TITLE: Data Mining In Computer Auditing

WXES3182

Hiromi Hong	WET000304
Siew Lan	WET000307
Valery Fred Lee	WET000309

SUPERVISOR: Mr. Teh Yin Wah

MODERATOR: Assoc. Prof. Dr. Zaitun Abu Bakar

Contents

Contents of Figure

Contents of Table

ABSTRACT

In the past two decades, database technology has been evolved in order to turn the abundance of data into useful information. This evolution provokes the emergence of data mining tools, which perform data analysis and extract data patterns.

Data mining in computer auditing, these two businesses areas seem to be able to integrate. In which, auditing needs something to uncover patterns or suspiciousness while data mining can fulfill that need.

This report will firstly cover the introduction of data mining in computer auditing, project objectives, data mining problems or issues, project scope, and expected outcome. The report will cover the literature review that started with an introduction to computer auditing, introduction to data mining, data mining techniques (classification, neural network and sequential analysis), and the existing data mining software. This report will explain the methodology that is used to develop this system, which is Waterfall Model with Prototyping. A research is conducted during the process of finding information that is related to this topic by examining the current data mining literature. The functional and non-functional requirements for this project also will be discussed. Beside this, the development tools that will be used to develop this project are also listed in the last chapter.

In this thesis, we'll also explain how our FRIDCA system is develop in the system analysis and design chapter. Also included how FRIDCA is an integrated system consisting of fraud detection, intrusion detection and credit card approval is developed. We will explain further on how FRIDCA system can adapt to the ever evolving world of technology. However we also included the future enhancement of how FRIDCA can be applied in the real world.

ACKNOWLEDGEMENT

First of all, we would like to take this opportunity to express my deepest appreciation and my utmost gratitude to Mr. Teh Ying Wah, our supervisor for this thesis project, for guiding us towards the process of writing this assignment, thanks for his unwavering support, encouragement, advice, and time. His explanation did help us a lot.

Secondly, we would like to express our appreciation to our moderator, Assoc. Professor Dr. Zaitun Abu Bakar for her useful advice and guidance for our project since the VIVA presentation session that had gone through smoothly. Not forgetting also for her precious time spent during the VIVA presentation as well.

Next, we would like to thank all those people who are directly or indirectly contributed to the success of making this assignment and helping us in solving our problems especially our friends, Lee Li Fei, Barau Eddy, Kieu Kee Ching, Juhaida and Nor Rafidah.

Finally, we would like to convey our special thanks to our family for their support throughout this project and life. Thank you all once again for your kind support, cooperation, invaluable guidance, precious advice, and many more.

Thank you.

Hiromi Hong	WET000304
Siew Lan	WET000307
Valery Fred Lee	WET000309

CHAPTER 1

INTRODUCTION

1.1 Introduction to the project

In years to come, data warehouse will become commonplace as most organizations intend to build one. To audit such an organization, auditors will have to deal with copiousness of data. It implies that auditors can no longer use only reporting or summarizing tools, but also have to combine tools that can automate and extract information from large amount of data. One of the possibilities is data mining.

Currently, auditors incorporate their primary manual audit processes with computerized tools including general-purpose and generalized audit software. However, such software allows auditors only to examine a company's data in a variety of formats by running varied query commands but not to extract any information from that data.

In the past two decades, database technology has been evolved in order to turn the abundance of data into useful information. This evolution provokes the emergence of data mining tools, which perform data analysis and extract data patterns.

These two businesses areas seem to be able to integrate. One, auditing, needs something to uncover patterns or suspiciousness while the other, data mining, can fulfill that need. In this thesis, I will try to find possibilities of using data mining as a tool in ameliorating audit performance and to evaluate the success of them.

Efficiency, accuracy, and an open architecture play an important role in data mining and are main requirement for data mining. The technique that will be use in this project is sequence pattern technique. The detail will be discussed in chapter 2.

1.2 Project Objective

The project objective of this thesis is to evaluate the usefulness of data mining techniques in supporting auditing works. The main objectives of developing this project are outline in the following:

- To builds intelligent system that can help business discover hidden patterns in their data.
- Identified high-risk and low-risk customers.
- Help business to understand the purchasing behavior of their key customers, detect likely credit card or insurance fraud, and predict credit application approval and detect any unauthorized activity through intrusion detection method.
- Large databases can analyze by users to solve business decision problems.
- The results of the decision can feed in to the appropriate touch point systems (email systems, web centers, call centers, etc.) So that the right offers received by the right customers.
- Enables customers to discover previously undetected facts present in their business-critical data.
- Protect computer system against intrusion from intruder and hence prevent fraudulent activity.

1.3 Data Mining Problems/Issues

Data mining systems are depend on databases to provide the raw data for input and his causes problems in that databases to be likely incomplete, large, dynamic, and noisy. Other problems that faced by data mining are:

- **Limited Information**

A database is usually designed in different purposes. A data will cause problems when some attributes that are very important to knowledge about the application domain are not present in the data and it may be impossible to reveal significant knowledge about a given domain.

- **Cost, Time and Effort**

The data mining setup can be expensive running into hundreds of thousands of dollars. Many man-hours of development are needed; involving complicated procedural steps and product choices. There is a need for data scrubbing or cleaning programs, and there is no single high-powered system that can handle this. Some of the data mining functions involve steep learning curves for the end-users, since higher computing power is directly related to the depth of knowledge on how the data mining system actually works. Writing SQL queries can be complex and difficult, even with a Windows-based front-end tool. Extensive training and practice are still needed for most users.

- **Low-end software**

Some of the lower-end software available for data warehouse analysis tools are available for thousands of dollars, but these are piece-meal modules, not the enterprise-wide solutions necessary for data mining operations and businesses. These have limited query capabilities and its inability to perform multidimensional analyses - impossible to ask open-ended questions to find associations between data items. Making additions, changes and other replacements to these smaller software systems can lead to integration and implementation problems. Many of the current data mining methods are not truly interactive and cannot incorporate prior knowledge about a problem except in simple ways.

- **Large databases**

The large size of business databases presents problems in terms of finding efficient algorithms for association rules. Large numbers of fields (or attributes) also increases the need for search space enormously, and in addition, it increases the chances that the data-mining algorithm will find patterns that are not valid in general. A characteristic of business databases is the dynamic nature of its data - variables maybe modified, deleted or augmented with new measurements over time.

- **Noise and missing values**

The data that contain in a database cannot assume as correct because database is usually contaminated by errors. Some of the data may even be miss-classified because of the attributes, which depend on measurement judgments or subjective, where it can rise to errors. Noise will happened when there is an error in the values of attributes or class information.

performance report or just to check other information. The administrator can assign certain users to log in to this system by create a new user name and password for them.

Auditors

Who can login to the system and manage the system. Besides that, the system enables them to view the statistics of the system and the behavior of the customers and so on.

1.5 Expected Outcome

The development of this project will be expecting some of the outcomes listed at the following:

- It must allow administrator, managers to sign-in and access to the system.
- It enables enterprise to identifying patterns, relationships and dependencies that impact on business outcomes.
- It should be able to fulfill business requirements and could achieve the objectives as proposed in this proposal report.
- Each module in the system must be clearly identified and has a specific direction that could provide a clear functions to the users and easy to maintain for the management.
- Provide a user-friendly interface with graphical user interface that is easy-to-use by everyone no matter what kind of computer background he or she has.
- Provide a manual or help module to assist both the administrators and customers in using the system.

1.6 Report Layout

The purpose of this project layout is to give an overall explanation of the major contents, which involved during the development of this project. Below is the report layout of this project.

Part 1: Introduction

This chapter gives and overview of the project, which includes the project introduction, objectives scope of project, project schedule and expected outcome.

Part 2: Literature review

This chapter gives a brief explanation on topic research that is relevant to this project. I'll focus on data mining process, techniques and algorithms especially using Classification, Sequential Pattern and Neural Networks algorithms. Some explanation of integrating data mining in computer auditing. Comparison to other existing system.

Part 3: Methodology

This chapter emphasized on the methodology and gives an explanation about the technique and research that can be used to solve on the project problems that already list out.

Part 4: System Analysis

The system analysis of the project explains how the requirements for this project were acquired and the analysis of the results. Besides that, it also analyst the development tools available and then choose the best tools or software to developed the system.

Part 5: System design

This chapter shows the interface of the FRIDCA System as well as some explanation of how the system works by showing through the data flow diagram.

1.7 Project Schedule

In order to reduce inherent uncertainty in determining the time estimations, the expected time of all the activities will be estimated optimistically. For the project schedule, please refer to table 1.1.

No	Task	Start Date	End Date	Duration
1	Research on Thesis Title	13/3/2003	3 /4/2003	3 weeks
2	Requirement Analysis	16/3/2003	5/4/2003	3
3	Literature review	16/3/2003	21/4/2003	6
4	System Analysis & design	20/3/2003	1/5/2003	10
5	System development	20/5/2003	16/8/2003	14
6	Implementation & Testing	3/7/2003	26/9/2003	10
7	Documentation	15/3/2003	1/10/2003	36

Table 1.1: Project Schedule

CHAPTER 2

LITERATURE REVIEW

2.1 Definition and Purposes of Literature Review

Recent advances in information technology have made the various industries in Malaysia to realize the need to upgrade or develop a new information system. Many companies in western country have used data mining in managing and analyzing their data especially business organizations.

To develop a data mining system in computer auditing using neural network algorithms, the literature review is an important process in system development where it indicates findings on the project or thesis, summarization or the findings, analysis of the findings as well as the synthesis of the system proposed.

Literature review will ensure the understanding of the system requirements that will be used to build or develop the system. Knowledge and information gained from the literature review will enable the best and most suitable development tools to be chosen to develop a system that achieve its objectives.

The following are some of the steps taken for literature review:

- Collect materials about data mining and computer auditing.
- Scan through the information obtained and understand the contents.
- Extract the important points and summarize it.
- Collate summary for analysis.
- Rewrite these document and present in documents.

2.2 An introduction to Computer Auditing

Most business processes are now automated. Regardless of whether they are in the private or public sector, companies are increasingly relying on Information Technology (IT) in all organizational areas. The profitability and the future viability of companies increasingly depend on the continued functioning of IT systems. Without them, there is often doubt if a company will survive. These IT systems also represent a considerable proportion of any company's capital budget.

Therefore, the Internal Auditor must participate in all aspects of IT to ensure that the company's assets are being protected and that suitable internal controls are in place to protect its information resources.

2.2.1 Internal Control

All internal audit work revolves around the concept of *internal control*. The following definition is drawn from the Institute of Internal Auditors' *Standards for the Professional Practice of Internal Auditing* and provides the basis for the work of the Computer Auditor, as well as the Internal Auditor.

Internal control is part of the management process. It is the actions taken by management to plan, organize and direct the performance of sufficient actions to provide reasonable assurance that the following objectives will be achieved:

- Accomplishment of established objectives and goals for operations and programs;
- The economical and efficient use of resources;
- The safeguarding of resources;
- The reliability and integrity of information;
- Compliance with policies, plans, procedures, laws and regulations.

2.2.2 Main Function of a Computer Auditor

Designs and monitors control systems, which ensure the integrity and security of data and reviews the organization's computing environment and usage of computer facilities.

Tasks include:

- Analyses information processing systems to assess their completeness, accuracy, validity, and efficiency
- Assesses whether business systems process authorized transactions completely, accurately, and in a timely manner
- Reviews application systems and associated business procedures to ensure that they meet desired business objectives in a timely and efficient manner
- Participates in the design of new systems to ensure that the resulting systems is efficient, effective and well controlled
- Reviews the computing environment of organizations to assess the operations, systems software, systems development and security procedures

- Reviews the acquisition of software and hardware in terms of economy, efficiency and ability to meet operating requirements
- Investigates the usage of computing facilities
- Liaises with data processing management and systems users, and prepares reports recommending improvements in the management of computing facilities

2.2.3 The Need for Computer Auditing

Computer auditing, a specialization of internal auditing, grew up in the early 1970s when it became clear that typical Internal Auditors did not possess the technological knowledge or skills required to access information stored on computer systems. Until then, the *black box* approach had been adopted. In other words, they audited what went into the systems and what came out. However, they paid no regard to what happened to the information while it was being processed and stored.

The first development in computer auditing was to create *audit programs* to extract information to be used as the basis for the audit. This moved on to carrying out reviews of the application programs and other general IT control to ensure that they were all adequately controlled. From there, the Computer Auditor began to look at the systems software, which provides the environment within which the application programs work.

With time, the specialization has been fine-tuned so that now there are two specialist qualifications for Computer Auditors: the Qualification in Computer Auditing (QiCA) from *the Institute of Internal Auditors - United Kingdom (IIA - UK)* and the Certified Information Systems Auditor (CISA) from *the Information Systems, Audit and Control Association (ISACA).*

Despite this, it is felt that all Internal Auditors should participate in the review of Information Technology, not just Computer Auditors. As it is increasingly difficult to clearly delineate between the computer and the rest of the organization, audit reviews of business areas and application systems must include a review of the automated process. In fact, all auditors must now be computer literate.

Having said this, computer audit skills are still very much in demand and the need for Computer Auditors will remain for the foreseeable future. This is emphasized by the rapid developments happening today in the world of Information Technology. These rapid changes also require that Computer Auditors be constantly updating their skills and technical knowledge. These changes are nowhere more evident than in the trend away from mainframes to the use of client/server technology and distributed processing. This move has also meant greater emphasis on control within the user areas. The Computer Auditor must be able to provide advice to a new and wider range of customers, most of whom do not have the knowledge of the IT professional.

It is perhaps worth mentioning that there is sometimes confusion between the roles of Computer Audit and Computer Security. The *Computer Audit* function is responsible for providing an organization with independent, objective views on the level of security that is applied over Information Technology. This often includes providing advice to line management. The *Computer Security* function is responsible for implementing security in the IT environment and will also provide advice. There is a crossover of tasks in the area of providing advice. It is the responsibility of user management to choose what advice to accept. The successful Computer Auditor will learn to co-exist with the Computer Security function and work together for the benefit of the whole organization, ensuring professional standards are maintained at all times.

Furthermore, given how widespread technology has been implemented across the whole of the business world, it would be impossible to be specific about Computer Audit requirements in any particular industry sector. The Computer Auditor must be prepared to work in many different environments: from organizations who are totally dependent on mainframe computers, through those who have distributed their processing through a network of client/server applications, to those who are merely users of standalone mini and microcomputers. The Computer Auditor will constantly be faced with new challenges as the newer, emerging technologies are implemented.

Whatever technology is implemented within a company, the actual process of Computer Auditing can be broken down into specific areas which do not change and which are independent of the type of technology used by the customers. This segment provides a brief description of each area and some of the technologies that will be encountered within companies. These basic areas can be summarized as:

- The organization's policies and standards, especially as they relate to IS or information processing
- The organization and management of the computer facilities
- The physical environment in which the computer systems operate and the controls over that environment
- Contingency planning
- The operation of the system software
- The application systems development process
- Review of the business applications
- User programming (or end-user computing)

Finally, the Internal Auditor is employed to provide assurance to management that the company's assets are secure and that the company's procedures allow for control to be exercised. The auditor's main occupation is not to look for fraud. Despite press reports to the contrary, the major computer-related losses to industry come from error and "mismanagement." It is this error, which is searched for and trying to protect against. Internal auditors are part of the company and should strive to eradicate the police officer image that many have built for themselves over the years. They are there to help management, not to hinder them. Therefore, a public relations is also an important part of the work of the auditor.

2.2.4 Audit Procedures

Audit procedures in each auditing firm are different from one to another. However, mainly, audit procedures can be divided into four major steps as follows.

1) Client Acceptance or Client Continuance

Client acceptance (or client continuance in case of the consecutive acceptance) step is to evaluate the client and the auditing firm itself in order to decide whether or not the firm should engage with this client. Major concerns are;

⇒ *Assessment of engagement risks:* Each client provokes different level of risk to the firm. Risks include disrepute and intolerable misstatement disregard, which would lead to legal suit from clients. Some auditing firms have basic requirements of favorable clients. On the other hand, some have the list of criterion of unfavorable ones. Unfavorable clients, for example, are in dubious businesses or have too complex financial structures.

⇒ *Relationship conflicts:* Independence is a key issue in relation to audit; of equal importance is the auditor's objectivity and integrity. It is these factors, which guarantee reliability and trust in the accounts.

⇒ *Requirements of the clients:* The requirements include, for example, the qualification of auditors, time boundary, extra reports and estimated budget.

⇒ *Sufficient competent personnel available*

⇒ *Cost-Benefit Analysis:* It is to compare the potential costs of the engagement with the audit fee offered from the client. The major cost of audit engagement is professional staff charge.

If the client is accepted, a written confirmation, generally on an annual basis, of the terms of engagement is established between the client and the firm.

2) Planning

The objectives of planning step are to design and to develop an audit plan. It includes the followings.

- **Mobilization**

This step is to form the engagement team and to communicate among team members. First, the key team members have to be identified. Team members include one or more independent auditors who will sign the auditor's opinion prescribed in the engagement contract, specialists and assistant auditors. The mobilization meeting, or pre-planning meeting, should be conducted to communicate all engagement matters including client requirements and deliverables, level of involvement, tentative roles and responsibilities of each team member and other relevant substances. The meeting should also cover the determination of the most efficient and effective process of information gathering. In case of client continuance, a review of the prior year audit to assess scope for improving efficiency or effectiveness should be identified.

- **Client's Information Gathering**

In order to perform this step, the most important thing is the cooperation between client and audit team. The meeting is arranged to update the client's needs and expectations as well as their perception of their own business in the respect of management perspective and controls environment.

Next, within audit team members, a meeting is arranged in order to perform the preliminary analytical procedures. In other words, the following tasks are performed.

⇒ *Obtaining background information:* It includes the understanding of client's business and industry, the business objectives, legal obligations and related risks.

⇒ *Understanding system structures:* System structures include the system and computer environments, operating procedures and the controls embedded in those procedures.

⇒ *Control assessment:* Based upon information about controls identified from the meeting with clients and understanding system structures process, all controls are updated, assessed and documented. The concerns include control environment, general computerized (or system) controls, monitoring controls and application controls.

Audit team members' knowledge, expertise and experiences are considered as the most valuable tools in performing this step.

♦ **Risk Assessment**

Risk, in this case, is some level of uncertainty in performing audit works. Risks identified in the first two steps are gathered and assessed. The level of risks assessed in this step is directly lead to the audit strategy to be used. In other words, the level of tasks is based on the level of risks. Therefore, auditors must be careful not to understate or overstate the level of these risks.

Level of risks is different from one auditing area to another. In planning the extent of audit evidences of each auditing area, auditors primarily use audit risk model. Such audit risk model is as follows:

$$\boxed{\text{Planned Detection Risk} = \frac{\text{Acceptable Audit Risk}}{\text{Inherent Risk * Control Risk}}}$$

- **Planned detection risk**: Planned detection risk is the highest level of misstatement risk that audit evidences cannot detect in each audit areas. Auditors need to accumulate audit evidences until the level of misstatement risk is reduced to plan detection risk level. For example, if the planned detection risk is 0.05, audit evidences needed to be obtained is ninety-five percent so that there is only five percent misstatement risk left.

- **Acceptable audit risk:** Audit risk is the probability that auditor will unintentionally render inappropriate opinion on client's financial statements. Acceptable audit risk, therefore, is a measure of how willing the auditor is to accept that the financial statements may be materially misstated after the audit is completed (Arens, 2000, 261).

- **Inherent risk***:* Inherent risk is the probability that there are material misstatements in financial statements. There are many risk factors that affect inherent risk including errors, fraud (it will be explained in detail later in the chapter), business risk, industry risk and change risk. The first two are preventable and detectable but others are not. Auditors have to ensure that all risks are taken into account when considering inherent risk probability.

- **Control risk***:* Control risk is the probability that client's control system cannot prevent or detect errors. Normally, after defining inherent risks, controls that are able to detect or prevent such risks are identified. Then, auditors will assess whether the client's system has such controls or not and, if it has, how much they can put reliance on those controls. The heavier control reliance is, the lower control risk is. In other words, control risk represents auditor's reliance on client's control structure.

It is the responsibility of auditors to ensure that no risk factors of each audit area are left unaddressed and the evidences obtained are sufficient to reduce all risks to acceptable audit risk level.

- **Audit Program Preparation**

The purpose of this step is to determine the most appropriate audit strategy and tasks for each audit objective within each audit areas based on client's background information about related audit risks and controls identified from the previous steps.

Firstly, the audit objectives of each audit areas have to be identified. A primary purpose of audit strategy and tasks is to ensure that those objectives are materially met.

After addressing audit objectives, it is time to develop overall audit plan. The audit plan should cover audit strategy of each area and all details related to the engagement including client's needs and expectations, reporting requirements, timetable. Then, the planning at the level of details has to be performed. This detailed plan is known as tailored audit program. It should cover tasks identification and schedule, types of tests to be used, materiality, acceptable audit risk and person responsible. Notice that related risks and controls of each area are taken into account for prescribing audit strategy and tasks.

The finalized general plan should be communicate with clients in order to agree upon significant matters especially deliverables and timetable. Both overall audit plan and detailed audit program need to be clarified among the team as well.

3) Execution and Documentation

Briefly, this step is to perform following the audit program. It includes audit tests execution and documentation. Generally, two basic types of audit approaches auditors can use during execution phase are tests of controls and substantive tests. Documentation includes summarize the results of audit tests, level of satisfactory, matters found during the tests and recommendations. If there is an involvement of specialists, the process performed and the outcome have to be documented as well.

Communication practices are considered as the most important skill to perform this step. Not only with the clients (or staffs working for the client), it is also crucial to communicate among the team. Normally, it is a responsible of more senior auditor to

After addressing audit objectives, it is time to develop overall audit plan. The audit plan should cover audit strategy of each area and all details related to the engagement including client's needs and expectations, reporting requirements, timetable. Then, the planning at the level of details has to be performed. This detailed plan is known as tailored audit program. It should cover tasks identification and schedule, types of tests to be used, materiality, acceptable audit risk and person responsible. Notice that related risks and controls of each area are taken into account for prescribing audit strategy and tasks.

The finalized general plan should be communicate with clients in order to agree upon significant matters especially deliverables and timetable. Both overall audit plan and detailed audit program need to be clarified among the team as well.

3) Execution and Documentation

Briefly, this step is to perform following the audit program. It includes audit tests execution and documentation. Generally, two basic types of audit approaches auditors can use during execution phase are tests of controls and substantive tests. Documentation includes summarize the results of audit tests, level of satisfactory, matters found during the tests and recommendations. If there is an involvement of specialists, the process performed and the outcome have to be documented as well.

Communication practices are considered as the most important skill to perform this step. Not only with the clients (or staffs working for the client), it is also crucial to communicate among the team. Normally, it is a responsible of more senior auditor to coach less senior auditor. Techniques used are briefing, coaching, discussing, and reviewing.

A meeting with clients in order to discuss the issues found during execution process and the recommendations of those findings can be arranged either formally or informally. It is a good idea to inform about those issues to the responsible client before the completion step and to leave only critical matters to the management.

4) Completion

This step is like the final step of every other kind of projects. The results of everything are summarized, recorded, assessed and reported. Normally, the assistant auditors report their work results to the more senior (or in-charged) auditors. The in-charged should perform the final review to ensure that all necessary works are performed and that the audit evidences for each audit area are adequate. Also, the critical matters left from the execution process have to be finalized. The resolution of those matters might be either solved by client's management or by auditors (disclosing them in the auditor's opinion).

The last fieldwork for auditors is review of subsequent events. Fundamentally, based on accumulated audit evidences and audit findings, the auditor's opinion can be issued. Types of auditor's opinion are unqualified, unqualified with explanatory paragraph or modified wording, qualified, adverse and disclaimer.

After everything is done, it is time to arrange the clearance meeting with clients. Generally, auditors are required to report results and all conditions to the audit committee or senior management. Although not required, auditors often make suggestions to management to improve its performance. On the other hand, auditors can get feedback from the client according to their needs and expectations as well.

Also, auditors should consider evaluating their own performances in order to improve their efficiency and effectiveness. The evaluation includes summarizing client's comments, bottom-up evaluation (more senior auditors evaluate the work of assistant auditors) and top-down evaluation (get feedback from field work staffs).

2.2.5. Audit Approaches

Primarily, audit approaches fall into one of these two categories:-

i) <u>Tests of Controls</u>

Generally, control objectives can be categorized into these four broad categories:

- Validity
- Completeness
- Accuracy
- Restricted access.

ii) **Substantive Tests**

Substantive tests include the followings:-

- Analytical Procedures
- Detailed Tests of Transactions
- Detailed Tests of Balances

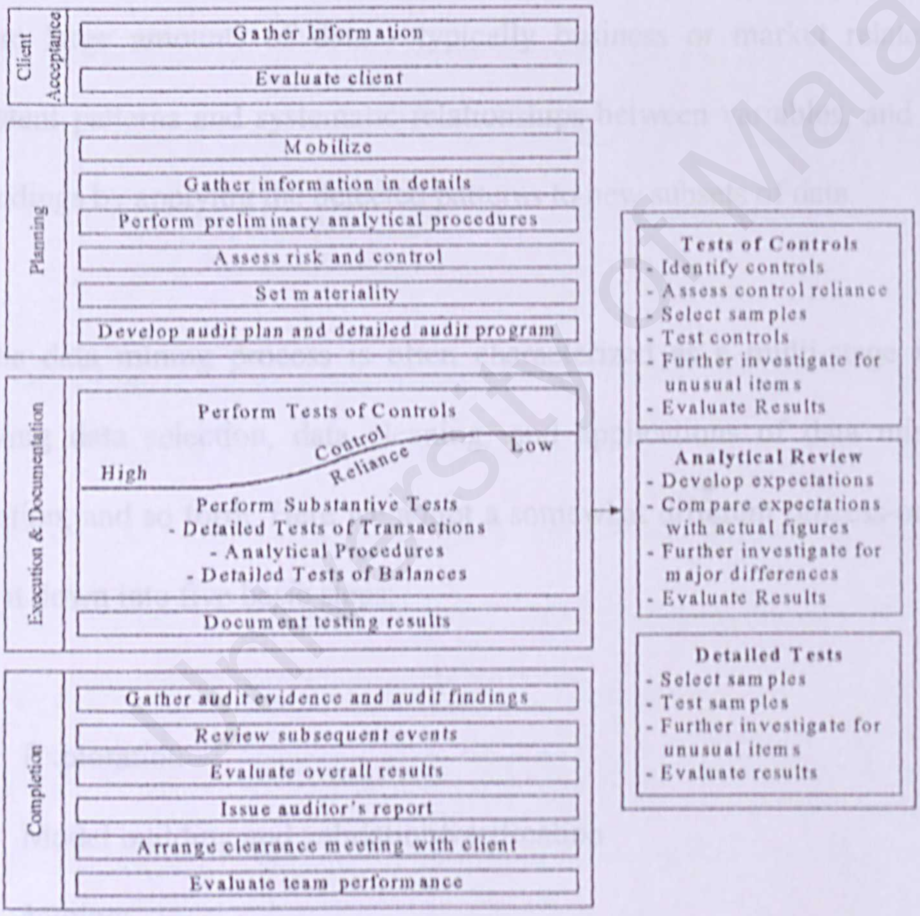

Client Acceptance
Gather Information
Evaluate client
Planning
Mobilize
Gather information in details
Perform preliminary analytical procedures
Assess risk and control
Set materiality
Develop audit plan and detailed audit program
Execution & Documentation
Perform Tests of Controls
High
Control Reliance
Low
Perform Substantive Tests
- Detailed Tests of Transactions
- Analytical Procedures
- Detailed Tests of Balances
Document testing results
Completion
Gather audit evidence and audit findings
Review subsequent events
Evaluate overall results
Issue auditor's report
Arrange clearance meeting with client
Evaluate team performance
Tests of Controls
- Identify controls
- Assess control reliance
- Select samples
- Test controls
- Further investigate for unusual items
- Evaluate Results
Analytical Review
- Develop expectations
- Compare expectations with actual figures
- Further investigate for major differences
- Evaluate Results
Detailed Tests
- Select samples
- Test samples
- Further investigate for unusual items
- Evaluate results

Figure 2.1: Summary of audit procedures

2.3 An Introduction to Data Mining

2.3.1 Overview

Data mining is the process of exploration and analysis, by automatic or semi-automatic means, of large quantities of data in order to discover meaningful patterns and rules. (Michael & Gordon, 2000).

Data Mining can also be defined as an analytic process, which is designed to explore large amounts of data - typically business or market related, in search of consistent patterns and systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data.

The data mining process is often characterized as a multi-stage iterative process involving data selection, data cleaning, and applications of data mining algorithms, evaluation, and so forth. Here we adopt a somewhat different process-oriented view and break it down into five basic steps:

- ❖ Exploration
- ❖ Model building and validation/verification
- ❖ Mining
- ❖ Evaluating
- ❖ Deployment

The Initial Exploration

This stage usually starts with data preparation which may involve cleaning data, data transformations, selecting subsets of records and - in case of data sets with large numbers of variables ("fields") - performing some preliminary feature selection operations to bring the number of variables to a manageable range (depending on the statistical methods which are being considered). Then, depending on the nature of the analytic problem, this first stage of the process of data mining may involve anywhere between a simple choice of straightforward predictors for a regression model, to elaborate exploratory analyses using a wide variety of graphical and statistical methods in order to identify the most relevant variables and determine the complexity and/or the general nature of models that can be taken into account in the next stage.

Model Building or Pattern Identification with Validation/Verification

This stage involves considering various models and choosing the best one based on their predictive performance (i.e., explaining the variability in question and producing stable results across samples). This may sound like a simple operation, but in fact, it sometimes involves a very elaborate process. There are a variety of techniques developed to achieve that goal - many of which are based on so-called "competitive evaluation of models," that is, applying different models to the same data set and then comparing their performance to choose the best. These techniques - which are often considered the core of predictive data mining- include: Bagging (Voting, Averaging), Boosting, Stacking (Stacked Generalizations), and Meta-Learning.

Mining

The step (often repeated) of actually running a particular data mining algorithm on a particular data set.

Evaluating

The step (often ignored) of critically evaluating the quality of the output of the data mining algorithm from step 3, both the predictions of the model and the interpretation of the fitted model itself.

Deployment (i.e., the application of the model to new data in order to generate predictions).

That final stage involves using the model selected as best in the previous stage and applying it to new data in order to generate predictions or estimates of the expected outcome.

2.3.2 The cycle of Data Mining

The virtuous cycle of data mining consists of four major business processes in which the success in data mining requires all four, which are:

- Business problem Identification
- Data transformation
- Act to result
- Measurement

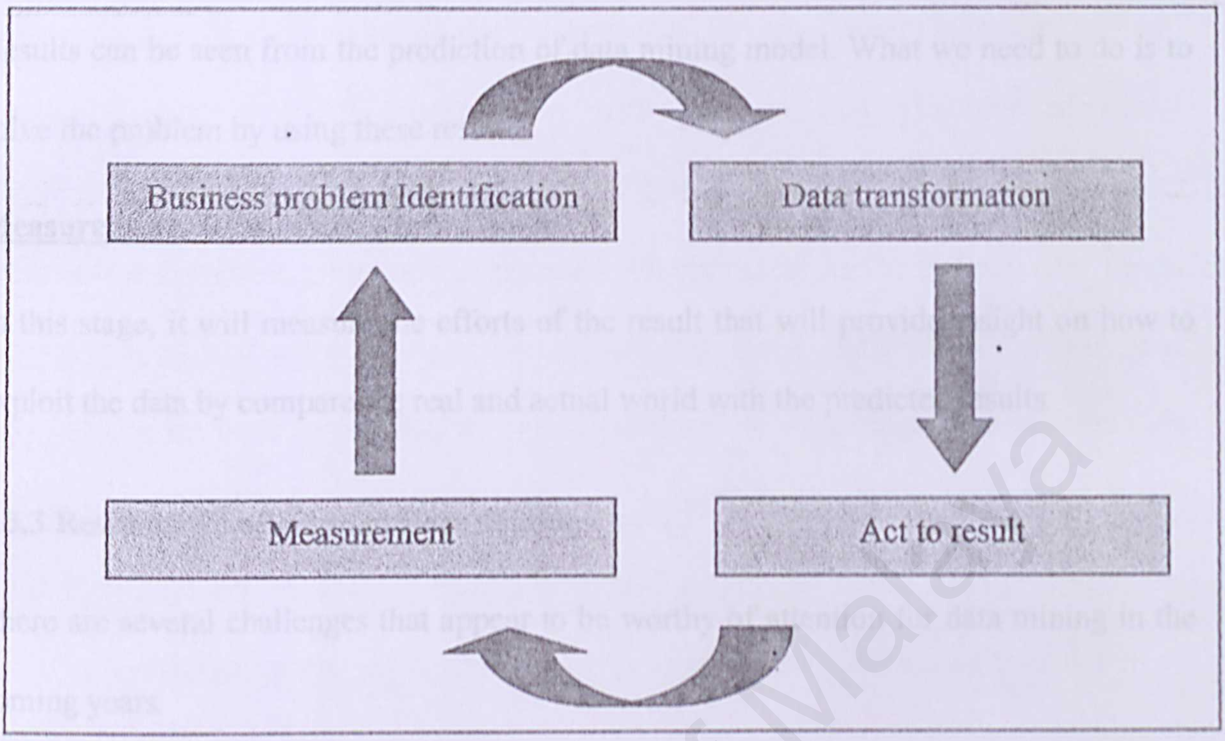


Figure 2.2: The cycle of the data mining leads to a learning organization

Business Problem Identification

In this stage, communication skill is needed to help those people in understanding and identify the problems that occur in business. They have to make sure whether the data mining effort really necessary, what is the relevant business rules, what is the data source, and so on.

Data Transformation

This is the most important stage for data mining where it will transform the data into actionable results or information. The steps that will be involved in this stage are identify and obtain data, validate and clean the data, recorder the data to the right level, add derived variables, prepare the model set, choose the modeling technique and train the model, and check performance of the models.

Acting on the results

Results can be seen from the prediction of data mining model. What we need to do is to solve the problem by using these results.

Measurement

In this stage, it will measure the efforts of the result that will provide insight on how to exploit the data by compare the real and actual world with the predicted results.

2.3.3 Research Challenges in Data Mining

There are several challenges that appear to be worthy of attention for data mining in the coming years.

A grand vision for data mining is the development of general-purpose data mining software environments that assist the user in the overall process of data mining. The software would ideally help the data miner to navigate through the space of possible exploratory steps, modeling steps, algorithm choices, evaluation metrics, and deployment options. The current state of affair is that for many applications the branching factors in terms of selecting specific method is so high that most novice users are bewildered by the space of possible choices that they can make in the data mining process. The conventional solution to date (e.g. in commercial data mining packages) is typically to support a few standard methods and algorithms at each step. Clearly this can severely constrain how we model our data and in the extreme may be entirely inappropriate for the scientific data miner where time and space are often important enough that they must be explicitly accounted for in any model.

Development of such a software environment is clearly a quite challenging problem. Statistician has been thinking of such approaches for quite some time, i.e. general-purpose environments for programming with data as well as graphical model environments that provide flexible and general-purpose high-level languages for model construction. However, these tools are primarily intended for use by statisticians. To get business person and science person domain experts to use statistical algorithms on a routine basis we need to develop a "next-generation" of interactive user-centered data exploration tools. If we don't, the current situation will continue where only a very small set of algorithms and models are widely used, and the broader spectrum of modeling and algorithmic techniques are accessible to only a small subset of data miners skilled in these techniques.

2.4 Data Mining Techniques

There are several data mining techniques, which are used mainly to solve specific problems or objectives. These techniques mentioned refer to associations, classifications, sequential patterns and clustering. However, in this segment, we only describe three techniques, which are related to our project:- classification, neural network that is under classification and sequential pattern.

2.4.1 Classification

Classification is the process of finding models, also known as classifiers, or functions that map records into one of several discrete prescribed classes. It is mostly used for predictive purpose. Additionally, classification is the most commonly task commonly used in data mining.

Typically, the model construction begins with two types of data sets, training and testing. The training data sets, with prescribed class labels, are fed into the model so that the model is able to find parameters or characters that distinguish one class from another. This step is called learning process. Then, the testing data sets, without preclassified labels, are fed into the model. The model will, ideally, automated assign the precise class labels for those testing items. In case that the results of test are poor, more training iterations are required. On the other hand, if the results are satisfactory, the model can be used to predict the classes of target items whose class labels are unknown. The following Table 2.1 shows an example of a training data set.

Outlook	Temp (F)	Humidity (%)	Windy?	Class
sunny	75	70	true	Play
sunny	80	90	true	Don't Play
sunny	85	85	false	Don't Play
sunny	72	95	false	Don't Play
sunny	69	70	false	Play
overcast	72	90	true	Play
overcast	83	78	false	Play
overcast	64	65	true	Play
overcast	81	75	false	Play
rain	71	80	true	Don't Play

Table 2.1: A training data set

This method is most effective when the underlying reasons of labeling are subtle. The advantage of this method is that the reclassified labels can be used as the performance measurement of the model. It gives the confidence to the model developer of how well the model performs. An example for classification would be target marketing. Any company, which intends to carry out promotional mailings and using a profile generator, a classification or profile is developed characterizing the people who had responded to the previous mailing. This profile is then taken as a predictor of response to the current mailing. The mailing list filtered such that the promotional materials are tagged towards those who match the profile. Besides target marketing, profile generator is used for attached mailings and treatment- appropriate determination.

There are four steps in the classification task: -

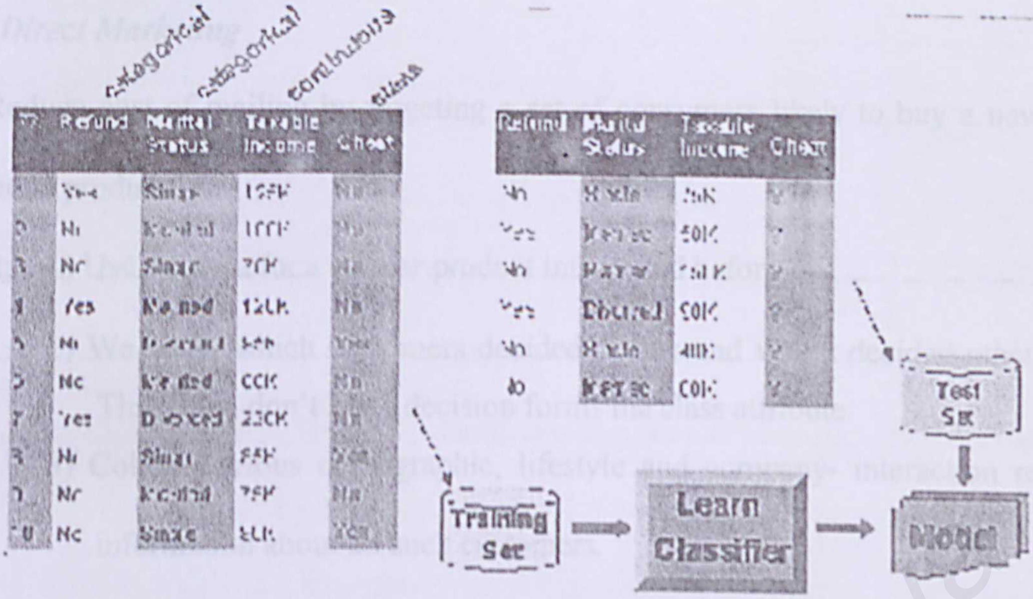
i) Collection of the relevant set of data and portioning of the data into training and testing data.

ii) Analysis of the relevance of the dimensions involved.

iii) Construction of the classification tree

iv) Testing the effectiveness of the classification using the test data set.

Appropriate techniques for classification include neural network (which will be explain later in section 2.4.1.1), relevance analysis, rule induction, decision tree (will be explain in section 2.4.1.2), case-based reasoning, genetic algorithms, linear and non-linear regression and Bayesian classification. The following Figure 2.3 is an example of classification.



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Refund	Marital Status	Taxable Income	Cheat
No	Single	75K	?
Yes	Married	50K	?
No	Married	150K	?
Yes	Divorced	90K	?
No	Single	40K	?
No	Married	80K	?

Figure 2.3: Example of Classification

Classification: Application

- *Fraud Detection*

Goal: Predict fraudulent cases in credit card transactions

Approach: 1) Use credit card transactions and the information on its account-holder as attributes.

-When does a customer buy, what does he buy, how often he pays on time, etc.

2) Label past transactions as fraud or fair transactions. This forms the class attribute.

3) Learn a model for the class of the transactions.

4) Use this model to detect fraud by observing credit card transactions on an account.

❖ *Direct Marketing*

Goal: Reduce cost of mailing by targeting a set of consumers likely to buy a new cell phone product.

Approach: 1) Use the data for a similar product introduced before

2) We know which customers decided to buy and which decided otherwise. This {buy, don't buy} decision forms the class attribute.

3) Collect various demographic, lifestyle and company- interaction related information about all such customers.

4) Use this information as input attributes to earn a classifier model.

❖ *Customer Attrition/ Churn*

Goal: To predict whether a customer is likely to be lost to a competitor.

Approach: 1) Use detailed record of transactions with each of the past and present customers, to find attributes

- How often customers calls, where he calls, what time-of the day he calls most, his financial status, etc.

2) Label the customers as loyal or disloyal.

3) Find a model for loyalty.

2.4.1.1 Neural networks

These are collections of connected nodes with inputs, outputs and processing at each node. A number of hidden processing layers exist between the visible input and output layers. The neural model has to train the net on a training dataset and then use it to make predictions. Neural nets typically cannot be trained on very large databases, but, with suitable sampling methods, the net can produce reasonable accuracy on small and medium sized data sets. The problem with neural networks is that no explanation of the results is provided (black box operation). This inhibits confidence, acceptance and application of results. However, there are some proprietary neural net products, which can translate the neural model into a set of understandable rules. This application is more often applied to pattern recognition, especially in handwriting and the interpretation of electrocardiograms.

The most well known neural network-learning algorithm is Back propagation. Like other learning algorithms, it consists of layers connected by adaptive weights. Typically, back propagation algorithms need at least 3 layers; input layer, hidden calculation layer(s) and output layer. The difference of back propagation algorithm is that it works backward which simply means it predicts the weighted algorithms by propagating the input from the output. The structure of neural network is shown in figure 2.4.

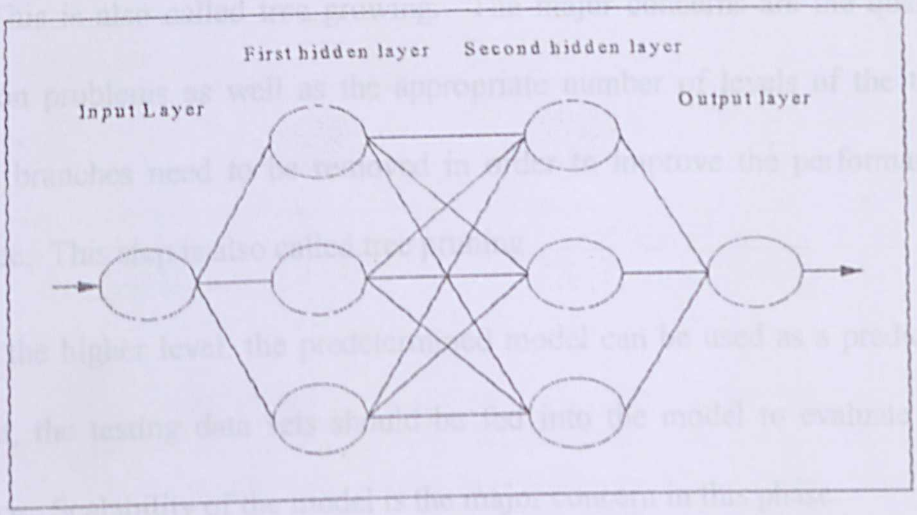


Figure 2.4: A neural network with 2 hidden layers

Neural networks are widely recognized due to its robustness. There is, however, criticism, which is its lack of self-explanation capability. Though the performance of the model is satisfactory, some feel uncomfortable and unconfident to rely irrationally on the model.

2.4.1.2 Decision trees

Decision tree is a predictive model with tree or hierarchical structure. It is used most in classification and prediction methods. It consists of nodes, which contained classification questions, and branch, or the results of the questions. At the lowest level of the tree, leave nodes, the label of each classification is identified. The structure of decision tree is illustrated in figure 2.5.

Typically, like other classification and prediction techniques, the decision tree begins with exploratory phase. It requires training data sets with labels to be fed. The underlying algorithm will try to find the best-fit criteria to distinguish one class from another. This is also called tree growing. The major concerns are the quality of the classification problems as well as the appropriate number of levels of the tree. Some leaves and branches need to be removed in order to improve the performance of the decision tree. This step is also called tree pruning.

On the higher level, the predetermined model can be used as a prediction tools. Before that, the testing data sets should be fed into the model to evaluate the model performance. Scalability of the model is the major concern in this phase.

The fundamental algorithms can be different in each model. Probably the most popular ones are Classification and Regression Trees (CART) and Chi-Square Automatic Interaction Detector (CHAID). For the sake of simplicity, I will not go into details of these algorithms; only perspectives of them are provided.

CART is developed by Leo Breiman, Jerome Friedman, Richard Olshen and Charles Stone. The advantage of CART is that it automates the pruning process by cross validation and other optimizers. It is capable of handing missing data; it sets the unqualified records apart from the training data sets.

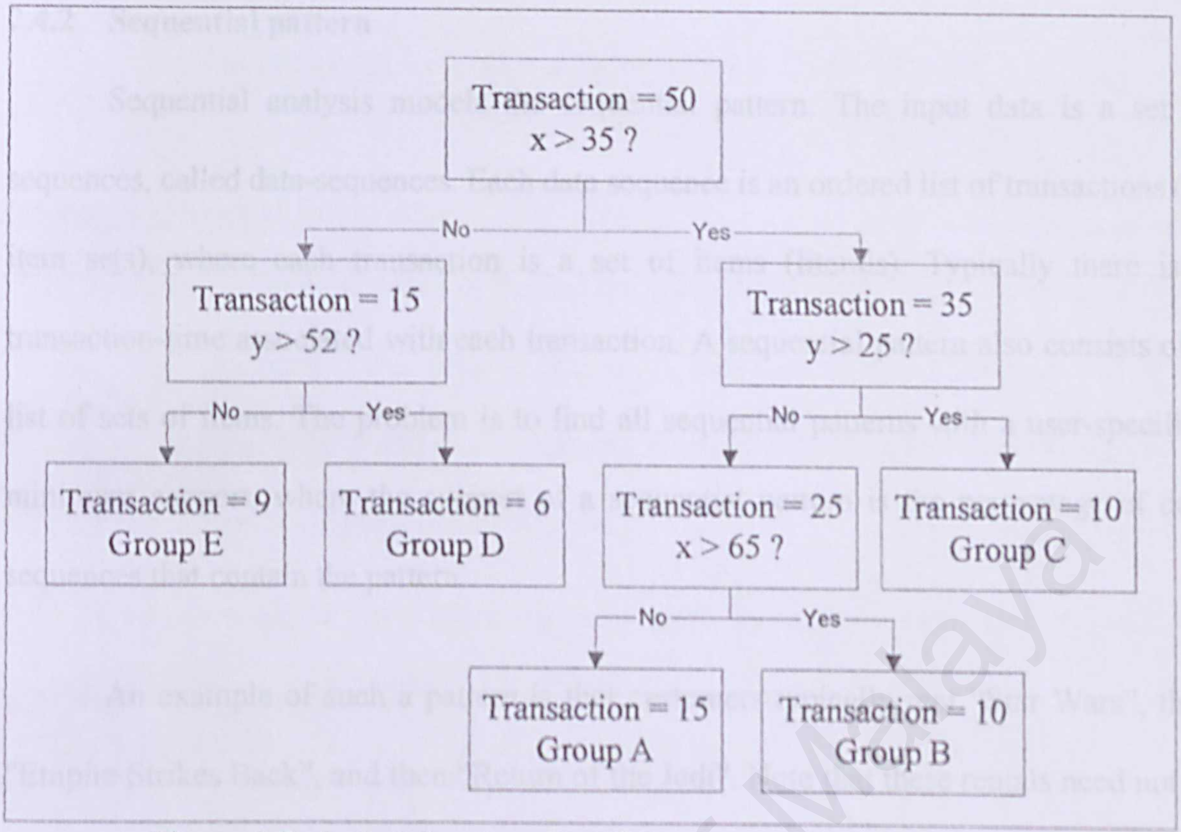


Figure 2.5: A decision tree classifying transactions into 5 groups

CHAID is another decision tree algorithm that uses contingency tables and the chi-square test to create the tree. The inferiority of CHAID comparing to CART is that it requires more data preparation process.

This application segregates the data based on values of the variables. This methodology uses a hierarchy of if-then statements to classify data. The major advantage in this application is that it is faster and more understandable than neural networks. However, the major drawback is that data type has to be interval or categorical. Continues data will then have to be recoded into these two data types, thus bringing out the possibility of concealing significant breakpoints in the data. The if-then statements could also be complex, especially if the condition list is long.

2.4.2 Sequential pattern

Sequential analysis models the sequential pattern. The input data is a set of sequences, called data-sequences. Each data sequence is an ordered list of transactions (or item sets), where each transaction is a set of items (literals). Typically there is a transaction-time associated with each transaction. A sequential pattern also consists of a list of sets of items. The problem is to find all sequential patterns with a user-specified minimum support, where the support of a sequential pattern is the percentage of data sequences that contain the pattern.

An example of such a pattern is that customers typically rent "Star Wars", then "Empire Strikes Back", and then "Return of the Jedi". Note that these rentals need not be consecutive. Customers who rent some other videos in between also support this sequential pattern. Elements of a sequential pattern need not be simple items. "Fitted Sheet and flat sheet and pillow cases", followed by "comforter", followed by "drapes and ruffles" is an example of a sequential pattern in which the elements are sets of items. This problem was initially motivated by applications in the retailing industry, including attached mailing, add-on sales, and customer satisfaction. But the results apply to many scientific and business domains. For instance, in the medical domain, a data-sequence may correspond to the symptoms or diseases of a patient, with a transaction corresponding to the symptoms exhibited or diseases diagnosed during a visit to the doctor. The patterns discovered using this data could be used in disease research to help identify symptoms/diseases that precede certain diseases.

2.5 Integration between Auditing and Data Mining

2.5.1 Possible Area of Integration

Recently, people in auditing field have started realizing that technologies can influence their practices. Data mining is one of those technologies, which they are referring to, especially in credit card approval, fraud detection, and intrusion decision field as it has proven to be very effective.

From our point of view, there are many audit steps that data mining techniques are likely to be capable of assisting or handling. Those steps as well as the presumptuous appropriate mining patterns required are summarized in table 2.2.

Audit Steps	Appropriate Mining Patterns That Are Involved
Client Acceptance or Client Continuance	Classification and prediction
Planning	Dependency analysis, classification and prediction
Execution and Documentation	Dependency analysis, classification, prediction, data description, outlier analysis, cluster analysis and evolution analysis
Completion	Classification and prediction
Other Possibilities • Fraud detection	•Outlier analysis, dependency analysis, classification

Neural networks are called machine-learning algorithms because changing these connections (training) causes the network to learn the solution to a problem. This differs from other artificial intelligence technologies, such as expert systems, fuzzy logic or constraint-based reasoning which must be programmed to solve a problem.

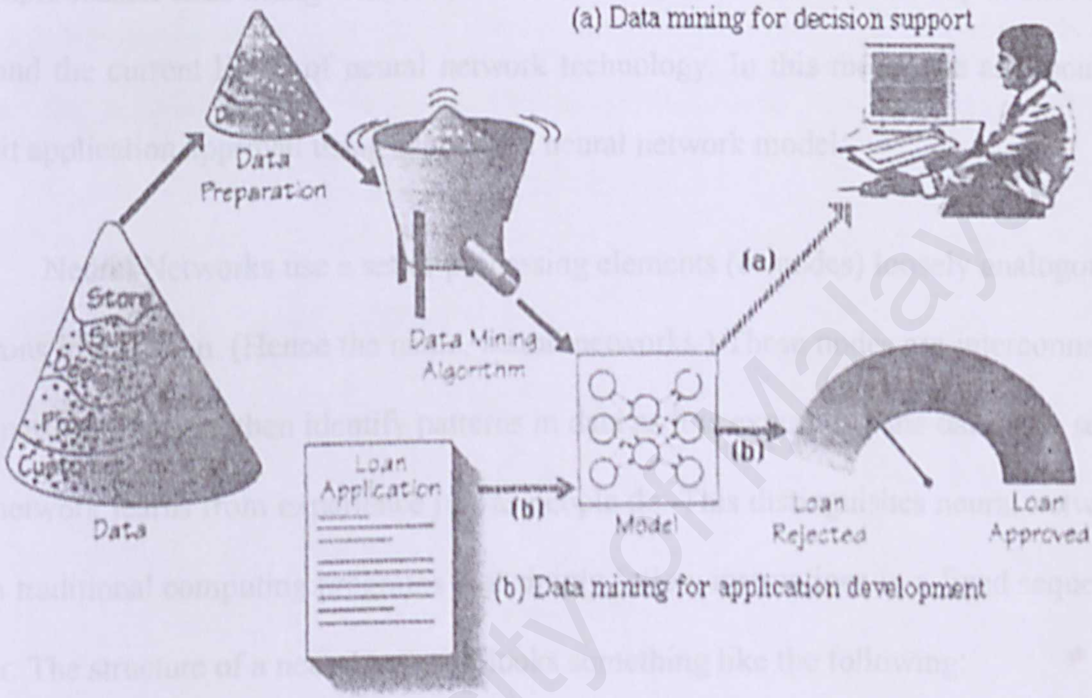


Figure 2.6: Steps of Machine Learning

Many different neural network models have been explored. These models are described as either unsupervised or supervised. Unsupervised neural networks, such as self-organizing feature maps, find relationships between input examples by examining the similarities and differences between the examples. Supervised neural networks, such as back propagation, are used for pattern recognition or prediction. For supervised neural networks, the input examples must be accompanied by the desired output.

Neural networks are well suited to a number of situations requiring approval decisions, whether for loans, leases, or credit cards. Networks can be trained to simply classify an application as acceptable (a yes or no decision) or to predict a value such as the revenue that will be generated. Networks with multiple outputs can be used to provide a simple reason code along with a credit evaluation, but more detailed explanations are beyond the current limits of neural network technology. In this thesis, we are focus on credit application approval using supervised neural network model.

Neural Networks use a set of processing elements (or nodes) loosely analogous to neurons in the brain. (Hence the name, neural networks.) These nodes are interconnected in a network that can then identify patterns in data as it is exposed to the data. In a sense, the network learns from experience just as people do. This distinguishes neural networks from traditional computing programs that simply follow instructions in a fixed sequential order. The structure of a neural network looks something like the following:

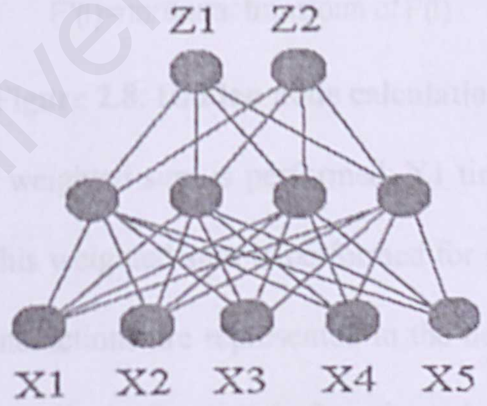


Figure 2.7: Neural Network Structure

The bottom layer represents the input layer, in this case with 5 inputs labeled X1 through X5. In the middle is something called the hidden layer, with a variable number of nodes. It is the hidden layer that performs much of the work of the network. The output layer in this case has two nodes, Z1 and Z2 representing output values we are trying to determine from the inputs. For example, we may be trying to predict sales (output) based on past sales, price and season (input).

Each node in the hidden layer is fully connected to the inputs. That means what is learned in a hidden node is based on all the inputs taken together. This hidden layer is where the network learns interdependencies in the model. The following diagram provides some detail into what goes on inside a hidden node.

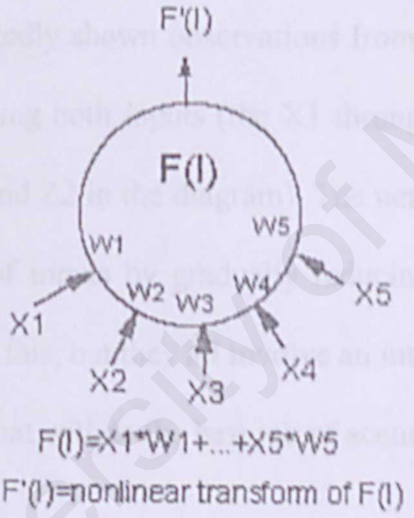


Figure 2.8: Hidden node calculation

Simply speaking a weighted sum is performed: X1 times W1 plus X2 times W2 on through X5 and W5. This weighted sum is performed for each hidden node and each output node and is how interactions are represented in the network. Each summation is then transformed using a nonlinear function before the value is passed on to the next layer.

The major elements in the processing of the network are:

<u>**Inputs:**</u> An input corresponds to a single attribute, eg. Age, income or ownership or land in value terms. Zero/one dummy variables can be used for qualitative variable.

<u>**Outputs:**</u> The solution to the problem. For a loan application, this may be "yes" or "no" where the NN assigns say +1 for "yes" and 0 for "no"

<u>**Weights:**</u> The relative importance of each input to a processing element. The networks "learn" through the repeated adjustment of the weights.

The network is repeatedly shown observations from available data related to the problem to be solved, including both inputs (the X1 through X5 in the diagram above) and the desired outputs (Z1 and Z2 in the diagram). The network then tries to predict the correct output for each set of inputs by gradually reducing the error. There are many algorithms for accomplishing this, but they all involve an interactive search for the proper set of weights (the W1-W5) that will do the best job of accurately predicting the outputs.

How do Neural Network compare to other artificial Intelligence Techniques?

Neural network typically cannot be trained on very large databases, but with suitable sampling methods, the network can produce reasonable accuracy on small and medium sized data sets. Some problems, such as loan approval decisions, can be approached as either expert system or neural network applications. In contrast to expert systems, neural networks can be developed where expertise is limited or the experts are unable to explain their reasoning process. Because of their speed, neural networks may be appropriate where rule-based processing is too slow. Developing a neural network, however, requires far more data than building an expert system. In general, the more complex the data and the more accurate the required response, the more data is required. Typical applications require hundreds to thousands of training examples. Neural networks alone are inappropriate in situations requiring detailed explanations of the output; situations where expert systems excel. The technologies do not compete so much as they complement one another. Each technology mimics a different type of human problem-solving behavior. Expert systems embody conscious, methodical reasoning while neural networks represent instantaneous, unconscious pattern recognition.

Neural Network in Data Mining

Data mining (also referred to as Knowledge Discovery in Databases or KDD) can be defined as "the nontrivial extraction of implicit, previously unknown, and potentially useful information of data". The goal of data mining, as illustrated in the picture below, is to discover knowledge out of data and present it in a form that is easily comprehensible to humans. The data mining toolbox includes techniques such as machine learning, statistics, neural networks and visualization methods. Applications of data mining are, among others, in retail/marketing, banking, insurance, transportation and medicine.

Figure 2.9: **Discover knowledge out of data**

Neural networks form just part of the data miner's toolbox. They are well suited to identify trends and patterns in the data and are therefore mainly used for prediction and forecasting needs. Examples are sales forecasting, industrial process control, customer research, data validation, risk management, and target marketing.

Many standard software packages for data mining contain neural network modules. However, these modules are extremely basic: most of the time just a simple multi-layered perception, trainable with inefficient and old-fashioned updating techniques such as standard back-propagation. They often fail to fulfill the important requirement of providing insight in the database. In fact, one could even argue whether these standard neural networks are truly methods for data mining as defined above, or at most classification, predictions and perhaps clustering tools.

Organizations in the late 1990's typically have large stores of data available due to advances in information technology and reduced costs for data storage. The question is how to best utilize data that has magnified in terms of volume and complexity? Companies that can more quickly and efficiently uncover useful information to help run their businesses have a distinct competitive advantage.

Neural networks are well suited for data mining tasks due to their ability to model complex, multi-dimensional data. As data availability has magnified, so has the dimensionality of problems to be solved, thus limiting many traditional techniques such as manual examination of the data and some statistical methods. Although there are many techniques and algorithms that can use for data mining, some of which can be used effectively in combination, neural networks offer the following desirable qualities:

- Automatic search of all possible interrelationships among key factors.

- Automatic modeling of complex problems without prior knowledge of the level of complexity.
- Ability to extract key findings much faster than many other tools.

Credit application approval using Neural Network

Neural networks are making big inroads into the financial worlds. Banking, credit card companies, and lending institutions deal with decisions that are not clear cut. They involve learning and statistical trends.

The loan approval process involves filling out forms, which hopefully can enable a loan officer to make a decision. The data from these forms is now being used by neural networks, which have been trained on the data from past decisions. Indeed, to meet government requirements as to why applications are being denied, these packages are providing information on what input, or combination of inputs, weighed heaviest on the decision. Credit card companies are also using similar back-propagation networks to aid in establishing credit risks and credit limits.

Predicting Credit Risk using Neural Network

A financial institution seeks to minimize loan defaults. Officers must be able to identify potential credit risks during the loan approval cycle. The problem is one of simple classification: to predict whether or not an applicant will be a good or poor credit risk. This dataset contains information about people to whom the institution previously loaned money. Risk is the dependent column (target variable). The other columns used to build the model are known as the independent columns.

Name	Debt	Income	Married?	Risk
Joe	High	High	Yes	Good
Sue	Low	High	Yes	Good
John	Low	High	No	Poor
Mary	High	Low	Yes	Poor
Fred	Low	Low	Yes	Poor

Table 2.3: Credit risk training dataset. Debt, income, and marital status are the independent variables. Credit risk is the dependent variable or the outcome.

Name	Debt	Income	Married?	Risk
Joe	1	1	1	1
Sue	0	1	1	1
John	0	1	0	0
Mary	1	0	1	0
Fred	0	0	1	0

Table 2.4: Credit risk data with column values converted to numeric values

•High debt and income, married= yes, and good risk were all replaced by the value 1.

•Low debt and income, married=no, and poor risk were replaced by 0.

•The neural net that we are going to use is shown in Figure 2.10.

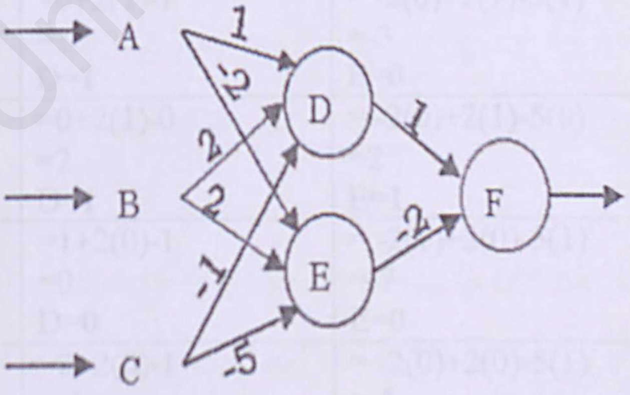


Figure 2.10: The Structure of Neural Network

This neural network contains six nodes labeled A through F. The yellow nodes (A, B and C) are input nodes, which correspond to the independent variable columns in the credit risk problem (Debt, Income, and Married). The red node (F) is the output node, which corresponds to Risk, the dependent column. The numbers on the arrows are weights. Each node has a squashing function that converts the weighted sum of the inputs to an output value. For our neural net we will use a very simple squashing function: if the weighted sum of the inputs is greater than zero, the output is 1, otherwise the output is 0.

- •D = If (A + 2B - C) > 0 Then 1 Else 0
- •E = If (-2A + 2B -5C) > 0 Then 1 Else 0
- •F = If (D- 2E) > 0 Then 1 Else 0

Name	D= A+2B-C	E= -2A+2B-5C	F= D-2E
Joe	=1+2(1)-1 =2 D=1	= -2(1)+2(1)-5(1) =-5 E=0	= 1-2(0) =1 F=1
Sue	=0+2(1)-1 =1 D=1	= -2(0)+2(1)-5(1) =-3 E=0	= 1-2(0) =1 F=1
John	=0+2(1)-0 =2 D=1	= -2(0)+2(1)-5(0) =2 E=1	= 1-2(1) =-1 F=0
Mary	=1+2(0)-1 =0 D=0	= -2(1)+2(0)-5(1) =-7 E=0	= 0-2(0) =0 F=0
Fred	=0+2(0)-1 =-1 D=0	= -2(0)+2(0)-5(1) =-5 E=0	= 0-2(0) =0 F=0

Table 2.5: Sample of weighted sum calculation

Node:	A	B	C		D	E	F
Name	Debt	Income	Married	Risk			
Joe	1	1	1	1	1	0	1
Sue	0	1	1	1	1	0	1
John	0	1	0	0	1	1	0
Mary	1	0	1	0	0	0	0
Fred	0	0	1	0	0	0	0

Table 2.6: Credit risk data sample, with the three independent variables, converted to numbers, the actual risk, and the computed values for nodes D, E, and F.

Table 2.6 shows the sample data, with the three independent variables, converted to numbers, the actual risk, and the computed values for nodes D, E, and F. The output value for F (1 on the first row) is the predicted value of risk for Joe. It equals the actual value, so the neural net made a correct prediction in this case. In fact, the net in Figure 2.10 makes correct predictions for all five rows in the training set, as shown in table 2.6.

Advantages and disadvantages of Neural Network

One of the major advantages of neural networks is that, theoretically, they are capable of approximating any continuous function, and thus the researcher does not need to have any hypotheses about the underlying model, or even to some extent, which variables matter. Besides that, there is no model, no logical or mathematical rules involved.

While an important disadvantages, however, is that the final solution depends on the initial conditions of the network, and, as stated before, it is virtually impossible to "interpret" the solution in traditional, analytic terms, such as those used to build theories that explain phenomena. Besides, when a network makes a mistake it is often against all expectations. Example, the mistaken case seems to be inside the generality of the training examples. When the network is redesigned or retained including the mistaken case example, then new surprising mistakes may appear. In other words there is no such thing as a perfect neural network (unless all possible inputs are trained and the network is sufficiently large).

Conclusion

Neural networks represent a proven, widely used technology for complex prediction and classification problems. Neural networks excel in problem domains where there are non-linear relationships, limited expertise, missing or incomplete data and fast processing is required. Many neural network applications have been successfully applied in a number of domains, from retail to insurance.

2.7 Fraud Detection Using Classification

Fraud Detection (Credit Card Transactions)

In this segment, we are going to focus on how data mining works in fraud detection, which is related to auditing.

Frauds have plagued telecommunication industries, financial institutions and other organizations for a long time. The types of frauds that can be found include cellular communication frauds, credit card transaction frauds, and computer intrusions. These frauds cost the businesses millions of dollars per year. As a result, fraud detection has become an important and urgent task for these businesses. At present a number of methods have been implemented to detect frauds, from both statistical approaches (e.g. data mining) and hardware approaches (e.g. firewalls, smart cards).

At present, data mining is a popular way to combat frauds because of its effectiveness. Hand et al define that data mining is "a well-defined procedure that takes data as input and produces output in the forms of models or patterns." In other words, the task of data mining is to analyze a massive amount of data and to extract some usable information that can be used in future. In doing so, it is the best solution to define the clear goal of data mining, and find out the right structure of possible model or patterns that fit to the given data set. Once the right model for the data has been chosen, the model can be used for predicting future events by classifying the data.

In terms of data mining, fraud detection can be understood as the classification of the data. Input data is analyzed with the appropriate model and determined whether it implies any fraudulent activities or not. Recognizing the patterns of former fraudulent behaviors develops a well-defined classification model. As a result, the model can be used to predict any suspicious activities implied by the new data set.

As informed, statistical approaches have been implemented to detect frauds. Statistical fraud detection methods may be 'supervised' or 'unsupervised'. In supervised (classification) methods, models are trained to discriminate between fraudulent and non-fraudulent behavior, so that new observations can be assigned to classes so as to optimize some measure of classification performance. Of course, this requires one to be confident about the true classes of the original data used to build the models; uncertainty is introduced when legitimate transactions are mistakenly reported as fraud or when fraudulent observations are not identified as such.

Supervised methods require that examples of both classes are provided, and they can only be used to detect frauds of a type that have previously occurred. Basically, supervised methods are only trained to discriminate between legitimate transactions and previously known fraud.

Nevertheless, these methods also suffer from the problem of unbalanced class sizes: in fraud detection problems, the legitimate transactions generally far outnumber the fraudulent ones and this imbalance can cause misspecification of models. Braise et al (1999) say that, in their database of credit card transactions, '*the probability of fraud is very low (0.2%) and has been lowered in a preprocessing step by a conventional fraud detecting system down to 0.1%.*' *Hassibi (2000)* remarks '*Out of some 12 billion transactions made annually, approximately 10 million – or one out of every 1200 transactions – turn out to be fraudulent.*' Also, one limitation of using data mining alone in fraud detection is its efficiency problem. Data mining and model construction require a lot of time, which prohibits it to detect frauds in real time. This is a serious drawback since, in many occasions such as online credit card transactions, we need to detect fraudulent activities in a very short period of time. Otherwise, the loss could be huge.

Fraud detection is a non-trivial task in this information explosion age. It is faced with three major challenges. First, fraud detection usually involves a large amount of data. Detecting frauds in such high volume of data is worse than finding a needle in a haystack. It is easy to differentiate a needle from hay but it is hard to tell fraudulent activities from legitimate ones since they look similar.

Second, fraud detection needs to be highly accurate. Although the sum of fraudulent activities is very high, the fraud rate is relatively low compared to the gigantic volume of legitimate operations. For credit card transactions in general, the fraud rate is 0.93%. And it is 1.97% for online credit card transactions Thus; a good fraud detection mechanism should be good at catching frauds and reducing false alarms as well.

Third, frauds need to be detected fast. A criminal can commit many frauds with high dollar amounts in a short period time. Moreover, legitimate users and customers will lose their patience if they wait too long for fraud check in an operation or transaction. Thus, we need to detect frauds in a very short period of time. Otherwise, the damage costs will be high and the business will lose customers

As we have mentioned above, there are many types of frauds such as frauds in mobile communications but in this segment, we are going to emphasize more on frauds on credit cards transactions.

Credit card fraud is perpetrated in various ways but can be broadly categorized as application, 'missing in post', stolen/lost card, counterfeit card and 'cardholder not present' fraud. Application fraud arises when individuals obtain new credit cards from issuing companies using false personal information; application fraud totaled £10.2 million in 2000 (Source: APACS) and is the only type of fraud that actually declined between 1999 and 2000. 'Missing in post' (£17.3m in 2000) describes the interception of credit cards in the post by fraudsters before they reach the cardholder. Stolen or lost cards accounted for £98.9 million in fraud in 2000, but the greatest percentage increases between 1999 and 2000 were in counterfeit card fraud (£50.3m to £102.8m) and 'cardholder not present' (i.e. postal, phone, internet transactions) fraud (£29.3m to £56.8m). To commit these last two types of fraud, it is necessary to obtain the details of the card without the cardholder's knowledge. This is done in various ways, including employees using an unauthorized 'swiper' that downloads the encoded information onto a laptop computer and hackers obtaining credit card details by intrusion into companies' computer networks. A counterfeit card is then made, or the card details simply used for phone, postal or Internet transactions.

Supervised methods to detect fraudulent transactions can be used to discriminate between those accounts or transactions known to be fraudulent and those known (or at least presumed) to be legitimate. For example, traditional credit scorecards (Hand and Henley, 1997) are used to detect customers who are likely to default, and the reasons for this may include fraud. Such scorecards are based on the details given on the application forms, and perhaps also on other details, such as bureau information. Classification techniques, such as statistical discriminate analysis and neural networks, can be used to discriminate between fraudulent and non-fraudulent transactions to give transactions a suspicion score.

As informed, it is known that the emphasis on fraud detection methodology is with supervised techniques particularly using neural networks. Researchers who have used neural networks for supervised credit card fraud detection include Ghosh and Reilly (1994), Aleskerov et al. (1997), Dorronsoro et al. (1997), and Brause et al (1999).

Lastly, here we include a decision tree (figure 2.11). A decision tree indicates whether or not a fraud is detected. Each internal (nonleaf) node represents a test on an attribute. Each leaf node represents a class (either fraud detected=yes or no). Also below, is a training data set.

Age	Income	Married?	Fraud detected
<=30	High	N	Y
<=30	High	Y	Y
31..40	High	Y	Y
>40	Low	Y	N
>40	High	Y	Y
31..40	Low	N	N

Table 2.7: A training data set for fraud detection

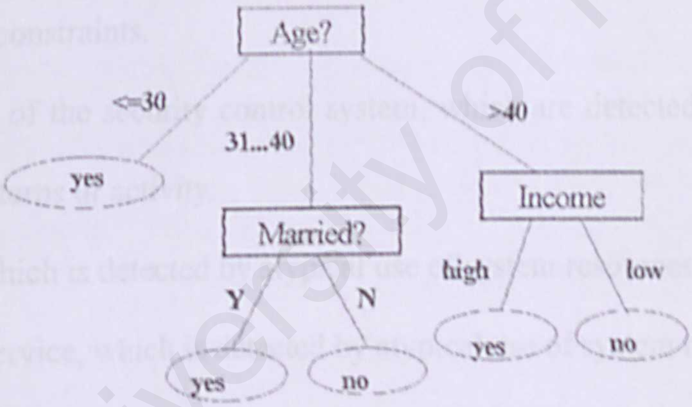


Figure 2.11: Fraud detection using decision tree.

2.8 Intrusion Detection using Sequential Pattern

Intrusion detection is one of the auditing steps that I will implement in my project in which I'll focus more on computer intrusion whereby using one of the data mining technique i.e. sequential analysis.

First of all, we need to know what define computer intrusion. Here intrusions can be divided into 6 main types:

1. Attempted break-ins, which are detected by atypical behavior profiles or violations of security constraints.
2. Masquerade attacks, which are detected by atypical behavior profiles or violations of security constraints.
3. Penetration of the security control system, which are detected by monitoring for specific patterns of activity.
4. Leakage, which is detected by atypical use of system resources.
5. Denial of service, which is detected by atypical use of system resources.
6. Malicious use, which is detected by atypical behavior profiles, violations of security constraints, or use of special privileges.

In the last three years, the networking revolution has finally come of age. More than ever before, we see that the Internet is changing computing, as we know it. The possibilities and opportunities are limitless; unfortunately, so too are the risks and chances of malicious intrusions.

It is very important that the security mechanisms of a system are designed so as to prevent unauthorized access to system resources and data. However, completely preventing breaches of security appear unrealistic. We can, however, try to detect these intrusion attempts so that action may be taken to repair the damage later. This field of research is called Intrusion Detection.

Intrusion detection is the art and science of sensing when a system or network is being used inappropriately or without authorization. An intrusion-detection system (IDS) monitors system and network resources and activities and, using information gathered from these sources, notifies the authorities when it identifies a possible intrusion.

If a firewall is like having a security guard at your office door, checking the credentials of everyone coming and going, then an intrusion-detection system (IDS) is like having a network of sensors that tells you when someone has broken in, where they are and what they're doing.

Firewalls work only at the point of entry to the network, and they work only with packets as they pass in and out of the network. Once an attacker has breached the firewall, he can roam at will through the network. That's where intrusion detection is important.

Intrusion detection technique can be divided into two type:

1. **Misuse detection systems**

Encode and match the sequence of "signature actions" (e.g., change the ownership of a file) of known intrusion scenarios. The main shortcomings of such systems are: known intrusion patterns have to be hand-coded into the system; they are unable to detect any future (unknown) intrusions that have no matched patterns stored in the system.

2. **Anomaly detection (sub) systems**

Establish normal usage patterns (profiles) using statistical measures on system features, for example, the CPU and I/O activities by a particular user or program. The main difficulties of these systems are intuition and experience is relied upon in selecting the system features, which can vary greatly among different computing environments; some intrusions can only be detected by studying the sequential interrelation between events because each event alone may fit the profiles.

The key ideas are to use data mining techniques to discover consistent and useful patterns of system features that describe program and user behavior, and use the set of relevant system features to compute (inductively learned) classifiers that can recognize anomalies and known intrusions.

How Sequential Technique is used Intrusion Detection

With the increase of internet connectivity, there is the ever increasing risk of attackers illicitly gaining access to computers over the network. Intrusion detection is often used as another wall to protect computer systems, in addition to the standard methods of security measures such as user authentication (e.g. user passwords), avoiding programming errors, and information protection (e.g. encryption).

We describe a data-mining framework for adaptively building Intrusion detection models specifically for use with Network Flight Recorder (NFR). The central idea here is to implement auditing programs to extract an extensive set of features that describe each network connection or host session, and apply data mining programs to learn rules that accurately capture the behavior of intrusions and normal activity.

Data mining generally refers to the process of (automatically) extracting models from large stores of data. The recent rapid development in data mining has made available a wide variety of algorithms, drawn from the fields of statistics, pattern recognition, machine learning, and databases. One type of algorithms is particularly useful for mining audit data is sequential analysis, which models the sequential patterns. These algorithms can discover what (time-based) sequence of audit events frequently occurs together. These frequent event patterns provide guidelines for incorporating temporal statistical measures into intrusion models. We use the frequent episodes algorithms for this analysis in which I'll discus further as we move on.

2.8.1 The need for Intrusion Detection Systems

A computer system should provide confidentiality, integrity and assurance against denial of service. However, due to increased connectivity (especially on the Internet), and the vast spectrum of financial possibilities that are opening up, more and more systems are subject to attack by intruders. There are two ways to handle subversion attempts. One way is to prevent subversion itself by building a completely secure system. We could, for example, require all users to identify and authenticate themselves; we could protect data by various cryptographic methods and very tight access control mechanisms. However this is not really feasible because:

- In practice, it is not possible to build a completely secure system. Since a compelling report on bugs in popular programs and operating systems that seems to indicate that (a) bug free software is still a dream and (b) no-one seems to want to make the effort to try to develop such software. Apart from the fact that we do not seem to be getting our money's worth when we buy software, there are also security implications when our E-mail software, for example, can be attacked. Designing and implementing a totally secure system is thus an extremely difficult task.
- The vast installed base of systems worldwide guarantees that any transition to a secure system, (if it is ever developed) will be long in coming.
- Cryptographic methods have their own problems. Passwords can be cracked, users can lose their passwords, and entire crypto-systems can be broken.

- ➢ *Even a truly secure system is vulnerable to abuse by insiders who abuse their privileges.*
- ➢ It has been seen that that the relationship between the level of access control and user efficiency is an inverse one, which means that the stricter the mechanisms, the lower the efficiency becomes.

We thus see that we are stuck with systems that have vulnerabilities for a while to come. If there are attacks on a system, we would like to detect them as soon as possible (preferably in real-time) and take appropriate action. This is essentially what an Intrusion Detection System (IDS) does. An IDS does not usually take preventive measures when an attack is detected; it is a reactive rather than pro-active agent. It plays the role of an *informant rather than a police officer.*

2.8.2 Future Work

These frequent patterns form an abstract summary of an audit trail, and therefore can be used to guide the audit data gathering process; provide help for feature selection; and discover patterns of intrusions.

- Implement a support environment for system builders to iteratively drive the integrated process of pattern discovering, system feature selection, and construction and evaluation of detection models;
- Investigate the methods and benefits of combining multiple simple detection models. We need to use multiple audit data streams for experiments;

- *Implement* a prototype agent-based intrusion detection system. Evaluate our approach using extensive audit data sets, some of which is presently under construction at Rome Labs.

2.8.3 Frequent Episodes

Sequence analysis models sequential patterns. These algorithms can help us understand what (time-based) sequence of audit events is frequently encountered together. These frequent event patterns are important elements of the behavior profile of a user or program.

While the association rules algorithm seeks to find intra- audit record patterns, the frequent episodes algorithm can be used to discover inter- audit record patterns. A frequent episode is a set of events that occur frequently within a time window (of a specified length). The events must occur (together) in at least a specified minimum frequency, min_fr, sliding time window. Events in a serial episode must occur in partial order in time; whereas for a parallel episode there is no such constraint. For *X* and *Y* where X+Y is a frequent episode, *X* --> *Y* with confidence=frequency (X+Y)/frequency (X) and support=frequency (X+Y) is called a frequent episode rule. An example frequent serial episode rule from the log file of a department's Web site is home, research --> theory; [0.2, 0.05], [30s] which indicates that when the home page and the research guide are visited (in that order), in 20% of the cases the theory group's page is visited subsequently within the same 30s time window, and this sequence of visits occurs 5% of the total (the 30s) time windows in the log file (that is, approximately 5% of all the records).

We seek to apply the frequent episodes algorithm to analyze audit trails since there is evidence that the sequence information in program executions and user commands can be used to build profiles for anomaly detection.

2.8.4 Intrusion detection Decision Tree

Decision trees are structures used to classify data with common attributes. Here, a decision tree is used to detect intrusive behavior based on the data in the given table.

In this example, the IP port, and System Name label the nodes, intrusion and normal label the leaves and the labeled arrows are the edges. The generated tree is shown in the figure below.

IP Port	system name	category
004020	Artemis	Normal
004020	Apollo	Intrusion
002210	Artemis	Normal
002210	Apollo	Intrusion
000010	Artemis	Normal
000010	Apollo	Normal

Table 2.8 : Example of Intrusion Data Set

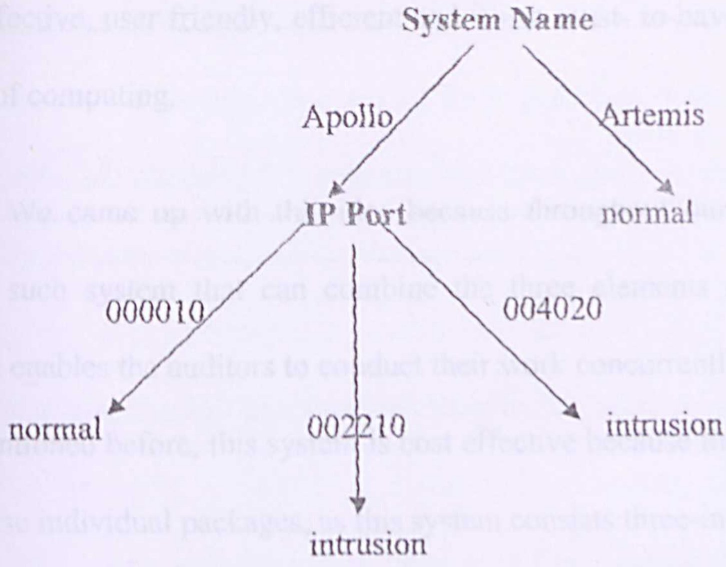


Figure 2.12 : Example Intrusion Decision Tree

2.8.5 Challenge of using Data Mining Approaches in Intrusion Detection

The biggest challenge of using data mining approaches in intrusion detection is that it requires a large amount of audit data in order to compute the profile rule sets. And the fact that we may need to compute a detection model for each resource in a target system makes the data mining task daunting. Moreover, this learning (mining) process is an integral and continuous part of an intrusion detection system because the rule sets used by the detection module may not be static over a long period of time.

2.9 FRIDCA System

FRIDCA is the combination of Fraud detection, Intrusion Detection and Credit Approval. As the name implies, this system combine the three elements, which are credit card fraud detection, credit application approval and intrusion detection. This system is cost effective, user friendly, efficient and it is a must- to-have tool in this fast- evolving world of computing.

We came up with this idea because throughout our research, we didn't come across such system that can combine the three elements together. Furthermore, this system enables the auditors to conduct their work concurrently and this help to save time. As mentioned before, this system is cost effective because the company does not need to purchase individual packages, as this system consists three-in-one.

Additionally, this is a trustworthy system since it has both fraud and intrusion detection elements, which can protect computer systems from unauthorized activity.

2.10 Comparison to the Existing System

2.10.1 Comparison to the existing neural network system

Our Neural Network model in FRIDCA System is better than the existing system because of the friendly user interface, which is simple and easy to use. For the existing system, they provide too many interfaces and auditors need to turn too many pages to get the result of the applicants, while our system is only have 3 to 4 interfaces to get the result. This means, auditors does not have to spend much work and time during auditing. Even new auditors who never use this function are able to use without any help module because all the buttons are general ones. Besides that, the existing system needs auditors to train and test the data before they can get the result, but our system is automatically trained and tested during the selecting fields where we programmed our system all in one and in the same time providing the intrusion detection. Instead of choosing the existing system, why not choosing our system which is easier to use and saving time which is also providing the same result as the existing one.

2.10.2 Existing Fraud Detection Software in the Market

Now, many company have developed fraud detection software in order to detect fraud and to prevent it. Here, we are going to tell briefly about one software, which we have found during our research that is *Fractals*.

Fractals, a product of Alaric, scan credit and debit card transaction sequences and raises alerts to fraud managers when it spots suspicious patterns. Fractals uses a rules-based approach to card fraud detection. Fractals utilize user-developed rules and system inferred strategies.

Fractals derives its own System Strategies automatically, providing the only card fraud detection system which is dynamically adaptive. Due to its efficient implementation, Fractals is able to cost effectively derive and update System Strategies on a regular basis, as frequently as weekly if necessary. This means that Fractals is highly adaptive to changing fraud patterns, much more so than neural network models, which can take months to retune, and retrains.

Fractals detects card fraud by efficiently evaluating incoming sequences of card transactions against its database of User Rules and System Strategies, flagging when a rule or strategy is a triggered and estimating the likelihood that the transactions are fraudulent

As a conclusion, after comparing existing fraud detection software in the market, we can say that our own system, which is FRIDCA, is more reliable and effective for the users to use. It also implements statistical approach that is data mining (in this case, classification method), which is the technology considered to be the most valuable tools in today's competitive, fast-evolving world. Our system is built to detect fraud in a short period of time in order to keep business going.

2.10.3 Intrusion Detection Software (IDS) in the Market

Perhaps the most famous Intrusion Detection System is Tripwire, a program written in 1992 by Eugene Spafford and Gene Kim. Tripwire exemplifies the host-based agent approach to intrusion detection: Installed on a host, it checks to see what has changed on the system, verifying that key files haven't been modified.

The agent is initially installed against a pristine host installation and records important system file attributes, including hashes of the files. The agent software then periodically compares the current state of those files to the stored attributes and reports any suspicious changes.

Another host-based approach monitors all packets as they enter and exit the host, essentially taking a personal firewall approach. Receipt of a suspicious packet triggers an alarm. Other commercial host-based products include Cupertino, Calif.-based Symantec

Corp.'s Intruder Alert and Issaquah, Wash.-based CyberSafe Corp.'s Centrax. To name one of the Open Source available on the web is Snort, which can be found in this website www.snort.org. This is a freely downloadable IDS agent featuring huge possibilities in attack detection.

IDS perform a continuous monitoring of events. The intrusion detection software monitors the server and logs any unauthorized access attempts and aberrant behavior patterns. Of course, IDS must be instructed to recognize such events. IDS can process various types of data. The most frequent are traffic eavesdropping, packets flowing into system logs, information on users' activities. In operational terms, three primary types of intrusion detection systems are available:

- Host-based systems – HIDS

  This is firewall software that is based on auditing whatever information it can glean as generated by the OS' activities. Such information can include system-generated logs, system events (e.g. unauthorized login attempts, aberrant file accesses, file status etc.).

- Network-based systems – NIDS

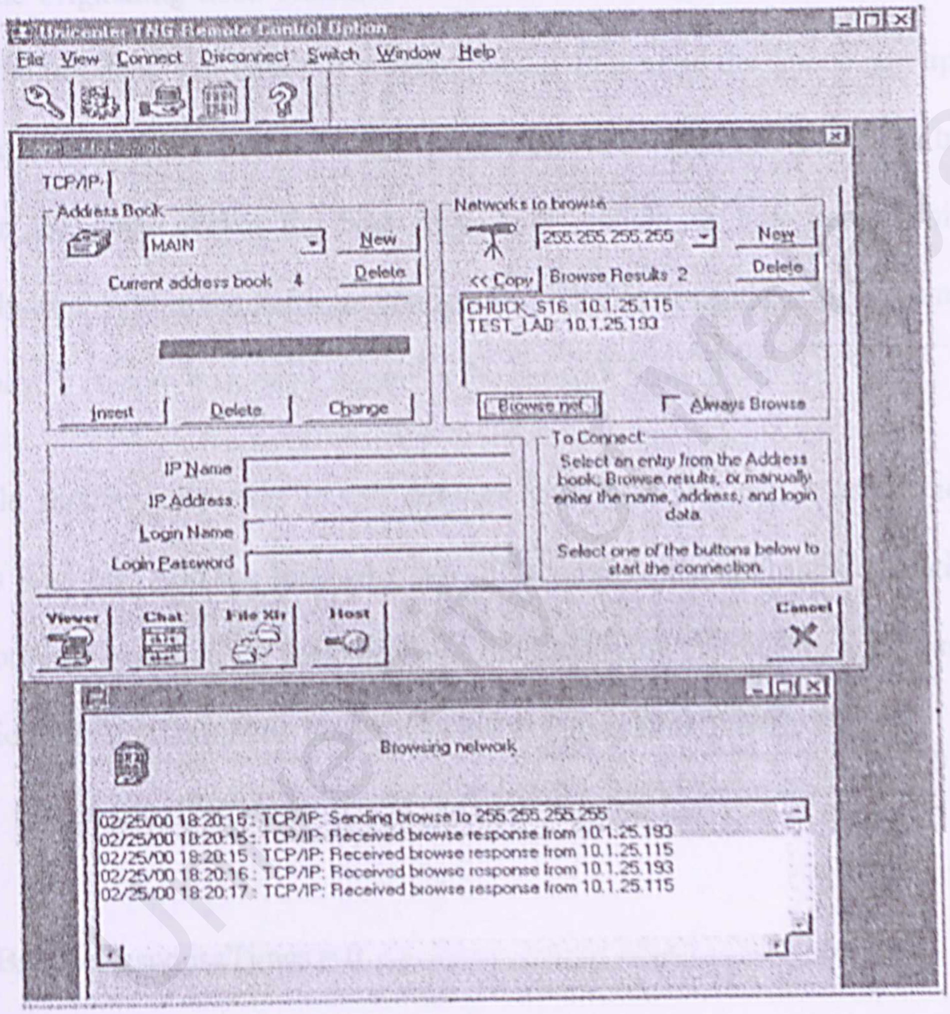
  Analyzes network packets. A NIDS agent places the network interface card into "promiscuous" mode and audits all traffic crossing the interface. As a general rule, it should be able to analyze all traffic within a specific network segment.

- Network node-based systems - NNIDS.

  A specific modification of NIDS.NNIDS is composed of micro agents distributed over each workstation within a network segment. Each micro agent monitors the network traffic directed to that workstation only, greatly reducing the capacity requirements needed by the NIDS.

Below is one of the featured intrusion detection software available on the web through purchase.

Network Browsing



Address Book: MAIN

Current address book 4

Networks to browse: 255.255.255.255

Browse Results 2

CHUCK_S16 10.1.25.115
TEST_LAB 10.1.25.193

IP Name
IP Address
Login Name
Login Password

To Connect
Select an entry from the Address book, Browse results, or manually enter the name, address, and login data.

Select one of the buttons below to start the connection.

Browsing network

```
02/25/00 18:20:15 : TCP/IP: Sending browse to 255.255.255.255
02/25/00 18:20:15 : TCP/IP: Received browse response from 10.1.25.193
02/25/00 18:20:15 : TCP/IP: Received browse response from 10.1.25.115
02/25/00 18:20:16 : TCP/IP: Received browse response from 10.1.25.193
02/25/00 18:20:17 : TCP/IP: Received browse response from 10.1.25.115
```

Figure 2.13: For Network Browsing

Computer Associates [CA] has several names for its remote control software i.e. Remotely Possible, TNG Remote Control Option and ControlIT. All of the CA remote control products have the same ability to scan an entire subnet for a host running their software. In the connect to remote screen you can enter the subnet you wish scanned by using either 0 or 255 as the last octet. The software will then provide a list of all hosts available.

This product does the subnet search by sending a UPD packet to the broadcast address of a subnet and then waiting for a response. By default the responding hosts send 2 packets back to the originating host. Attached at the end of this document are network traces showing the full details of how this program works. In general the source port appears to vary by host installation however the destination port is always set to 800 in the broadcast packet. In the reply packet the host appears to include its host name. This is not necessarily true. Within the software configuration there is a configuration setting so that you can have a custom host name appear in the network browse window.

To disable sending responses to this network broadcast you will need to modify the Windows Registry. Add the following DWORD value in the appropriate location based on your product based on the table below. After modifying the registry and rebooting the workstation/server it should no longer respond to the network browse requests.

sBrowseResponseTimes = 0

Product	Registry Key Location:
RP/32	HKEY_LOCAL_MACHINE/Software/Avalan/RemotelyPossible32/Host
ControlIT	HKEY_LOCAL_MACHINE/Software/ComputerAssociates/ControlIT/Host
RCO	HKEY_LOCAL_MACHINE/Software/ComputerAssociates/RemoteControl

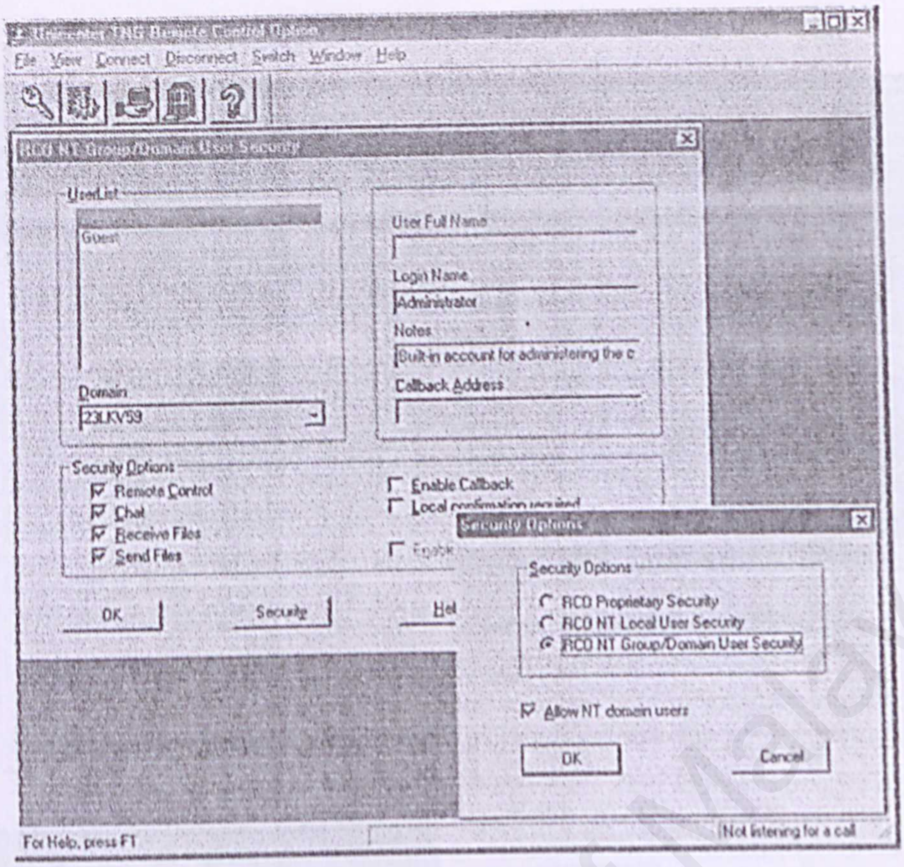
	Option/Host

Software Security

After installing this product it is very important to verify the security settings. The default security setting for this product is to utilize its own proprietary security system, which has a default login and password of "default". The default settings also do not have any logging enabled.

If you are running on a Windows workstation, which is not a member of a NT domain, you will need to use their Proprietary Security. Make sure you change the login id and password to something more secure.

If you are running on a Windows workstation, which is a member of a NT domain, you should configure the software to use the NT Group/Domain User Security. This will allow you to have the same security settings for this software that you use for logging into the network (this includes items like time restrictions and account intrusion lockouts).


Unicenter TNG Remote Control Option
File View Connect Disconnect Switch Window Help
RCO NT Group/Domain User Security

UserList
Guest

User Full Name

Login Name
Administrator

Notes
Built-in account for administering the c

Callback Address

Domain
23LKV59

Security Options
☑ Remote Control
☑ Chat
☑ Receive Files
☑ Send Files

□ Enable Callback
□ Local confirmation required
□ Enable

OK
Security
Help
Security Options

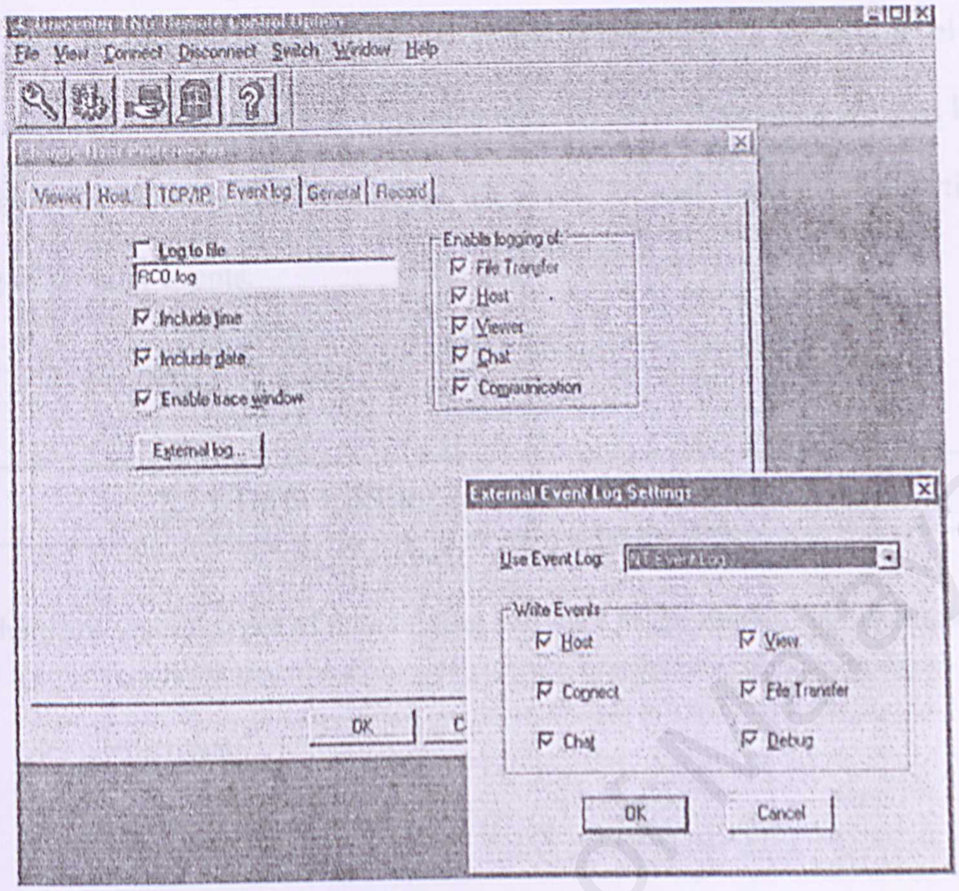
Security Options
○ RCO Proprietary Security
○ RCO NT Local User Security
● RCO NT Group/Domain User Security

☑ Allow NT domain users

Figure 2.14 : Configuring Software Security

Software Logging

If you are running Windows 95/98 enable logging to a text file and make sure all options are checked off. If you are running Windows NT Workstation or Server make use of the External Log capability. This will allow you to set the software to log all entries to the NT Event Log. Again make sure all options are checked.


File View Connect Disconnect Switch Window Help
Viewer Host TCP/IP Event log General Record
External log...
OK
Cancel

☐ Log to file

RCO.log

☑ Include time

☑ Include date

☑ Enable trace window

Enable logging of:
☑ File Transfer
☑ Host
☑ Viewer
☑ Chat
☑ Communication

External Event Log Settings

Use Event Log: NT Event Log

Write Events:
☑ Host ☑ View
☑ Connect ☑ File Transfer
☑ Chat ☑ Debug

Figure 2.15 : Software Logging

An additional safety factor can be added in by not running the remote control software at startup on user workstations. In most cases the user of a workstation can always manually start the software when needed.

After analysing the existing intrusion detections software availbale in the market, each of this marketed product deserve a compliment in their design and application. Our system however is designed using data mining technique i.e. sequential technique, in which this intrusion detection software that we design applied more of the anomaly detection technique. Our system will detect any intrusion event going on and hence will inform the user of the occuring event via email. How does FRIDCA works? It works by retrieving on a real time or schedule basis all the event and determines the security level of the event and alert user when there's an important security event depending on what level of security the event is. After this, it archives the event, for easy centralised reporting and reviewing of security events.

2.11 Data mining software in Market

There are several types of data mining software in the market. For example:-

- Clementine
- DBMiner
- PolyAnalyst

2.11.1 Clementine

Clementine was developed by Integral Solutions Ltd.(ISL). It provides an integrated data mining development environment for end users and developers. Multiple data mining algorithms, including rule induction, neural networks, classification, and visualization tools, are incorporated in the system. A distinguishing feature of Clementine is its object-oriented, extended module interface, which allow user's algorithms and utilities to be added to Clementine's visual programming environment. Clementine has been acquired by SPSS Inc.

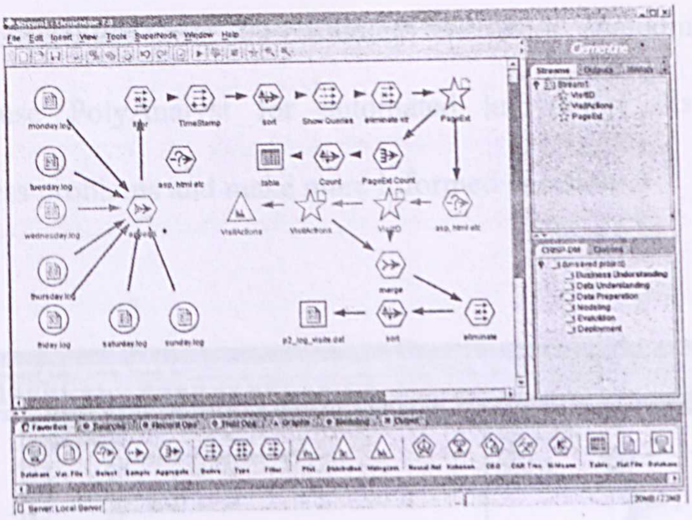


Figure 2.16: Clementine screening shot

2.11.2 DBMiner

DBMiner was developed by DBMiner Technology Inc. It provides multiple data mining algorithms including discovery-driven OLAP analysis, association, classification, and clustering. A distinct feature of DBMiner is its data-cube-based on-line analytical mining, which includes efficient frequent-pattern mining functions, and integrated visual classification methods. 5.3 Microsoft's OLE DB

2.11.3 PolyAnalyst

PolyAnalyst is the world's most comprehensive and versatile suite of advanced data mining tools. PolyAnalyst incorporates the latest achievements in automated knowledge discovery to analyze both structured and unstructured data. The PolyAnalyst platform offers a complete end-to-end analytical solution - from data importing, cleaning, manipulation, visualization, modeling, scoring, and reporting. The intuitive interface and an online tutorial smooth the learning curve to enable analysts to reach conclusions comfortably and confidently. Over 300 customers worldwide, including several Fortune 100 companies, use PolyAnalyst for automated knowledge discovery to solve complicated business problems and make more informed decisions

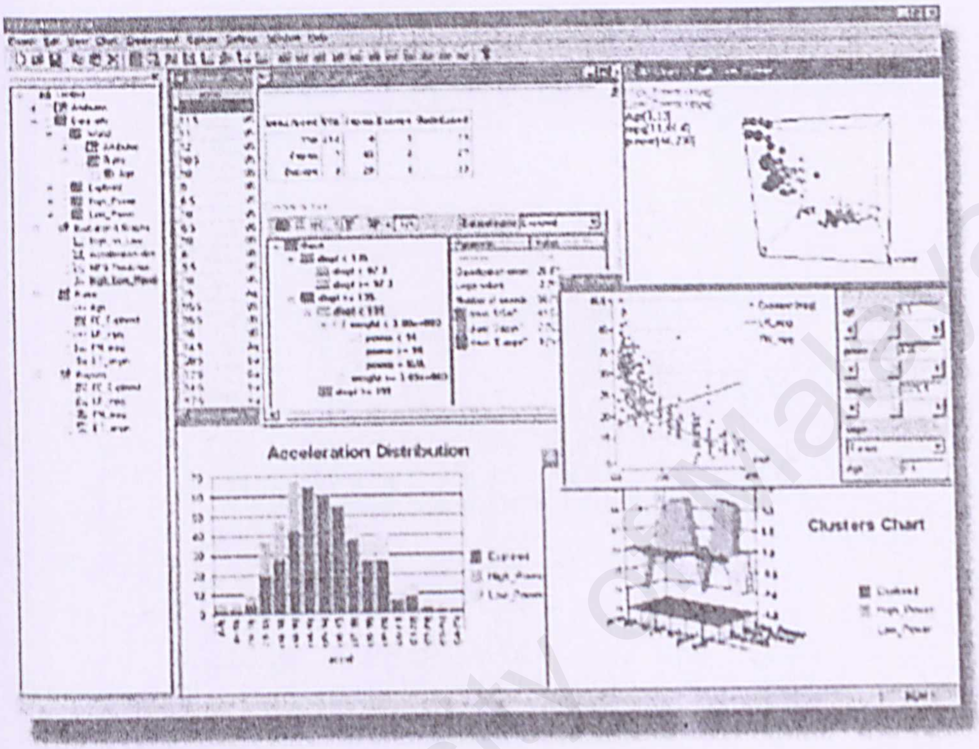


Figure 2.17: PolyAnalyst 4.5 Screen Shots

Solutions & Applications

The solutions and applications for PolyAnalyst are as follows:-

- **Database marketing …** improve mailing response
- **Cross selling …** maximize customer wallet-share
- **Predict customer behavior …** maximize loyalty and minimize churn
- **Call-center notes analysis …**understand customer feedback

- **Survey analysis …**decipher open-ended questions

- **Scientific research …** fit non-linear empirical equations

- **Fraud detection …** identify and predict fraudulent behavior

- **Customers retention …** identify satisfaction and retention driver

CHAPTER 3

METHODOLOGY

After analyzing the survey and having gathered information from the literature review in Chapter 2, this chapter will give a specific description about the technique and procedures that will be applied to gather the system requirement and also will specify the justification for the chosen methodology. The modules for the proposed system will also be discussed in this chapter.

3.1 Fact Finding

To determine the requirements of this project, it is important to gather all the relevant information related to this project. There are varieties of technique that can be used to determine the system requirement such as interview, research, sampling, surfing the internet and investing hard data, and so on. However, not all the technique can be used at the same time. Among the methods or techniques that been used are:

Research

Research will begin with an analysis of the current systems, it will give a good understanding on how the current system works and also provide the groundwork for the system designs. Research has been done through reviewing books, references, and journals that contain relevant information needed in this system. The main library in University of Malaya is visited frequently to get the relevant materials for the system development. Besides that, the document room in faculty of Computer Science and Information Technology (FSKTM) is a helpful place to gather more information and also to review the structure of the report. The document room provides previous thesis from the seniors, which is useful in giving a basic guideline and idea on how to generate a good report by evaluating the strengths and weaknesses of their work.

Internet Surfing

Internet surfing is another efficient way to find or gather useful information, which is relevant to this project. Internet is the largest knowledge and information repository in the world. It is more easier to find information through surfing than finding information from a book manually because all we need is just giving the keyword and the search engine will help us to gather the information. Therefore, information can be capture from the Internet by using the web portals and search engines such as Yahoo, Google, Amazon and many other. This may be the popular and easiest way of retrieving information.

3.2 Methodology

3.2.1 Introduction and Concept Methodology

According to FOLDOC(Free On-Line Dictionary of Computing), Methodology can be defined as a documented set of procedures, an organized, and guidelines for one or more phases of the software life cycle, such as the analysis or design phase. Whereas, software engineering methodology can also be defined as a process for the organized production of software, using a collection of predefined techniques and notational conventions (Shari, 2001). A methodology is usually presented as a series of steps, with techniques and notation associated with each step.

Process model is very important during software development process or software life cycle, because a set of activities, automated tools, best practices, deliverables and methods can be defined for system developers to use as to maintain and develop all or most information system and software.

There are several types of system development models. For instance, Waterfall Model, V Model, Prototyping Model, Spiral Model, and Transformation Model. These models provide guidance on the order in which a project should carry out its major tasks.

3.2.2 System Development Methodology

In order to analyze a database in computer auditing using data mining technique – neural network, the Waterfall Model with prototyping has been chosen as the development methodology for this system. Following are the reasons why Waterfall Model is choose:

- It enables me to examine some aspect of the proposed system and decide if it is suitable or appropriate for the finished product.
- It is widely used, easily understood and implemented in a system development.
- With the prototyping, it can enhance understanding by controlling the thrash.
- It supports good process visibility as each activity produces deliverable. The deliverable will prove useful when the system evolves in the future.
- It enables maintenance or changes to be carried out at each stage due to its interactive nature. The iteration process can be carried out as many times as

needed and this produce a fine system with high quality that meets a user's requirements.

- Prototyping is useful for verification and validation, where verification ensures that each function works correctly and validation ensures that the system has implemented the entire requirement in the specification

From figure 3.1, we can see there are five stages of Waterfall Model with Prototyping, which are the requirement analysis, system design, implementation, integration and system testing, operation and maintenance. The Waterfall Model with Prototyping is very important in order to make sure that the project has been well planned from the beginning of the stage until the end of this project.

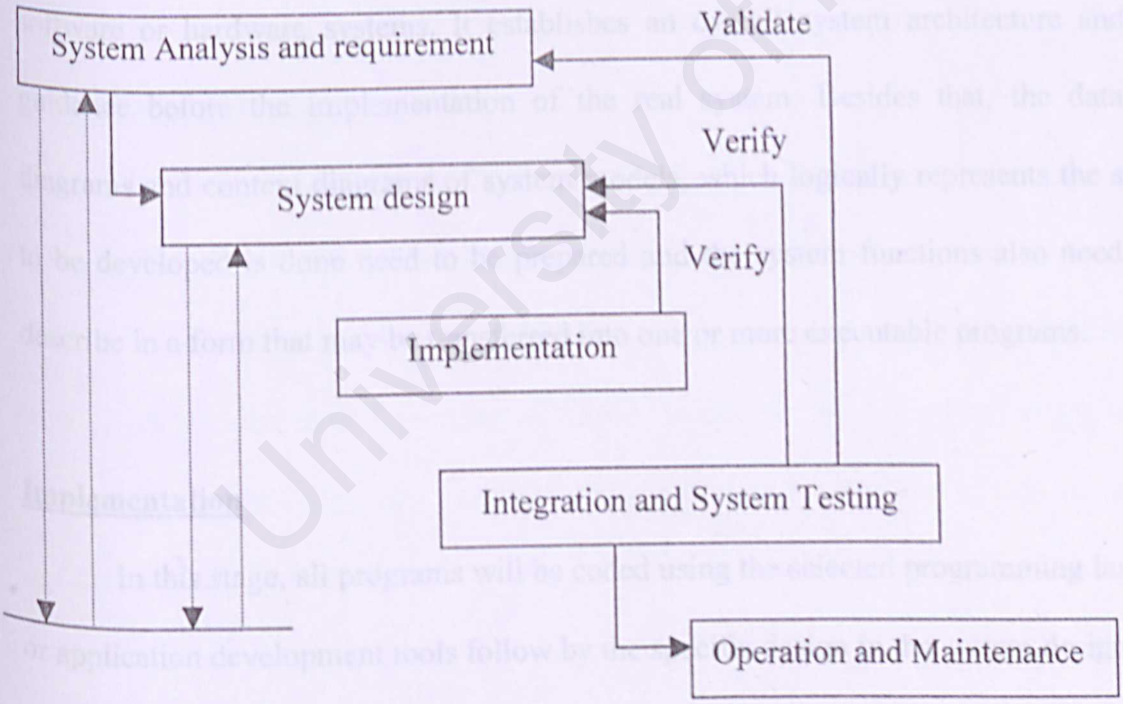

System Analysis and requirement
Validate
Verify
System design
Verify
Implementation
Integration and System Testing
Operation and Maintenance

Figure 3.1: Waterfall Model with Prototyping

System Analysis and Requirement

This stage has been done in chapter 2 and it is the first phase where it used to identify the problems, objectives and the scope of developing data mining in computer auditing using neural network algorithms. All the information regarding to this project is gathered through Internet, and reading materials such as, books and journal. This step is very important because addressing the wrong objectives and scope of this system will pretty much affect the outcome of the project. In this stage of project, requirements, constraints, and needs have to be identified.

System Design

At this stage, the project's design process partitions the requirements to either software or hardware systems. It establishes an overall system architecture and as a guidance before the implementation of the real system. Besides that, the data flow diagrams and context diagrams of system models, which logically represents the system to be developed is done need to be prepared and the system functions also need to be describe in a form that may be transferred into one or more executable programs.

Implementation

In this stage, all programs will be coded using the selected programming language or application development tools follow by the specific design in the system design stage. Implementation is a procedure to integrate the entire system that is being developed, which includes all the hardware and software in order for it to function properly and as a complete system. At this stage, each functions of the program will be tested to make sure whether it is working according to its specifications.

System testing and Integration

System testing is very important to assure the quality of the system. All the units that are separated are combined and tested as a whole system to ensure that the software requirements have been met and ensures the usefulness of the development. System testing verifies that the whole system meets its specification. If the system testing was fail, the system design stage will be reprocessed again or system prototyping will be redefined again. The main objectives of system testing are detecting the faults or errors in developed system so that it can be corrected before the system is fully operational.

Operation and Maintenance

This stage can be defined as the longest life cycle phase and it also can be described as the process of changing the system after it is under operation. The system will be installed and put into practical use. The maintenance may involves correcting errors, which were not discovered in earlier stages of the life cycle, improving the implementation of system units and enhancing the system's services as new requirement are discovered.

CHAPTER 4

SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, identify or find out the problems, and using the information to improve the system. All of the analyses done on this phase are very crucial and important for the following step, which is the system design. An extensive analysis is needed in order to get an overview of the system requirement. The system analysis will used to determine functional and non-functional requirements of this project, programming language, databases and hardware needs of this project.

4.1 Requirements Analysis

A requirement is a feature of the system or a description of something the system is capable of doing in order to fulfill the system's purpose and defines the requirement for a new system. The requirement analysis covers the area of functional and non-functional requirements of this project where it will be discussed in next session.

4.1.1 Functional Requirements

A functional requirement will describe an interaction between the system and its environments and also describes how the system should behave given the certain stimuli (Shari, 2001). Functional requirements are subsystems features, and functions that must be included in an information system and are frequently identified in terms of processes, outputs, inputs, and stored data that are needed to satisfy the system needs and be accepted to the users. The absence of the functional requirements will make the whole system incomplete. The followings are the functional requirements of this project:

Login Module

Request users (authorized users and administrator) to login this system with a valid user name and password. This will ensure that the information in the database is secured form authorized users. This module also enables the administrator to change their password frequently and create user name and password for new users. After the administrator login to this system, he or she will be able to upload any personal user's particulars and identifying the total number of them who have the permission to access this system.

Knowledge Base Module

All the results that already generated from mining neural network module can be saved in this module and allows authorize users to view it as a record in any time.

Print Module

Allows administrator to print out a hard copy of the reports by selecting the report that already generated and save in the knowledge base module.

Help Module

Provide guidelines to administrator about how to administer the system, how to analyze the selected data using neural network algorithms, and etc.

Intrusion Detection System Module

First of all, we'd like to give a brief explanation on the module use in this IDS. Basically, the current generation of IDS programs has a modular architecture. In its most basic form, IDS architecture consists of the following modules:

- IDS agents that collect information. These are software programs that reside on servers (HIDS), within critical network segments (NIDS), or on each network node (NNIDS). Agents are key issues for IDS functioning. They may generate alerts for malicious activities.

- **Database**

Here, all data collected by agents is stored. By auditing data gathered by all agents, certain attacks that are threats for the entire network can be detected and also attack trends and patterns on the network may be tracked.

- **Manager**

This is a console that manages all modules. The manager is the administrator's interface. First of all, we'd like to give a brief explanation on the module use in this IDS. Basically, the current generation of IDS programs has a modular architecture. In its most basic form, IDS architecture consists of the following modules:

- IDS agents that collect information. These are software programs that reside on servers (HIDS), within critical network segments (NIDS), or on each network node (NNIDS). Agents are key issues for IDS functioning. They may generate alerts for malicious activities.

- **Alert Generator**

This module is responsible for notifying the administrator about a potential threat. There are a variety of currently available IDS approaches. Certain IDS are limited to generate alerts (which may be logged) or others which may be placed on the management console only (Snort, Cisco RealSecure) and are based on outside software for information processing purposes. Other solutions can take the form of a wide range of sophisticated notifications (e-mail, SNMP trap, displaying a console box, fax, SMS, sending messages to the managing software, launching any deliberate program). The alerting module may be included either in an agent or in the central data acquisition system.

- **Report Generator**

Basically I didn't provide any interface for this as I consider its job as a 'behind-the-scene' kind of thing.

Often, (particularly with IDS) the database, the manager and the reporting software are integrated within a single console.

4.1.2 Non-functional requirements

Non-functional requirements are as important as functional requirements. A non-functional requirement describes a restriction on the system that limits the choices for constructing a solution to the problem (shari, 2001). It also can be defined as a constraint under which the system must operate and standard, which must be met by the delivery system soon. The non-functional requirements that have to be embedded into this project include following aspects:

Security

The system should be equipped with sufficient security to protect the data form falling into the wrong hands and should not show any potential for information leakage. Therefore, a username and password are needed to ensure that only authorized user can access the system periodically where the password should be encrypted. Only the authenticated users shall have the access rights to view and modify the data in the specified database and communication with the system need to be established with validation control to ensure authenticity of the data transfer.

Accuracy

The system must meet the objective and requirement of user started earlier and data mining(neural network algorithms) should provide and accurate result, where it should be able to show the neural network pattern for a data(Computer auditing database).

User Friendly and Usability

The Graphic User Interface (GUI) used to provide a better visual effect for user and this system will offer user friendly interface as well as a simple and ease-to-use features or applications to the user where the user able to use the system in shortest learning curve. The data-mining model can also be customized to meet the need of changing requirement. The usage of suitable and meaningful icons or buttons and menu will help the user to use the system with more confidence. Confirmation message and error message for any non-trivial process such as updating or deleting any records should be displayed to make sure that the user could do final decision before certain action is taken.

Response time

The response time to retrieve the information can be considered within a reasonable interval time. It means that when a user selects any data mining technique to analyze their data, the result should be come out to users any point in time.

Reliability

The system to be developed must be reliable because reliability is one of the essential software qualities. The system shall maintain its reliability and integrity in performing its functions and operations and shall not cause any unnecessary or unplanned actions of the overall environment. For example, users of different access authority will have different functional access, thus controls needs to be enforced to prevent any unusual access.

Flexibility

This system will be able to incorporate with new technologies in the future and in fast changing environment.

Legislation

All software, including platform used will be assured a lessened copy. None of any pirated software will be use.

Availability and Manageability

The system shall be available to users especially the administrator and the administrator should manage and operate the system. The system shall be capable of being managed and developed in the most operating system environment to maximize adaptability of the application to the computer system available to the administrator.

Modularity

Software is divided into separately named and addressable components, called module that is integrated to satisfy problem requirements. This is done to isolate functions codes form one another. This characteristic will makes testing, debugging, and maintenance much easier.

Maintainability and Expandability

The system must also designed to be understood, corrected, adapted, and be able for enhancement without difficulties. Architecture components, algorithm, data structure, and procedure design should be able to extend and modify with ease so that any future enhancements and expansion can be done easily.

4.2 Development Analysis

After reviewing and analyzing the requirements, the tools for developing the system will be decided. The following section discusses all the tools that will be used in this system.

4.2.1 Operating system

Why Microsoft Windows 2000 Professional is the chosen Operating System?

Microsoft Windows 2000 Professional was chosen as an operating system for this project because:

- It is ease to use and helps to reduce costs and increase productivity through improved management and reliability.
- The Microsoft Windows 2000 Professional is reliability and can modify the operating system core to prevent crashes.
- The Windows 2000 Professional is mobility, where it allows us to work at anywhere in anytime, help to save time, and increasing productivity because Windows 2000 Professional offers time-saving and mobile users key productivity features. This includes the ability to easily take files and folders offline and the ability to hibernate and restart the system without a reboot.
- Windows 2000 Professional is easy to support, manage, and deploy. The performance of Windows 2000 Professional is faster than Windows 95/98 and Windows NT 4.0.
- There is a security features in windows 2000 professional and it used to protect the sensitive data.
- Windows 2000 Professional is usability because it combines with the traditional ease of use of Windows 98, combines the power and security of its predecessor, and Windows NT Workstation.

4.2.2 Database

MySQL was chosen as a database for this project because:

- ❖ Each MySQL client gets a dedicated thread in the MySQL server, which allows different users to access the same tables at the same time. All MySQL operations are atomic: no other users can change the result for a running query.
- ❖ MySQL has a compact fast design (the code size of the server is less than 1MB on an i386), which normally uses very little memory, but can be configured to take advantage of large amounts of memory.
- ❖ MySQL has a user configurable key cache and a record cache to quickly scan tables. Open tables are cached in a table cache.
- ❖ SQL functions are implemented through a highly optimized class library. Almost all parsing and calculating is done in a local memory store. No memory overhead is needed for small items, and the normal slow memory allocation and freeing is avoided.
- ❖ MySQL can be used from many popular languages.
- ❖ MySQL can runs on different platforms.

4.2.3 Application Programming Language

Why Java is the chosen Programming Language?

Java Programming language was chosen for this project because it is a perfect programming language for building applications and it is an independent platform where a single application can run on a computer that running Windows 95/98/2000/NT and so on. Below shows some of the reasons why Java language was chosen:

- The Java language will makes the programming easier because it provides the object-oriented feature and has automatic garbage.
- Java can be easily to download form the net.
- Java programming language is powerful development tools for a programmer compared with C, C++, as well as ASP.
- Java is a more dynamic language and Java language.
- Java allows creation of applets that allows animation of a graphics, play audio clips, and interact with the users through graphical user interface.
- Java enables the construction of virus-free and tamper-free systems.
- Java can eliminate the possibility of corrupting data and overwriting memory.

4.3 Software Requirement

The table below shows the software requirements that will be used or needed in order to developed this project.

Description	Software
Operating system	Microsoft 2000 Professional
Database	MySQL
Programming Language	JAVA

Table 4.1 : Software requirements

4.4 Hardware Requirements

The table below shows the minimal hardware requirement for developing this project.

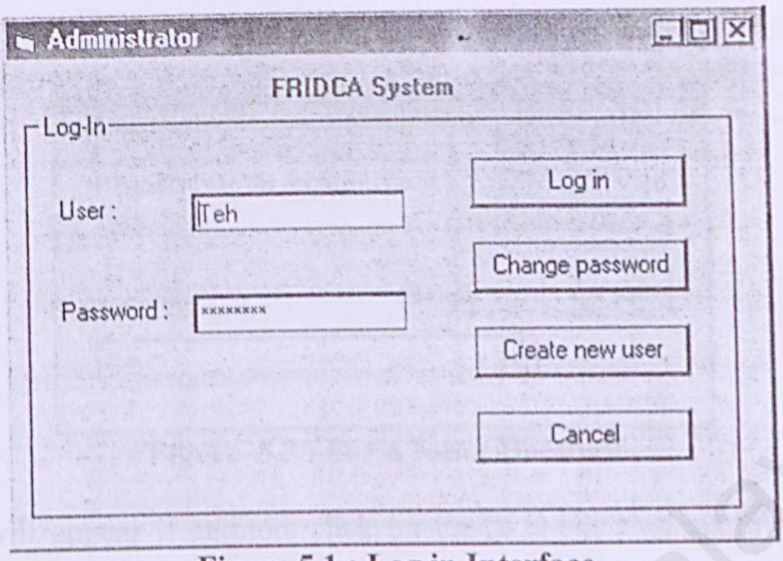
Description	Component
Microprocessor	133MHZ or higher Pentium compatible
RAM	64MB
Storage	2GB of hard disk
Input devices	Mouse, Keyboard
Output devices	Printer

Table 4.2: Hardware Requirements

CHAPTER 5

SYSTEM DESIGN

FRIDCA System Interface Design



User : Teh

Password : xxxxxxxx

Figure 5.1 : Log in Interface

This module enables auditors authorized users and administrator) to login this system with a valid user name and password. This will ensure that the information in the database is secured form-authorized users. This module also enables the administrator to change their password frequently and create user name and password for new users.

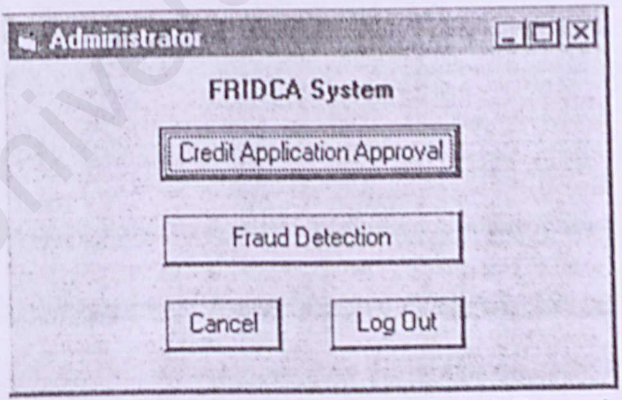


Figure 5.2: Credit Application Approval and fraud Detection choose button

This module provides two buttons for auditors to choose. If auditors need to check the status or the result of the applicants, then he or she should click on Credit Application

Approval button. While Fraud Detection button is for auditors to check or detect the credit card from fraud details.

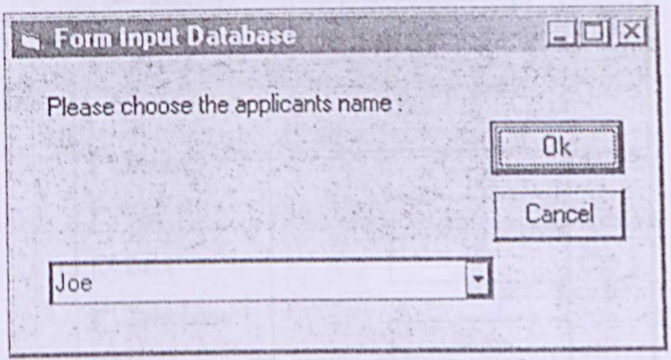
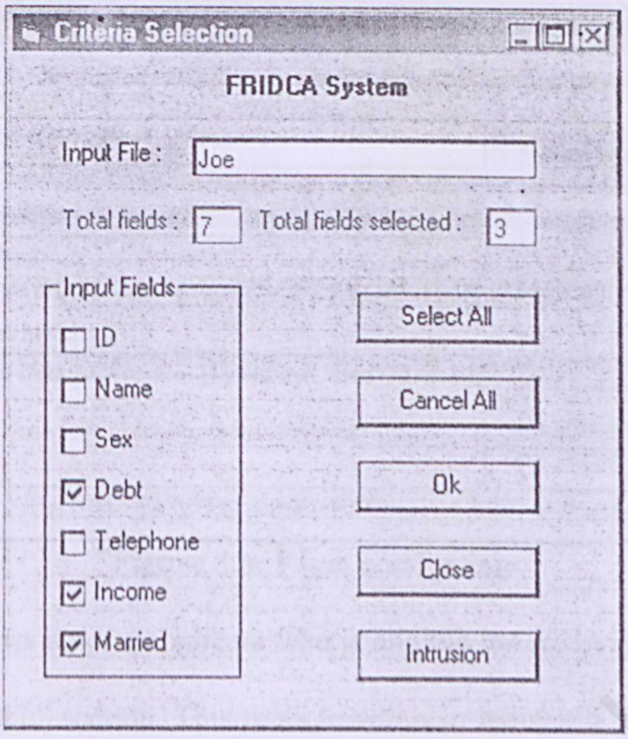


Figure 5.3 : Form Input Database

This Module will appear if auditors click on credit application approval button. This module enables auditors to choose the applicants name from the database store. After choosing the applicants name, clicking Ok button enables system to turn to another interface as shown in figure 5.4, and automatically bringing all information that is related to the refer applicant from the database.



Input File : Joe

Total fields : 7 Total fields selected : 3

Input Fields

□ ID

□ Name

□ Sex

☑ Debt

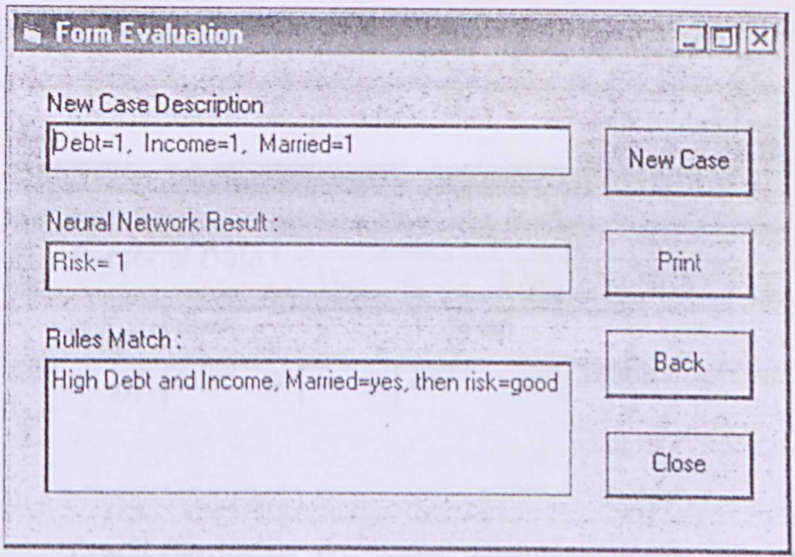
□ Telephone

☑ Income

☑ Married

Figure 5.4 : Criteria Selection form

This module automatically display the refer input file of all the information about the applicant from database. Total fields are referring to total description fields, which is the related information about the applicant. Total fields selected are referring to how many input fields is selected from the check box. Auditors need to select these input fields because it depends on the problem domain, the evaluation criteria might not be the same. These input fields will be the inputs for the neural network rules match. This module enables auditors to directly select all if deciding to take all the input fields as neural network rules match and also enables auditors to cancel all and start selecting manually. Intrusion button is for the system to start detecting if there is intrusion when the information is uploading. However, user needs to verify the setting before this system can detect any unauthorized activity, in which we discuss further as we go on.



New Case Description

Debt=1, Income=1, Married=1

Neural Network Result :

Risk= 1

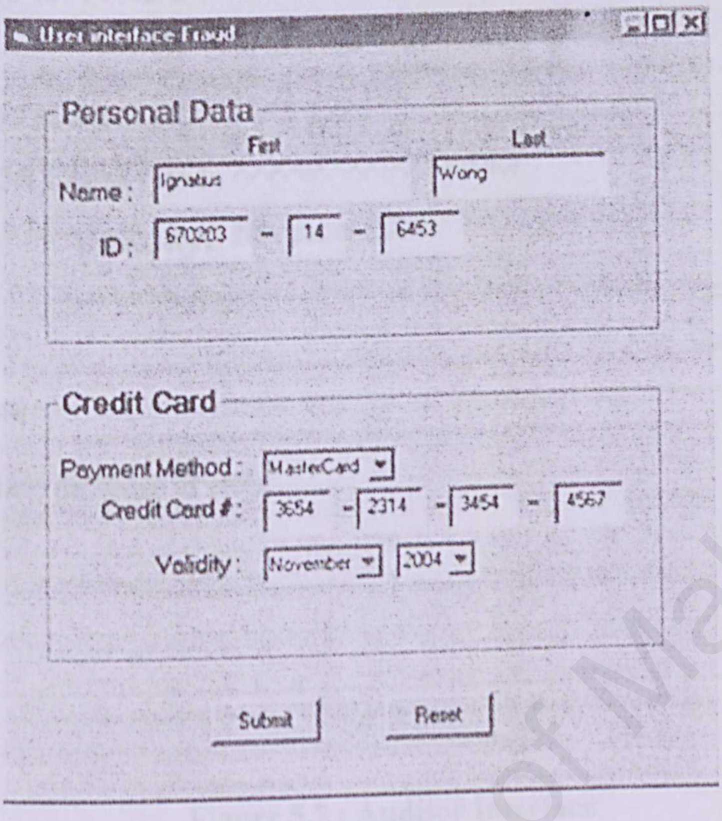
Rules Match :

High Debt and Income, Married=yes, then risk=good

Figure 5.5: Form Evaluation

This module will shows a good applicant who is eligible for credit approval for the given reason in "Rules Match" section. This module will also indicate a poor creditworthiness applicant as inferred by Neural Network. Risk=1 as shows in section "Neural Network Result" is refer to a good applicants and his or her application is approved. If this section showing that Risk=0, this refer to a poor applicant which his or her applications cannot be approved.

Proposed Interface Design for Fraud Detection



Personal Data

Name: First: Ignatius Last: Wong

ID: 670203 - 14 - 6453

Credit Card

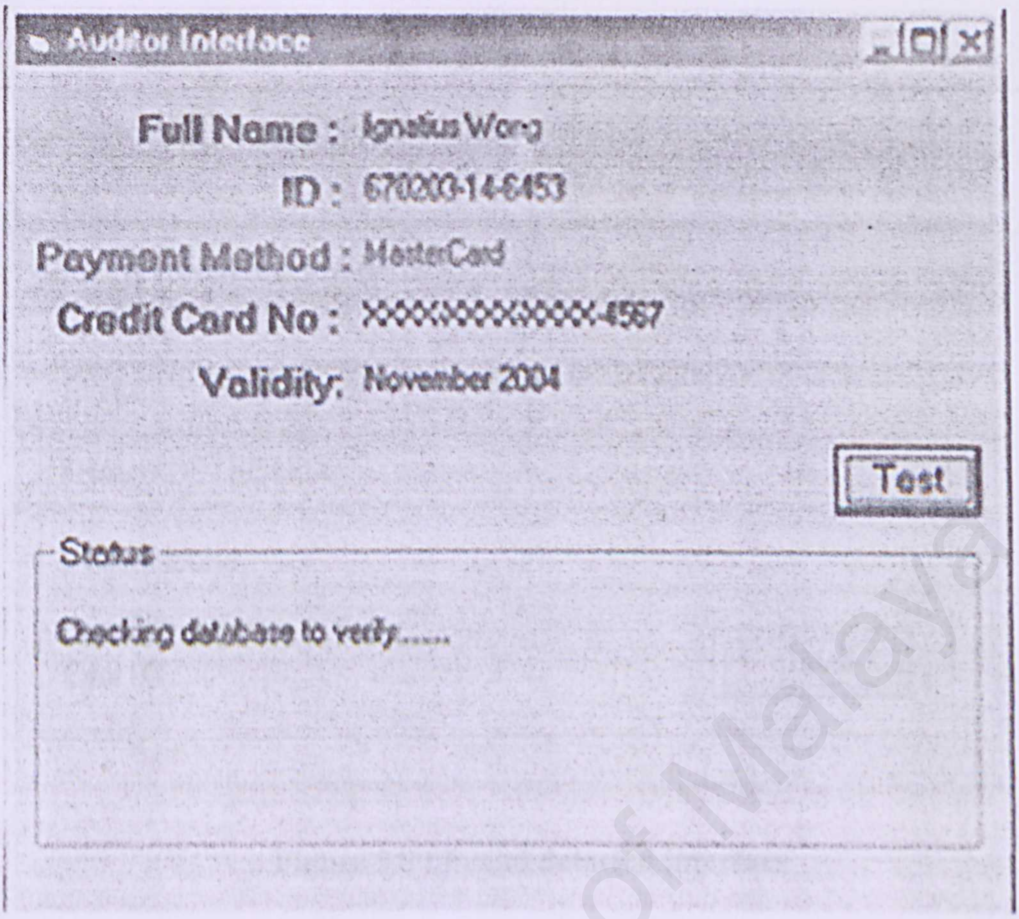
Payment Method: MasterCard

Credit Card #: 3654 - 2314 - 3454 - 4567

Validity: November 2004

Figure 5.6: User interface for fraud detection

Above is the user interface design for FRIDCA, focusing on fraud detection. This is a simple interface and it wouldn't be a problem for user to use. All the users have to do is fill up the empty space with valid information. After putting in their information, the user click on the "Submit" button or if they want to re-enter their information, the "Reset" button is used. Please be reminded that this is the interface which will be displayed to the user.



Full Name : Ignatius Wong

ID : 670200-14-6453

Payment Method : MasterCard

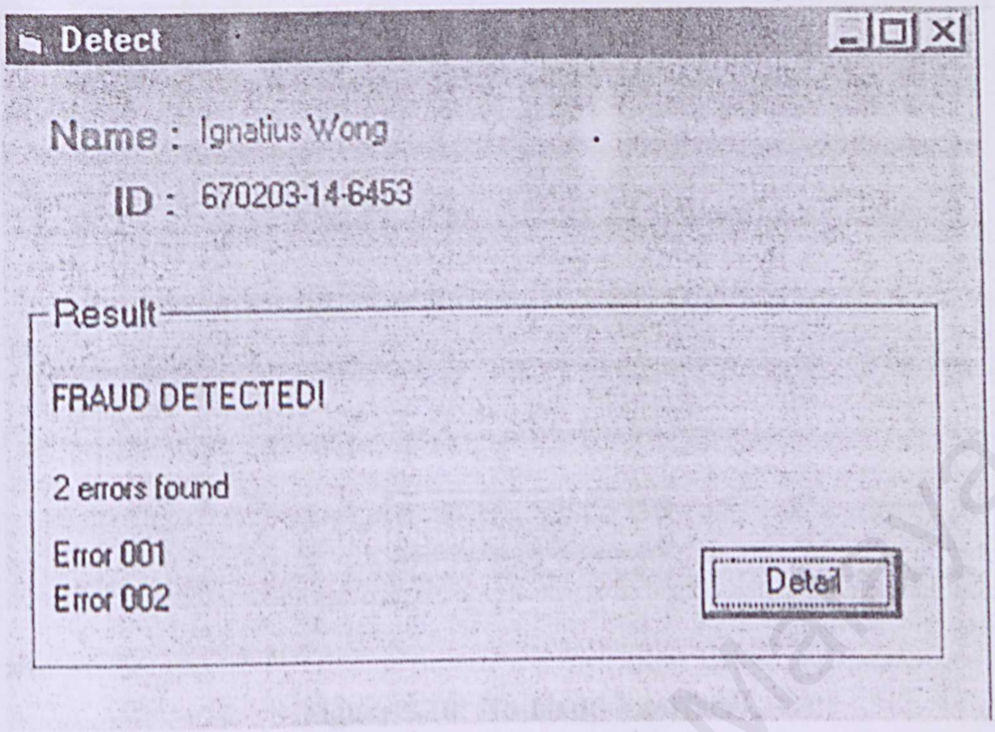
Credit Card No : XXXXXXXXXXXX-4567

Validity: November 2004

Checking database to verify......

Figure 5.7 : Auditor Interface

This is the interface, which will be displayed to the auditor. The auditor can see all the information given by the user, except for the credit card number that only show the last four digits. This is because the credit card number is a unique entity and this should be kept unknown and hidden from outsiders. The "Test" button is used when the auditor wants to test the information given by the user, to see if it is valid or not. The status column in the interface indicates that the system is checking the database to verify the information.



Name : Ignatius Wong

ID : 670203-14-6453

Result

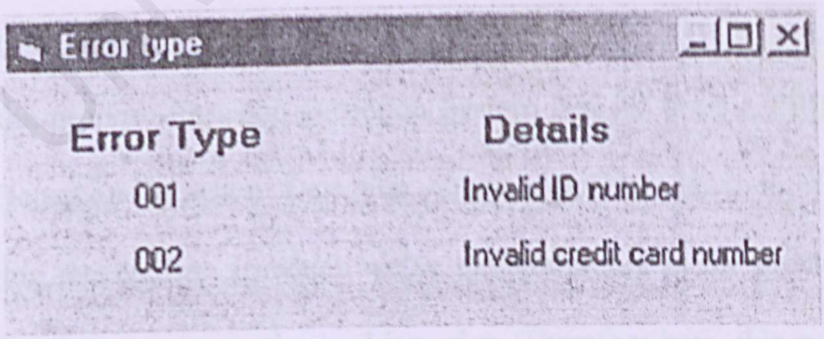
FRAUD DETECTED!

2 errors found

Error 001
Error 002

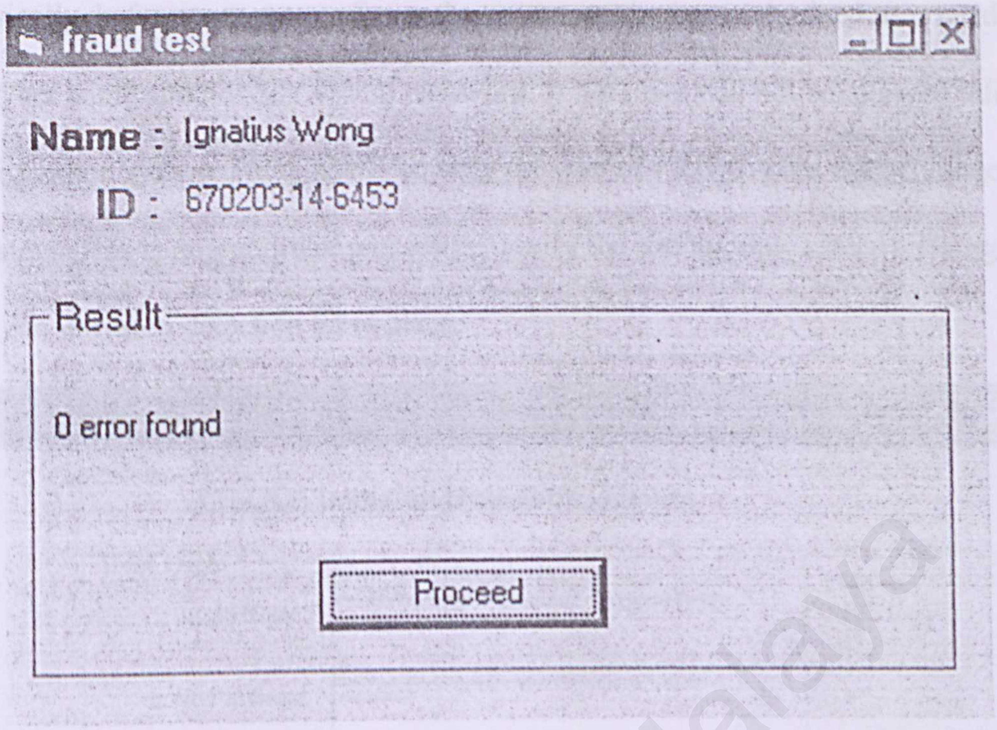
Figure 5.8 : Fraud detected interface

This is the interface displayed, when fraud is detected. It occurs when the system detects anomalies while it was searching in the database. The "Detail" button will show the details of the error. The interface below will explain this.



Error Type	Details
001	Invalid ID number
002	Invalid credit card number

Figure 5.9: Interface displaying error details



fraud test

Name : Ignatius Wong

ID : 670203-14-6453

Result

0 error found

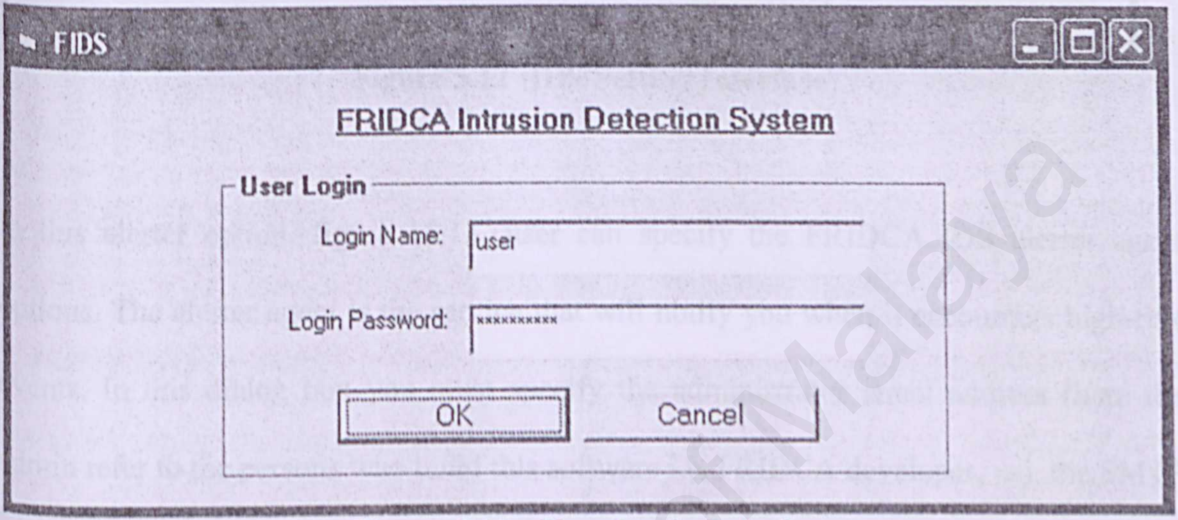
Figure 5.10: No-fraud interface

Lastly, if the system did not detect any faults, the above interface will be displayed. As the system did not detect any anomalies, the transaction will then proceed. All the auditor has to do is click on the "Proceed" button. Therefore, we can say that no fraud is detected.

FRIDCA Intrusion Detection System (FIDS)

This is the propose Intrusion Detection System that we design. FRIDCA which means Fraud Detection Intrusion Detection Credit Approval have in it system an intrusion detecting mechanism. In which we've explain before in the previous chapter, IDS is designed to detect any unauthorized intrusion. Therefore here, this system is used to detect any anomalies via sequential technique and alert the user if any intrusion occurs during transaction.

Basically, before user can configure the system setting, an authorized user needs to login his/her name and provide a valid password. If not, user will not be allowed into FIDS configuration system. Here, however, only the user of FRIDCA has the privilege to use this system. This is an important process i.e. verify the setting, only after verifying the setting can intrusion detection takes place.



FIDS

FRIDCA Intrusion Detection System

User Login

Login Name: user

Login Password: xxxxxxxxxx

Figure 5.11 : User Login Interface

After successfully login to the system, the interface below will pop up. As you can see, there are five buttons, the alert button, action button, TCP/IP button, help button and logout button.

User can click any of the following buttons that he/she wish to configure the setting. I'll explain further the function of each button as we move on. As for the help button, it simply helps the user if he/she encounter any problem. And if user doesn't wish to configure the setting he/she can just click the logout button.

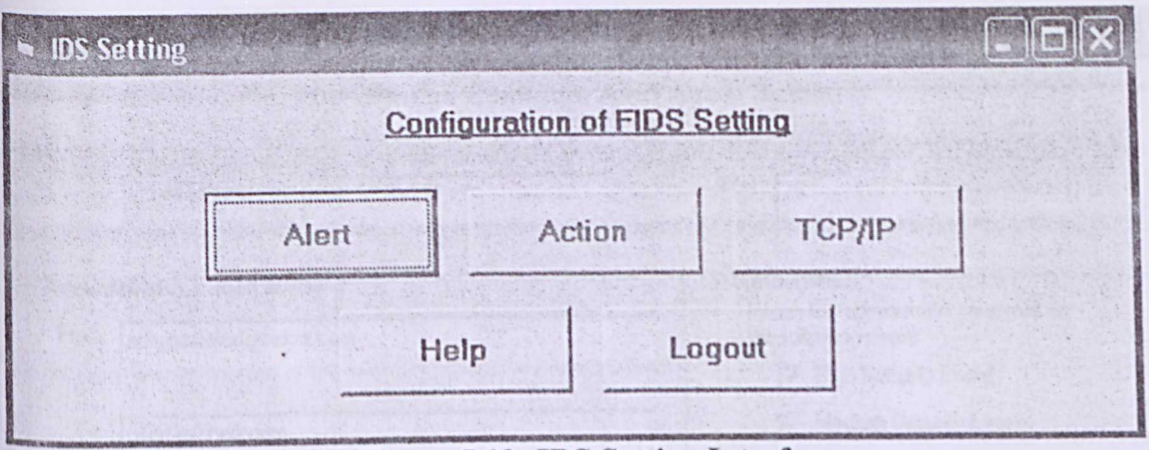
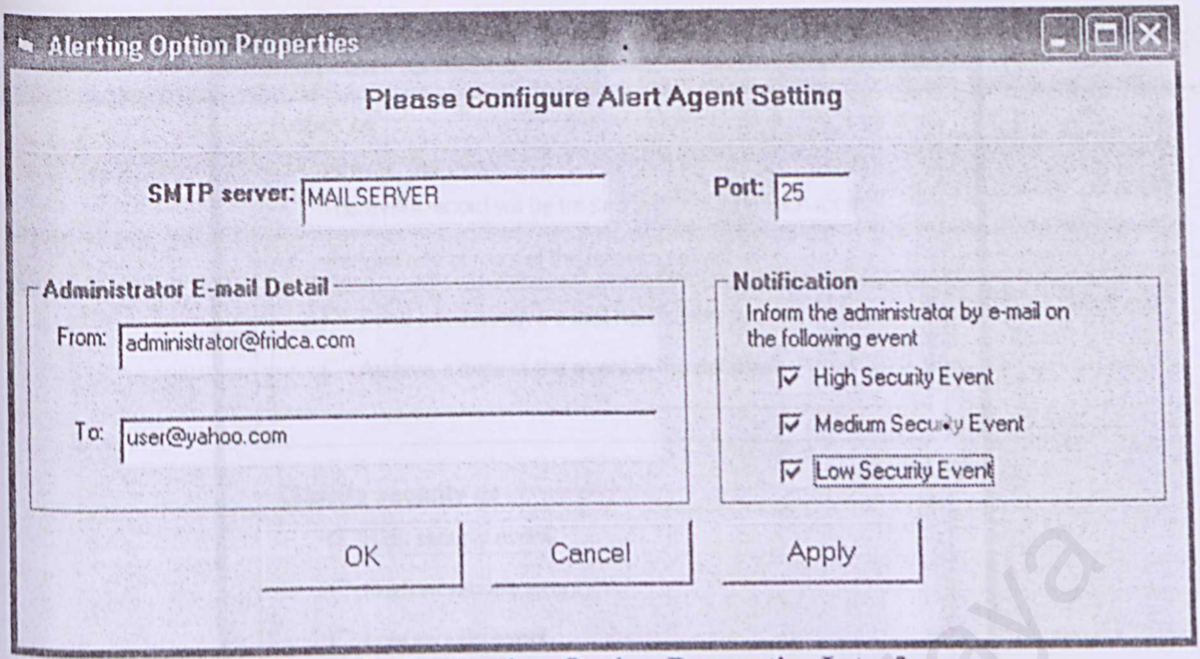


Figure 5.12 :IDS Setting Interface

In this alerter option, figure 15.13, user can specify the FRIDCA IDS alerter agent options. The alerter agent is the service that will notify you when it encounters high-risk events. In this dialog box you must specify the administrator email address (here the admin refer to the persons who build this software i.e. FRIDCA developer, us), the SMTP server name and the port. When you receive an administrative alert email, user will be presented with both the original details of the event as well as extended information explaining to you what may have caused that event and how to deduce whether that event indicates an intruder or not.

This is a very difficult task to for us to build in which it involves sending direct mail to the system user.



Alerting Option Properties

Please Configure Alert Agent Setting

SMTP server: MAILSERVER

Port: 25

Administrator E-mail Detail

From: administrator@fridca.com

To: user@yahoo.com

Notification

Inform the administrator by e-mail on the following event

☑ High Security Event

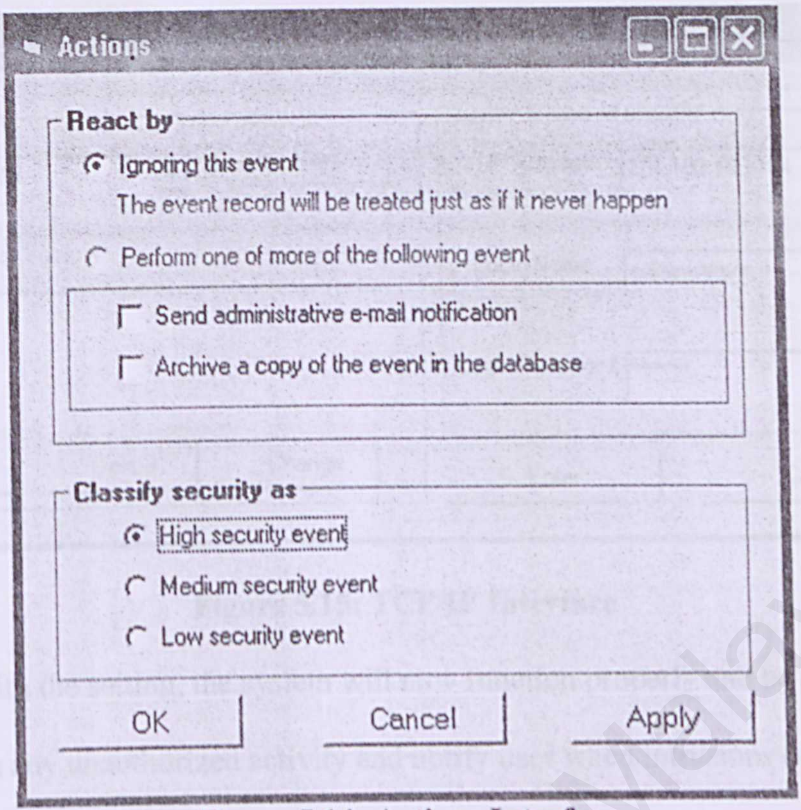
☑ Medium Security Event

☑ Low Security Event

Figure 5.13 :Alerting Option Properties Interface

Figure 15.14 is the action properties, here user can specify the level of intrusion. User can react by ignoring the event or send an administrative e-mail notification or archive the event in the database.

Apart from that, user can classify the security level as high, medium or low. So that administrator can keep a record of the event.



React by

● Ignoring this event

The event record will be treated just as if it never happen

○ Perform one of more of the following event

□ Send administrative e-mail notification

□ Archive a copy of the event in the database

Classify security as

● High security event

○ Medium security event

○ Low security event

Figure 5.14: Actions Interface

As for the TCP/IP dialog box figure 5.15, user can specify which IP address is considered valid to the user computer system so as to detect any unknown intrusion from outside intruder when he/she is connected to the Internet. Here, as you can see, user can choose from the address book to specify the known IP address that he/she has previously stored.

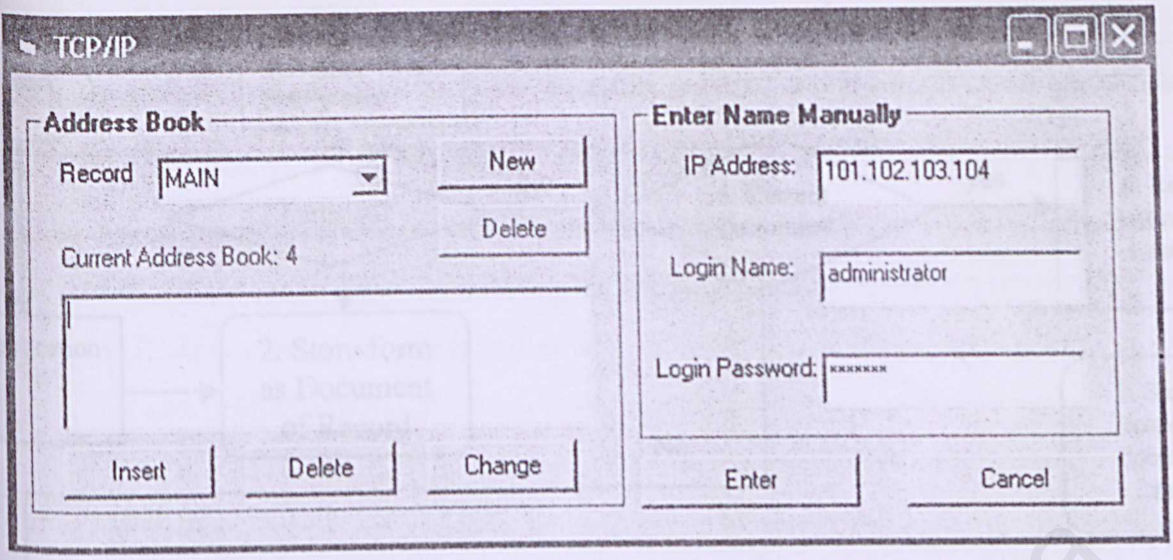
New button: Insert a new address

Delete button: To delete any address that user previously stored

Insert button: Insert the new address to the address book

Change button: To edit the previously stored address

User can also choose the 'Enter-Name-Manually' box to enter the IP address manually, however user need to login his/her name and password as wells.



Record: MAIN

Current Address Book: 4

IP Address: 101.102.103.104

Login Name: administrator

Login Password: xxxxxxx

Figure 5.15: TCP/IP Interface

After finish with the setting, the system will now function properly and be able to protect user data from any unauthorized activity and notify user when intrusions occur.

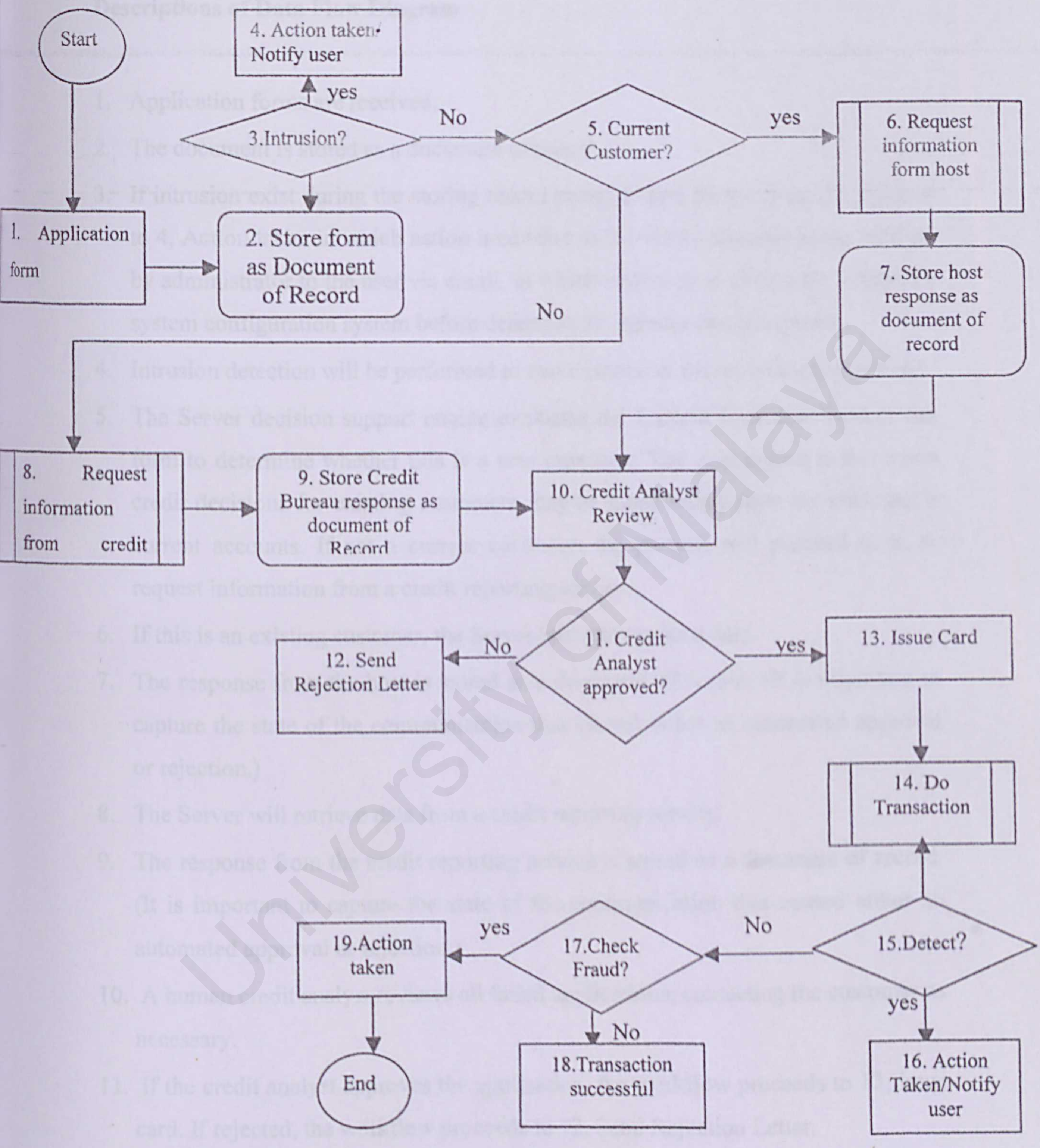

Start
4. Action taken/ Notify user
yes
3.Intrusion?
No
5. Current Customer?
yes
6. Request information form host
1. Application form
2. Store form as Document of Record
No
7. Store host response as document of record
8. Request information from credit
9. Store Credit Bureau response as document of Record
10. Credit Analyst Review
12. Send Rejection Letter
No
11. Credit Analyst approved?
yes
13. Issue Card
14. Do Transaction
19.Action taken
yes
17.Check Fraud?
No
15.Detect?
yes
No
16. Action Taken/Notify user
End
18.Transaction successful

Figure 5.16: FRIDCA System Data Flow Diagram

Descriptions of Data Flow Diagram

1. Application forms are received.
2. The document is stored as a document of record.
3. If intrusion exist during the storing record process, then the process will proceed to 4, Action taken in which action here refer to the intrusion event being notified by administrator to the user via email. In which user need to previously verify the system configuration system before detection for intruder can take place.
4. Intrusion detection will be performed to avoid intrusion and unauthorized activity.
5. The Server decision support engine evaluates the Current Customer field of the form to determine whether this is a new customer. The assumption is that some credit decisions for existing customers may be made based upon the customer's current accounts. If not a current customer, the process will proceed to 8, to request information from a credit reporting service.
6. If this is an existing customer, the Server will retrieve host data.
7. The response from the host is stored as a document of record. (It is important to capture the state of the communication that caused either an automated approval or rejection.)
8. The Server will retrieve data from a credit reporting service.
9. The response from the credit reporting service is stored as a document of record. (It is important to capture the state of the communication that caused either an automated approval or rejection.)
10. A human credit analyst reviews all failed applications, contacting the customer as necessary.
11. If the credit analyst approves the application, the workflow proceeds to 13, issue card. If rejected, the workflow proceeds to 12, Send Rejection Letter.
12. The Server sends an automated letter.
13. The Server initiates a process to issue card.
14. Applicants will use the credit card given to do their transaction over the net or ATM machine.

15. If intrusion exists during the storing record process, then the process will proceed to 16, Action taken. Intrusion detection is an ongoing process in which it will detect any unauthorized behavior.

16. Intrusion detection is performed to avoid intrusion.

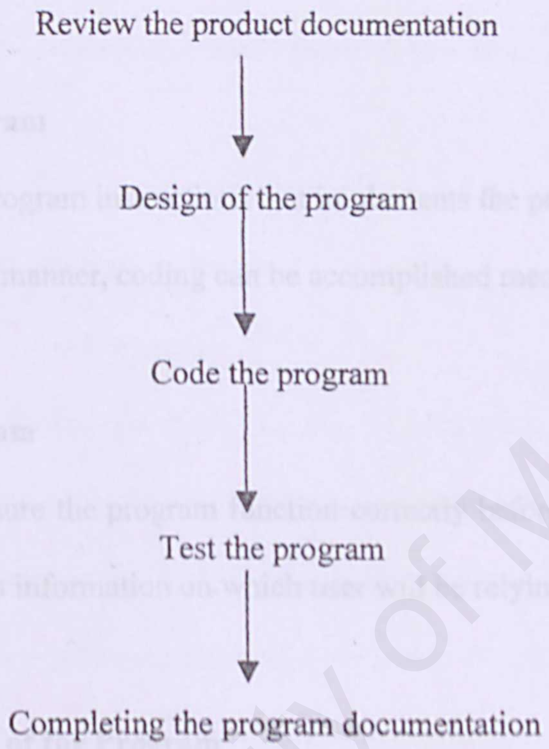
17. If no intrusion were detected, then the system will automatically check fraud. If there is no fraud, then the transaction is successfully performed. But if the system identifies a fraud during the transaction, then the process will proceed to 19 Action taken.

18. A transaction is successfully done.

19. Fraud detection system will take place in order to detect credit card fraud.

CHAPTER 6
SYSTEM DEVELOPMENT

6.1 5 Steps of System Development



Review the product documentation

Design of the program

Code the program

Test the program

Completing the program documentation

Figure 6: System development

6.1.1 Review The Product Documentation

Review the product documentation that was prepared during the previous phases. To understand the work better.

6.1.2 Design of The Program

This is the process of what it must do by developing a logical solution to the programming problem. The logical solution is a step-by-step solution to a programming pattern.

6.1.3 Code The Program

Process of writing the program instructions that implements the program design. If design is performed in detailed manner, coding can be accomplished mechanically.

6.1.4 Test The Program

Program is tested to ensure the program function correctly before the program processes actual data and produces information on which user will be relying on.

6.1.5 Documentation of the Program

Completing the program is essential for the successful operation and maintenance of the information system. Documentation includes the system's user manual that may be needed by customers.

6.1.6 Coding Approach

An easy to real source code makes the system easier to maintain and enhance. This coding process translates the system design into programming language, which is a machine-readable form.

6.1.7 Database Connection

Microsoft Access 2000 is used in this for database connection. Microsoft Access provides the means by which program code access the database.

6.2 The Adjustment For Implementation

During the development of our system, we have changed the tools and technology used. Among the changes that have been made are:-

- **Programming language**

  We choose Visual Basic 6.0 as our programming language to build our system instead of the previously proposed Java. It is because Visual Basic is easier to learn and the fact that we have learned it before. We realized that Visual Basic combines extraordinary ease of use and great power and flexibility. It can be used in many ways and at many levels from beginners to expert user. It is the easiest computer languages to work with and understand.

- **Database**

  We choose Microsoft Access 2000 as our system database because it is already installed in the computer and thus, it is easier to learn. In addition, it saves our time in learning a new database program such as MySQL. Microsoft Access 2000 is used in order to retrieve input or data from the users and also the auditors. Therefore, the auditor will be able to have a view of the record that is transparent to the users.

- **Other Adjustments**

Previously we have proposed an email function and also to detect IP address from the internet but due to lack of time and knowledge, we are unable to apply this function in our FRIDCA system. Also, the proposed Help module which we have mentioned before is not available because mostly our system is conducted by the auditors that have knowledge of the system built. Besides, our system is user friendly so it wouldn't be any problem for the users to use it.

CHAPTER 7

SYSTEM TESTING

7.1 System Testing

Testing is a process of executing a program with the intention of finding an error. Therefore changes and adjustment can be taken care of immediately.

7.1.1 Unit Testing

Unit testing was done where control paths are tested to uncover error. The first step is to examine the program code by reading through it. All of the coding is made sure there's no debug so that the paths and the flow of the website still fluently browsed. Finally, test cases are developed to show that the input is properly converted to the desired output.

7.1.2 Integrating Testing

This is an approached where the program structure was constructed at the same time test are conducted to uncover errors associated with interfacing. Testing the interfaces explore how components interact with each other. Error will be corrected before processing to the next integration.

7.1.3 System Testing

It is the final phase. This process ensures that all units in the module will function accordingly when integrated and have fully satisfied its functional requirements. It reveals bugs that cannot be attributed to individual components or to the interaction among component or to the interaction among components and other objects.

7.1.4 *System Evaluation*

Evaluation was implemented to consider carefully before effectiveness can be concluded. Field test evaluation was carried out when the information system was believed to be of the final draft quality. If problems were identified, additional changes may be made. However the informal evaluation conducted at this point should ensure that the information system is completed or minimal changes will be required.

7.2 Coding

```
/* to declare the variables*/

Public db As Database
Public rs As Recordset
Public sqlstr As String
Public sdata As Integer
Public pvalue As String
Public pvalue1 As String
Public pvalue2 As String
Public weekday1 As String

/*to open the database and the selected file*/

Public Function OpenDatabase(TableName As String)

   Set db = DBEngine.Workspaces(0).OpenDatabase(App.Path & "\db.mdb")
   Set rs = db.OpenRecordset(TableName, dbOpenDynaset)

End Function

Public Sub dbsetup(str As String)

'Open the DataBase
Set db = DBEngine.Workspaces(0).OpenDatabase(App.Path & "\db.mdb")
'Open the File
Set rs = db.OpenRecordset(str)

End Sub
```

```
/*to classify the pattern*/

If rs1.EOF Then
   isEnd = True
End If
Counter = 0
While Not (rs1.EOF)
      SQL = "Select count(*) from past where firstname='" & txtFirstName & "' and
lastname='" & txtLastName & "' And amt='" & rs1!amt & "'"

      dbsetup (SQL)
      amount = amount & rs1!amt & "="
      frequency = frequency & rs(0) & "="

      rs1.MoveNext
      Counter = Counter + 1
      If rs1.EOF And Counter < 10 Then
         isEnd = True
      End If
Wend

/*(frequency & "---" & amount)*/

If isEnd Then
   largestsum = txtamt
Else
   j = Split(amount, "=")
   f = Split(frequency, "=")
   largest = f(0)
   largestsum = j(0)
   For I = 0 To UBound(j) - 1
      If CDbl(largest) < CDbl(f(I)) Then
         largest = f(I)
         largestsum = j(I)
      ElseIf CDbl(largest) = CDbl(f(I)) And CDbl(largestsum) < CDbl(j(I)) Then
         largest = f(I)
         largestsum = j(I)
      End If
   Next
End If

getfrequency = largestsum


End Function
```

```
/* to generate the neural network result*/

If txtPmt <= 2000 Then
   pmt = 1
Else
   pmt = 0
End If

If txtIcm >= 2000 Then
   income = 1
Else
   income = 0
End If

If txtMrd = "Yes" Then
   status = 1
Else
   status = 0
End If

If income = "1" Then
   If status = "0" Then
      risk ="poor"
   Else
      risk = "Good"
   End If
ElseIf income = "0" Then
   risk = "poor"
End If
```

```
/*to return the number of total support count*/

Private Sub cmdcheck_Click()
   Dim small
List = Split(returnCount, "-")
   small = -1

   For i = 0 To UBound(List)
      If List(i) = 0 Then
         small = 0
      ElseIf small <> 0 Then
         small = List(0)
         If small > List(i) Then
            small = List(i)
         End If
      End If
   Next
   /*to return the result of the count */
   txtcount = small
   If small = 0 Then
      txtsuspect = "Not Suspected"
   ElseIf small = 1 Then
      txtsuspect = "PreCaution"
   ElseIf small = 2 Then
      txtsuspect = "Suspected"
   ElseIf small = 3 Then
      txtsuspect = "Suspected"
   ElseIf small >= 4 Then
      txtsuspect = "Give warning"
   End If

End Sub

/* to combine the frequency of days with staff during the occurrence of highcount
intrusion*/

Private Function returnCount()
   Dim calcTotal, daycount

   For i = 0 To chkday.Count - 1
      If chkday(i) = Checked Then
         First = i
         incValue = incValue + i

         weekday1 = chkday(i).Caption
         /*For staff1*/
```

```
        sqlstr = "SELECT count(*) from time_frame where Staff1='" & txtstaff & "' and
day='" & weekday1 & "'"
        dbsetup (sqlstr)
        calcTotal = calcTotal + rs(0)

        /*For staff2*/
        sqlstr = "SELECT count(*) from time_frame where Staff2='" & txtstaff & "' and
day='" & weekday1 & "'"
        dbsetup (sqlstr)
        calcTotal = calcTotal + rs(0)

        /*For staff3*/
        sqlstr = "SELECT count(*) from time_frame where Staff3='" & txtstaff & "' and
day='" & weekday1 & "'"
        dbsetup (sqlstr)
        calcTotal = calcTotal + rs(0)

        If First = incValue And Not (isfirst) Then
           daycount = daycount & calcTotal
           isfirst = True
        Else
           daycount = daycount & "-" & calcTotal
        End If
        calcTotal = 0
      End If
   Next
   returnCount = daycount
End Function

/*to count the number of intrusion for that day*/

If rs.RecordCount = 0 Then
         MsgBox "Please enter the right login name and password", vbCritical
         'count number of intrusion
         sdata = sdata + 1
         Call intrusion
      Else
         If Text1 = rs(0) And Text2 = rs(1) Then
            Command1.Enabled = True
            Command4.Enabled = True
            cmdok.Caption = "Proceed"
         Else
            sdata = sdata + 1
            'count number of intrusion
            Call intrusion
            MsgBox "Wrong Login Name or Password! Try Again", vbCritical
        End If
      End If

/*return the number of intrusion and also the date, day and month*/

Private Sub intrusion()
Dim temp, monthname, dayname

daynames  =  Array("Sunday",  "Monday",  "Tuesday",  "Wednesday",  "Thursday",
"Friday", "Saturday")
dayname = "the current day is" & daynames(Weekday(Now) - 1)
monthnames = Array("Jan", "Feb", "Mac", "Apr", "May", "Jun", "July", "Aug", "Sept",
"Oct", "Nov", "Dec")
monthname = "the current month is" & monthnames(Month(Now) - 1)

  Set db = DBEngine.Workspaces(0).OpenDatabase(App.Path & "\db.mdb")
 /*Open the File*/
  sqlstr = "Select * from intrusion_frequency  where date like '" & Date & "' "
  Set rs = db.OpenRecordset(sqlstr)

  If rs.RecordCount = 0 Then
    Set db = DBEngine.Workspaces(0).OpenDatabase(App.Path & "\db.mdb")
    sqlstr = "Select * from intrusion_frequency where date like '" & Date & "'"
    Set rs = db.OpenRecordset(sqlstr)
    /*updating new count of intrusion*/
    rs.AddNew
    rs!frequency = sdata
    rs!Date = Date
    rs!Day = daynames(Weekday(Now) - 1)
    rs!Month = monthnames(Month(Date) - 1)
    rs.Update

  Else
    temp = rs(0)
    /*open file*/
    Set db = DBEngine.Workspaces(0).OpenDatabase(App.Path & "\db.mdb")
    sqlstr = "Select * from intrusion_frequency where date like '" & Date & "'"
    Set rs = db.OpenRecordset(sqlstr)

    rs.Edit
    temp = temp + sdata
    rs!frequency = temp
    rs.Update
  End If

End Sub
```

CHAPTER 8

SYSTEM EVALUATION AND CONCLUSION

8.1 Problems Encountered And Solutions

A lot of system analyses need to be done on technologies and programming concepts before starting to develop FRIDCA System. The basic knowledge needed as a foundation in building an application of this nature involves studies in fields such as Visual Basic, information System and others. Throughout the development of FRIDCA, a few problems were encountered. However, most of them were resolved eventually. Some of the problem encountered was:

8.1.1 Lack of knowledge In the Programming Language

Due to time constraints the learning and developing process was done in parallel. Without a strong base of the language, we need to spend a lot of time looking for solutions to solve problems encountered that occurred during the development. This happened mostly in situations related to the concept of programming language that are new.

8.1.2 Slow Response Time

Some of the modules need to be able to response on minimum amount of time. If all the information input of each user is stored in database, the response time will be very slow and thus favorable. In order to speed up response time, each form's information to be filled out by the user in the credit card application process was stored in a separate table rather than having to store all information of 1 user into 1 table. It will be much easier this way and the administrator could view information much more conveniently as well.

8.1.3 Difficulty in Choosing An Appropriate Operating System

There were some difficulties in choosing the appropriate OS to host. Because of limited facilities in the faculty and problems as well as lack of resources, therefore windows 2000 Professional was used for it is considered a stable and robust OS available.

8.2 System Strength

8.2.1 User Friendly

In overall, FRIDCA could be evaluated as a simple to use application. FRIDCA provides simple, user friendly and graphical based interface for user to deal with it. Besides, sufficient instructions and guidance are provided to guide and assist users in saving their time to learn on using the system. For example, error messages will be displayed to guide users whenever invalid user inputs are encountered by the system.

8.2.2 Transparency

The system is transparent to the users, as they do not need to know where the database resides, how the systems is structured. For example, users do not need to know how to retrieve and insert records into the database. All they need to do is submit data and then view necessary information required.

8.2.3 Error Messaging

The error messages are immediately displayed after a button is clicked if the user has input in the wrong information required. Error pages or message box will be displayed to allow users to identify their errors effectively and make appropriate corrections.

8.3 System Weaknesses

There are some limitations due to time constraints, facilities, limitations and constraints of the programming language itself including.

8.3.1 Mailing Capabilities

There is no function available to mail the credit card customer's requirements or inquiries to the administrator.

8.3.2 Online System

There is no online transaction that available for user interface and FRIDCA system to be connected.

8.3.3 Help Module

There is no help module available here in FRIDCA system for new user.

8.4 Problems face during system development

During the early development of our FRIDCA system, we face problems in integrating our system because it consists of three different auditing steps. Besides that, since data mining is a very new subject so there's not much reference can be found in the library or the internet. In addition to that, we also face problems when using visual basic to connect Microsoft access database since the visual basic software that we use is not compatible with the Microsoft access that is available in our computer. Also, we face some problem with adapting the data mining algorithm with our system. However, we manage to solve our problem through guidance from the book and Mr.Teh, our supervisor.

8.5 Future Enhancements

Some of the future enhancements that should be considered to be included:

8.5.1 Software Upgrade

Database development tools used is Microsoft Access 2000. In future, a higher performance and more stable database platform such as Microsoft SQL server. In addition, we have use Visual Basic 6.0 as our programming language and to improve it, the usage of latest programming language which is, PHP would be implemented.

8.5.2 Security Enforcement

To enforce Secure Socket Layer(SSL) so that all customers identification will not be viewed by anyone else. The possibility of frauds will be reduced.

8.5.3 Online Connection

In the coming future, we will make this system web-based because most of the system nowadays can be found online. By putting it online, users can access it 24/7 and they can also check their application themselves to see if it is approved or not. And fraud can be detected anytime during the transaction. Also, we will be able to provide the email function by putting it online so that auditors can inform the users with related matters.

8.5.4 User Interface

To make the interface more realistic and professional, we would create it in way that it is applicable in the real working environment.

8.6 Achievement Of Objectives

The primary goal of this project is to evaluate the usefulness of data mining techniques in supporting auditing works. We have build intelligent system that can help business discover hidden patterns in their data. Identified high-risk and low-risk of credit card customers. Help business to understand the purchasing behavior of their key customers, detect likely credit card or insurance fraud, and predict credit application approval and detect any unauthorized activity through intrusion detection method. Besides that, we also provide a computer system protection against intrusion from intruder and hence prevent fraudulent activity.

8.7 Conclusion

Overall, the requirements of this project as determined during the system analysis phase were done eagerly. FRIDCA uses the database management system to do maintenance. Database is setup to record all the records of the customers and the administrator. For example the customer, database will record their transaction meanwhile for administrator, database will record the updating processes like edited administrator password or user and the status of the credit card application whether it is approved or not. The aim of this project is to develop a system for the use of auditors to audit all data and identify the unusual pattern and also to check frequently occur episode.

Summary

FRIDCA is an auditing organization towards the effort of reducing the time and effort of the auditor by implementing data mining technique and concept. While developing the whole system is not an easy task because as we all know, data mining is a very new subject and not many organization has implement the use of data mining in their business, but it can very still be considered as a contemporary effort to achieve the goal.

In the process of analyzing how to develop the system, insights was gain into complexities that we faced throughout the whole process. The development schedule is very important in order to get a job or task done on time. There are still much enhancements needed to improve this whole system in order to make it become more reliable and effective in a global market.

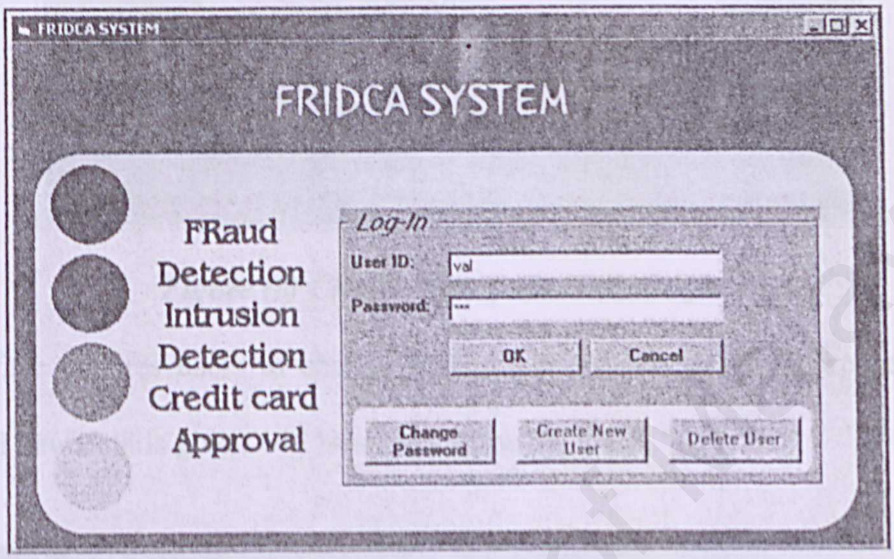
As the competitive market rising, people will tend to go ahead with using data mining technique in their business as it can effectively identify pattern. To satisfy this standard, a more detail analysis should be carried out or organize a survey and compile to be more detail and not just implementing the concept but also using the technique wholly.

With hope that FRIDCA has successful development keeping future enhancements in mind, it would be a great outcome towards the future development. The problems and experiences gained during the system development should be useful in future endeavors. It is hope that this system can provide a foundation upon which many more innovative and comprehensive system maybe built to perform multiple task and fulfill various user requirements.

User Manual

Below are the guidance for using FRIDCA system :-

The Main Interface

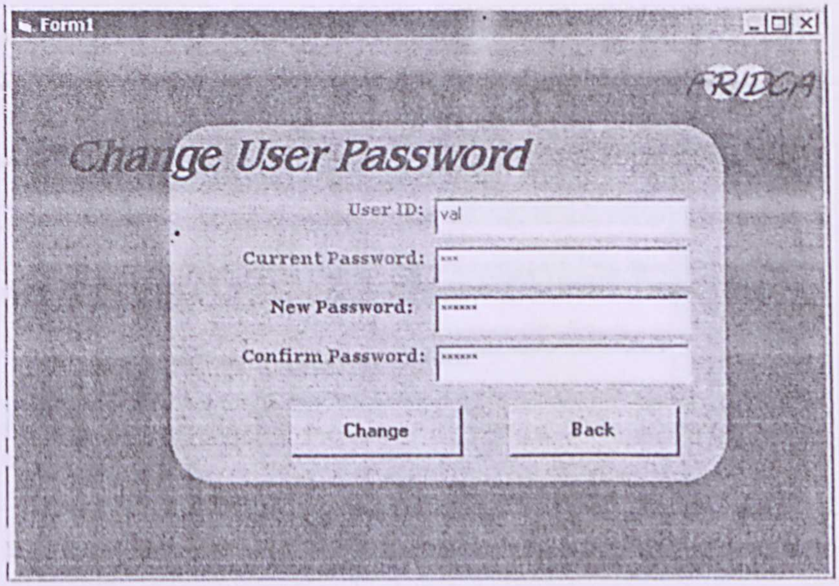


User ID: val

Password: ***

Figure (a) The interface for Login page

This is the interface that will appear as the auditors enter our system. Users that have already registered have to type in their user id and password in order to gain access into the system, after typing in the user id and password, user click on the ok button, if the user wish to change password, click on the change password and the interface below i.e. figure (b) will appear.



User ID: val

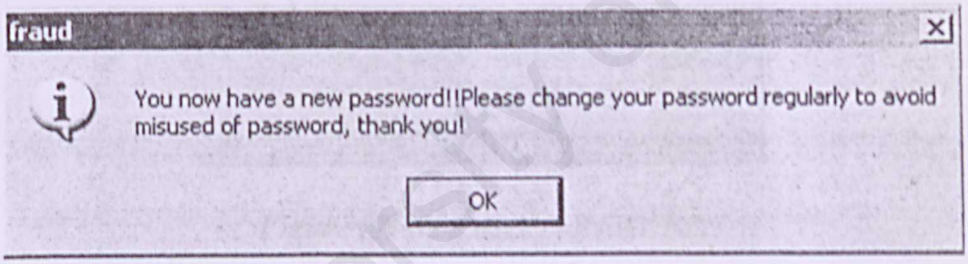
Current Password: ***

New Password: ******

Confirm Password: ******

Figure (b) Change user password interface

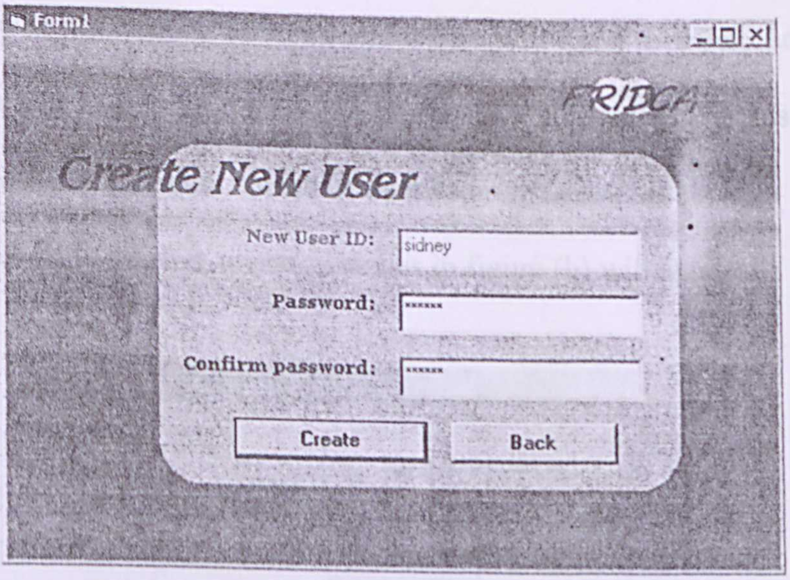
This interface enables users to change their password for security purposes, click on change and new details given will be stored in the database.



You now have a new password!!Please change your password regularly to avoid misused of password, thank you!

Figure (c) New password message box

This message box will appear after the new password has been verified.



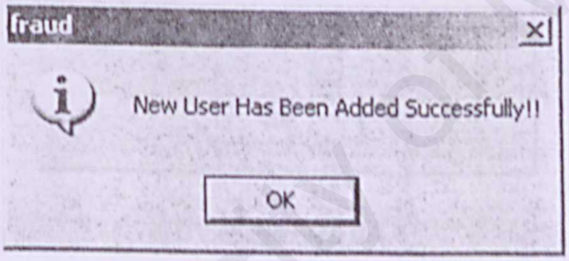
New User ID: sidney

Password: ******

Confirm password: ******

Figure (d) Create new user interface

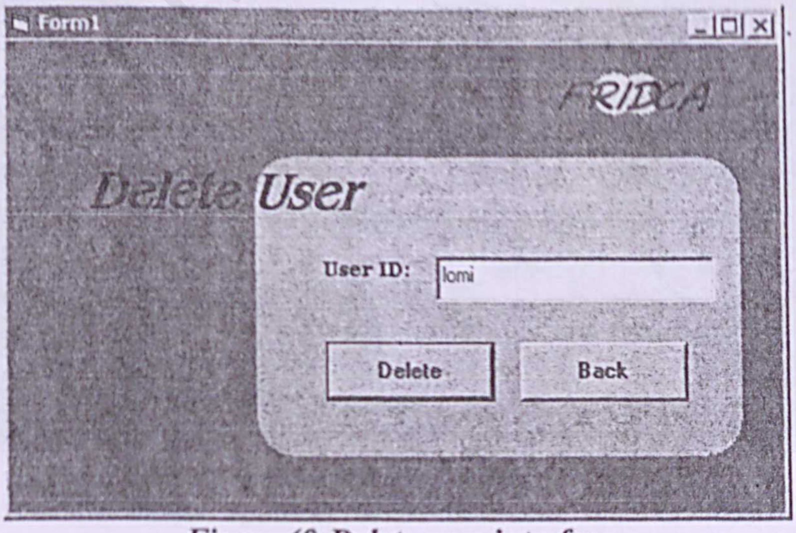
This interface provides the existing user to create new user. Click on create and new user will be added.



New User Has Been Added Successfully!!

Figure (e) New user message box

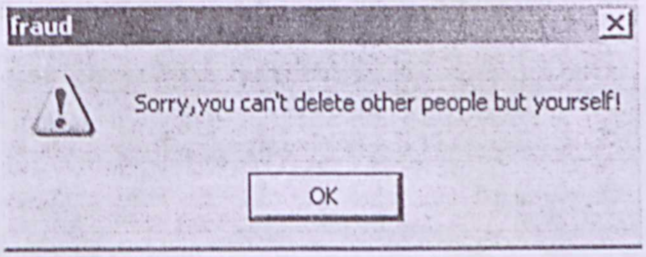
When user is added successfully this message box will appear.



User ID: lomi

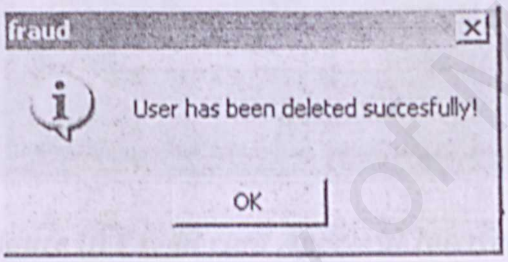
Figure (f) Delete user interface

When user is no longer using this system, he/she will need to delete herself from the admin database. So they will not be able to access to the system. The user will have to type in their own user id and not other people or the message box in figure (g) will appear. If successfully deleted, message box as in figure (h) will appear.



Sorry,you can't delete other people but yourself!

Figure(g)



User has been deleted succesfully!

Figure(h)

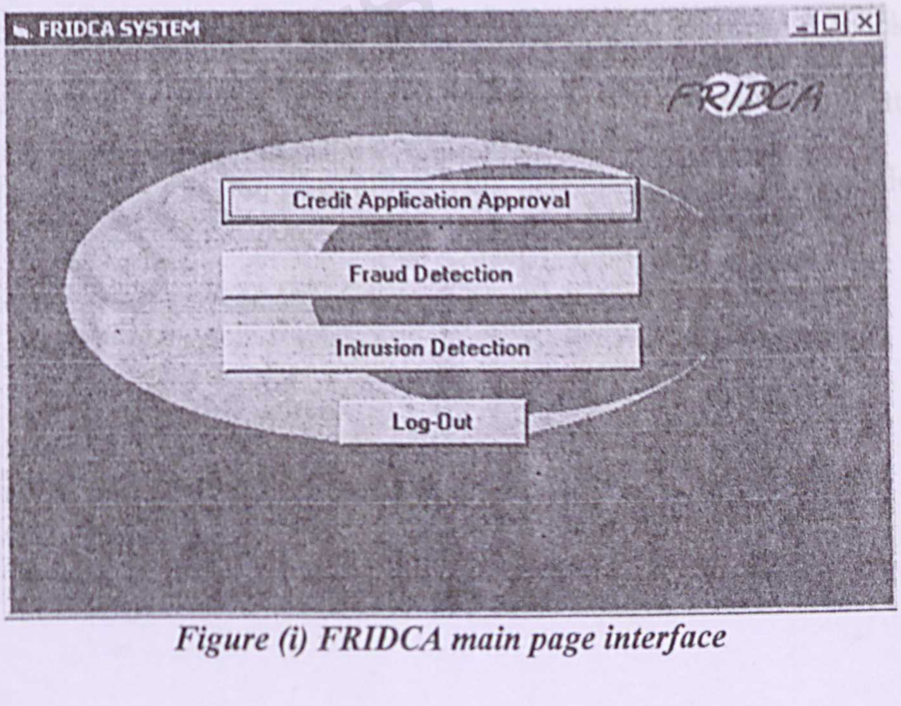


Figure (i) FRIDCA main page interface

This interface enables auditor to choose field that they want to check. And the button logout provides a logout function. When click on credit Application Approval button, the interface as in figure (j) will appear.

The Credit Card Application Approval Interface

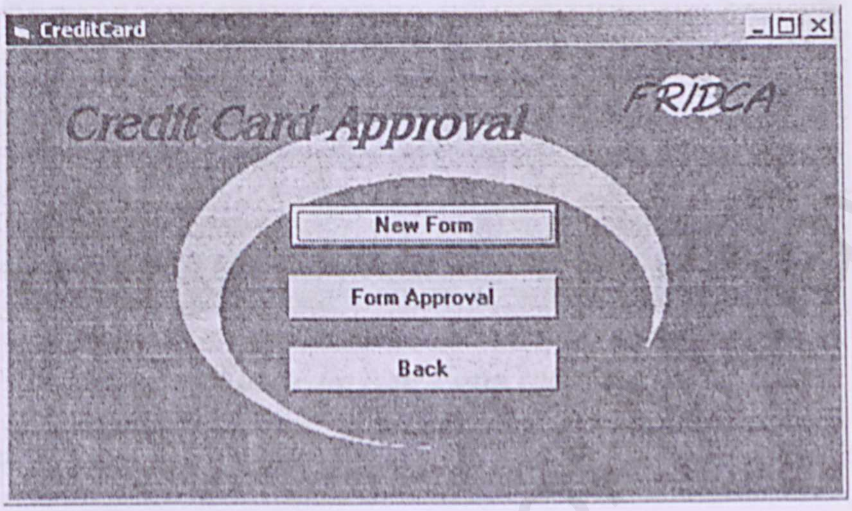
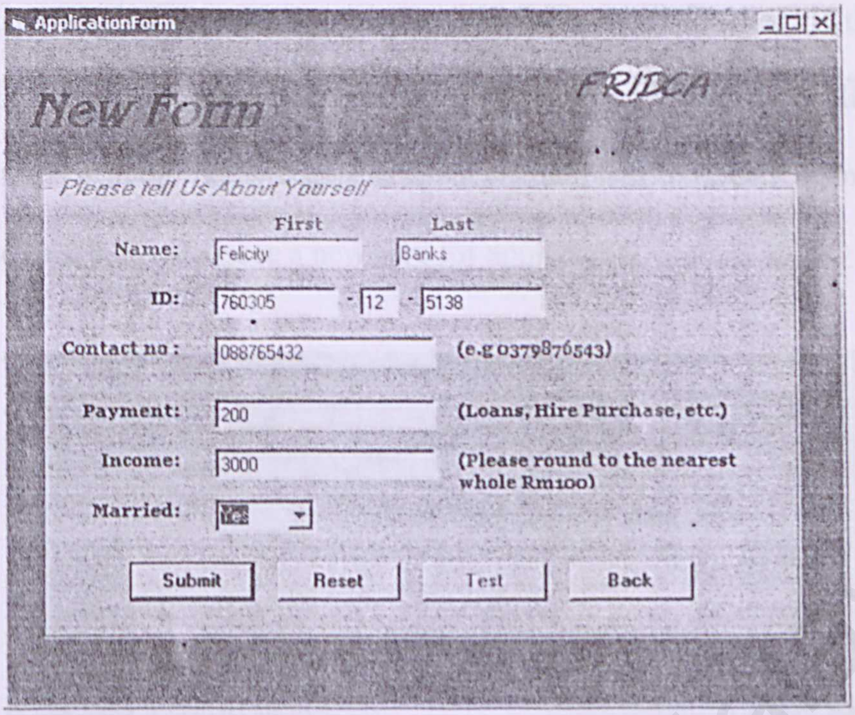


Figure (j) Credit card Approval Interface

From this interface, auditors can choose to click on New Form or Form Approval. When click on New Form the interface below as in figure (k) will appear. If click on Form Approval button, the interface of figure (n) will appear. While back button enable auditors to return to figure (i) to go to either fraud detection or intrusion detection.



Please tell Us About Yourself

	First	Last
Name:	Felicity	Banks

ID: 760305 - 12 - 5138

Contact no : 088765432 (e.g 0379876543)

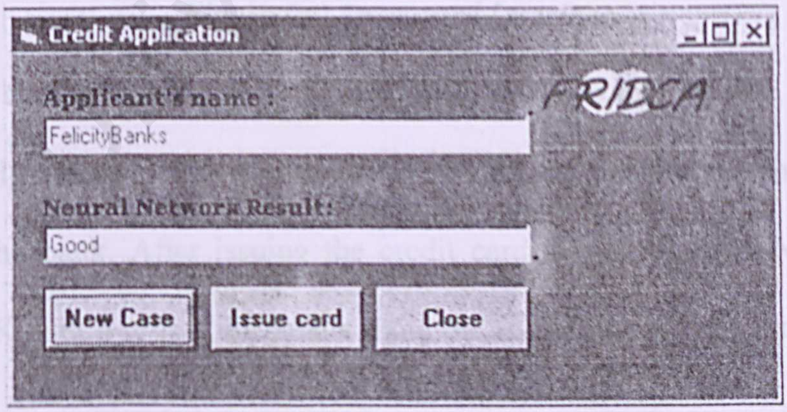
Payment: 200 (Loans, Hire Purchase, etc.)

Income: 3000 (Please round to the nearest whole Rm100)

Married: Yes

Figure (k) New Form Interface

This interface requires the auditor to input the applicant's relevant details in order to check whether the application is approved or not. After submit the form, auditor will be able to test the application whether it is 'good' or 'poor' which our system will show the result in figure (l).

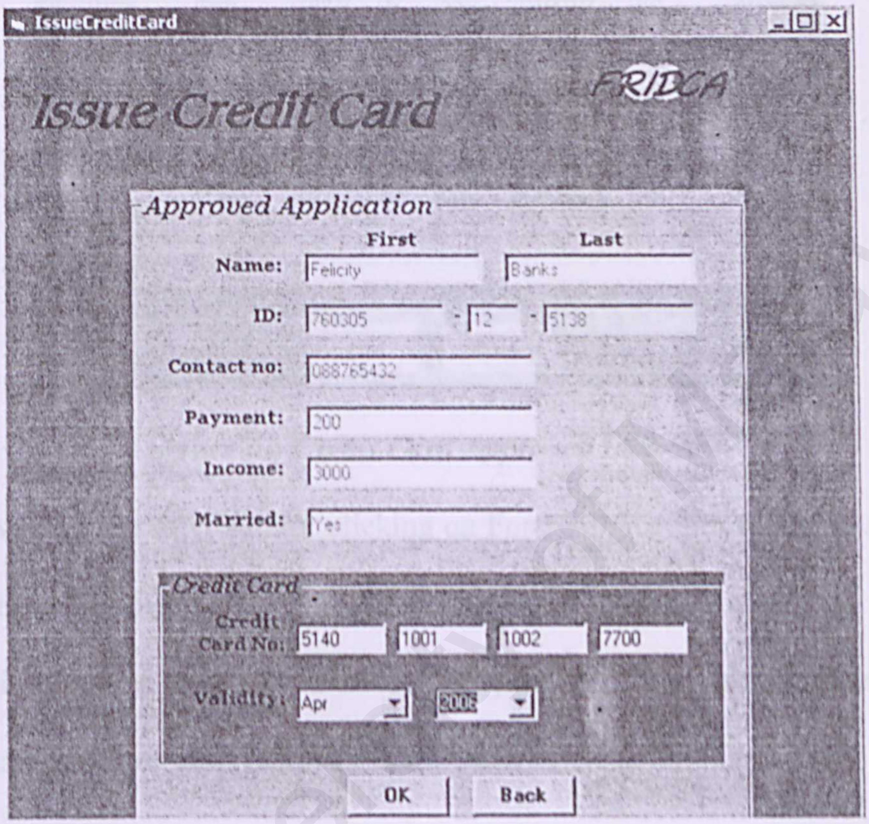


Applicant's name : FelicityBanks

Neural Network Result: Good

Figure (l) Result's interface

This is the interface, which will appear after the test button is clicked. If the applicant result is 'bad' then New Case button enable auditor to check another applicants which new form interface figure (k) will appear again. If the result is 'good' then auditor can click on Issue card button to issue a new card for applicants.



Approved Application

	First	Last
Name:	Felicity	Banks

ID: 760305 - 12 - 5138

Contact no: 088765432

Payment: 200

Income: 3000

Married: Yes

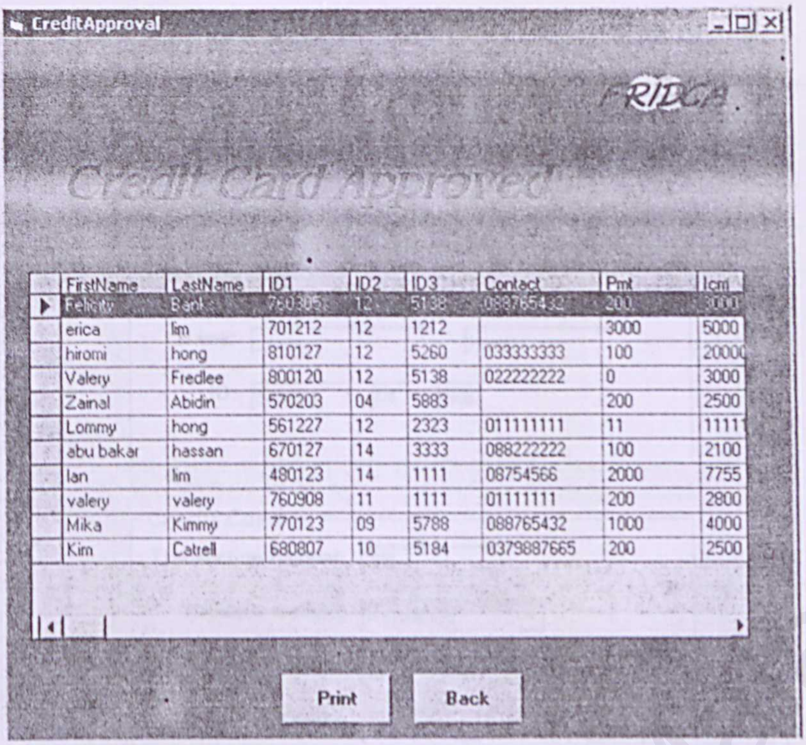
Credit Card

Credit Card No: 5140 1001 1002 7700

Validity: Apr 2008

Figure (m) Issue card Interface

After the issue card button has been clicked on the previous interface i.e. Figure (l), here the auditor will type the credit card manually as this is a confidential method, which is protected by the bank. After issuing the credit card number and the validity, auditor should click OK button so that the details about the applicants will be send to database.

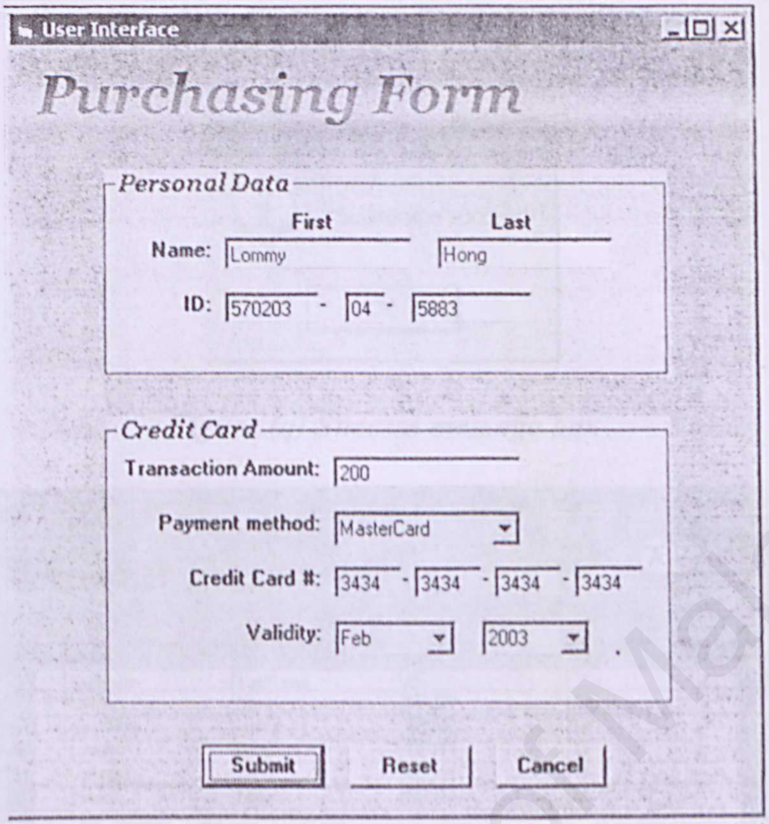


FirstName	LastName	ID1	ID2	ID3	Contact	Pmt	Icm
Felicity	Bank	760305	12	5138	088765432	200	3000
erica	lim	701212	12	1212		3000	5000
hiromi	hong	810127	12	5260	033333333	100	20000
Valery	Fredlee	800120	12	5138	022222222	0	3000
Zainal	Abidin	570203	04	5883		200	2500
Lommy	hong	561227	12	2323	011111111	11	11111
abu bakar	hassan	670127	14	3333	088222222	100	2100
Ian	lim	480123	14	1111	08754566	2000	7755
valery	valery	760908	11	1111	01111111	200	2800
Mika	Kimmy	770123	09	5788	088765432	1000	4000
Kim	Catrell	680807	10	5184	0379887665	200	2500

Figure (n) Credit Card Approved Interface

This report will appear after auditor clicking on Form Approval in Figure (j). This report shows the newly approved applicants particulars, which is useful for auditors to check the applicant's information anytime in need. The print function on this interface allows auditor to make a copy of this report for future use.

The Credit Card Fraud Detection Interface



User Interface

Purchasing Form

Personal Data

	First	Last
Name:	Lommy	Hong

ID: 570203 - 04 - 5883

Credit Card

Transaction Amount: 200

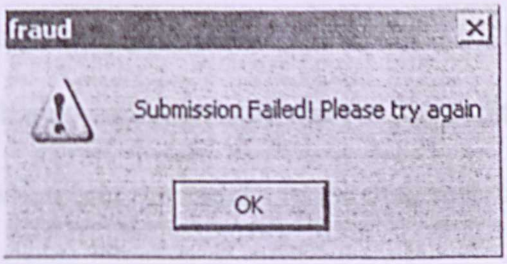
Payment method: MasterCard

Credit Card #: 3434 - 3434 - 3434 - 3434

Validity: Feb 2003

Figure (o) User's Interface –this is not part of FRIDCA system, it is merely an interface for the user.

In addition to FRIDCA, we have provided with a user interface, which is specifically for the user, example of usage is like online ticketing. Here, the user has to fill in all the main detail as it is required. After submitting, if there are any discrepancies or if the user has put in the amount, which is not in the regular pattern, the message box will appear.



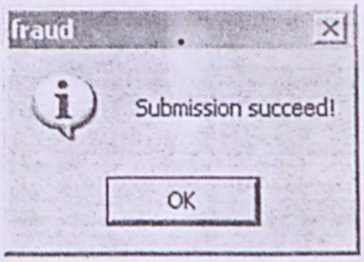
fraud

Submission Failed! Please try again

Figure (p) Submission fail message box.

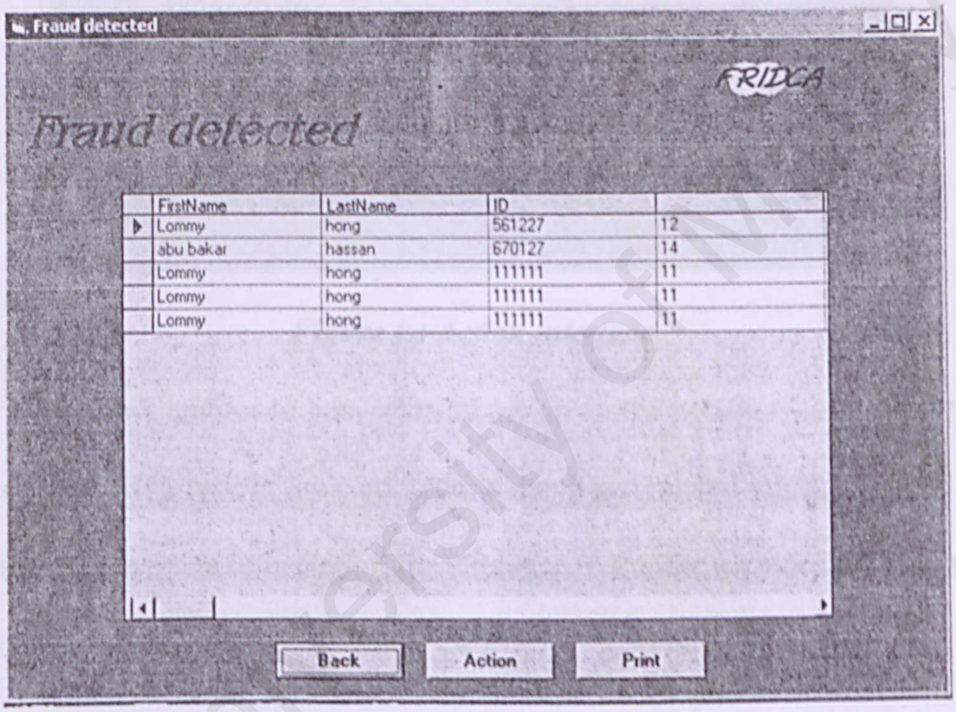
User will be able to key in their input three times before they're barred from the system.

Otherwise if everything goes well, the message box below will appear.



Submission succeed!

Figure (q) Success message box

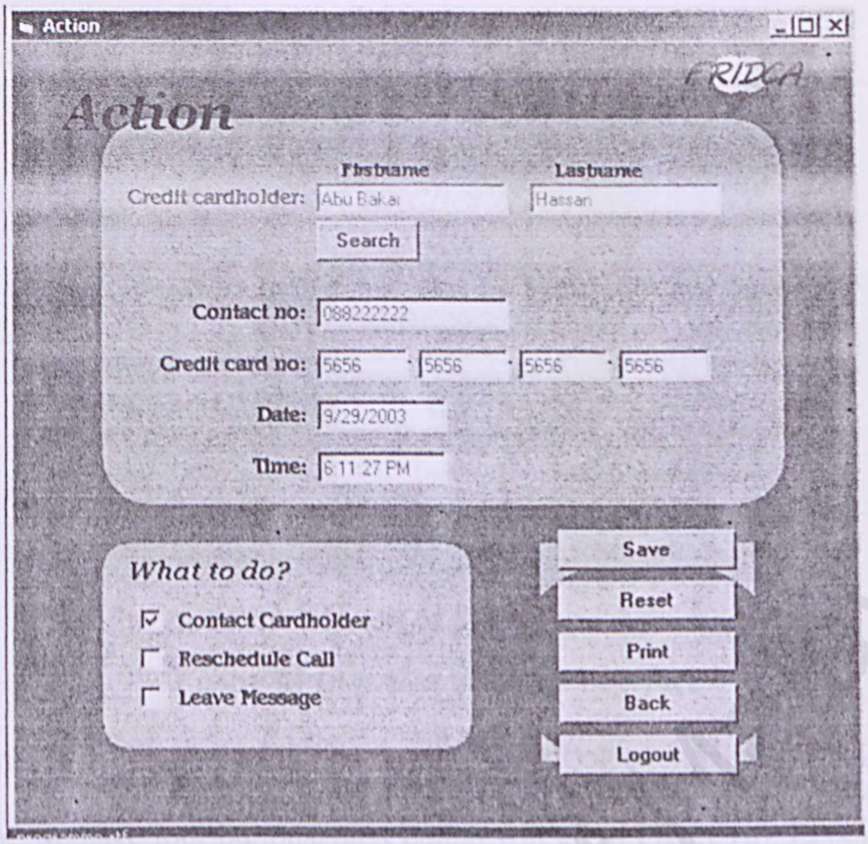


Fraud detected

FirstName	LastName	ID	
Lommy	hong	561227	12
abu bakar	hassan	670127	14
Lommy	hong	111111	11
Lommy	hong	111111	11
Lommy	hong	111111	11

Figure (r) Fraud detected interface

This is the interface which will appear after auditors click on fraud detection button in Figure (i). If there are any discrepancies or fraud detected found, this interface will display the database which also can be printed out for the auditors to use in the future to come. The action button will direct the auditor to the next interface.



	Firstname	Lastname
Credit cardholder:	Abu Bakai	Hassan

Contact no: 088222222

Credit card no: 5656 - 5656 - 5656 - 5656

Date: 9/29/2003

Time: 6:11:27 PM

What to do?

☑ Contact Cardholder

□ Reschedule Call

□ Leave Message

Figure (s) Action interface

This interface allows auditor to take relevant action on the person who is the cardholder. The auditor just has to type in the cardholder's name and related information will appear. And auditor can save this information for reference. Also, the auditor is able to print this information for future use.

The Intrusion Detection Interface

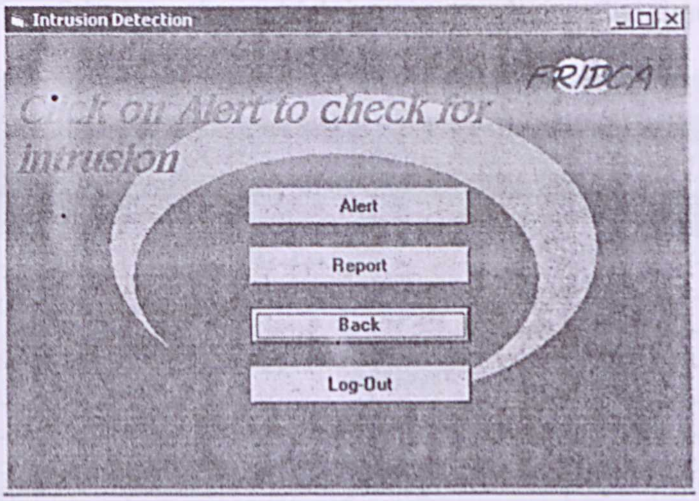
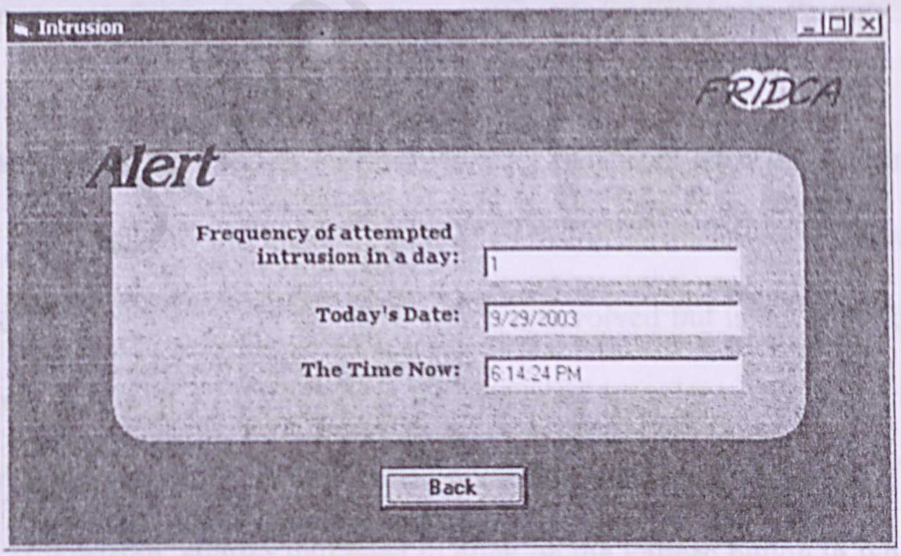


Figure (t) Intrusion detection interface

After clicking on the FRIDCA interface i.e. Intrusion Detection button, this interface will appear. Here, we have four buttons, alert button will go to Figure (u), Report button will go to Figure (v), and back button will go to previous interface which is Figure (i) and logout will logout the user automatically.



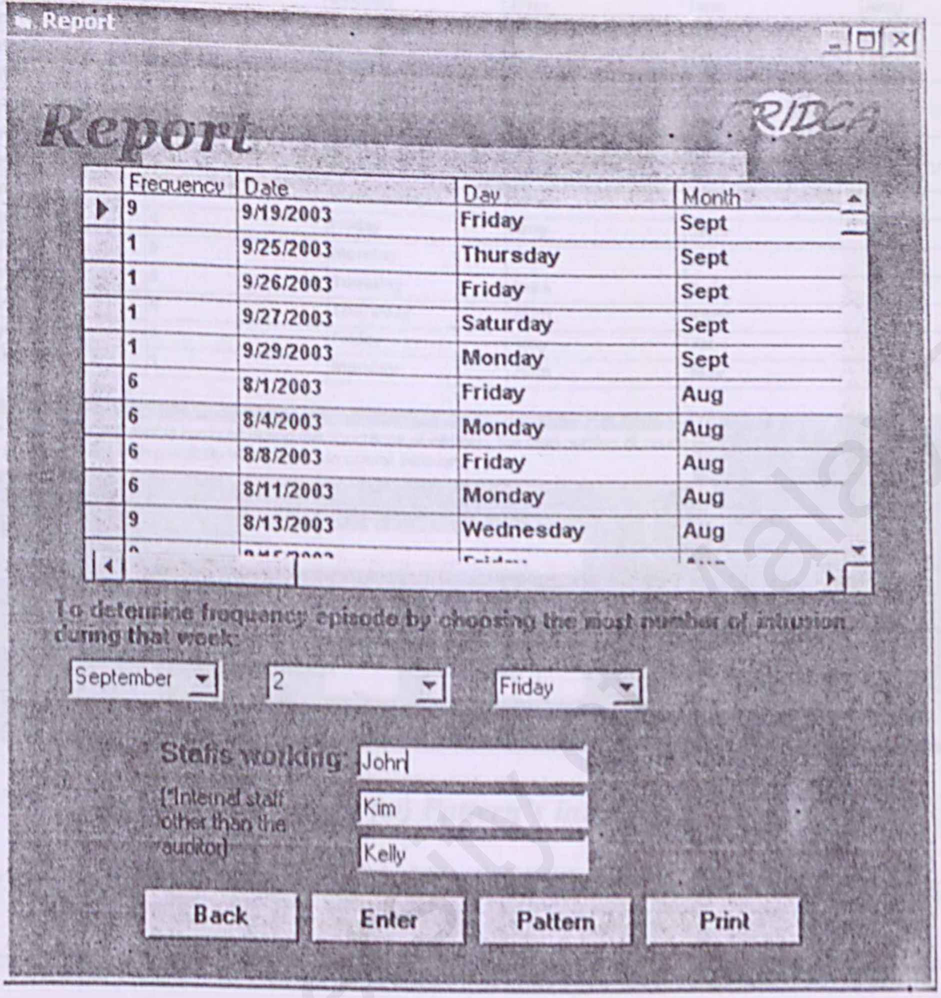
Frequency of attempted intrusion in a day: 1

Today's Date: 9/29/2003

The Time Now: 6:14:24 PM

Figure (u) Alert Interface

This page will display the frequency of intrusion for that date and time. So that auditor will have knowledge of the intrusion, which is going on during that time.



Report

Frequency	Date	Day	Month
9	9/19/2003	Friday	Sept
1	9/25/2003	Thursday	Sept
1	9/26/2003	Friday	Sept
1	9/27/2003	Saturday	Sept
1	9/29/2003	Monday	Sept
6	8/1/2003	Friday	Aug
6	8/4/2003	Monday	Aug
6	8/8/2003	Friday	Aug
6	8/11/2003	Monday	Aug
9	8/13/2003	Wednesday	Aug

To determine frequency episode by choosing the most number of intrusion during that week:

September ▾ 2 ▾ Friday ▾

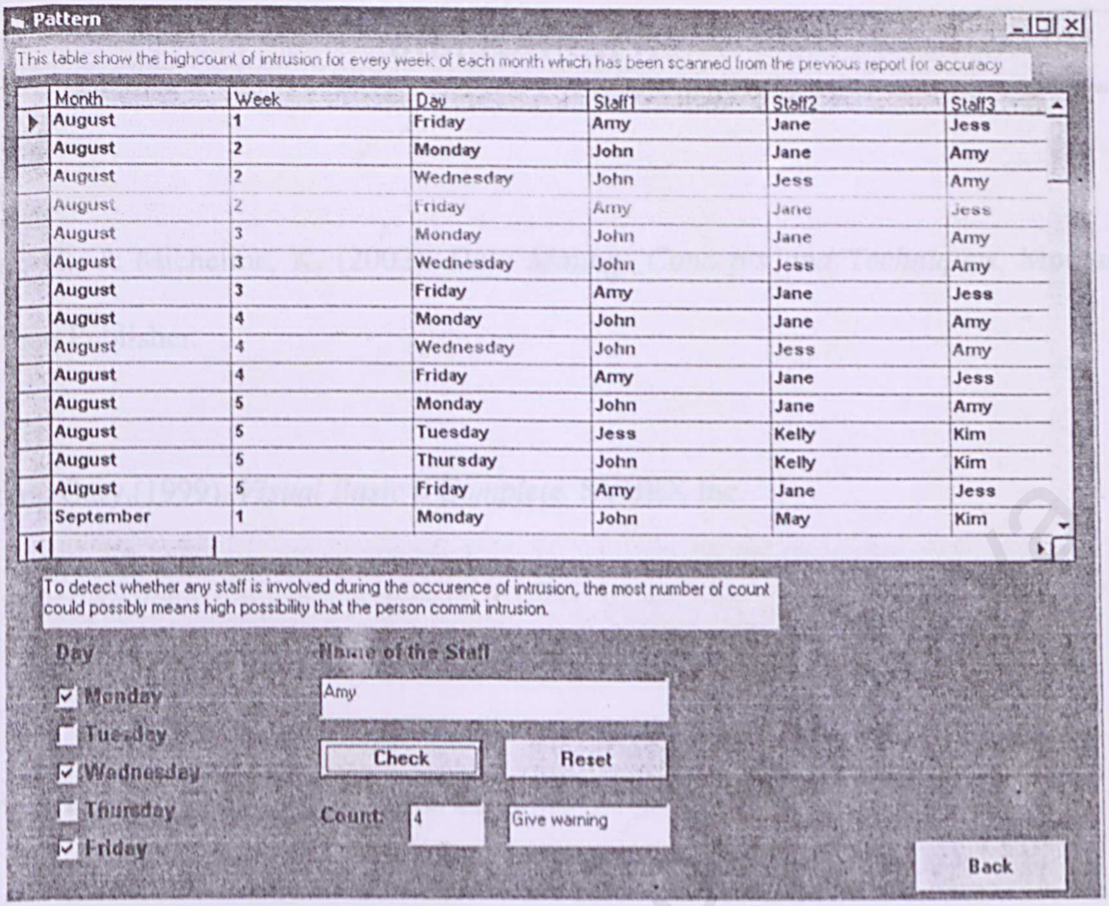
Staffs working: John

(*Internal staff other than the auditor) Kim

Kelly

Figure (v) Report Interface

This interface shows the frequency of intrusion for the stated date, day and month. Here the auditor will check for the highest count of intrusion for the days of the week of that month. And auditor will type in the staff, which is involved but is not an authorized user of this system, to check whether they're the suspected intruder or not. Then auditor will click 'enter' button in which this information will be stored in the database. To check for the pattern of intrusion, auditor will need to click on the Pattern button in which this will direct the auditor to the interface below i.e. Figure (w).



This table show the highcount of intrusion for every week of each month which has been scanned from the previous report for accuracy

Month	Week	Day	Staff1	Staff2	Staff3
August	1	Friday	Amy	Jane	Jess
August	2	Monday	John	Jane	Amy
August	2	Wednesday	John	Jess	Amy
August	2	Friday	Amy	Jane	Jess
August	3	Monday	John	Jane	Amy
August	3	Wednesday	John	Jess	Amy
August	3	Friday	Amy	Jane	Jess
August	4	Monday	John	Jane	Amy
August	4	Wednesday	John	Jess	Amy
August	4	Friday	Amy	Jane	Jess
August	5	Monday	John	Jane	Amy
August	5	Tuesday	Jess	Kelly	Kim
August	5	Thursday	John	Kelly	Kim
August	5	Friday	Amy	Jane	Jess
September	1	Monday	John	May	Kim

To detect whether any staff is involved during the occurence of intrusion, the most number of count could possibly means high possibility that the person commit intrusion.

Day

☑ Monday
☐ Tuesday
☑ Wednesday
☐ Thursday
☑ Friday

Name of the Staff: Amy

Count: 4 Give warning

Figure (w) Pattern's interface

This interface shows the week of each month and also the staffs involved during the high occurrence of intrusion. Here we use the frequency item set method, in which it will detect whether the staff happen to be there during the specified days of high count intrusion or not. Choose the days and type in the staff name, to check whether this staff is an intruder or not, click on 'check' button, it will generate the total support count of the staff being present during the specified days, the highest count would mean the staff is a possible intruder.

BIBLIOGRAPHY

Books

Jiawei, H., & Micheline, K. (2002). *Data Mining: Concepts and Techniques*. Morgan Kaufman Publisher.

Masters, Gary.(1999). *Visual Basic 6 Complete.* SYBEX Inc.

Davis, Harold.(2000). *Visual Basic For Windows*. Peachpit Press

Shari, L. P. (2001). *Software Engineering: Theory and Practice*. Prentice Hall Inc.

Agrawal, R., & Srikat, R (1995). Mining Sequnetial Pattern. *In proc. 1995 Int. Conf. Data Engineering (ICDE'95)*, pages 3-14, Taipei, Taiwan.

Harvey, M.D.,& Paul, J.D.(1999). *Java How To Program.* 3rd edn. Prentice Hall Inc.

Berson, Alex, Smith, Stephen & Kurt, Thearling (2000), Building Data Mining Applications for CRM, McGraw-Hill Companies Inc.