without them realizing they are using agents by doing so.”*

The adaptivity of an agent means the agent is capable of reacting flexibly to changes in its environment; taking goal-directed initiative when appropriate; and learning from its own experience, its environment, and interactions with others. This is the important criteria to be embedded in the agent application in order to make it very useful.

2.7 SEARCH ENGINE AND AGENT

Bjom Hermans said that *"The information and information services on the Internet are very heterogeneous. It means information on the Internet is being offered in many different kinds of formats and in many different ways. This makes it very difficult to search for information automatically, because every information format and every type of information service requires a different approach"*

There are several ways to deal with the problems that have just been described. Most of the current solutions are of a strong ad hoc nature. The gathered information, characterized by a number of keywords (references) and perhaps some supplementary information, is then put into a large database. Anyone who is searching for some kind of information on the Internet can then try to localize relevant information by giving one or more query terms (keywords) to such a search engine.

Although search engines are a valuable service at this moment, they also have several disadvantages. To overcome the disadvantages, the use of the agent is the ideal answer.

"In the future, it [agents] is going to be the only way to search the Internet, because no matter how much better the Internet may be organized, it can't keep pace with the growth in information..." - Bob Johnson, analyst at Dataquest Inc.

Using agents when looking for information has certain advantages compared to current methods, for instance, Agents are capable of searching information more intelligently, for instance because tools (such as a thesaurus) enable them to search on related terms as well, or even on concepts. Furthermore, in the future agents will be able to communicate and cooperate with other agents (such as middle layer agents). This will enable them to perform tasks, such as information searches, quicker and more efficient, reducing network traffic. They will also be able to perform tasks such as searches directly at the source or service, leading to a further decrease of network traffic.

2.8 MARKET ANALYSIS AND RESEARCH

Dr. Bryan Taylor, President, Global Financial Data, Inc, said, *“The only thing certain about the stock market is that you can’t predict what is going to happen in the future. This puts investors in a difficult position. Investors want to maximize their return while minimizing their risk, but of course, increasing return increases risk.”*

According to Dr. Bryan Tailor, investors are always playing with risks. It means that, trader the may either gain profits or suffer losses from their investment. This is resulting form the mistake such as fail to study the past historical data regarding stock or else, they did not monitor well the trend of the stock that their invest.

As I mention in the earliest section, not all trader has time to monitor the trend of the stock that they invest for the whole day. Some of them have other important task to look for. For an example the trader might be student, doctor and others. Their professions limit their time to fully concentrate on the stock that they invest. They can’t do the transaction effectively. To solve the problem, some of them may think delegate the task to monitor and do transaction of their stock to stock broker. However, they should aware on how effective the stock broker will work for them. This is because; stock broker may have many clients to deal with.

To solve the problem, stock trader agent application may be the best solution. This is because, the agent application what develop has fewer tendencies to make mistake. The application basically work based on the program that written.

2.9 THE IMPORTANT OF USING AGENT IN THE STOCK MARKET

As we know, an agent technology is going to be applied in every aspect of life. This is because an agent can bring greater benefits to people surrounding. This statement is support by Graham Kendall.

Graham Kendall, school of computer science and IT from university of Nottingham in his article said *“In recent years, using multi-agent based models to study the stock market has become a promising research area.”*

From the statement we can see that the importance of using agent technology to study stock market. This is because people are recognizing the benefits of an agent from every aspect. People are willing to use an agent because some of reason as mention below:

- Computer based agent is able to work 24 hours round the clock. This is because they are differing from human. They don't need to have rest, sleeping and others things that human must have.
- Secondly, computer base agent is design to follow the rules or program that has been written to them. They are able to perform works that delegate to them constantly without commit any error.
- In term of cost, the agent base system is profitable for long time run. The initial cost to develop an agent may be quite high. However for the long time basis, it is beneficial.

2.10 TRADITIONAL SEARCH ENGINE AND AGENT SEARCH ENGINE

Using agent search engine to search certain information has several advantages if compared to current methods, which is using a traditional search engine. Below are some of the advantages that user can obtain by using agent search engine.

- Agents are capable of searching information more intelligently, for instance because tools (such as a thesaurus) enable them to search on related terms as well, or even on concepts. Agents will also use these tools to fine-tune, or even correct user queries on the basis of a user model, or other user information. It is more effective than using traditional searching engine. Traditional search engine conduct the search based on one or more keywords that given by a user. This presupposes that the user is capable of formulating the right set of keywords to

retrieve the wanted information. Querying with the wrong, too many, or too little keywords will cause many irrelevant information (*'noise'*) to be retrieved.

- Individual user agents can create their own knowledge base about available information sources on the Internet, which is updated and expanded after every search. When information (example documents) has moved to another location, agents will be able to find them, and update their knowledge base accordingly.
- Agents can relieve their human user of the need to worry about "clerical details", such as the way the various Internet service have to operate. Instead, he or she will only have to worry about the question what exactly is being sought (instead of worrying about where certain information may be found or how it should be obtained). The user's agent will worry about the rest. For the traditional search engine, the search for information is often limited to a few Internet services, such as the WWW. Finding information that is offered through other services (e.g. a 'Telnet-able' database), often means the user is left to his or her own devices
- As a user agent resides on a user's computer, it is always available to the user. An agent can perform one or more tasks day and night, sometimes even in parallel. As looking for information on the Internet is such a time-consuming activity, having an agent do this job has many advantages, one of them being that an agent does not mind doing it continuously. A further advantage of agents is that they can detect and avoid peak-hours on the Internet. It is differ from the search engine. Search engines cannot always be reached: the server that a service resides on may be 'down', or it may be too busy on the Internet to get a connection. Regular users of the service will then have to switch to some other search engine, which probably requires a different way to be operated and may offer different services.
- Software agents will be able to search information based on contexts. They will deduce this context from user information (built-up user model) or by using other

services, such as a thesaurus service. For the traditional search engines, they are domain-independent in the way they treat gathered information and in the way they enable users to search in it. Terms in gathered documents are lifted out of their context, and are stored as a mere list of individual keywords. A term like "information broker" is most likely stored as the two separate terms "information" and "broker" in the meta-information of the document that contains them. Someone searching for documents about an "information broker" will therefore also get documents where the words "information" and "broker" are used, but only as separate terms (example: as in "an introductory *information* text about stock *brokers*").

- User agents can adjust themselves to the preferences and wishes of individual users. Ideally this will lead to agents that will more and more adjust themselves to what a user wants and wishes, and what he or she is usually looking for, by learning from performed tasks (i.e. searches) and the way users react to the results of them. Furthermore, agents are able to continuously scan the Internet for newly available information about topics a user is interested in. For the traditional search engine,

CHAPTER 3

METHODOLOGY

3.0 INTRODUCTION FOR METHODOLOGY APPROACH

Methodology is defined as a step by step approach that is essential in every system development. In other words, methodology can be referred to as the system development life cycle. There are many types of methodology such as the waterfall model, spiral model, rapid application development model, and others. However, these methodologies cannot easily be chosen and used for every project.

A developer has to do a detailed analysis on each type of methodology to ensure it is suitable with the nature of the project. There are several factors that should be a concern for the developer in order to choose the best methodology. The most important factors that should be underlined are, time constraints, cost of the project, the level of user involvement, and knowledge on the methodology itself. A developer also has to do research on similar projects that have been performed by other people.

In this chapter, the author will list three types of methodology which have potential for the project. Then follow by describing some important aspects in selecting methodology and come out with the best methodology to be used in the project. Finally, the discussion on the selected methodology will take place.

3.1 POTENTIAL TYPES OF METHODOLOGY

There are three types of methodology that the author wants to compare in order to find out the best and appropriate one for the project. The candidates' methodologies are waterfall model, Spiral Model and Rapid application development model. There are two reasons for choosing these three methodologies to compare. The first reason is to compare the practicability of each methodology in term of time and cost if applying to this project. The second reason is to compare the suitability of each methodology with the project. For instance, the project is stress more on the communication between developer and user.

Several factors or criteria are chosen and assume as the standard for the comparison purposes. The factors to be considered are time constraints, cost for system development, user involvement and finally the number of human resource available to develop the system.

3.2 FACTORS FOR CHOOSING METHODOLOGY

3.2.1 Time Constraint

Time constraint is very crucial factor to be considered in the process of selecting appropriate methodology. Basically the most time consume methodology is the spiral type model. Follow by the waterfall model and finally the rapid application development model (RAD).

The spiral model is concerning about the final product of the project. It stress on the performance and perfection of the output. In order to obtain the perfection, the repetition may occur on some phases. For instance, if the deliverables or output from implementation phase is not good or some changes have occurs in user requirement, then the process will turn back to design phase. The system will be redesign and

develop based on new requirement. The process will run continuously until the system owner is really satisfied with the output. Other than that, this type of methodology also consists of many testing phases which require time and effort. The repetition processes that happen are wasting the development time and increase the cost of the project.

The waterfall model is consumed less time compared to the spiral model. This type of model also stress on the perfection of the final product. However, it is not as critical as spiral model are. The waterfall model is step by step approach which is consisting of five phases. There are preliminary investigation, system design, system development, system testing and system installation and documentation. The methodology emphasizes on completing an individual phase, before proceed with other phases. It is not too complicated as spiral model and more systematic.

The Rapid application development (RAD) is the methodology that required the less time for project to accomplish. The major element for this type of methodology is prototype. The development of system is dependent on the prototype to ensure the system can be completed and test in short time. However, there is a limitation on this type of the project. The methodology is not too emphasis on the performance of the system if compare to the goal of spiral model and waterfall model.

3.2.2 Cost for System Development.

The cost for the system development can be regarded as the major aspect to be considered for every project. From my knowledge, the cost is basically associated with the time. The reason is, people that work in the project must be paid for the work that they done. It means, the longer the project than the higher the cost. When the development time increase, then the cost needed to carry out the project also increase. From the previous section, we know that the spiral model is the methodology that requires the longer time to accomplish.

From our first perception, we may say that the spiral model associate much cost because it consume much time if compared to two others methodology. However, the perception is absolutely wrong. The most costly project is the project that uses rapid application development (RAD) methodology. It is true that the methodology is consuming less time compared to other; however we cannot deny the cost to purchase the prototype since the developer that using this type of methodology is too dependent on it. This is major disadvantages of the methodology.

For this project, the scenario is different. The cost to develop the system is small because the author will develop the system for free in order to accomplish the mission of the final year project. So, there is no payment that needs to be done for the compensation purpose. As a result, we can say that the cost to develop a system is even, regardless which methodologies to use.

3.2.3 Degree of User Involvement

The involvement of user in project is very important. User is the entity for the developer to get user requirement. Thus, working closely with user is very helpful in order for developer to get the fast feedback from them and perform any changes instantly. Different methodologies have different level of user involvement. By comparing all those methodologies that define above, the rapid application development is the type of methodology that requires more involvement from user.

Rapid application development (RAD) is designed for developer and user to join together and work intensively toward their goal which is to shorten the time duration of the project. Rapid application development encourages the involvement of user from the analysis phase until the end phases of system development life-cycle. During that period, the developer is more on listening, while the user will express their need. The methodology practice is focus on the incremental development of prototype. User will see the progress of the project as well to test the system during the development period.

Spiral model also concern will the involvement of user. However, it is not too critical as rapid application development does. Spiral model encourage user to comment about the current system during the testing phase. If user does not satisfy with the result, then the enhancement will be done. Or in other word, we can say that the developer will redesign the system based on the comment from user. The installation phases and documentation will only start when user fully agrees with the performance of the system. Else, the redesigning phase will take place.

Waterfall model is the methodology that is more leisure in term of user involvement. Basically, user will be more involve during the preliminary investigation and testing phase. They do not involve during construction of the system. During preliminary investigation, user will ask about their requirements and preferences. After that phase is over, the involvement of user is decrease. Development team will focus on the analysis, design and development phase without intervention of user. When the system is fully complete, the user will involve again. They will test the system and if the system works well and follow their intention, then the installation and documentation will take place.

3.2.4 Human Resource

The number of man power in every project is very important. It can bring great impact such as determine the duration and the cost of the project and the performance of the system. When the number of the man power is large, then the project will have the capability to complete earlier than the estimated time.

For this project, the author will work solely to develop the system. Thus, the selection of the methodology can be regard as very important. As discuss earlier, the spiral model is consuming the longest time, followed by waterfall model and rapid application development model. The crucial analysis is necessary in order to choose the best

methodology. The result to choose the best methodology for the project will be discussed below.

3.3 The Result and Discussion

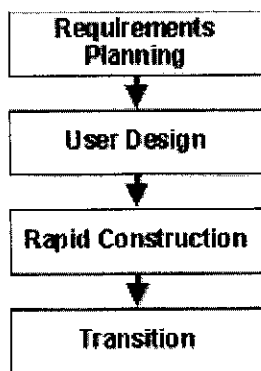
After study and analyze all of methodologies above and look at the current scenario, the author has decided to choose Rapid Application Development (RAD) as the methodology for this project. The reasons are discuss below:

- **Time allocation** - The most crucial thing to be considered is the time's allocation for completing the project. Basically there is about 15 weeks to complete the project. During that particular time, the author has to do analysis, research, designing, developing and documentation alone. It is impossible to develop an agent application system within that period unless the shorter methodology is applied.
- **Prototype** – For this project, the author will use available HTML and PHP code to develop prototype system. From the prototype, the incremental enhancement will be done in order to accomplish the goal of the project. Without the prototype, it is difficult to be done. This is because the agent technology can be regard as the new technology in Malaysia and to the author itself.
- **Joint Application Development (JAD)** – The Rapid Application Development is stress on the communication between project team and system owner or user. The constant communication for both parties is important in order to have better end product. From the intensive communication, the developer able to understand well about the user requirements. Developer also can do changes or incremental development if they receive new requirements from the user. For the system owner side, they can see how the project progress and the way the

system function. If they do not satisfy with the outcome of the system, they may change the requirements to propose the change.

3.4 Process Flow

The following diagram depicts the dependency relationships between the stages in the Rapid Application Development Process model.



Based on the above diagram, we see four stages or phases of rapid application development life cycle. It begins with requirements planning phase, followed by user design phase, rapid construction phase and finally the transition or cut over phase. Each of the stage are dependent to the phase that before them. It means that the user design phase is dependent on the planning phase and so on.

For the rapid application development life cycle, the user design phase and rapid construction phase are highly interrelated. Both phases are recycling for several times until the system owner are fully satisfied with the whole of the system. For instance, after the construction process, the system will be evaluate and test by system owner. If they not agree with the functionality or features of the system, the developer and system

owner have to redesign the system based on new requirements that request by the system owner. It means that the cycle of the methodology will move a step backward.

After the redesign process is complete, it will move to construction phases. The process will keep continue until the system owner is fully agree or accept the system. In the next page there was the detail explanation of each phase that involve in rapid application development life cycle.

Requirements Planning Phase

The requirements planning phase requires that high level or knowledgeable end-users determine what the functions of the system should be. It should be a structured discussion of the business problems that need to be solved. It can often be done quickly when the right users and executives are involved.

User Design Phase

The user design phase requires the users to participate strongly in the no technical design of the system, under the guidance of I.S. professionals. User design is done in a **Joint Application Design** (JAD) workshop. In the first two phases the users and executives should play a larger part than the I.S. professionals. Prototyping is used to aid in requirements specification and design. The user does not sign off a paper design, they sign off a CASE representation.

Construction Phase

The design created during the User Design Phase is added to using I-CASE tools. As each transaction is built it may be demonstrated to the end-users for revision. The CASE environment allows for the continuous changes in design. End-users are closely involved in the construction phase. Testing occurs throughout the process. The I-CASE toolset

should generate the code as well as the database descriptions for the final product. Code optimizers may be used to improve the performance of the generated code.

Cutover Phase

When the cutover phase occurs, a variety of actions are needed, comprehensive testing, training of the end-users, organizational changes and operation in parallel with the previous system until the new system settle in.

3.5 Deliverables for Each RAD Phase

In this section, the deliverable of each phases will be discuss. Every phase is necessary to produce output or deliverable that will be use as the input for the next phase. Without deliverable, the process of development can't be proceeding. Below is the list of deliverables that will be producing.

Deliverable 1 – Project scope and objective of the project should be the output for the requirement and planning phases. During this phase, the scope and objectives of the project must be defined in order to have the clear picture of the project. The feasibility study also has to be carried out in order to ensure the project is worthwhile or not to be carried out.

Deliverable 2 – Complete user requirement must be completed at the end of the first phase. This is the important as the document can be regard as guide line to developer in designing the system.

Deliverable 3 – Analyzed data and information regarding the system also need to be completed at the first phase. This data and information is useful for project team member in decision making process.

Deliverable 4 – Complete design of the system should be completed at the end of design phase. This deliverable then is use as the output for the next phase which is constructing phase.

Deliverable 5 – Complete system is the deliverable for the constructing phase. This is the final product for the system. Complete system means the system has been tested by user and they are agreeing to accept the system.

Deliverable 6 – Complete project documentation is the final deliverable for the final phase of the rapid application methodology (RAD). This not means that the output is not going to be used. The documentation will be keep for references for other project that have the similarity with this project.

CHAPTER 4

RESULT AND DISCUSSION

4.0 RESULT OF THE SYSTEM

In this section, the result that obtained from the project will be explained in detail. The main purpose of this chapter is to tell the audience about the final product of the system such as how the system work and what functions that offer by the system. For better understanding, this chapter will be divided into two major parts. The first part is result part and the second part is discussion part. For the result part, the topics that will be present are as below:

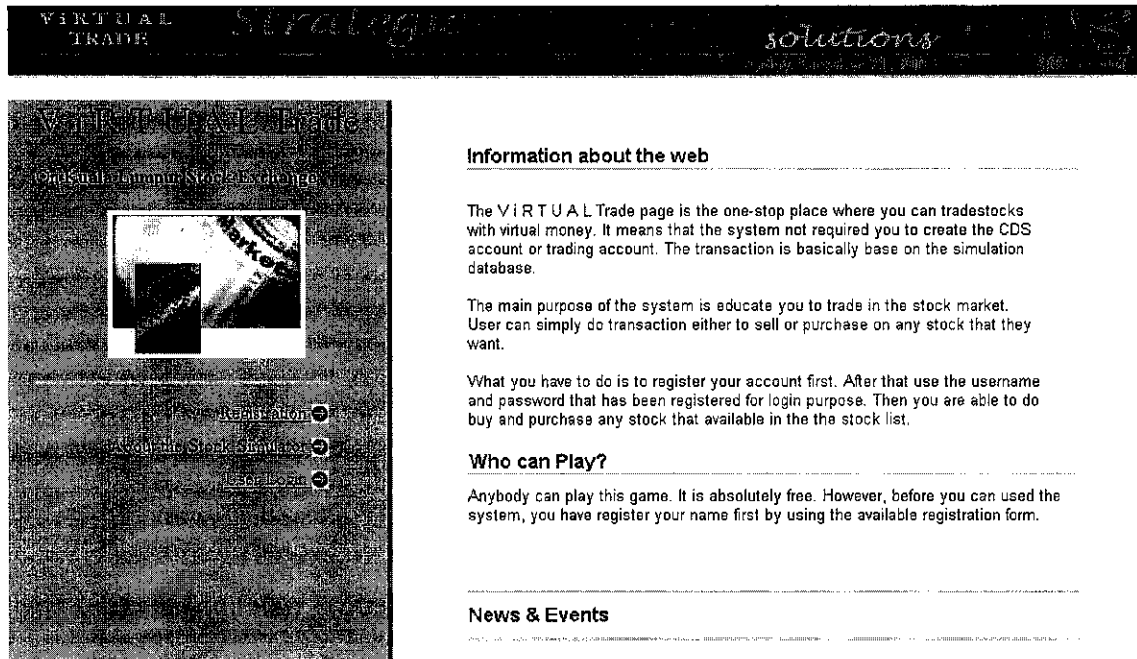
- System Architecture
- Agent Architecture and functionality
- Database architecture

4.1 SYSTEM ARCHITECTURE

The overall system architecture will be discussed here. Basically, the total interface of the system is seven pages. The discussion will be started on the main page or the first interface of the system, follow by the page that describe the rules and functionality of the system, the registration page, login page, transaction page, list of the stock page and finally the agent page.

4.1.1 First Interface – Main Page

Figure 1



This is the main page of the system that user will see when they first navigate or visit to the system. The purpose of the main interface is to give brief explanation about the system. For instance, the audience of the system will know the purposes of the system, the benefits that they will gain by using it, the way to be one of user of the system and finally they can read on latest news that release by the administration side regarding the system. It is important to user to fully understand the purpose and functionality of the system before they use the system.

From this page, user can navigate through three more pages which is registration page for registration purpose, stock simulator page for understanding the concept of simulator and finally the login page for login purpose. The main interface is designed based on human – computer interaction concept where it should be simple, balance and interactive. The term “simple” means that the interface is not too crowded with information. The term “balance” means the layout of the interface is stable or position well and finally the term “interactive” means the interface is equipped with acceptable graphic element or simple decoration that is acceptable. Other than that, interactive

means that the user and the agent application inside the system able to interact with each other. For instance, when user of the system delegates certain job to the agent, then the agent must able to interpret and understand the order and perform the job accordingly. Other than that the agent also must clarify it status to the user by telling the user it current job or status such as waiting to buy or sell certain stock that has been assigned to them.

4.1.2 Second Interface – About stock simulator

Figure 2

VIRTUAL TRADE Strategis solutions

VIRTUAL Trade
Or Kuala Lumpur Stock Exchange

Home
Registration
About the Stock Simulator
User Login

About VIRTUAL Trade Simulator

This simulator is created to help the beginner in the stock trading or the people that have the deep interest in stock trading. By using this simulator, you have the opportunity to invest in certain stock that you like. There are no capital limitation to spend on every transaction.

As the investor, you have the right to invest in any stock that available in the stock list. Only the stock that available in the list can be purchase. If you able to make profit during the transaction, the amount of the profit will be added to your account. On the other hand, if you suffer losses, the the amount that you loss will be deducted from the current value of your account

The 5 top investors which is most profitable will be listed

How Do You Get Started?

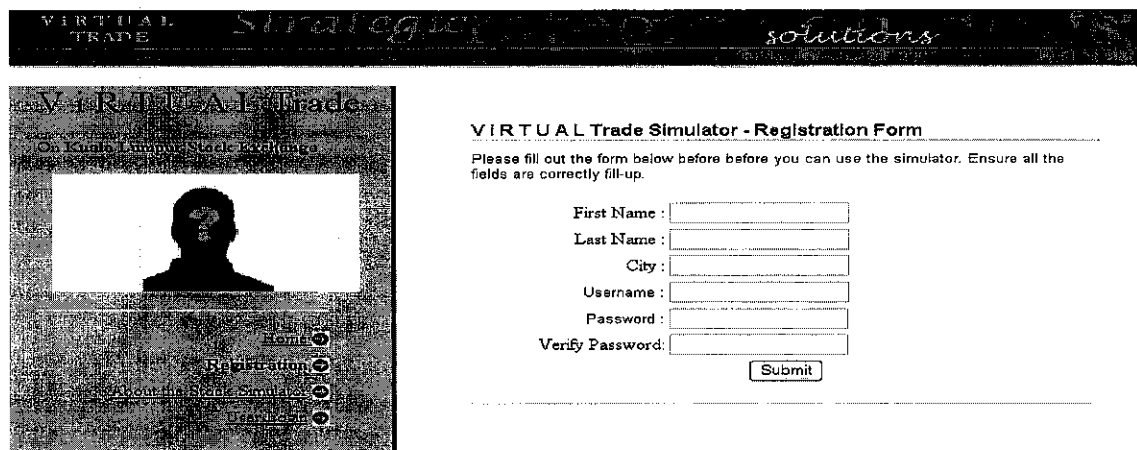
1. First, you must register as a user. Click sign-up and fill out the online registration form to get started. Then, login into your account for your first transaction.
2. You must know the stock symbol for every company that trade in the stock market. For Instance, if you want to buy habib jewel stock, you have to know code and share name. (code - 7045, share name - habib)
3. There are two options for you to trade. First, you can trade any of the available stock by yourself or you can delegate the work to the agent. If you choose to use agent to trade

The name of the second interface is “About Stock Simulator”. This page is design to inform the audience about the rules that they should comply with. The first rules that user has to follow is to register their name in the registration form that prepared for them. If they ignore the first rules, then the access for them will be denied. This can be considering as the security element for the system.

The other purpose for registration is to set up the individual account for them. This account will be link with database and it will be used to update their data. The second rules for the system is the user can only select stock from the stock list that available in other page. They should type the correct stock code in order for the transaction to be valid. Else, the transaction will consider void. The last rules, the user is giving the options to trade by their own or delegate the job to the agent. It is beneficial to use the agent service because they can use the time to do other job. From this page, the user can navigate to other four pages by click the link that available at the left hand side.

4.1.3 Third Interface - Registration

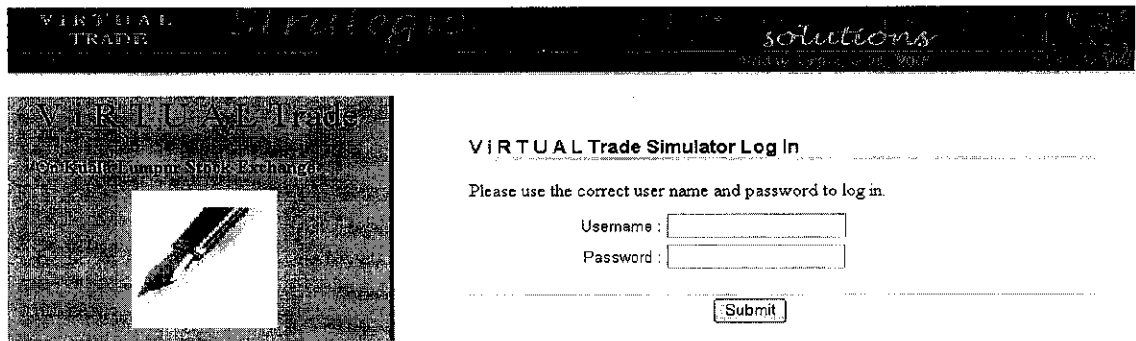
Figure 3



The above image is the registration interface. The interface consists of the registration form for user to fill up. User has to fill up the entire field that provide to them. If they fail to do so, then the registration process will be consider not succeed and they can not access to the transaction page. The main purpose for registration is to store data that provide by user in database and to create an account for them.

4.1.4 Fourth Interface - Login

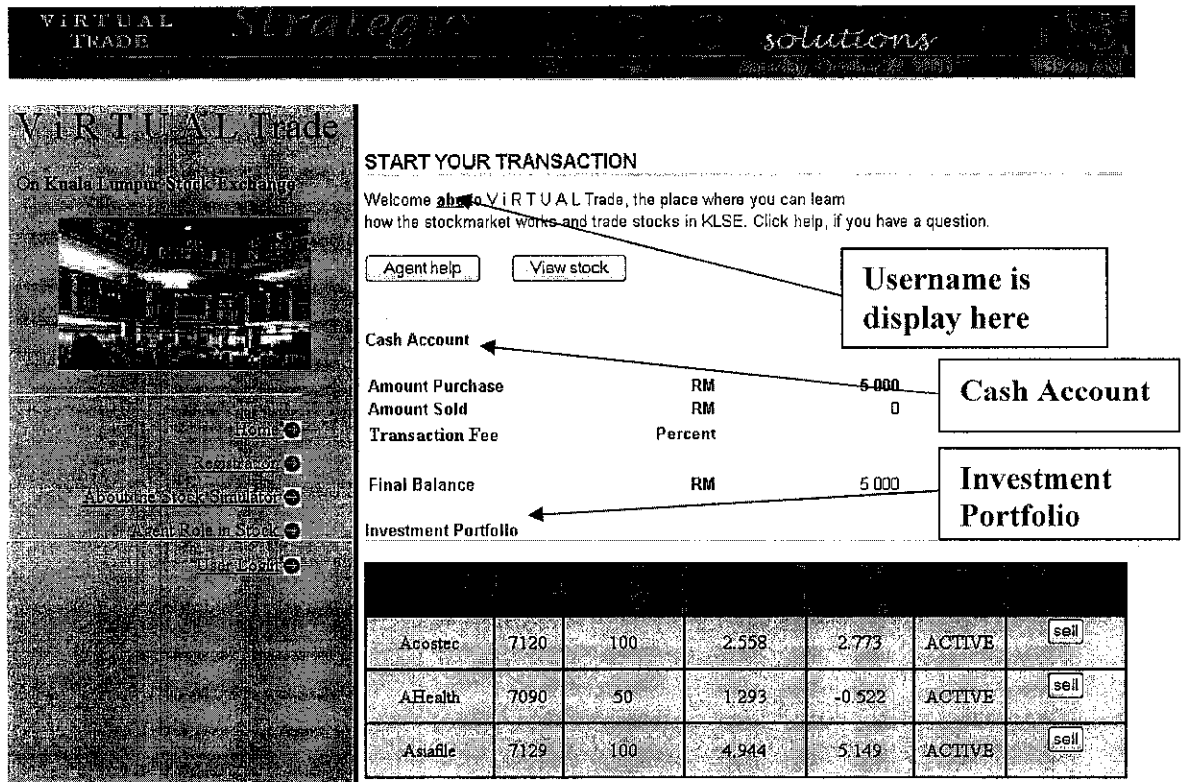
Figure 4



This is the login interface where user can use for login purpose. They should use the correct username and password in order to access to the transaction page. The purpose for user to login is to ensure that they are directed to their own account. All the information regarding their past transaction will be displayed at the transaction page.

4.1.5 Fifth Interface – Transaction Page

Figure 5



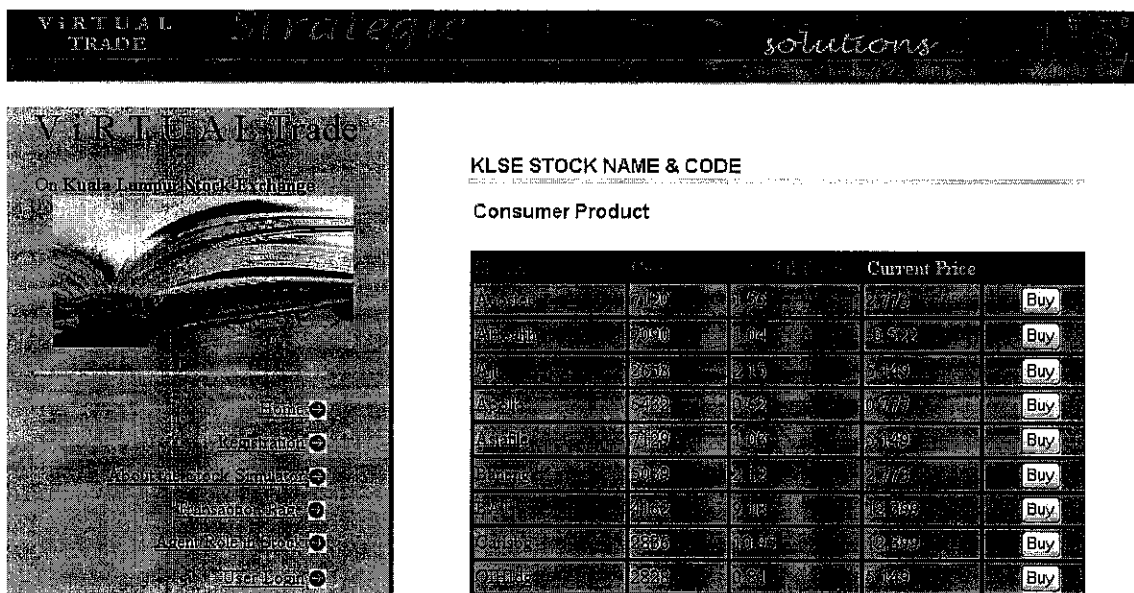
The fifth interface for the system is the transaction page. This page is can be regard as the user’s account. The username that use by user will appear at the first line after the page heading. The main purpose of this interface is show to the user about the state of their current account. User can see the progress of their account by referring to the “Cash account” and “Investment portfolio”.

From the cash account, user can see the total amount of money that they spend to buy stock and the amount that they gain from selling stock. At the final line of the cash account, user can see the final balance of the transaction, which is either they gain profits or suffer losses.

From the investment portfolio table, user can see what stock that they purchased before, the purchased quantity, buying price and the current price of the stock. They also can sell the stock by clicking the sell button at the last column of the table.

4.1.6 Sixth Interface – Stock Name

Figure 6



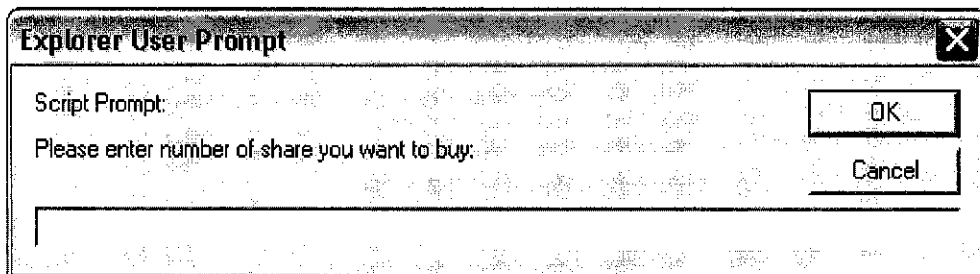
This is the interface where user can view and buy the stock. That is the main purpose to have the interface. The table is designed to have five columns. There are share or stock

name, stock code, initial price for the stock when it opens to the public on the particular day and the current price of the stock.

In the first column, user can see the name of the stocks that they can purchase. The second column is the list of stock code. The stock code is the unique attribute for stock and it should be different with other stock. The user has to know the code for any stock that they want to purchase if they decide to use agent's service. This is vital requirement in order to use an agent. In the third column, there is stock's initial price for particular day. Next to it, there was the current price that is update for every 15 seconds.

For the final column, there was buy button. The purpose of the button is to help user to buy the stock that they want. To buy stock, user just has to select the stock name from rows and click the buy button at the end of the row. Whenever the button has been clicked, the explorer user prompt as shown in figure 7 will appear.

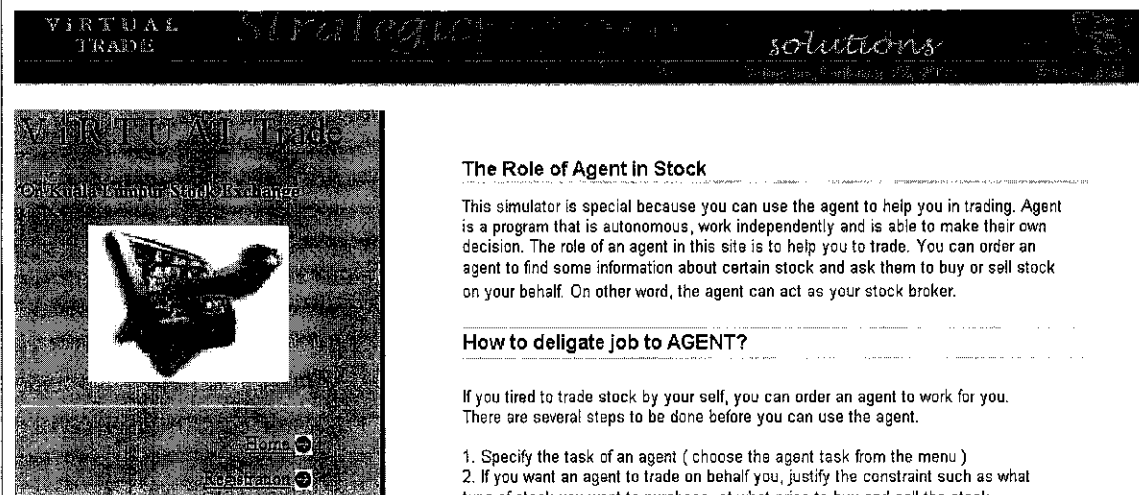
Figure 7 – Script Prompt – Buying Stock



This explorer user prompt is design the captured data about the quantity of shares that user wish to buy. The data that captured will be kept in the database for updating purpose. The transaction detail will also appear at the user's investment portfolio which is located in the transaction page (figure 5). After the user confirms the transaction by clicking ok, they will be directed to the transaction page.

4.1.7 Seventh Interface – Agent Help

Figure 8



This is the final interface for the system. This interface is designed with the purpose to help the user to trade by using the agent. However, to use the agent service is optional. User can choose to use the agent to do transaction on their behalf or they can do the transaction by themselves. The page is divided into three sections. The first section is designed to explain the purpose to have an agent in the system as well the agent's functionality and capability. The next section is the explanation the way to use the agent. User has to follow all the rules that mention in the section in order for the agent to execute well and successfully.

The main purpose to have an agent is to trade stock on behalf the user. The agent has been designed to have the ability to purchase and sell stock. To buy the particular stock via agent, user has to define the stock code, the quantity of the stock they want to buy and finally the price that they willing to buy. The agent will wait until the stock price equal to the price that defines by user. To sell stock, user has to choose stock code that they want to sell from the list menu. Then they have to define the conditions to sell the stock. Basically there are four conditions that associated with the selling transaction. The agent will sell the stock that they select base on the condition that they decide on. In the next section, I will mention about the conditions that associated with selling options under the title of agent architecture.

4.2 AGENT ARCHITECTURE AND FUNCTIONALITY

The agent architecture will be discussed in this section. The agent is designed to help user in performing stock trading transaction. It having the ability to buy and sell the stock by following the criteria that set up by user. The agent is develop with the purpose to solve several problems that confront by certain users such as user does not have sufficient time to monitor the trend of stock price as well to purchase and sell the price at the correct price and time. Thus, by create the agent that can react in the stock environment, the problem that mentions earlier can be solved. To help the audience to understand the way agent work, please look at the illustration (see figure 9) with the explanation below.

Figure 9 – Selling and Buying Stock Using Agent Interface

The screenshot shows a window titled "Ask Agent" with a dark background. It contains two main sections for user input. The first section, on the left, has a red checkmark and is for buying stock. It includes labels for "Stock Code", "Buying Price", and "Buy Quantity", each followed by a white input field. The second section, on the right, has a red checkmark and is for selling stock. It includes a label for "Stock Code" followed by a dropdown menu labeled "Select Stock to Sell". Below these sections is a note: "Note: You can only choose one from the available selling options". This is followed by four radio button options: "Stock price at", "Return from stock at", "Losses from stock at", and "Date and Time at". Each option has associated input fields: the first three have white input fields followed by the word "percent", and the last one has three dropdown menus for day, month, and year, followed by "Hours" and "Minutes" labels with their respective dropdown menus. At the bottom of the interface are two buttons: "Confirm" and "Reset".

Figure 8 show the interface which user can interact with the agent. From the exhibit, we can see two check boxes which are labeled in red color. There are buy stock and sell stock check box.

4.2.1 Buying Stock Using Agent

To ask the agent to buy stock, user must tick the “buy stock” check box first. After that they should define which stock to buy by entering the correct stock code into the correct text field that provided to them. The process should be followed by defining the price and the quantity of stock that they want the agent to buy. Figure 10 shows the example of the buying transaction using agent. Finally, they should click on “confirm” button to validate the transaction.

Figure 10 – Buying Stock Using Agent Interface

<input checked="" type="checkbox"/>	Stock Code	7120
	Buying Price	2.558
	Buy Quantity	100

As a result, the agent will take over the task to buy the stock. Agent will monitor the trend of current price of the selected stock and purchase them when the price is equal or lower than price that declare by user. During the waiting period for agent to buy stock, user can monitor the progression of the agent by refer to the agent log (see figure 11) in the transaction page.

Figure 11 – Agent Log Interface – Pending To Buy Stock

Agent Log:

Stock pending to buy: ----- Stock code: 7120 Quantity: 100 Buy at: 2.558
--

When the buying transaction that performs by agent is completed, the result of the transaction will be displayed in the investment portfolio (see figure 12) which is located at the transaction page. At this moment the agent task to buy the stock is consider completed.

Figure 12 – User Account Interface

Investment Portfolio

Acostec	7120	100	2.558	2.773	ACTIVE	sell
---------	------	-----	-------	-------	--------	------

4.2.2 Selling Stock Using Agent

The second functionality of the agent is to sell the stock on behalf user. The vital requirement is users must have the stock in their account. If they do not have any of stock in their account, then the agent will ignore the selling transaction that delegate to them. To delegate the task to agent, user must tick the “sell stock” check box. Then select the stock code from menu for the particular stock that they want sell (see figure13). Next, they have to specify the condition to sell the stock.

Figure 13 - Selling Stock Using Agent Interface

Sell Stock

Stock Code Select Stock to Sell ▼

Select Stock to Sell
 7120
 7090
 7120

Note: You can only choose one from the available selling options

Stock price at
 Return from stock at percent
 Losses from stock at percent
 Date and Time at - - -
 Hours Minutes

There are four conditions for the stock to be sold. The user has the right to choose more than one conditions from the entire conditions that available. If the user decides to choose more that one condition, the agent will only response or execute the condition

that firstly fulfills its rules. If user wishes to choose the first condition, they must define the selling price. The agent will sell the stock when the current price of the particular stock is equal or exceed the price that defines by user.

If user chooses the second and third conditions, they must define the amount of stock's return and loss repetitively in percentage. The agent will calculate the price and sell the stock when the current price is equivalent or surpass the percentage that defines by user. If the user which to choose both of the conditions which are the return and the loss percentage in order to sell their stock, then the agent will only responding to the first value that equal or exceeds the percentage that they defined.

The last condition is user must set the date and time to the agent. The agent will sell the stock based on the value that set to it. The reason to have this condition is to help user to sell their stock in the correct date. There are certain traders that trade stock in the locus contra basis. Locus contra means that the user do not need to pay money on the day (T1) they buy stock. However, on the transaction day three (T3), they must clear off their debt. If they willing to continue to keep the stock, they have to pay the entire transaction cost. In the other hand, if they decide to sell the stock when the current price is lower that than purchased priced, they have to pay the amount of loss and vise versa.

After select the condition, user has to validate the transaction by clicking on the "confirm" button. If user ignore this rules then the delegation process is consider void. During the waiting period for agent to sell the stock, user can monitor the progression of the agent through the agent log in the lowest part of transaction page (see figure 14).

Figure 14 - Agent Log Interface – Pending To Sell Stock

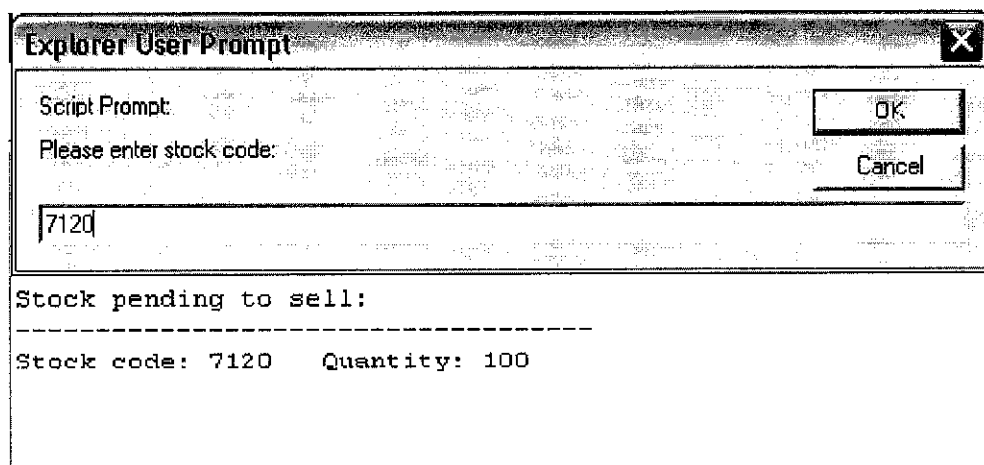
```
Stock pending to sell:
-----
Stock code: 7120   Quantity: 100
```

When the transaction is completed, the stock code and the quantity of the stock that shown in the agent log will be automatically disappear and the status of the stock that display at the investment portfolio will be changed from the “active” to “sold”.

4.2.3 Take over Job from Agent

User has right to take over the transaction task when ever they want. In other word, we may interpret it as cancellation of the task. Sometimes, trader may change their mind and set up new condition to the agent. To take over the task from agent, user has to click on the “take over” button and specify which stock they want to take over. Please refer to figure 15 for better understanding.

Figure 15 – Script Prompt To Buy Stock



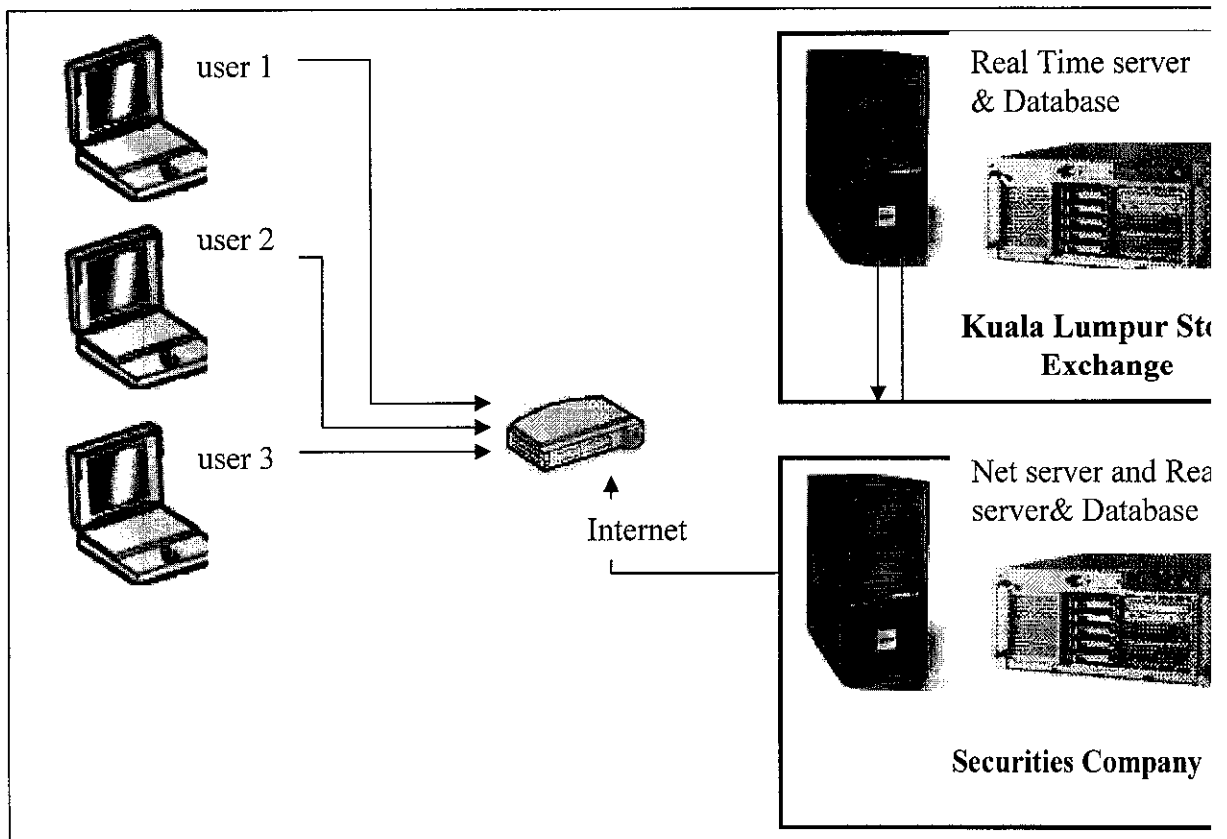
For instance, if user wishes to take control over Acostec stock from the agent, user must click the take over button and enter the correct stock code which is 7120 into explorer user prompt and then click ok button. The system will immediately stop the agent service from that particular stock. As a result, the stock data that appear in the agent log will automatically disappear.

4.2.4 Architecture of the System

Figure 16 below shows the system architecture of the stock trading system. From the architecture, we can see that the user can access the system that resides in the security company server via internet or World Wide Web (www). Users have to connect their personal computer (PC) to the router in order to access the internet and enable them to access stock trading application.

The security company should have net server in order to hosting or publish the web based application through the internet. This server will maintain the connection between the user and the web application. The net server is connected to the securities company's database where the information about user is store and update. The securities' server also must connect to the Kuala Lumpur Stock Exchange server in order to retrieve some important data such as the current stock price and others data regarding stock market.

Figure 16 – System Architecture



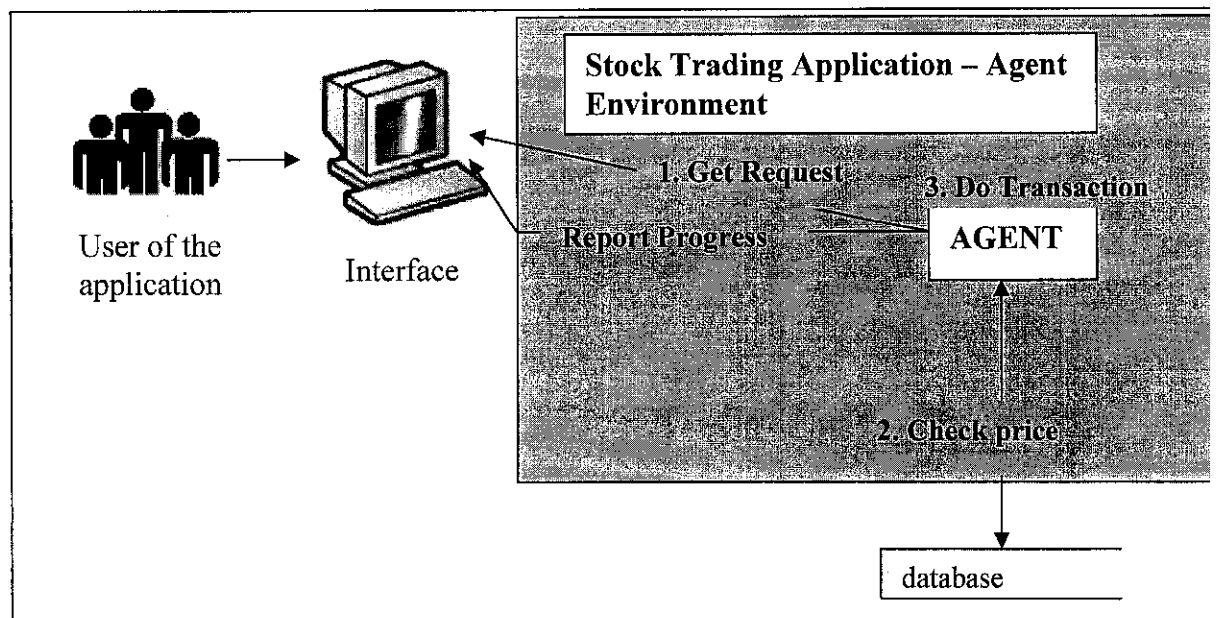
Basically, the Securities Company and Kuala Lumpur Stock Exchange (KLSE) must use the real time communication server in order to ensure the integrity, performance and efficiency of the system is always at high level. The main reason is the stock trading application it self must is run in the real time mode.

4.2.5 Architecture of the Agent

In this section, the author will discuss about the architecture of the agent in order to give better understanding to audience. In the stock trading using agent application, the role of agent is focusing on buying and selling stock on behalf of user. In this case the agent will receive the order from the user, analyze it and finally perform some action based on the order it received.

For the system, the agent is working as single entity. Meaning that the agent is working alone in it environment and do not have any communication with other agent. The agent will do every single job from the beginning until the end of the transaction as mention in above paragraph. Figure 17 below shows the architecture of the agent that working inside the application

Figure 17 – Agent Architecture



As can we see, the agent which is situated inside the web application will receive the request from user through the user interface. For this system, the request that will be passing to agent is regarding the stock transaction.

After receiving user's request, the agent will interpret the type of transaction. For instance, let say the user want the agent to buy certain stock. The user will key in the stock code, buying price and the amount of shares to buy. The agent then will look inside the database to find the available stock code that is matching with the code that user enters. If the agent find the matching code, it will monitor the current stock price of that particular stock that always changing over time and buy the stock when the current price is equal or exceed the buying price that define by user. However, if the agent cannot find the matching code, then it will ignore the transaction.

During the waiting period to wait for the current moving price to equal or exceed the buying price that user define, the agent will issuing the status message to the user. The status message will tell the user what is the status of the agent, such as the agent is pending to buy the stock or the agent is successfully buy the stock. During the condition where the agent is pending to buy stock, the user can cancel the agent operation at any time. If user chooses to cancel the transaction, the task that was given to the agent previously will be terminated automatically.

Diagram 1 – Context Diagram

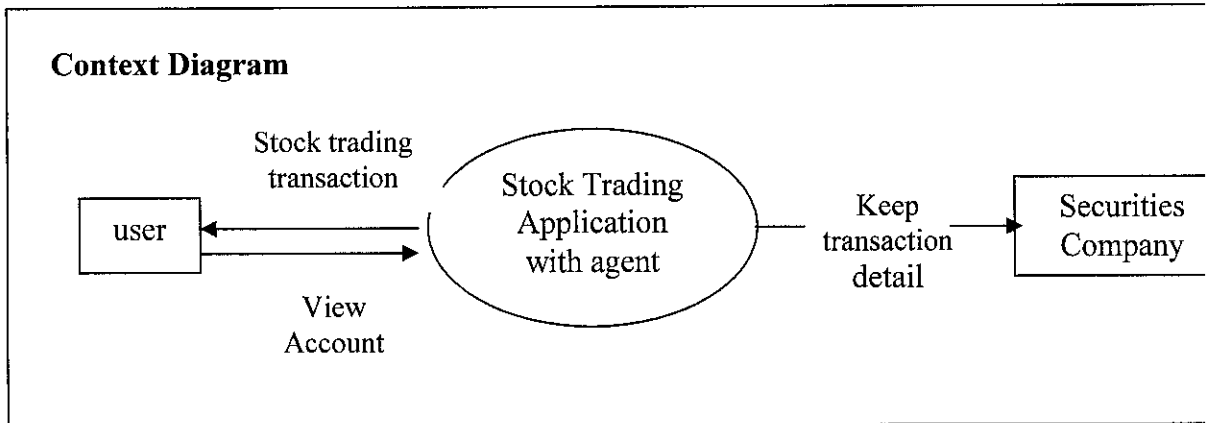
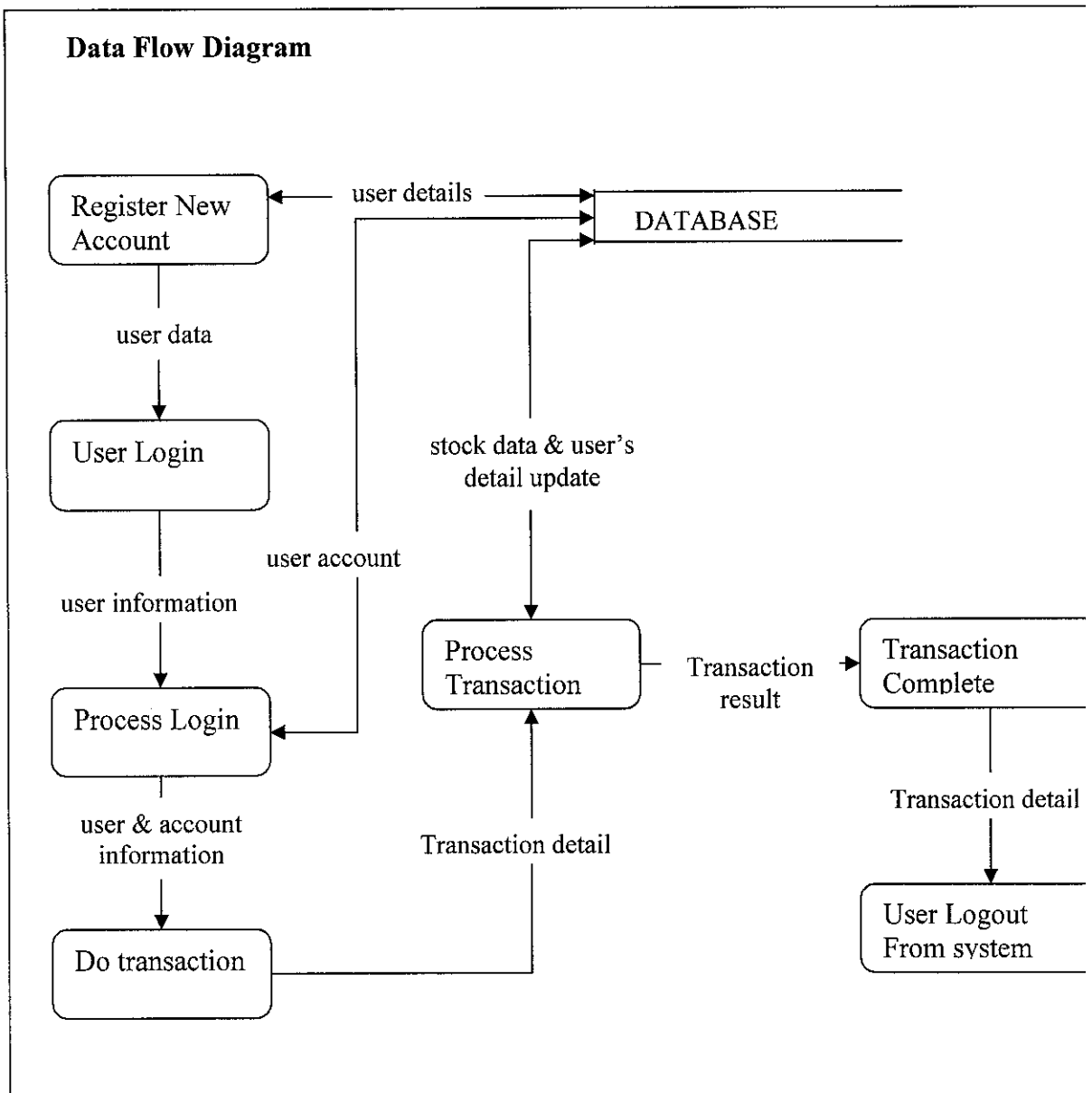


Diagram 2 – Data Flow Diagram



4.3 DATABASE ARCHITECTURE

























Database is the important part for this system. The main purpose to have database is to store data that captured from user, to update stock data as well as user data and finally to query data from database. For this project MYSQL database has been used.

There are four tables inside the database like shown in figure 18. There are agent, stockcode, user and userstock. The table agent is used to stored data about the buyer, stock code, the quantity of shares to buy using agent and the price of stock.

Figure 18 – Database Table

Server: localhost Database: stock

Structure SQL Export Search Query Drop

Table	Action	Records	Type	Collation	Size	Overhead
<input type="checkbox"/> agent	     	0	MyISAM	latin1_swedish_ci	1.1 KB	148 Bytes
<input type="checkbox"/> stockcode	     	76	MyISAM	latin1_swedish_ci	5.6 KB	-
<input type="checkbox"/> user	     	3	MyISAM	latin1_swedish_ci	2.1 KB	-
<input type="checkbox"/> userstock	     	1	MyISAM	latin1_swedish_ci	1.8 KB	756 Bytes
4 table(s)	Sum	80	--	latin1_swedish_ci	10.6 KB	904 Bytes

↑ Check All / Uncheck All / Check overheaded With selected: ▼

The second table is the stockcode table. This table is design to stored data regarding the stock such as stock code, stock name, the initial price and current price of the stock. The third table is user table. In this table, the data about user will be stored. For instance, to stored first name data, last name data, username and password data. All the data about user is taken during user did the registration process. The final table is userstock table. This is the composite table that resulting from many to many relationship between user table and stock table. The user stock table stored data about the conditions of the transactions that user want agent to perform. For instance, to keep data about the selling price, date to sell stock, status of stock and others.

4.4 DISCUSSION OF THE SYSTEM

In this section, the author will first discuss about the result that obtained from system testing, follow by explaining benefits of using agent in stock environment and finally discuss about future enhancement for the system. The purpose is to show to the audience about the impact of using the agent in stock trading environment.

4.4.1 Result from System Testing

System testing is the testing that performs on the system as a whole. Means that, the testing is carrying out when the sub-units of the system are integrated to be one complete system. Before the system testing can be performed, developer has to do the unit testing first. The unit testing means to test the sub-unit of the system with the purpose to ensure it work satisfactorily as planned. Unit testing is basically done by developer.

After the unit testing is done, the sub-unit of the system must be integrated to be single unit of system. Then the system testing should take place. For the system testing the author has test the system first before asking external testers to test the system. The external testers are the student of Universiti Teknologi Petronas that minor in corporate finance. They have basic knowledge about the stock market and some of them are traders that invest in the Kuala Lumpur Stock Exchange. The reasons to select students that have basic knowledge about stock environment are to ensure the testing process become easier and to get fast and relevant feedback from them.

There are ten testers that selected for system testing purpose. They have been divided into two groups which are group A and group B. For the first session, testers from group A are assigned to doing transaction manually. It means they have to purchase and sell the stock by their own self. The other group is assigned to use an agent for

transaction motive. For the second session, they have asked to change their role. Below is the description of testing.

4.4.1.1 Description of System Testing

1. System to be tested

Stock trading system using an agent

2. Purpose of System Testing

- To test the system to ensure it functioning well.
- To get tester’s feedback regarding the system.

3. Venue and date

Building 1, Lab 4 (Final Year Project Lab), October 24, 2005

4. Number of tester

10 Corporate Finance’s students

5. Number of Group – 2

6. Duration for each session 10 minutes

4.4.1.2 Data gathered from the first session of system testing

Group A – Manual Transaction

Table 1 – First Session Testing – Manual Transaction

	Tester 1	Tester 2	Tester 3	Tester 4	Tester 5
Stock selected (code)	7120	7090	2658	6432	7120
Transaction start (time)	0900	0900	0900	0900	0900
Buying price	1.55	1.04	2.15	0.62	1.04
Forecasting Selling price	1.70	1.19	2.30	0.77	1.19
Transaction end (time)	0907	0910	0910	0910	0910
Result	complete	incomplete	incomplete	incomplete	incomplete
Transaction duration	7 minutes	10 minutes	10 minutes	10 minutes	10 minutes

Group B – Agent Transaction

Table 2 – First Session Testing – Agent Transaction

	Tester 1	Tester 2	Tester 3	Tester 4	Tes
Stock selected (code)	7120	7090	2658	6432	71
Transaction start (time)	0915	0915	0915	0915	09
Buying price	1.61	1.01	2.17	0.57	1.
Forecasting Selling price	1.76	1.16	2.32	0.72	1.
Transaction end (time)	0917	0918	0918	0919	09
Result	incomplete	incomplete	incomplete	incomplete	incor
Transaction duration	2 minutes	3 minutes	3 minutes	4 minutes	2 mi

For the first session of testing group A has been assigned to do manual transaction while group B using agent for the transaction. Each of them has been given the stock code for the transaction. The transaction begins at 9 am for group A and 9 15 am for group B. The duration for the transaction is 10 minutes. Each of the testers has to buy the stock when the price is increase by 15 cent.

At the end of the transaction, we can see that only tester 1 from group A is manage to sell the stock successfully. The time taken for the transaction is around 7 minutes. The other four fail to complete the transaction. They have fully used the time that allocated to them. The average time that used by group A is 9 minutes and 23 second. For the group B, all testers are not able to sell the stock. This is because the price of the current stock is not exceeding the selling price that they set to the agent. The average taken by group B is 2 minutes and 48 second.

The author can conclude that the group that performs transaction manually spends much time more than the group that uses the agent to work for them. From the observation during the first testing session, the trading system runs satisfactorily.

4.4.1.3 Data gathered from the second session of system testing

Group A – Agent Transaction

Table 3 – Second Session Testing – Agent Transaction

	Tester 1	Tester 2	Tester 3	Tester 4	Tester 5
Stock selected (code)	7120	7090	2658	6432	7120
Transaction start (time)	0930	0930	0930	0930	0930
Buying price	1.57	1.00	2.18	0.58	1.00
Forecasting Selling price	1.72	1.15	2.33	0.73	1.15
Transaction end (time)	0932	0932	0933	0932	0932
Result	incomplete	complete	incomplete	incomplete	incomplete
Transaction duration	2 minutes	2 minutes	3 minutes	2 minutes	3 minutes

Group B – Manual Transaction

Table 4 – Second Session Testing – Manual Transaction

	Tester 1	Tester 2	Tester 3	Tester 4	Tester 5
Stock selected (code)	7120	7090	2658	6432	7120
Transaction start (time)	0945	0945	0945	0945	0945
Buying price	1.56	1.02	2.14	0.60	1.00
Forecasting Selling price	1.71	1.17	2.29	0.75	1.15
Transaction end (time)	0955	0955	0955	0954	0955
Result	incomplete	incomplete	incomplete	complete	incomplete
Transaction duration	10 minutes	10 minutes	10 minutes	9 minutes	10 minutes

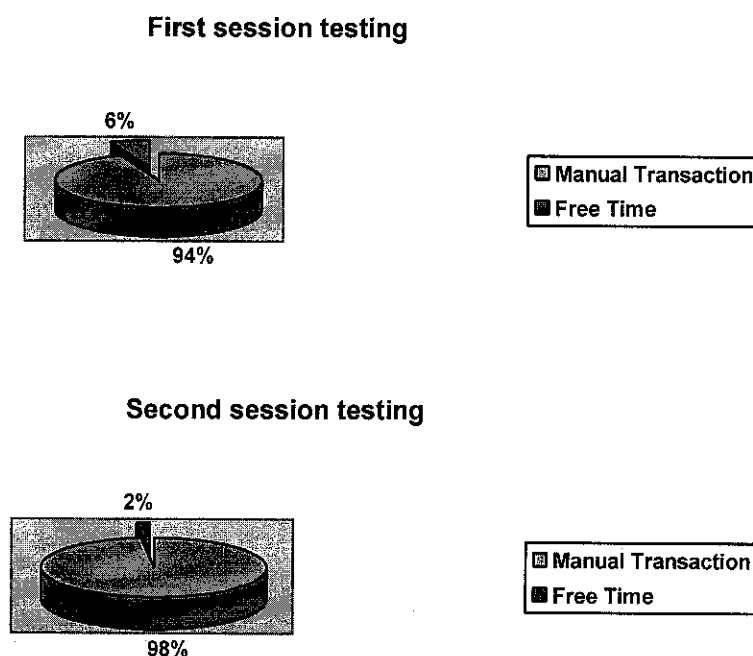
For the second session, tester from group A has been assigned to trade stock by using agent, while testers from group B has to trade manually. The other rules are same as the first session. During this session, the average time that used by group A is 2 minutes and 23 second, while the average time taken by group B is 9 minutes and 48 seconds. The result shows that the group that used agent for transaction manages to complete their transaction earlier than the other group. The system works well during the second testing session.

4.4.1.4 Discussion on the System Testing Result

The system testing that conducted on October 24, 2005 is run successfully. During the testing period, the system that being used by tester works well. All Testers are satisfied with the system and the way it works.

From the first session system testing table (4.4.1.2) and second session system testing table (4.4.1.3) we can see clearly that the testers used much time when they do transaction manually. However, by using agent to trade for them, the time that they used is lesser. To be more precise, let look at the chart below (figure19).

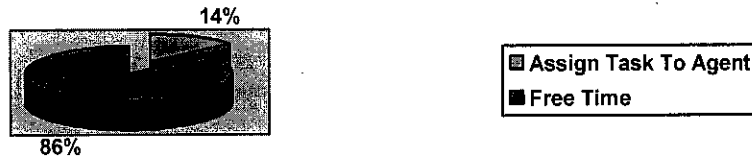
Figure 19 – Chart - Manual transactions for both sessions



Testers have used an average of 94 percents and 98 percents time that allocated to them repetitively when they asked to do transaction manually. From the observation, testers used most of their time to monitor the trend of stock price. They will wait until the stock price equal or exceed the forecasting buying price in order to buy that particular stock. However, different scenario occurs when they use an agent to trade stock for them. Please refer to figure 20.

Figure 20 – Chart - Using Agent to do transaction for both sessions

First session testing



Second session testing



From The chart above, we can see that the average time taken to assign task to agent is about 14 percents and 12 percents repetitively for both session. The testers just used small percentages of the time allocated to them to complete their task. After completed their task, the agent will take over the task for the transaction.

From the testing, the author has the grounds to prove that by using the agent in stock trading environment, the transaction process become easier and lot of time can be saved. Other than that the transaction is less prone to error. The agent will work exactly to the request or order from the user.

The second thing that has been proven is the agent has the ability to complete the task. From the table 4.4.1.3, which is the data that gathered from the second session of testing, we can see that the agent has completed the task that given by the number two (2) tester. The agent has successfully bought the defined stock within the duration of 10 minutes that allocated to for that particular session. The agent also has the chance to complete other tasks that delegate by other testers if the duration of the testing is

extending until the current price of the stock is equal or exceed the defined buying price.

In term of efficiency, using agent to do transaction is more efficient that do the transaction manually. The term efficiency refers to the time that allocate by tester in performing the transaction. Basically, the user will use the time allocated to them to monitor the trend of the stock price in order to buy the particular stock with the right price. However, by using the agent, the testers only used 20 percents – 30 percents of their total allocated time to delegate the job to agent and then waiting for the agent to complete the task and report the result.

4.4.2 Future Enhancement

Future enhancement is needed in order to ensure the system can be reconstructed to be more interactive and effective. There are four opinions regarding this matter. These opinions come from author and some of the testers that performed system testing. The opinions will be explained in detail with some relevance points as below.

1. System should be integrated with Kuala Lumpur Stock Exchange (KLSE). The purpose to integrate the system with KLSE database is to get the actual data regarding stock information, stock price, KLSE announcement and others.
2. Agent must have intelligent features. The main purpose to have intelligent agent is to ensure the agent can work independently and able to make their own decision. In the current system, the agent is work by referring to the data that set up by user. For instance, agent should have the ability to gather and analysis data by their own and advice user in certain aspects.
3. Program agent with more functionality. To have an interactive system, the agent should be programmed to have variety of functionality. For instance, the agent

should have the ability to travel into network and find information regarding the stock market, do analysis on stock data based on previous five years data and forecasting the future stock price. In the current system, the agent just has the ability to do buying and selling transaction for user.

4. System should be design more interactive and informative. The interface of the system should be designed more interactive by adding some multimedia elements such as animated graphic and updated stock price chart. Other than that, some educational information should also be included in the system in order to attract more traders to use the system.

CHAPTER 5

CONCLUSION

5.0 CONCLUSION OF THE PROJECT

The project has been completed within the time frame. It means that the whole durations for completing the project is about four months. The successful or failure of the project is dependently on the ability to accomplish all of the objectives that defined during the beginning of the project.

For this project, there are two objectives that have been set up. The first objective is to perform small scale study regarding the agent. To satisfy this objective, the author has gather information from variety of sources such as internet, books and asking some advice and idea from expert. The purpose to perform small scale of study is to find out the information about agent such as the features that agent should comprise the environment that appropriate to use agent service and the best programming language to used for agent programming purpose. Beside that, the study on the existing system that using agent technology is also being done.

The first objective of the project is important and prerequisite for the second objective to be executed. The second objective of the project is to develop simple stock trading application that using an agent. The knowledge regarding agent that gathered to fulfill the first objective is then used for system development purpose. This is the key factor that determines the achievement of the project. From the testing that was conducted, it shows that the second objective of project is fulfilled. The testers that involved in the system testing had given the positive impression and comment regarding the system especially on the agent. However, there are some aspects that should be taking into considerations for future enhancement.

It is difficult to cope up with user requirements that always change over the time. The incremental development or system enhancement must progress constantly in order to satisfy them.

The selection of the suitable system development life cycle is also important. Developer has to analyze the characteristics of the project such as the time, cost, user involvement and other issue that related to the project in order to choose the best methodology. If all of the important criteria or characteristic is not carefully analysis, the project might face several problem such as it can not be completed in the period that has been defined or the cost of the project may be exceed the planned cost. For this project, the author has select the Rapid Application Development methodology as the system development life cycle after analyzing all the important criteria in order to ensure the project is able to complete within the time and fulfill all the objectives that defined in the planning phase.

When the development phase is completed, the product or the output of the project should be analyzed carefully in order to ensure it is meet the entire requirements that defined in the planning phase. In this phase, the system testing should be conducted to test the capability of the system and to ensure the system is free from error. All the data regarding the testing and the result should be analyze carefully and transform it into meaningful information. That information must be kept as future reference in case the system facing some problem or to expand the functionality of the system.

There are many important thing that author has realized and learned from the implementation of the project. The most important thing is to clearly understand the scopes and objectives of the project. This is to ensure end product of the project meet all the requirements and expectations. The second thing is to determine the time to be taken to complete each task and follow the schedule that has been set up during the planning phase. This is to ensure the progression of the project run smoothly and able to be completed as scheduled. Next, is to ensure all information and data regarding the project is properly documented and stored for future reference. The selection of the suitable methodology is also important. Developer can not easily employ any

methodology that they like without performing proper analysis and investigation. This is because every methodology has difference purpose and can resulting different impact.

As the conclusion of the project, the author is really satisfying with the product and the way the project progress. The project run smoothly without any major problems and it is able to complete within the time frame.

REFERENCES

- Journal – Intelligent Agent, Market and Competition - Kees Jonkheer
- www.agentland.com
- Text Book – Developing Intelligent agent system – practical guideline
(Lin Padgham, Micheal Minikoff)

- www.planet-source-code.com

- Text Book - Mobile Agent for Telecommunication Applications
(Ahmed Karmouch and Roger Impey)

- Text Book – Constructing Intelligent Agent With Java
- Intelligent Agent and Their Applications
(Lakhmi C. Jain, Zhengxin Chen and Nikhil)

- Text Book – PHP – By Example (Toby Butzon)

- Other source obtain through www.yahoo.com and www.google.com

APPENDICES

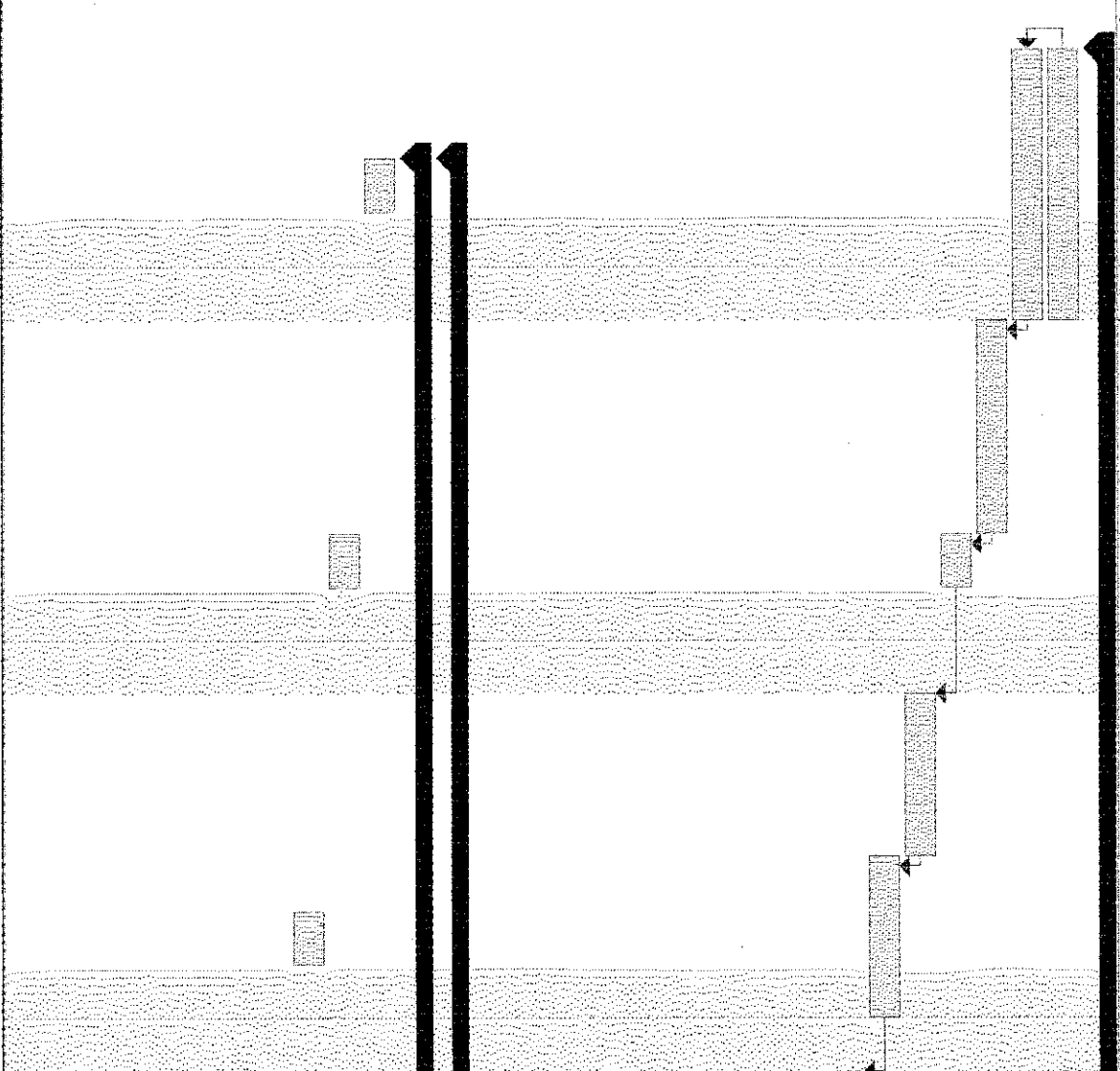
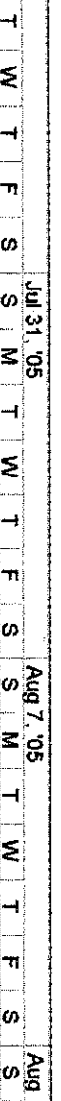
ID	Task Name	Duration	Start	Finish
1	1.0 Requirement Planning Ph	19 days	Wed 7/27/05	Mon 8/22/05
2	1.1 Define project scope	3 days	Wed 7/27/05	Sun 7/31/05
3	1.2 Define project constrai	3 days	Wed 7/27/05	Sun 7/31/05
4	1.3 Prepare preliminary rep	4 days	Mon 8/1/05	Thu 8/4/05
5	1.4 Submit preliminary repi	1 day	Fri 8/5/05	Fri 8/5/05
6	1.5 Identify data	3 days	Mon 8/8/05	Wed 8/10/05
7	1.6 Identify process	2 days	Thu 8/11/05	Sat 8/13/05
8	1.7 Identify interface	3 days	Mon 8/15/05	Wed 8/17/05
9	1.8 Tools and equipment t	3 days	Thu 8/18/05	Mon 8/22/05
10	2.0 Design Phase	6 days	Tue 8/23/05	Tue 8/30/05
11	2.1 Design Interface	6 days	Tue 8/23/05	Tue 8/30/05
12	3.0 Construction Phase	32 days	Thu 9/1/05	Sun 10/16/05
13	3.1 Build system	28 days	Thu 9/1/05	Mon 10/10/05
14	3.2 Test system	2 days	Tue 10/11/05	Wed 10/12/05
15	3.3 Debug errors	2 days	Thu 10/13/05	Sun 10/16/05
16	4.0 Cutover Phase	1 day	Thu 10/27/05	Thu 10/27/05
17	4.1 Product Deliverable	1 day	Thu 10/27/05	Thu 10/27/05
18	4.2 Final Presentation	1 day	Thu 10/27/05	Thu 10/27/05
19	5.0 Documentation	66 days	Fri 7/29/05	Fri 10/28/05
20	5.1 Weekly Report	66 days	Fri 7/29/05	Fri 10/28/05
21	5.1.1 Week 1	1 day	Fri 7/29/05	Fri 7/29/05
22	5.1.2 Week 2	1 day	Fri 8/5/05	Fri 8/5/05
23	5.1.3 Week 3	1 day	Fri 8/12/05	Fri 8/12/05
24	5.1.4 Week 4	1 day	Fri 8/19/05	Fri 8/19/05
25	5.1.5 Week 5	1 day	Fri 8/26/05	Fri 8/26/05
26	5.1.6 Week 6	1 day	Fri 9/2/05	Fri 9/2/05
27	5.1.7 Week 7	1 day	Fri 9/9/05	Fri 9/9/05
28	5.1.8 Week 8	1 day	Fri 9/16/05	Fri 9/16/05
29	5.1.9 Week 9	1 day	Fri 9/23/05	Fri 9/23/05
30	5.1.10 Week 10	1 day	Fri 9/30/05	Fri 9/30/05
31	5.1.11 Week 11	1 day	Fri 10/7/05	Fri 10/7/05

Project: GanttChartFYBackup
Date: Wed 12/14/05

Task Split

Milestone Summary

External Tasks External Milestone Deadlines



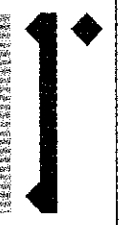
ID	Task Name	Duration	Start	Finish	Calendar																					
					T	W	T	F	S	Jul 31 '05	S	M	T	W	T	F	S	Aug 7 '05	S	M	T	W	T	F	S	Aug
32	5.1.12 Week 12	1 day	Fri 10/14/05	Fri 10/14/05																						
33	5.1.13 Week 13	1 day	Fri 10/21/05	Fri 10/21/05																						
34	5.1.14 Week 14	1 day	Fri 10/28/05	Fri 10/28/05																						
35	5.2 Preliminary Report	1 day	Fri 8/5/05	Fri 8/5/05																						
36	5.3 Progress Report	1 day	Fri 9/9/05	Fri 9/9/05																						
37	5.4 Final Report	1 day	Fri 10/28/05	Fri 10/28/05																						

Project: GanttChartFYBackup
Date: Wed 12/14/05

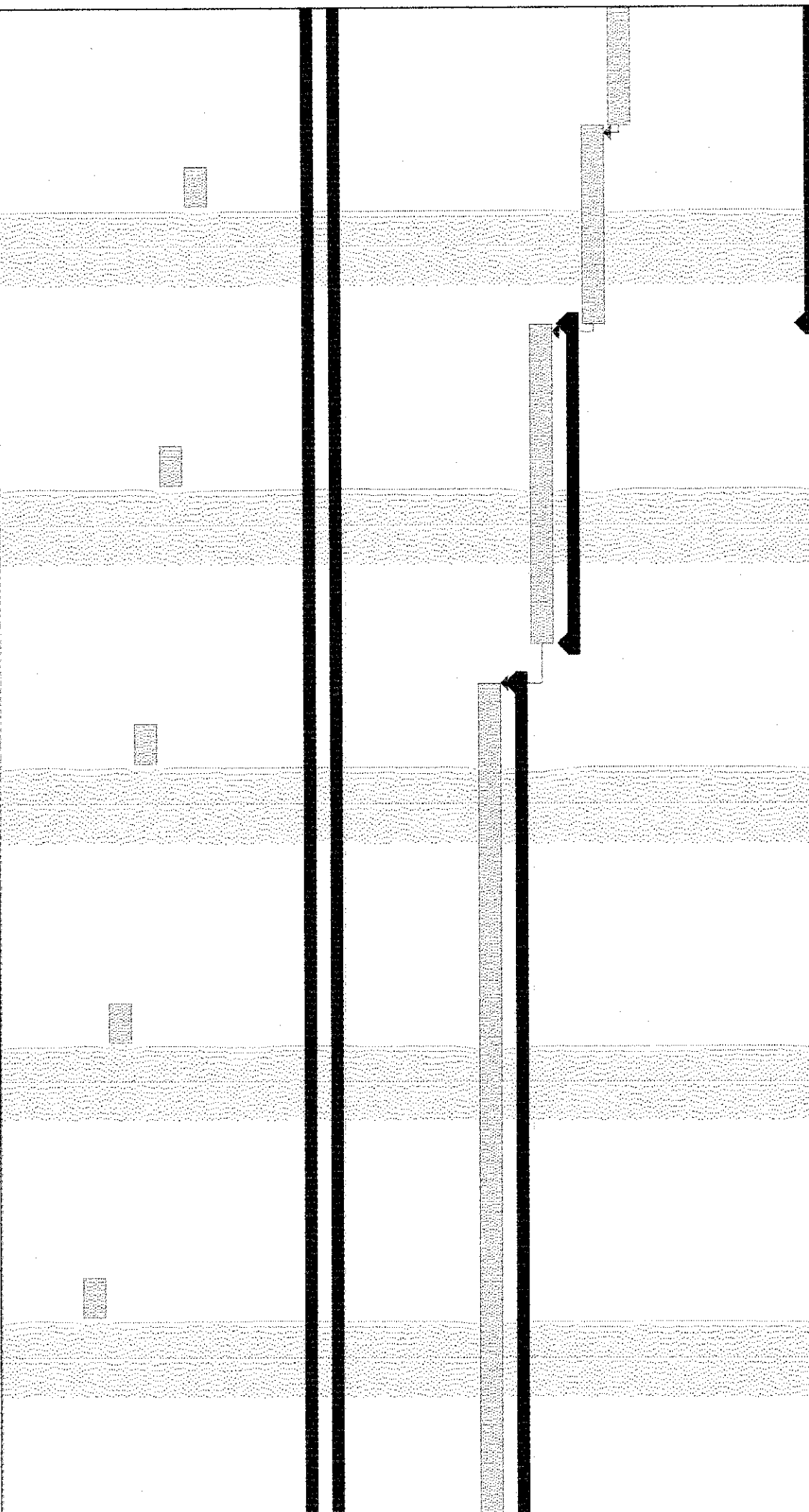
Task
Split

Milestone
Summary


External Tasks
External Milestone





4, '05 M T W T F S S Aug 21, '05 M T W T F S S Aug 28, '05 M T W T F S S Sep 4, '05 M T W T F S S Sep 11, '05 M T W T F S S Sep 18, '05 M T W




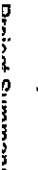
Project: GanttChartFYBackup
Date: Wed 12/14/05


Task 


Milestone 

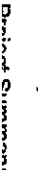
External Tasks 

Split 

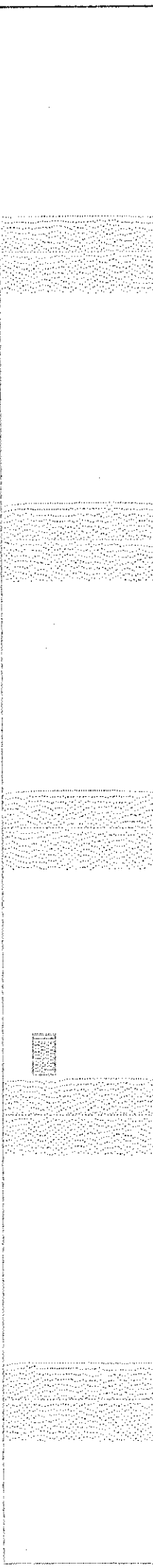
Summary 

External Milestone 

Deadline 

Deadline Summary 

4, '05 M T W T F S S Aug 21, '05 M T W T F S S Aug 28, '05 M T W T F S S Sep 4, '05 M T W T F S S Sep 11, '05 M T W T F S S Sep 18, '05 M T W



Task



Milestone



External Tasks



Split



Summary



External Milestone



Summary



Dryland Summary

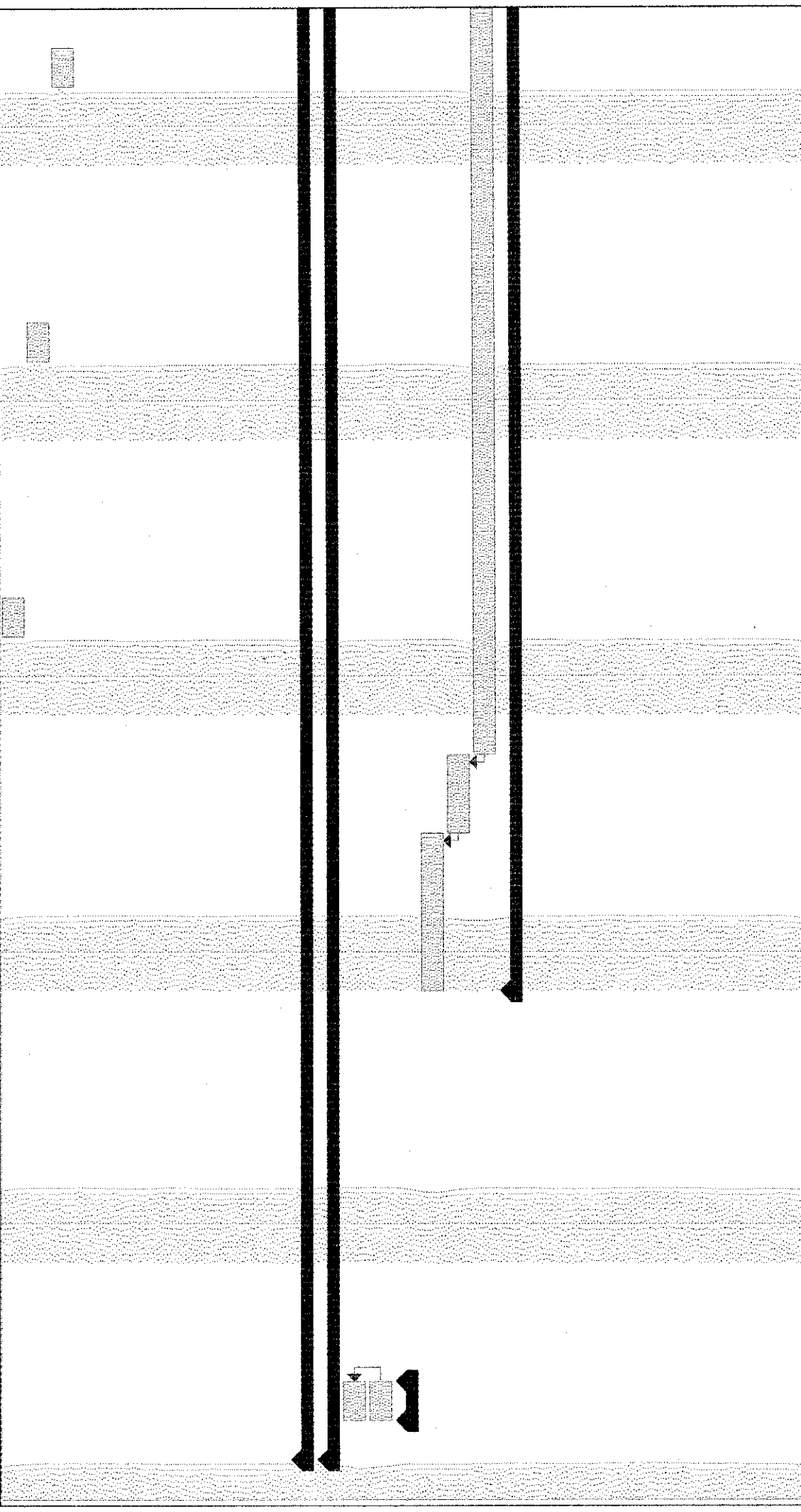


Deadline



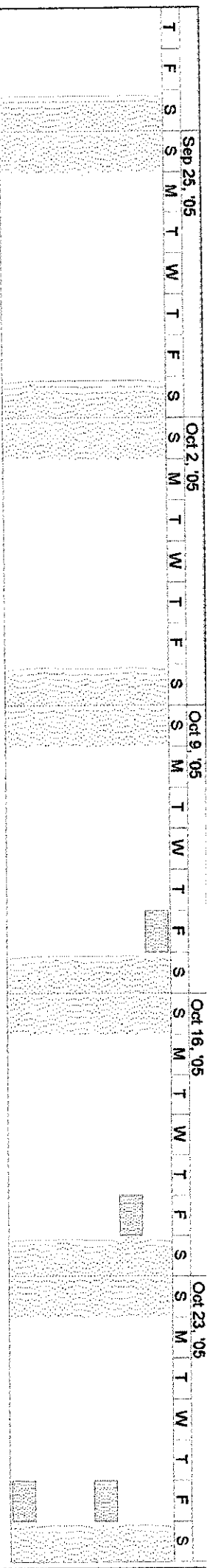
Project: GanttChartFYBackup
Date: Wed 12/14/05

T F S S M T W T F S S M T W T F S S M T W T F S
 Sep 25, '05 Oct 2, '05 Oct 9, '05 Oct 16, '05 Oct 23, '05



Project: GanttChartFY\Backup
 Date: Wed 12/14/05

Task **Milestone** **External Tasks** **External Milestone**
Split **Summary** **Deadline**



Project: GanttChart\F\pbackup
 Date: Wed 12/14/05

Task		Milestone		External Tasks	
Split		Summary		External Milestone	
Drwnracs		Project Summary		Deadline	

AGENTHELP CODING

```
<?php
session_start();
session_register("login");
session_register("user");
session_register("level");
?>
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
<head>
<title>V I R T U A L Trade</title>
<meta name="description" content="" />
<meta name="keywords" content="" />
<link rel="stylesheet" type="text/css" href="/style/camstyle.css" />
<link type="" rel="" title=""
href="" />
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></head>

<body bgcolor="#FFFFFF" onLoad="clock()">
<span id="pendule" style="position:absolute;left:850;top:55;"></span>
<?php
$conn=mysql_connect(localhost);
mysql_select_db(stock,$conn);

$sql = "select scode from userstock where buyer=" . $_HTTP_SESSION_VARS ["user"] . " and status=""";
$result=mysql_query($sql);

?>

<!-- header -->
<form name="form1" method="post" action="Transactionpage.php">

<table cellpadding="0" cellspacing="0" border="0" width="94%" summary="navigation bar">
<tr>
<td colspan="3" bgcolor="#0099cc" ></td>
<td width="3%" rowspan="3" align="right" bgcolor="#0099CC" ><div align="justify">
</div>
<div align="right"> </div></td>
</tr>
<tr>
<td bgcolor="#0099cc" width="63%" ></td>
<td bgcolor="#0099cc" width="23%" ><script language="JavaScript">

<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->
<!-- Author: www.cgiscript.net -->

<!-- Begin

// Get today's current date.
var now = new Date();

// Array list of days.
var days = new Array('Sunday','Monday','Tuesday','Wednesday','Thursday','Friday','Saturday');

// Array list of months.
```

```

<table width="915" border="0" cellpadding="0" cellspacing="0" summary="page content">
<tr>
<td width="282" height="840" align="right" valign="top" bgcolor="#66CCFF">
<table width="277" border="0" cellpadding="0" cellspacing="0" summary="University shield and montage">
<tr><td width="296" height="50"><p><font color="#006633" size="6">V i R T U
A L Trade</font></p>
<p>On <font color="#000033" size="3"><strong>Kuala Lumpur Stock Exchange</strong></font> </p></td>
</tr>
<tr>
<td>&nbsp;</td>
</tr>
<tr>
<td><div align="center"></div></td>
</tr>
</table>
<table width="277" border="0" align="right" cellpadding="0" cellspacing="0" summary="content left-hand
panel">
<tr>
<td align="right"> <p class="homeheading">&nbsp;</p></td>
<td width="38">&nbsp;</td>
</tr>
<tr>
<td bgcolor="#cccccc"></td>
</tr>
<tr valign="baseline">
<td align="right">&nbsp;</td>
</tr>
<tr valign="baseline">
<td align="right"> <a href="MainPage.php">Home </a> </td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
<tr valign="baseline">
<td align="right"> <a href="registration.php">Registration</a> </td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
<tr valign="baseline">
<td align="right"><a href="aboutStockSimulator.php">About the Stock
Simulator</a> </td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
<tr valign="baseline">
<td height="23" align="right"><strong>Agent Role in Stock</strong> <font size="2" face="Arial">decision.
  The role of an agent in this site is to help you to trade. You can
  order an </font></td>
</tr>
<tr>
  <td height="19" nowrap="nowrap"><font size="2" face="Arial">agent to
    find some information about certain stock and ask them to buy or sell
    stock</font></td>
</tr>
<tr>
  <td height="19" nowrap="nowrap"><font size="2" face="Arial">on your
    behalf. On other word, the agent can act as your stock broker. </font></td>
</tr>
<tr>
  <td>&nbsp;</td>
</tr>
<tr>
  <td bgcolor="#cccccc" ></td>
</tr>
<tr>
  <td height="24"><font color="#000000" size="2" face="Arial"><font color="#003300" size="3"
face="Arial"><strong>How
  to deligate job to AGENT?</strong></font></font></td>
</tr>
<tr>
  <td bgcolor="#cccccc" ></td>
</tr>
<tr>
  <td height="3"><p><strong></strong></p></td>
</tr>
<tr>
  <td>&nbsp;</td>
</tr>
<tr>
  <td height="15"><font size="2" face="Arial">If you tired to trade stock
  by your self, you can order an agent to work for you. </font></td>
</tr>
<tr>
  <td height="15"><font size="2" face="Arial">There are several steps
  to be done before you can use the agent. </font></td>
</tr>
<tr>
  <td height="16"><font size="2" face="Arial">&nbsp;</font></td>
</tr>
<tr>
  <td height="9"><font size="2" face="Arial">1. Specify the task of an
  agent ( choose the agent task from the menu )</font></td>
</tr>
<tr>
  <td height="10"><font size="2" face="Arial">2. If you want an agent
  to trade on behalf you, justify the constraint such as what </font></td>
</tr>
<tr>
  <td height="19"><font size="2" face="Arial"> type of stock you want
  to purchase, at what price to buy and sell the stock. </font></td>
</tr>
<tr>
  <td height="4">&nbsp;</td>
</tr>
<tr bordercolor="#000000" bgcolor="#66CCFF">

```

```

<td height="22">&nbsp;</td>
<td colspan="2">&nbsp;</td>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<td height="22" colspan="5"><font color="#FFFF00"><strong>Note:
  You can only choose one from the available selling options</strong>
</font></td>
</tr>
<tr>
<td height="24"><font color="#FFFF00">&nbsp;</font></td>
  <td colspan="2">
    <input type="checkbox" name="checkbox3" value="sprice">
    <strong><font color="#FFFF00">Stock price at</font></strong>
  </td>
<td><font color="#FFFF00">
  <input name="sellat" type="text" id="price2" size="10">
  </font></td>
<td><font color="#FFFF00">&nbsp;</font></td>
</tr>
<tr>
<td>
  <input type="checkbox" name="checkbox4" value="percentgain">
  <strong><font color="#FFFF00">Return from stock at</font></strong><strong></strong></td>
<td><font color="#FFFF00">
  <input name="percentgain" type="text" id="return2" size="10">
  <strong>percent</strong></font> <font color="#FFFF00">&nbsp;</font></td>
<td>&nbsp;</td>
</tr>
<tr>
<td>
  <input type="checkbox" name="checkbox5" value="lostpercet">
  <strong><font color="#FFFF00">Losses from stock at</font></strong>
  <strong></strong></td>
<td><font color="#FFFF00">
  <input name="percentlost" type="text" id="time2" size="10">
  <strong>percent</strong></font> <font color="#FFFF00">&nbsp;</font></td>
<td>&nbsp;</td>
</tr>
<tr>
<td height="26">&nbsp;</td>
  <td colspan="2">
    <input type="checkbox" name="checkbox6" value="date">
    <font color="#FFFF00"><strong>Date and Time at</strong></font>
  </td>
<td>
  <font color="#FFFF00">
  <select name="day" size="1" id="day">
    <option>-</option>
    <option value="01">01</option>
    <option value="02">02</option>
    <option value="03">03</option>
    <option value="04">04</option>
    <option value="05">05</option>
    <option value="06">06</option>
    <option value="07">06</option>
    <option value="08">08</option>
    <option value="09">09</option>
    <option value="10">10</option>
  </font>

```

<option>-</option>
<option value="00">00</option>
<option value="01">01</option>
<option value="02">02</option>
<option value="03">03</option>
<option value="04">04</option>
<option value="05">05</option>
<option value="06">06</option>
<option value="07">07</option>
<option value="08">08</option>
<option value="09">09</option>
<option value="10">10</option>
<option value="11">11</option>
<option value="12">12</option>
<option value="13">13</option>
<option value="14">14</option>
<option value="15">15</option>
<option value="16">16</option>
<option value="17">17</option>
<option value="18">18</option>
<option value="19">19</option>
<option value="20">20</option>
<option value="21">21</option>
<option value="22">22</option>
<option value="23">23</option>
<option value="24">24</option>
<option value="25">25</option>
<option value="26">26</option>
<option value="27">27</option>
<option value="28">28</option>
<option value="29">29</option>
<option value="30">30</option>
<option value="31">31</option>
<option value="32">32</option>
<option value="33">33</option>
<option value="34">34</option>
<option value="35">35</option>
<option value="36">36</option>
<option value="37">37</option>
<option value="38">38</option>
<option value="39">39</option>
<option value="40">40</option>
<option value="41">41</option>
<option value="42">42</option>
<option value="43">43</option>
<option value="44">44</option>
<option value="45">45</option>
<option value="46">46</option>
<option value="47">47</option>
<option value="48">48</option>
<option value="49">49</option>
<option value="50">50</option>
<option value="51">51</option>
<option value="52">52</option>
<option value="53">53</option>
<option value="54">54</option>
<option value="55">55</option>
<option value="58">56</option>
<option value="57">57</option>
<option value="58">58</option>
<option value="59">59</option>
</select>

CODING ON FOUR OPTIONS TO USE AGENT

```
<body><br>
<?php
$conn=mysql_connect(localhost);
mysql_select_db(stock,$conn);

if ($_POST["checkbox"] == "buy")
{
$sql = "insert into agent values (" . $HTTP_SESSION_VARS ["user"] . ", " . $_POST["ascode"] . ", " .
$_POST["bquantity"] . ", " . $_POST["buyprice"] . ")";
mysql_query($sql) or die($sql);
}

if ($_POST["checkbox2"] == "sell")
{
if ($_POST["radiobutton"] == "sprice")
{
$sql="update userstock set sellat=" . $_POST["sellat"] . " where buyer=" . $HTTP_SESSION_VARS ["user"] . " and
scode=" . $_POST["scode"] . """;
mysql_query($sql) or die("error");
echo $sql;
}
if ($_POST["radiobutton"] == "percentgain")
{
$sql="update userstock set gainpercent=" . $_POST["percentgain"] . " where buyer=" . $HTTP_SESSION_VARS
["user"] . " and scode=" . $_POST["scode"] . """;
mysql_query($sql) or die("error");
echo $sql;
}
if ($_POST["radiobutton"] == "lostpercent")
{
$sql="update userstock set lostpercent=" . $_POST["percentlost"] . " where buyer=" . $HTTP_SESSION_VARS
["user"] . " and scode=" . $_POST["scode"] . """;
mysql_query($sql) or die("error");
$myid = date("m") . date("d") . date("Y") . date("H") . date("i");
echo $myid;
}
if ($_POST["radiobutton"] == "date")
{
}

$sql="update userstock set date2sell=" . $_POST["month"] . $_POST["day"] . $_POST["year"] . $_POST["hour"] .
$_POST["minute"] . " where buyer=" . $HTTP_SESSION_VARS ["user"] . " and scode=" . $_POST["scode"] . """;
mysql_query($sql) or die("error");
$myid = date("m") . date("d") . date("Y") . date("H") . date("i");
echo $myid;
}
}

?>
</p>

</body>
</html>
```

```

<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body>
<?php

$conn=mysql_connect("localhost");
mysql_select_db("stock",$conn);

$username = $_POST["username"];
$password = $_POST["password"];

$sql="SELECT username FROM user WHERE username=' . $username . ' AND password=' . $password . '";
$rs=mysql_query($sql);

if (mysql_num_rows($rs) > 0)
{
$result = mysql_fetch_array($rs);
$_SESSION["login"] = TRUE;
$_SESSION["user"] = $username;
$_SESSION["level"] = $result["level"];

echo "<Script Language=\JavaScript\>";
echo "window.location='transactionpage.php'";
echo "</script>";

}
else
{
echo "<Script Language=\JavaScript\>";
echo "alert('\Invalid User Name or Password\');";
echo "history.back()";
echo "</script>";
}
?>
</body>
</html>

```

SCRIPT FOR REGISTRATION

```

<script language="javascript">

function validate()
{
if (document.form1.fname.value == "")
{
alert("please enter first name");
document.form1.fname.focus();
return(false);
}
if (document.form1.lname.value == "")
{
alert("please enter last name");
document.form1.lname.focus();
return(false);
}
if (document.form1.city.value == "")

```

```

echo "</script>";
}
else
{
$$SQL="INSERT into user (fname,lname,city,username,password) values (" . $fname . ", " . $lname . ", " . $city . ", " .
" . $username . ", " . $password . ")";
mysql_query($$SQL);
}
echo "<Script Language=\"JavaScript\">";
echo "window.location = 'login.php'";
echo "</script>";
?>
</body>
</html>

```

CODING TO TAKE OVER ACTION FROM AGENT

```

<?php
session_start();
session_register("login");
session_register("user");
session_register("level");
?>
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body>
<?php
$conn=mysql_connect("localhost");
mysql_select_db("stock",$conn);

$sql = "delete from agent where buyer=" . $_HTTP_SESSION_VARS["user"] . " and scode=" . $_GET["scode"] . "";
mysql_query($sql) or die($sql);
$sql = "update userstock set sellat=", gainpercent=", lostpercent=", date2sell=", status=" where buyer=" .
$_HTTP_SESSION_VARS["user"] . " and scode=" . $_GET["scode"] . "";
mysql_query($sql);
?>
</script>
window.location="Transactionpage.php";
</script>
</body>
</html>

```

MAINPAGE CODING

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
<head>
<title>V I R T U A L Trade</title>
<meta name="description" content="" />
<meta name="keywords" content="" />
<link rel="stylesheet" type="text/css" href="/style/camstyle.css" />
<link type="" rel="" title=""
href="" />

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></head>

<body bgcolor="#FFFFFF" onLoad="clock()">
<span id="pendule" style="position:absolute;left:850;top:55;"></span>

<!-- header -->

<table cellpadding="0" cellspacing="0" border="0" width="94%" summary="navigation bar">
<tr>
<td colspan="3" bgcolor="#0099cc" ></td>
<td width="3%" rowspan="3" align="right" bgcolor="#0099CC" ><div align="justify">
</div>
<div align="right"> </div></td>
</tr>
<tr>
<td bgcolor="#0099cc" width="63%" ></td>
<td bgcolor="#0099cc" width="23%" ><script language="JavaScript">

<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->
<!-- Author: www.cgiscript.net -->

<!-- Begin

// Get today's current date.
var now = new Date();

// Array list of days.
var days = new Array('Sunday','Monday','Tuesday','Wednesday','Thursday','Friday','Saturday');

// Array list of months.
var months = new
Array('January','February','March','April','May','June','July','August','September','October','November','December');

// Calculate the number of the current day in the week.
var date = ((now.getDate()<10) ? "0" : "")+ now.getDate();

// Calculate four digit year.
function fourdigits(number) {
    return (number < 1000) ? number + 1900 : number;
}

// Join it all together
today = days[now.getDay()] + ", " +
    months[now.getMonth()] + " " +
    date + ", " +
    (fourdigits(now.getYear()));
```

```

        <td width="279"><div align="center"></div></td>
        <td width="3">&nbsp;</td>
        <td width="2">&nbsp;</td>
    </tr>
</table>
<table width="277" border="0" align="right" cellpadding="0" cellspacing="0" summary="content left-hand
panel">
    <tr>
        <td colspan="2" align="right"> <p class="homeheading">&nbsp;</p></td>
        <td width="38">&nbsp;</td>
    </tr>
    <tr>
        <td colspan="2" bgcolor="#cccccc" ></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" ></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2"><a href="registration.php">Registration
        </a></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" ></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2"><a href="aboutStockSimulator.php">About
the Stock Simulator </a></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" ></td>
    </tr>
    <tr valign="baseline">
        <td height="25" colspan="2" align="right"><a href="login.php">User Login
        </a></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" ></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2">&nbsp;</td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" ></td>
    </tr>
</table>
</td>
<td width="2" bgcolor="#666666">&nbsp;</td>

```

```

        <td nowrap="nowrap"><p>&nbsp;</p></td>
    </tr>
    <tr>
        <td height="19">&nbsp;</td>
    </tr>
    <tr>
        <td bgcolor="#cccccc" ></td>
    </tr>
    <tr>
        <td></td>
    </tr>
    <tr>
        <td nowrap="nowrap"> <div class="hplist"><strong><font color="#000033" face="Arial">News
& amp; Events </font></strong></div></td>
    </tr>
    <tr>
        <td></td>
    </tr>
    <tr>
        <td bgcolor="#cccccc" ></td>
    </tr>
    <tr>
        <td></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td><div class="hplist">
            <p class="homenews"><br />
            <br />
            <br />
            </p>
            <!-- end news story -->
        </div></td>
    </tr>
    <tr>
        <td></td>
    </tr>
    <tr>
        <td bgcolor="#cccccc" ></td>
    </tr>
</table>
</td>
</tr>
</table>
<!-- end content -->

<!-- footer -->
<br /><br />
<table cellpadding="0" cellspacing="0" border="0" width="98%" summary="footer">
<tr>
<td bgcolor="#0099cc" colspan="2" ></td>

```

```

if ($count <= 2)
{
$sSQL="UPDATE stockcode set cprice=" . ($rs2["cprice"]-0.006) . " WHERE cprice=" . $rs2["cprice"] . """;
mysql_query($sSQL);
$sql = "select * from agent order by buyer";
$result2 = mysql_query($sql);
while ($rs3 = mysql_fetch_array($result2))
{
echo $rs2["scode"] . $rs3["scode"];
if ($rs2["scode"] == $rs3["scode"] && ($rs2["cprice"]+0.005) <= $rs3["price"])
{

$sql = "insert into userstock values (" . $rs3["buyer"] . ", " . $rs2["scode"] . ", " . $rs3["quantity"] . ", " .
($rs2["cprice"]+0.005) . ", , , , ,)";
mysql_query($sql) or die($sql);
$sql = "delete from agent where buyer=" . $rs3["buyer"] . " and scode=" . $rs3["scode"] . """;
mysql_query($sql) or die($sql);
}
}

//echo $sSQL;
$count++;
}
if ($count >= 3)
{
$sSQL="UPDATE stockcode set cprice=" . ($rs2["cprice"]+0.004) . " WHERE cprice=" . $rs2["cprice"] . """;
mysql_query($sSQL);
$sql = "select * from agent order by buyer";
$result2 = mysql_query($sql);
while ($rs3 = mysql_fetch_array($result2))
{
echo $rs2["scode"] . $rs3["scode"];
if ($rs2["scode"] == $rs3["scode"] && ($rs2["cprice"]+0.005) <= $rs3["price"])
{

$sql = "insert into userstock values (" . $rs3["buyer"] . ", " . $rs2["scode"] . ", " . $rs3["quantity"] . ", " .
($rs2["cprice"]+0.005) . ", , , , ,)";
mysql_query($sql) or die($sql);
$sql = "delete from agent where buyer=" . $rs3["buyer"] . " and scode=" . $rs3["scode"] . """;
mysql_query($sql) or die($sql);
}
}

//echo $sSQL;

if ($count == 5)
{
$count=0;
}
$count++;
}
}

if ($counter == 3)
{
if ($count==0)
{

$sSQL="UPDATE stockcode set cprice=" . ($rs2["cprice"]-0.002) . " WHERE cprice=" . $rs2["cprice"] . """;
mysql_query($sSQL);
$sql = "select * from agent order by buyer";
$result2 = mysql_query($sql);

```

PRICEUPDATE CODING

```
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body>
<?php
$conn=mysql_connect("localhost");
mysql_select_db("stock",$conn);

$sSQL="SELECT * FROM stockcode ORDER BY sname ASC";
$rs=mysql_query($sSQL);

$counter = $_GET["counter"];
if ($counter=="")
{
$counter=0;
}
else
{
$counter++;
}

if ($counter==4)
{
$counter=0;
}
$count = 0;

while ($rs2 = mysql_fetch_array($rs)){

if ($counter==0)
{
if ($count == 0)
{

$count=$count+1;
$sSQL="UPDATE stockcode set cprice=" . ($rs2["cprice"]+0.005) . " WHERE cprice=" . $rs2["cprice"] . "";
mysql_query($sSQL);
$sql = "select * from agent order by buyer";
$result2 = mysql_query($sql);
while ($rs3 = mysql_fetch_array($result2))
{
echo $rs2["scode"] . $rs3["scode"];
if ($rs2["scode"] == $rs3["scode"] && ($rs2["cprice"]+0.005) <= $rs3["price"])
{
$sql = "insert into userstock values (" . $rs3["buyer"] . ", " . $rs2["scode"] . ", " . $rs3["quantity"] . ", " .
($rs2["cprice"]+0.005) . ", ", ", ", ", ", ", ", ";
mysql_query($sql) or die($sql);
$sql = "delete from agent where buyer=" . $rs3["buyer"] . " and scode=" . $rs3["scode"] . "";
mysql_query($sql) or die($sql);
}
}
//echo $sSQL;
}
else
{
$count=0;
$sSQL="UPDATE stockcode set cprice=" . ($rs2["cprice"]-0.005) . " WHERE cprice=" . $rs2["cprice"] . "";
```



```
window.location=targetURL;  
return  
}  
setTimeout("countredirect()",1000)  
}
```

```
countredirect()  
!-->  
</script>  
</body>  
</html>
```

```

<table cellpadding="0" cellspacing="0" border="0" width="94%" summary="navigation bar">
  <tr>
    <td colspan="3" bgcolor="#0099cc" ></td>
    <td width="3%" rowspan="3" align="right" bgcolor="#0099CC" ><div align="justify">
      </div>
      <div align="right"> </div></td>
  </tr>
  <tr>
    <td bgcolor="#0099cc" width="63%" ></td>
    <td bgcolor="#0099cc" width="23%" ><script language="JavaScript">

```

```

<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->
<!-- Author: www.cgiscript.net -->

```

```

<!-- Begin

```

```

// Get today's current date.
var now = new Date();

```

```

// Array list of days.
var days = new Array('Sunday','Monday','Tuesday','Wednesday','Thursday','Friday','Saturday');

```

```

// Array list of months.
var months = new
Array('January','February','March','April','May','June','July','August','September','October','November','December');

```

```

// Calculate the number of the current day in the week.
var date = ((now.getDate()<10) ? "0" : "")+ now.getDate();

```

```

// Calculate four digit year.
function fourdigits(number) {
    return (number < 1000) ? number + 1900 : number;
}

```

```

// Join it all together
today = days[now.getDay()] + ", " +
    months[now.getMonth()] + " " +
    date + ", " +
    (fourdigits(now.getYear()));

```

```

// Print out the data.
document.write(today);

```

```

// End -->
</script></td>
  <td bgcolor="#0099cc" width="11%" >
    <script language="JavaScript">
<!-- Original: Xavier R. (xav@lougaou.com) -->
<!-- Modified: Benjamin Wright, Editor -->
<!-- Web Site: http://www.lougaou.com/ -->

```

```

<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->

```

```

<!-- Begin
function clock() {
if (!document.layers && !document.all) return;
var digital = new Date();

```

```

        <td align="right" colspan="2"> <a href="MainPage.php">Home </a> </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2"> <a href="registration.php">Registration</a>
        </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2"><a href="aboutStockSimulator.php">About
the Stock Simulator</a> </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>
    <tr valign="baseline">
        <td align="right"><a href="Transactionpage.php">Transaction Page</a> </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="2"></td>
        <td></td>
    </tr>
    <tr valign="baseline">
        <td height="25" colspan="2" align="right"><a href="agenthelp.php">Agent
Role in Stock</a> </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>
    <tr valign="baseline">
        <td align="right" colspan="2"><a href="login.php">User Login</a> </td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>
    <tr>
        <td colspan="3"></td>
    </tr>

```

```

$spec = $rs2["spec"];

$spec = split("\r\n",$spec);

?>
<tr bgcolor="#9999FF">
  <td><font color="#000033"><?php echo $rs2["sname"] ?></font></td>
  <td><font color="#000033"><?php echo $rs2["scode"] ?></font></td>
  <td><?php echo $rs2["sprice"] ?></td>
  <td><?php echo $rs2["cprice"] ?></td>
  <td><div align="center">
    <input type="button" value="Buy" onClick="buyStock('<?php echo $rs2["scode"] ?>','<?php echo
$rs2["sname"] ?>','<?php echo $rs2["cprice"] ?>');">
  </div></td>
</tr>

                                <?php
                                }
                                ?>

</table>
<p>&nbsp;</p></div>
</div>
</div></td>
</tr>
<tr>
  <td height="18" colspan="4" nowrap="nowrap">&nbsp;</td>
</tr>
<tr>
  <td colspan="4" bgcolor="#cccccc" ></td>
</tr>
<tr>
  <td colspan="4"> <div class="hplist"><strong></strong></div>
  <strong></strong></td>
</tr>
<tr>
  <td height="24" colspan="4">&nbsp;</td>
</tr>
<tr>
  <td colspan="4" bgcolor="#cccccc" ></td>
</tr>
<tr>
  <td colspan="4"><p></p></td>
</tr>
<tr>
  <td height="2" colspan="4" bgcolor="#cccccc" ></td>
</tr>
</table>
<p>&nbsp;</p></div>
</tr>
</table>
<!-- end content -->

<!-- footer -->
<br /><br />
<table cellpadding="0" cellspacing="0" border="0" width="99%" summary="footer">
<tr>
<td bgcolor="#0099cc" colspan="2" ><img

```

TRANSACTIONPAGE CODING

```
<?
session_start();
session_register("login");
session_register("user");
session_register("level");
?>
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
<head>
<title>V I R T U A L Trade</title>
<meta name="description" content="" />
<meta name="keywords" content="" />
<link rel="stylesheet" type="text/css" href="/style/camstyle.css" />
<link type="" rel="" title=""
href="" />
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></head>

<body bgcolor="#FFFFFF" onLoad="clock()">
<script>
function over()
{
var scode;
scode = prompt("Please enter stock code:", "");
if (!scode)
return(false);
window.location = "takeover.php?scode=" + scode;
}

</script>
<script language="JavaScript">
function sell(scode,sname,sprice)
{
var yes;
document.form4.stock2sell.value=scode;
document.form4.sprice.value=sprice;
yes = confirm("Are you sure want to sell: " + sname);

if (!yes)
{
return(false);
}

document.form4.action="transactionpage.php?stock2sell=" + scode;
document.form4.submit();
}

</script>
<?php
$conn=mysql_connect("localhost");
mysql_select_db("stock",$conn);

if ($_GET["stock2sell"] != "")
{
$sql = "update userstock set sellprice=" . $_POST["sprice"] . ", status='SOLD' where scode=" . $_POST["stock2sell"]
. """;
mysql_query($sql)or die($sql);
}
$result4 = "select * from userstock where buyer=" . $_HTTP_SESSION_VARS["user"] . """;
$result4 = mysql_query($sql);
```



```

<tr>
  <td align="center"></div></td>
</tr>
</table>
<table width="282" border="0" align="right" cellpadding="0" cellspacing="0" summary="content left-hand
panel">
  <tr>
    <td colspan="2" align="right"> <p class="homeheading">&nbsp;</p></td>
    <td width="43">&nbsp;</td>
  </tr>
  <tr>
    <td colspan="2" bgcolor="#cccccc" ></td>
    <td></td>
  </tr>
  <tr valign="baseline">
    <td align="right" colspan="2">&nbsp;</td>
    <td></td>
  </tr>
  <tr valign="baseline">
    <td align="right" colspan="2"> <a href="MainPage.php">Home </a> </td>
    <td></td>
  </tr>
  <tr>
    <td colspan="3" ></td>
  </tr>
  <tr valign="baseline">
    <td align="right" colspan="2"> <a href="registration.php">Registration</a>
    </td>
    <td></td>
  </tr>
  <tr>
    <td colspan="3" ></td>
  </tr>
  <tr valign="baseline">
    <td align="right" colspan="2"><a href="aboutStockSimulator.php">About
the Stock Simulator</a> </td>
    <td></td>
  </tr>
  <tr>
    <td colspan="3" ></td>
  </tr>
  <tr valign="baseline">
    <td align="right" colspan="2"><a href="agenthelp.php">Agent
Role in Stock</a> </td>
    <td></td>
  </tr>
  <tr>
    <td colspan="3" ></td>
  </tr>
  <tr valign="baseline">

```



```

<td width="47" nowrap="nowrap">&nbsp;</td>
<td width="47" nowrap="nowrap">&nbsp;</td>
<td width="47" nowrap="nowrap">&nbsp;</td>
<td width="47" nowrap="nowrap">&nbsp;</td>
<td width="46" nowrap="nowrap">&nbsp;</td>
</tr>
<tr>
<td height="9" colspan="10" nowrap="nowrap">&nbsp;</td>
</tr>
<tr>
<td colspan="10"><strong><font color="#006600" size="2" face="Arial">Cash
Account</font></strong></td>
</tr>
<tr>
<td colspan="10">&nbsp;</td>
</tr>
<tr>
<td colspan="10"><form name="form1" id="form1" method="post" action="">
<table width="71%" border="0">
<tr bgcolor="#FFFFFF">
<td width="36%" height="18"><strong><font color="#660066" size="2" face="Arial">Amount
Purchase </font></strong></td>
<td width="29%"><div align="right"><font color="#660066" size="2"
face="Arial"><strong>&nbsp;</strong></font></div></td>
<td width="35%"><div align="right"><font color="#003300" size="2" face="Arial"><strong><?php echo
$ampurchase; ?></strong></font></div></td>
</tr>
<tr bgcolor="#FFFFFF">
<td><strong><font color="#660066" size="2" face="Arial">Amount
Sold </font></strong></td>
<td><div align="right"><font size="2" face="Arial">&nbsp;</div><div align="right"><font size="2" face="Arial"><b><?php echo $amtsold; ?></b></font></div></td>
</tr>
<tr bgcolor="#FFFFFF">
<td><strong><font color="#660066">Transaction Fee</font></strong></td>
<td><div align="right"><font size="2" face="Arial">&nbsp;</div><div align="right"><font size="2" face="Arial"><b><?php echo $fee; ?></b></font></div></td>
</tr>
<tr bgcolor="#FFFFFF">
<td><strong></strong></td>
<td><font size="2" face="Arial">&nbsp;</font></td>
<td><div align="right"></div></td>
</tr>
<tr bgcolor="#FFFFFF">
<td><strong>Total Profit / Loss</strong></td>
<td><div align="right"><font size="2" face="Arial">RM</font></div></td>
<td><div align="right"><?php echo $profitloss; ?></div></td>
</tr>
</table>
</form></td>
</tr>
<tr>
<td colspan="10"><font color="#006600" size="2" face="Arial"><strong>Investment
Portfolio</strong></font></td>
</tr>
<tr>
<td colspan="10" bgcolor="#cccccc" ></td>
</tr>
<tr>

```

```

        <?php
                }
            ?>
        </table></td>
    </tr>
    <tr>
        <td height="24" colspan="10"><strong> </strong></td>
    </tr>
    <tr>
        <td colspan="10" bgcolor="#cccccc" ></td>
    </tr>
    <tr>
        <td height="6" colspan="10"><p><font color="#000066"><strong>Agent Log:</strong></font></p></td>
    </tr>
    <tr>
        <td height="30" colspan="10"><form name="form4" method="post" action="">
            <p>
                <textarea name="textfield" cols="75" rows="10">
Stock pending to buy:
-----
<?php
while ($myrs = mysql_fetch_array($myresult))
{
echo "Stock code: " . $myrs["scode"] . " Quantity: " . $myrs["quantity"] . " Buy at: " . $myrs["price"] . "\n";
}
echo "\n\n";
echo "Stock pending to sell:\n";
echo "-----\n";
$$sql = "select * from userstock where buyer=" . $_SESSION_VARS["user"] . " and status=ACTIVE";
$result = mysql_query($sql);
while ($rs5 = mysql_fetch_array($result))
{
echo "Stock code: " . $rs5["scode"] . " Quantity: " . $rs5["quantity"] . "\n";
}
?>
                </textarea>
            <p>
            <p>
                <input type="button" name="Submit2" value="Take Over" onClick="over();">
            <p>
                <input type="hidden" name="stock2sell">
                <input type="hidden" name="sprice">
            </form></td>
    </tr>
    <tr>
        <td height="2" colspan="10" bgcolor="#cccccc" ></td>
    </tr>
    </table>
    <p>&nbsp;</p></td>
</tr>
</table>
<!-- end content -->

<!-- footer -->
<br /><br />
<table cellpadding="0" cellspacing="0" border="0" width="99%" summary="footer">
<tr>
<td bgcolor="#0099cc" colspan="2" ></td>
</tr>

```