WXES 3182

Name : **Cheong Hon Tsong**

Metric No. : **WEK 010041**

Thesis Title : **Business Advisory System**

Supervisor : Pn. Norisma bt Idris

Moderator : Prof. Madya Dr. Roziati Zainuddin

Abstract

Business Advisory System (BAS), as the name suggest, is an advisory system in he world of business. Since the business domain is so huge, this project will only focus on the area of Human Resources Management to provide 4 major services: recruitment, selection, retrenchment and salary cut down.

Although BAS is a stand alone system, it is able to communicate with other web application through web services. It basically receives inputs either forms the user or the web application mentioned and provides advices to the user.

The major objective of BAS is for complying the company's objective, which is maximise the profit and minimize the cost. While other objectives are to reduce the heavy workload of HR staffs and help in decision making especially at the management level when recruitment and retrenchment occur.

The problem solving approach used in BAS is Case-based Reasoning (CBR). CBR is an intelligent reasoning methodology used to solve problem based on the past cases stored in a case library (Kolodner, 1993). CBR's concept is to solve a new case by retrieving a similar case from the case library and adapting the solution for the new case.

Acknowledgements

The past few months have been extremely challenging and receding time as I undertook this final year project or thesis. There have been several people, who have shown endless interest and effort in making this thesis a success, whom that I would like to thank with all my heart. They are:

...Pn. Norisma bt Idris, for introducing me to the world of Case-based Reasoning as well as the world of business. Not to be forgotten her endless advices and supervision to ensure a successful completion of this thesis.

...Prof. Madya Dr. Roziati Zainuddin, for a fair evaluation on this project.

...Miss Cheong Wai Yan, my sister, who study in Financial Accounting, for her endless help and explanation on how this system should perform.

...family and friends, for their constant love and support.

Cheong Hon Tsong

18th, February 2004

Contents

List of Table

List of Figure

Chapter 1: Introduction

Chapter 1: Introduction

1.1 Overview

Nowadays, every well established company owns a Human Resource Department. The role played by the HR department is to calculate the payroll of each employee, select and recruit new employee, keep tracks of employee's data, supervise the performance of the workers, and even making the decision to retrench the employee.

Therefore, the workload of the HR department is very heavy. Just imagine a small department with around dozens of people but needs to handle the resources of a few hundreds or even thousands of people.

The aim of one company is to maximise the profits and minimise the cost. Hence, the company will not allow the excess of human resources or not fully utilise the resources. For this reason, you can see that everyday, one person quits from a company and another person is being hired by the company. This is just a normal phenomenal in the society nowadays.

Therefore, the HR department really needs a comprehensive computer advisory system in order to reduce the heavy workload that should be managed everyday.

1.2 Project Motivation

Since 1997, the economy abruptly collapsed and many companies had begun to retrench their employees. During that period, we can say that almost every hour there will be employees losing their job.

After a few years until today, the world's economy had recovered substantially. Many industries need to hire people to support their large quantity of jobs. Therefore, the Human Resource Department plays an important role either in recruitment or retrenchment.

Although the technology has been expanding in a fast pace these days, there is still a lack of comprehensiveness in the Business Advisory System to provide help in recruitment, selection and retrenchment. Thus, the development of this system is to assist the HR Manager decision making.

Moreover, it was found out that the HR staffs always feel revolted when the company needs to retrench or cut down the salary.

Even though there is an Employee Management System (EMS) or even Human Resources Management System (HRMS) to help calculate the payroll and keep track of employee's record, there is seldom an advisory system to help integrate the entire system in the market for the HR.

1.3 Project Definition

This project is mainly for the Human Resources Department in helping them to reduce their heavy workload. Therefore, this project is made to provide advices to the HR department especially the management level regarding the tasks they perform and the problems that they face.

If the company needs to recruit new employees, they will certainly receive a lot of application forms from many applicants. After that, the HR department which only has slightly more than a dozen of staffs may probably need to analyze a few dozens or even hundreds applications in order to abandon the disqualified applications.

With the objective to solve this problem, the Business Advisory System requires that a user satisfies certain requirements to be a qualified applicant. Then, the list of 10 or 20 of the best qualified applicants will be selected for the interview. This is indeed a hard work for the HR staffs. It is because they have to analyze the application forms before making a decision to select the best applicants for an interview based on their professions, experiences and knowledge. It is impossible for the company to interview all the qualified applicants due to the expensive cost and time constraints. Thus, this system will help save the company's cost and the company does not have to hire extra staffs.

Besides the selection and recruitment, the Business Advisory System can also helps in cutting down the salary expenses. The company's salary expenses' budget is set into the system and then the system will provide an appropriate advice on how many percent of employees' salary should be cut down.

If the company is really facing a serious financial problem or the company's financial situation has already incurred a deficit amount, then the company needs to retrench their employees. At this moment, the HR manager will always face problems not knowing which employee should be retrenched. However, with this system; based on the performance of the employees and also other factors, a list of the employees to be retrenched is produced.

1.4 Scope

The scope of this project is to build up a business advisory system, which helps to reduce the workload of Human Resource Department, especially the management level.

This project mainly solves the problems faced by each company's HR department. This system focuses on:

- Producing a list of suitable candidates based on the applicant's qualification.
- Suggesting the minimum requirement when recruiting a new employee.
- Giving advices on salary cut down to reduce expenses.
- Giving advices on how many employees and which employee to be retrenched.

1.5 Limitation

The limitations of Business Advisory System are:

- Only communicates in English.
- Still needs the HR Manager to conduct the interview and other recruitment processes.

1.6 System Objective

This Business Advisory System is basically built to comply with the company's objective, which is to maximise the profit and minimize the cost. It helps the Human Resources Department to provide useful advices in decision making during selection, recruitment, retrenchment and salary cut down.

The objectives aimed to achieve are as follows:

- **Reduce the workload of the HR Staffs.**

  Application form analysis process will be taken over by the system, thus the HR Staffs will not need to spend their time on the particular process. As a result, their heavy workload on processing hundreds of application form is significantly reduced.

- **Reduce the human factor during retrenchment and salary cut down process.**

  This system will carry out the retrenchment process base on the employee's performance and suggest who will be retrenched. Since no human will be involve throughout the whole retrenchment process, the retrenchment will be carried out in a proper manner under non-bias and non-discretionary condition.

- **Reduce the company's expenses.**

  The company may not need extra staffs to carry out the selection process. In addition, the user's company will not need to spend a large amount of money to hire a special management company that helps in the retrenchment process. Thus, the company's expenses are significantly reduced.

- **Reduce the lost of quality employee.**

  Normally a company will need at least one month to process hundreds of the application forms and the selection process. However, some quality applicants might already obtain a job elsewhere. Hence, a more effective way of the selection processing is needed to reduce the lost of quality employee.

- **Increase the efficiency of recruitment process.**

  Provided with a more effective way of selection process, the further processes such as interviews is able to be carried out as soon as the list of selected applicants is ready.

- **Fast recovery of the company's financial problem.**

  As the system is able to give advices for the retrenchment and salary cut down process within a short period of time, proper actions are able to be carried out swiftly.

1.7 Project Milestone

In general, the project milestone for developing the Business Advisory System is divided into a few phases:

1. Literatures Review phase.
2. Requirement Analysis Phase.
3. Design Phase.
4. Coding Phase.
5. Testing and Implementation.
6. System Documentation.

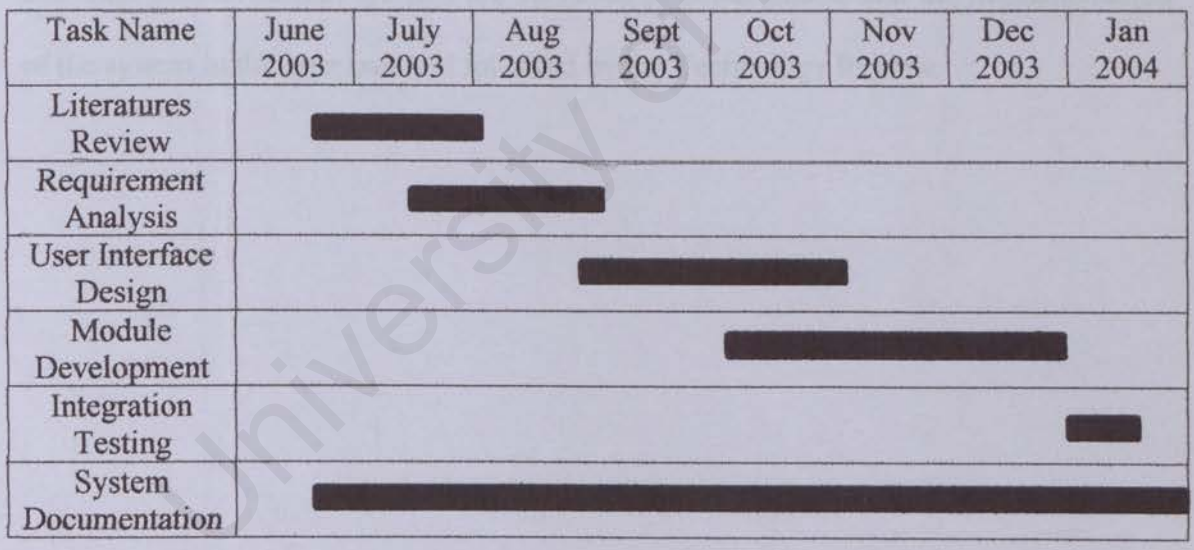


Figure 1.1: The Business Advisory System development time line.

1.8 Chapter Summary

This chapter focuses mainly on the introduction of this project. A brief introduction and definition are stated in the first part of this chapter, which is the Project Overview. Apart from that, relevant information and topics are also being discussed consequentially. Topics included are Project Motivation, Project Definition, Scope, Limitation, System Objective and Project Milestone. The research and development of this proposed system will take about 8 months.

The next chapter, Literature Review will be carried out by introducing what is Artificial Intelligence (AI) and some of its development techniques such as Expert System and Case-based Reasoning together with the comparison between Rule-based and Case-based. Current systems are surveyed to better understand the implementation of the system in the later part and followed by the Technology Review.

Chapter 2: Literature Review

Chapter 2: Literature Review

2.1 Artificial Intelligence

2.1.1 Overview of Artificial Intelligence

Artificial Intelligence (AI) can be defined as a field of study in computer science that pursues the goal of making a computer reason in a manner similar to human intelligence (Luger, 2002). AI is a broad field, and means different things to different people. It is concerned with getting computers to do tasks that require human intelligence. However, it is difficult to define intelligence.

The major goal of AI is to make computer more useful for human. This can be achieved by producing computer programs that assist humans in decision making, intelligence information search or simply by making computers easier to use with natural language interfaces. The other reason of why people want to automate human intelligence is to better understand human intelligence. This is because by building an intelligence computer system will require us to understand how human capture, organize and use knowledge during problem solving.

2.1.2 Application of Artificial Intelligence

"The Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform."

By Ada Byron, Countess of Lovelace

Artificial Intelligence (AI) is widely used in various application areas due to its inherent flexibility in adapting to any situation especially in problem domain that require qualitative reasoning rather than arithmetic calculations for their solution. The major applications of AI in computing are:

- **Games** – Much of the early research in state space search was done using common board game. Until now, AI is commonly used to simulate an opponent in games.
- **Expert System** – AI programs that use to replace or assist an expert in a particular domain. It will normally act as an intelligence advisor or consultants.
- **Natural Language Processing** – AI programs that understand and generate human language when communicating with users.
- **Pattern Recognition** – AI program that is able to analyze and evaluate visual information (image) in order to identify the particular object.
- **Robotic** – An effort to design robots that mimic the physical capability of human being to perform their tasks with some degree of flexibility and responsiveness to outside world.
- **Interactive computer-based training** – An interactive educational or training program that adjust its tutoring methods according to the user's progress.

2.2 Expert System

2.2.1 Introduction

"An Expert System is a computer system that encapsulates specialist knowledge about a particular of expertise and is capable of making intelligence decisions within that domain."

By Richard Forsyth, March 1984

An Expert System, for example, uses knowledge specific to a problem domain to provide "expert quality" performance in that application area. Generally, expert system emulates the human expert's methodology and performance by acquire the expert's knowledge that is gained through problem-solving experience. As with skilled humans, expert systems tend to be specialists, focusing in a narrow set of problems. Also, like humans, their knowledge is both theoretical and practical.

The Expert Systems bandwagon was given a push start by the Japanese Fifth Generation initiative (JIPDEC, 1981) and received further momentum from the various western responses to that plan (e.g. Alvey, 1982). They are built to solve a wide range of problems in domains such as medicine, mathematics, engineering, chemistry, geology, computer science, business, law, defence and education. These programs address a variety of problems; the following list, from Waterman (1986), is a useful summary of general Expert System problem categories.

- *Interpretation* – forming high-level conclusions from collections of raw data.
- *Prediction* – projecting probable consequences of given situation.

- *Diagnosis* – determining the cause of malfunctions in complex situation based on observable symptoms.
- *Design* – finding a configuration of system components that meets performance goals while satisfying a set of design constrains.
- *Planning* – devising a sequence of actions that will achieve set of goals given certain starting conditions and run-time constrains.
- *Monitoring* – comparing a system's observed behaviour to its expected behaviour.
- *Instruction* – assisting in the education process in technical domains.
- *Control* – governing the behaviour of a complex environment.

2.2.2 The Architecture of Expert Systems

Most people know that Expert Systems consist of a knowledge base (to store expertise) and an inference engine (to put it to work). But, as a matter of fact, there are four essential components to fully fledged Expert Systems (Forsyth, 1984):

1. the knowledge base
2. the inference engine
3. the knowledge-acquisition module
4. the explanatory interface

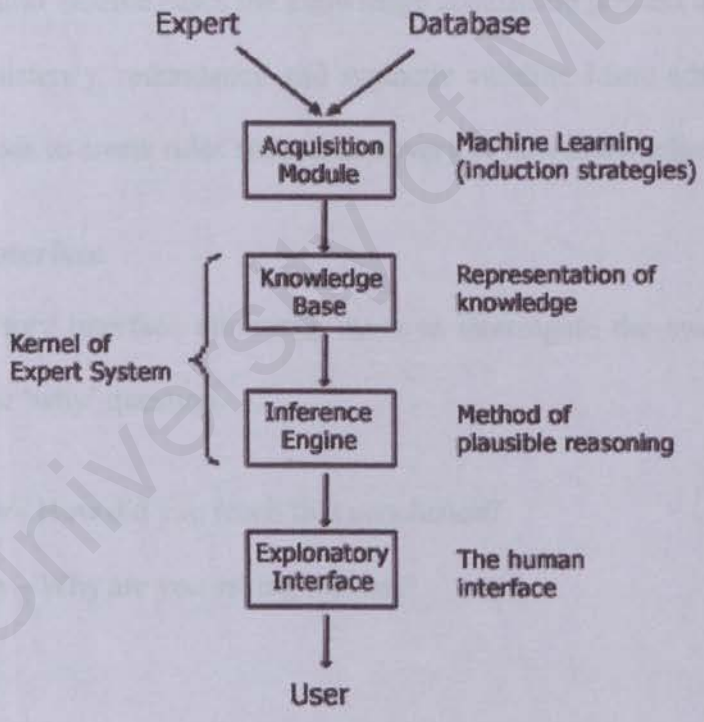


Figure 2.1: Expert System's Model (Forsyth, 1984)

Knowledge Base

Knowledge base consists of information structures for encoding expertise. In particular it is more active than database. That is, it contains rules for deducing facts that are not stored explicitly.

Inference Engine

Inference engine uses the facts and rules in the knowledge base to derive new conclusion leading to a recommendation or diagnosis. It typically follows one of the two top-level reasoning strategies – *forward chaining* or *backward chaining*.

Acquisition Module

Acquisition module eases the knowledge acquisition process by testing proposed rules for inconsistency, redundancy and syntactic validity. More advanced system use inductive methods to create rules such as discovery of new knowledge.

Explanatory Interface

Explanatory interface allow the users to interrogate the system. Normally by posting 'how' or 'why' question.

- **How -** How did you reach that conclusion?
- **Why -** Why are you asking me that?

2.2.3 Rule-based Expert Systems

Rule-based Expert Systems can be defined as a computer program that processes problem-specific information contained in the working memory with a set of rules contained in the knowledge base, using an interface engine to infer new information (Durkin, 1994).

The idea behind the Rule-based Expert Systems is simple but effective. It offers the argument that humans solve problems by combining problem-specific information contained in their short-term memory with a set of rules representing problem solving knowledge contained in their long-term memory. Rule-based Expert Systems consist of three main components as illustrated in figure 2.2.

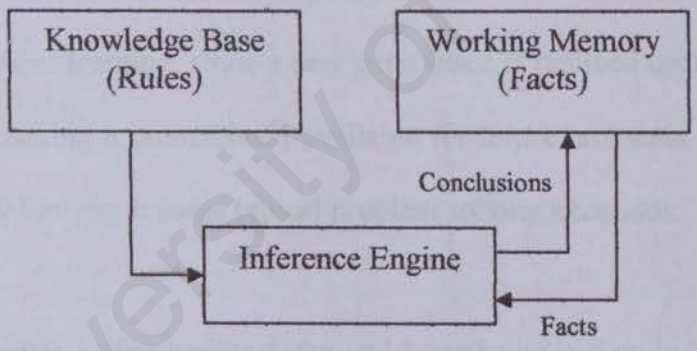


Figure 2.2: Rule-based Expert System model (Durkin, 1994)

- **Knowledge base** – models a human long-term memory as a set of rules.
- **Working memory** – models a human's short-term memory and contains problem facts both entered and inferred by the firing of the rules.
- **Inference Engine** – models human reasoning by combining problem facts contained in the working memory with rules contained in the knowledge base to infer new knowledge.

2.3 Case-Based Reasoning

2.3.1 Introduction

"Reasoning is a process of reaching conclusion by using one's reason"

~ Oxford Advanced Learner's Dictionary

Case-based reasoning (CBR) is a recent approach to problem solving and learning that has got a lot of attention over the last few years. The basic idea of CBR is to adapt solutions that have been used to solve old problem for used in solving new problem. It uses an explicit database of problem solution to address new problem-solving situation. These solutions may be collected from human experts through the knowledge engineering/acquisition process or may reflect the results of previous search-based successes or failures. A second important of CBR is that CBR is an approach to incremental, sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems. For instance, let us illustrate this by looking at some typical problem solving situations:

- A physician - after having examined a particular patient in his office - gets a reminding to a patient that he treated two weeks ago. Assuming that the reminding was caused by a similarity of important symptoms (and not the patient's hair-color, say), the physician uses the diagnosis and treatment of the previous patient to determine the disease and treatment for the patient in front of him.

- A drilling engineer, who have experienced two dramatic blow out situations, is quickly reminded of one of these situations (or both) when the combination of critical measurements matches those of a blow out case. In particular, he may get a reminding to a mistake he made during a previous blow-out, and use this to avoid repeating the error once again.

- A financial consultant working on a difficult credit decision task, by using a reminding to a previous case, which involved a company in similar trouble as the current one, to recommend that the loan application should be refused.

2.3.2 The Architecture of Case-Based Reasoning

There are two main models in the CBR:

- A process model of the CBR cycle
- A task-method structure for case-based reasoning

These two models are complementary and represent two views on case-based reasoning. The first is a dynamic model that identifies the main sub-processes of a CBR cycle, their interdependencies and products. The second is a task-oriented view, where a task decomposition and related problem solving methods are described.

The CBR cycle

At the highest level of generality, a general CBR cycle may be described by the following four processes:

1. **RETRIEVE** the most similar case or cases
2. **REUSE** the information and knowledge in that case to solve the problem
3. **REVISE** the proposed solution
4. **RETAIN** the parts of this experience likely to be useful for future problem solving

A new problem is solved by *retrieving* one or more previously experienced cases, *reusing* the case in one way or another, *revising* the solution based on reusing a previous case, and *retaining* the new experience by incorporating it into the existing

knowledge-base (case-base). The four processes each involve a number of more specific steps, which will be described in the task model. In figure 2.3, this cycle is illustrated.

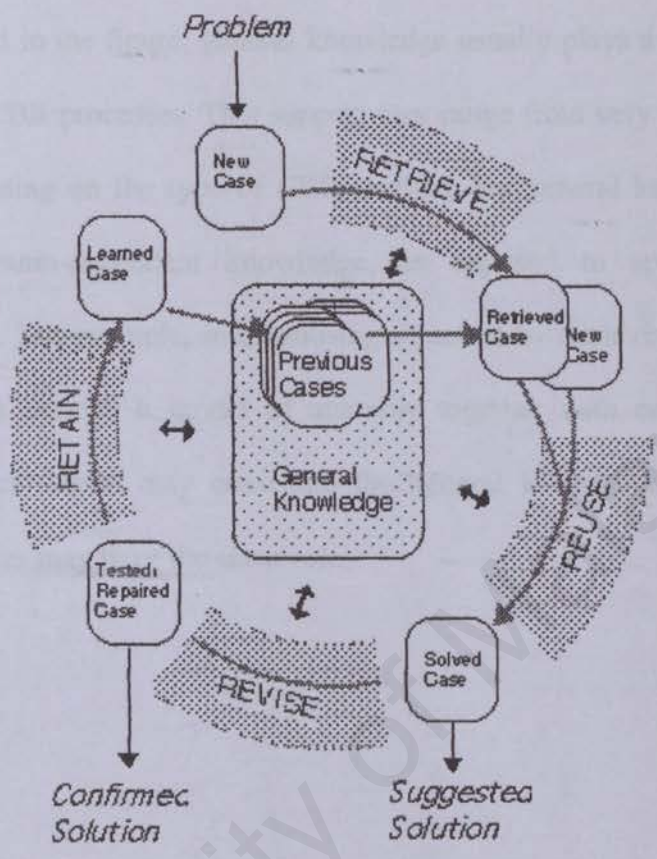


Figure 2.3: The CBR Cycle (Aamodt & Plaza, 1994)

An initial description of a problem (top of figure) defines a *new case*. This new case is used to RETRIEVE a case from the collection of *previous cases*. The *retrieved case* is combined with the new case - through REUSE - into a *solved case*, i.e. a proposed solution to the initial problem. Through the REVISE process this solution is tested for success, e.g. by being applied to the real world environment or evaluated by a teacher, and repaired if failed. During RETAIN, useful experience is retained for future

reuse, and the case base is updated by a new *learned case*, or by modification of some existing cases.

As indicated in the figure, general knowledge usually plays a part in this cycle, by supporting the CBR processes. This support may range from very weak (or none) to very strong, depending on the type of CBR method. By general knowledge we here mean general domain-dependent knowledge, as opposed to specific knowledge embodied by cases. For example, in diagnosing a patient by retrieving and reusing the case of a previous patient, a model of anatomy together with causal relationships between pathological states may constitute the general knowledge used by a CBR system. A set of rules may have the same role.

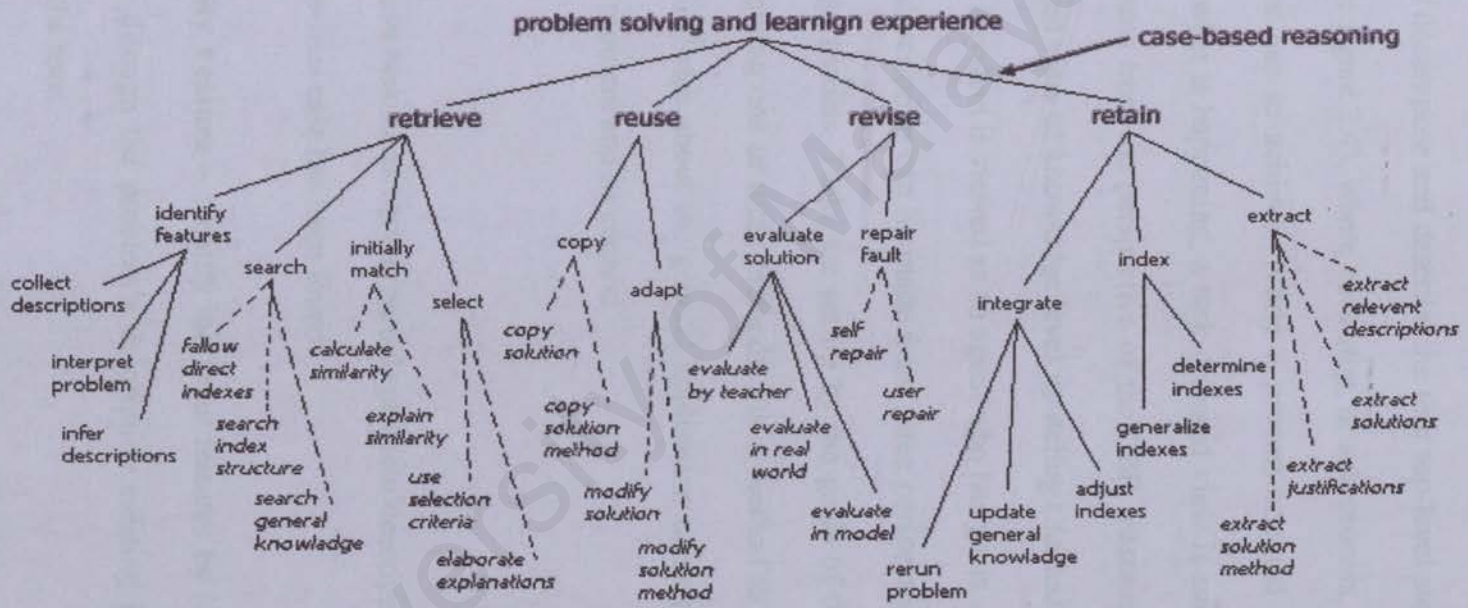


Figure 2.4: Task-model decomposition of CBR (Aamodt & Plaza, 1994)

The Task-method Structure

To further decompose and describe the four top-level steps, we switch to a task-oriented view (see figure 2.4), where each step, or sub-process, is viewed as a task that the CBR reasoner has to achieve. While a process-oriented view enables a global, external view to what is happening, a task oriented view is suitable for describing the detailed mechanisms from the perspective of the CBR reasoner itself. This is coherent with a task-oriented view of knowledge level modeling (*Aamodt & Plaza, 1994*). At the knowledge level, a system is viewed as an agent who has goals, and means to achieve its goals. A system description can be made from three perspectives: Tasks, methods and domain knowledge models. Tasks are set up by the goals of the system, and a task is performed by applying one or more methods. For a method to be able to accomplish a task, it needs knowledge about the general application domain as well as information about the current problem and its context.

Case Retrieve

The Retrieve task starts with a (partial) problem description, and ends when a best matching previous case has been found.

- **Identify Feature** – Identify the input features by 'understand' the problem either through the problem's description existing in the system or asking from the user.

- **Initially Match** – Finding a set of matching cases by using the problem descriptors (input features) as indexes to the case memory in a direct or indirect way.
- **Select** – Select the best match case base on the degree of initially match.

Case Reuse

The reuse of the retrieved case solution in the context of the new case focuses on two aspects: The differences among the past and the current case and what part of a retrieved case can be transferred to the new case.

- **Copy** – Copy the solution class of the retrieve case to the new case as its solution class.
- **Adapt** – Adapt the reuse past cases to the new case by reuse the past case solution (transformational reuse) or reuse the past method that constructed the solution (derivational reuse).

Case Revision

When a case solution generated by the reuse phase is not correct, an opportunity for learning from failure arises. This phase is called case revision.

- **Evaluate Solution** - The evaluation task takes the result from applying the solution in the real environment (asking a teacher or performing the task in the real world).
- **Repair Fault** - Case repair involves detecting the errors of the current solution and retrieving or generating explanations for them.

Case Retain - Learning

This is the process of incorporating what is useful to retain from the new problem solving episode into the existing knowledge. The learning from success or failure of the proposed solution is triggered by the outcome of the evaluation and possible repair.

- **Extract** - In CBR the case base is updated no matter how the problem was solved. If failure occurs, information from the Revise task may also be extracted and retained, either as separate failure cases or within total-problem cases.
- **Index** - The 'indexing problem' is a central and much focused problem in case-based reasoning. It amounts to deciding what type of indexes to use for future retrieval, and how to structure the search space of indexes.
- **Integrate** - This is the final step of updating the knowledge base with new case knowledge. If no new case and index set has been constructed, it is the main step of Retain.

2.3.3 Conclusion

As a conclusion, case-based reasoning (CBR) puts forward a paradigmatic way to attack AI issues, namely problem solving, learning, usage of general and specific knowledge, combining different reasoning methods, etc. In particular we have seen that CBR emphasizes problem solving and learning as two sides of the same coin: problem solving uses the results of past learning episodes while problem solving provides the backbone of the experience from which learning advances.

Further more, CBR has proved to be an extremely effective approach in complex cases (Kolonder, 1993). According to Riesbeck and Schank (1989), the basic justification for the use of this approach is that human thinking does not use logic or reasoning from first principles. It is basically a processing of the right information retrieved at the right time. So the central problem is the identification of pertinent information whenever needed. This can be done in CBR.

2.4 Rule-based Reasoning verses Case-based Reasoning

Table 2.1: Comparison between Rule-based Reasoning and Case-based Reasoning

Criterion	Rule-based Reasoning	Case-based Reasoning
Knowledge Unit	Rule	Case
Granularity	Fine	Coarse
Knowledge acquisition unit	Rules, hierarchies	Cases, hierarchies
Explanation mechanism	Backtrack of rule firings	Precedent cases
Characteristic output	Answer and confidence measure	Answer and precedent cases
Knowledge transfer across problems	High if backtracking; low if deterministic	Low
Speed as a function of knowledge base size	Exponential if backtracking; linear if deterministic	Logarithmic if index tree is balanced
Domain requirements	Domain vocabulary Good set of inference rules Either few rules or rules apply sequentially Domain mostly obeys rules	Domain vocabulary Database of example cases Stability (a modified good solution is probably still good) Many exceptions to rules
Advantages	Flexible use of knowledge Potentially optimal answer	Rapid response Rapid knowledge acquisition Explanation by examples
Disadvantages	Computationally expensive Long development time Black-box answers	Suboptimal solutions Redundant knowledge base

2.5 Existing System Review

There are lot of Human Resource Management like system, which include Employee Management System, Employee Performance System, Payroll System, Recruitment Management System, and so on. However, the market still lack of an advisory system that included with the functionality of select, recruit, retrench and cut down. Thus, a few systems that have some similarity to this project are chosen to be reviewed.

2.5.1 Case Study 1

Descriptions Now version 5.1.2 – KnowledgePoint (http://www.knowledgepoint.com)

Currently this software only performs some functionalities of recruitment process.

These functionalities are:

- **Job Description**

  The major functionality of this software is to create offered job's description for publication/advertising purposes. Both wizard and custom method are included. A feature called Job Library, which stores thousands of jobs with their essential duties, will help the user obtain a detailed description. The user can select the job description template form this Job Library according to its Categories or Industries. After that, the user will need to continue with a serial of steps (15 – 20 steps at least) and finally come out a full offering of job's description (see appendix 1).

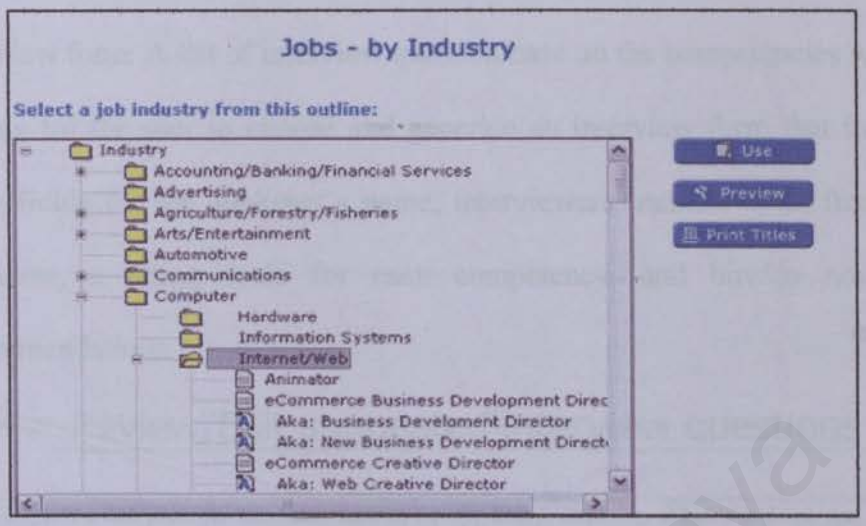


Figure 2.5: Create job description form Job Library - wizard

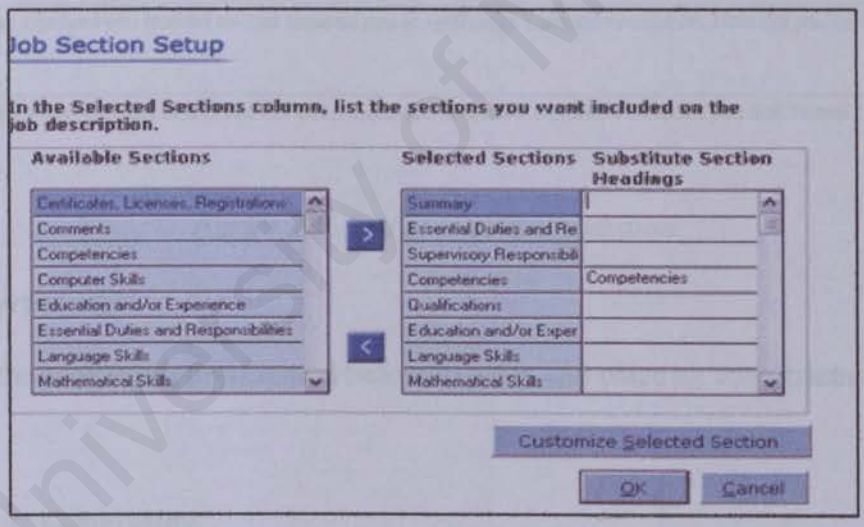


Figure 2.6: Create job description by the user's define - custom

- **Interview Question and Form**

  This system will help the user in generating the interview question and the interview form. A list of interview question base on the competencies will be display for the user to choose and generate an interview form that include some fields for the applicant's name, interviewers' names, notes from the interview, a rating scale for each competency, and hire/do not hire recommendations.

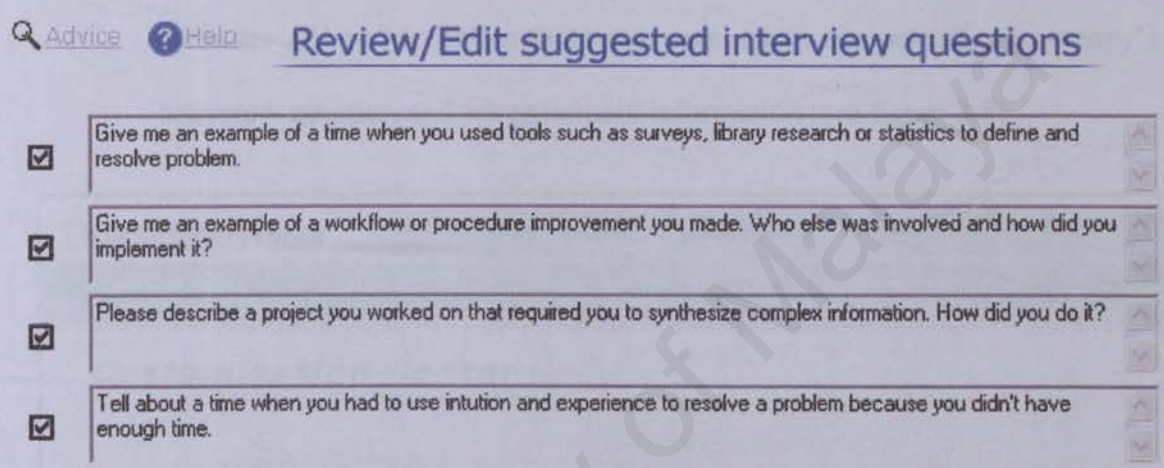


Figure 2.7: List of interview question.

- **Advertisement**

  Use the job description created before to write and place an advertisement.

- **Report Generating**

  Generate report according to the list below:

  - Title (with Job Information)
  - Title (with Summary)
  - Job Code
  - Division/Department

- Department
- Salary Level
- Prepare Date
- Approved Date

- **Customization Centre**

  Place for the user to create the job description in a custom way. Besides, the users are allowed to define their own company's competencies, company's interview question and advertisement information (see figure 2.8).

Figure 2.8: Customization Centre provides the functionalities to create the job description in a custom way.

- **HR tools.com Hiring Toolkit**

  This is an internet service that is provided for this application. The information on the site covers:

  - Hiring and recruiting
  - Human resource forms
  - Performance Management
  - Employee screening
  - Relocation
  - Training
  - Benefits and compensation
  - Personnel policy

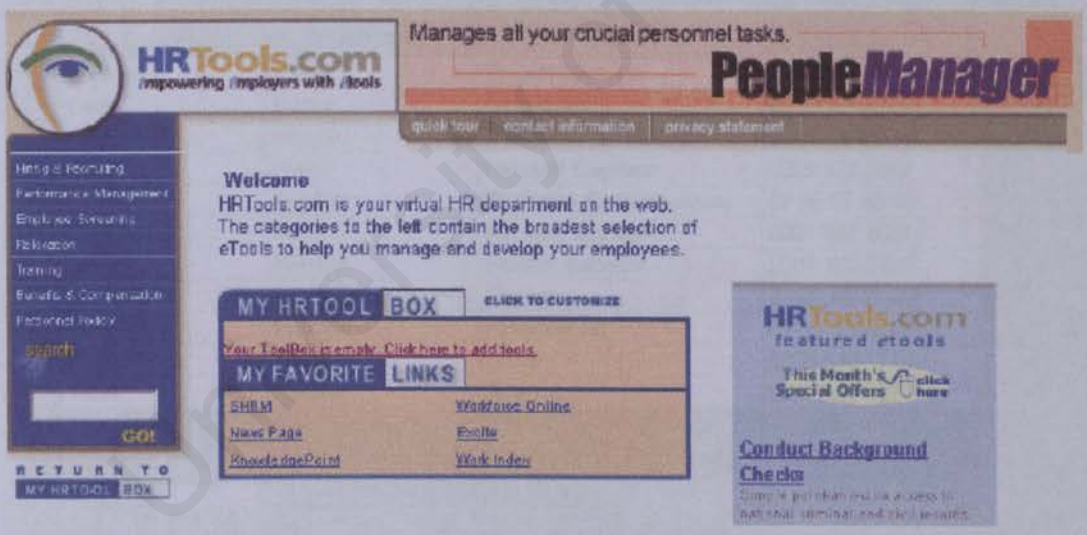


Figure 2.9: HR tools.com Hiring Toolkit

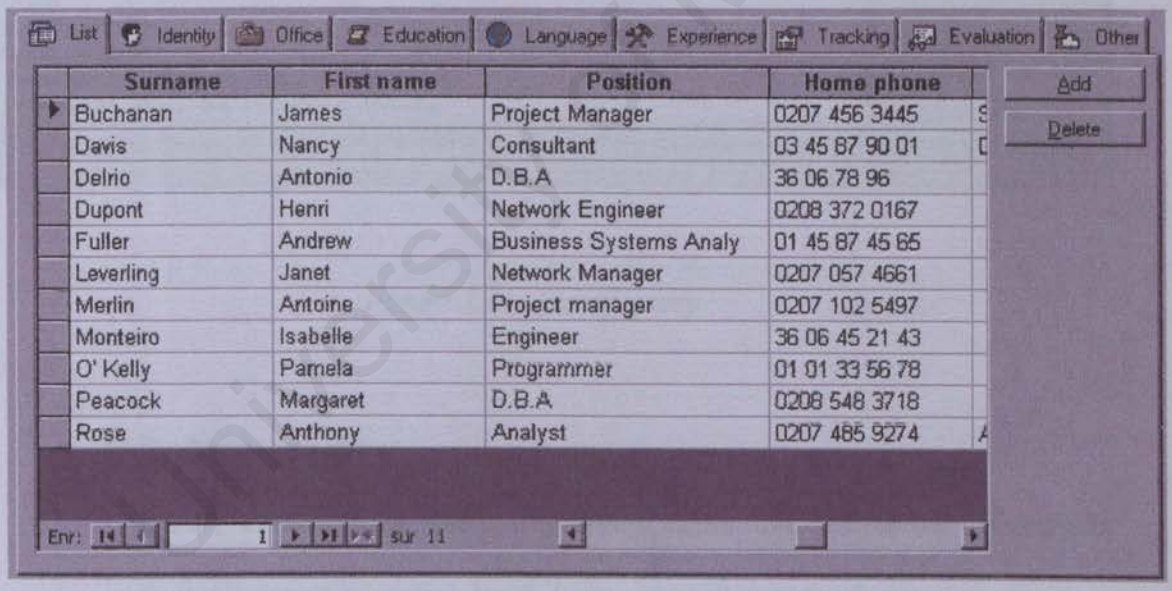
2.5.2 Case Study 2

Visual Recruitment 2000 – Task (http://www.visualrecruitment.com)

Visual Recruitment offers a set of applications and services for the Recruitment Specialists. These applications and services include:

- **Applicant database building and candidates tracking**

  This function support the full input of CVs with extensive database. The user is able to input all the necessary information about the applicants including personal detail, education level, experience, language skills, and so on. Thus, keep tracking the candidate information becomes much simpler and time saving.



Surname	First name	Position	Home phone
Buchanan	James	Project Manager	0207 456 3445
Davis	Nancy	Consultant	03 45 87 90 01
Delrio	Antonio	D.B.A	36 06 78 96
Dupont	Henri	Network Engineer	0208 372 0167
Fuller	Andrew	Business Systems Analy	01 45 87 45 65
Leverling	Janet	Network Manager	0207 057 4661
Merlin	Antoine	Project manager	0207 102 5497
Monteiro	Isabelle	Engineer	36 06 45 21 43
O' Kelly	Pamela	Programmer	01 01 33 56 78
Peacock	Margaret	D.B.A	0208 548 3718
Rose	Anthony	Analyst	0207 485 9274

Figure 2.10: List of the applicant

- **Multiple Criteria Selections**

  This allows the user specifically select a group of applicants from a list of query base grouping. For instance, a group of applicants that have a minimum of 2 years experience. Besides, the query is allowed to be defined by the user.

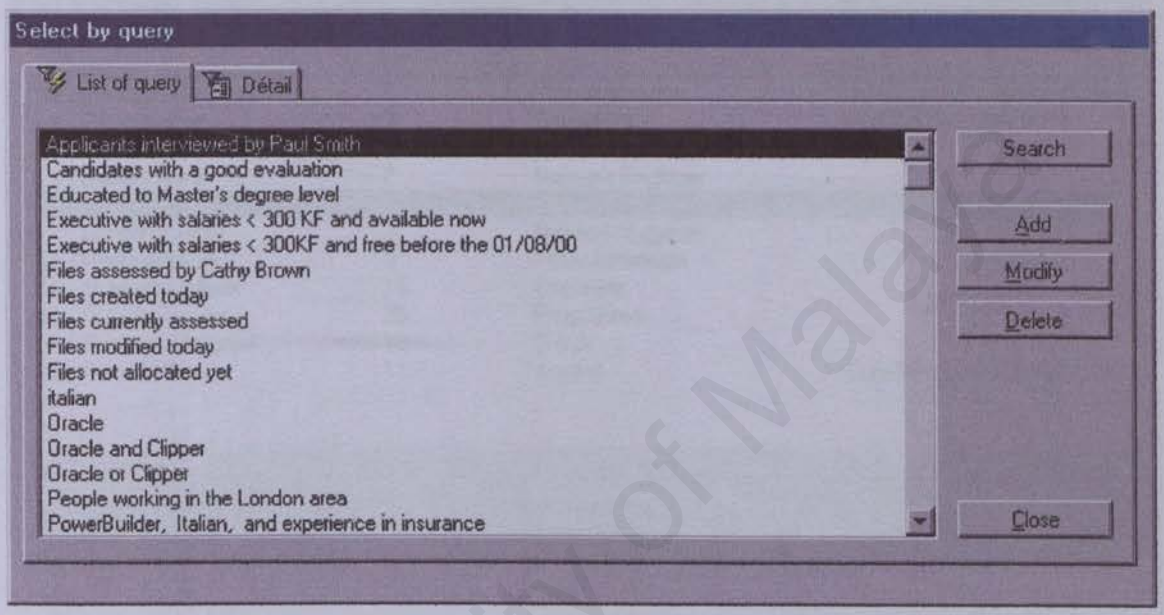


Figure 2.11: List of query

- **Skills Management**

  Users are allowed to define the minimum skills required for the offered job such as minimum education level, language skill, length of experiences, etc.

- **Search**

  Search module for fast and easy retrieving of information.

- **Sharing Facilities and Control of Access**

  Support multiple users. Several level of access right can be defined such as administrator, HR user, user, etc. Administrators have full access right.

- **Milling**

  Users are allowed to organize the mailings to pre-selected candidates.

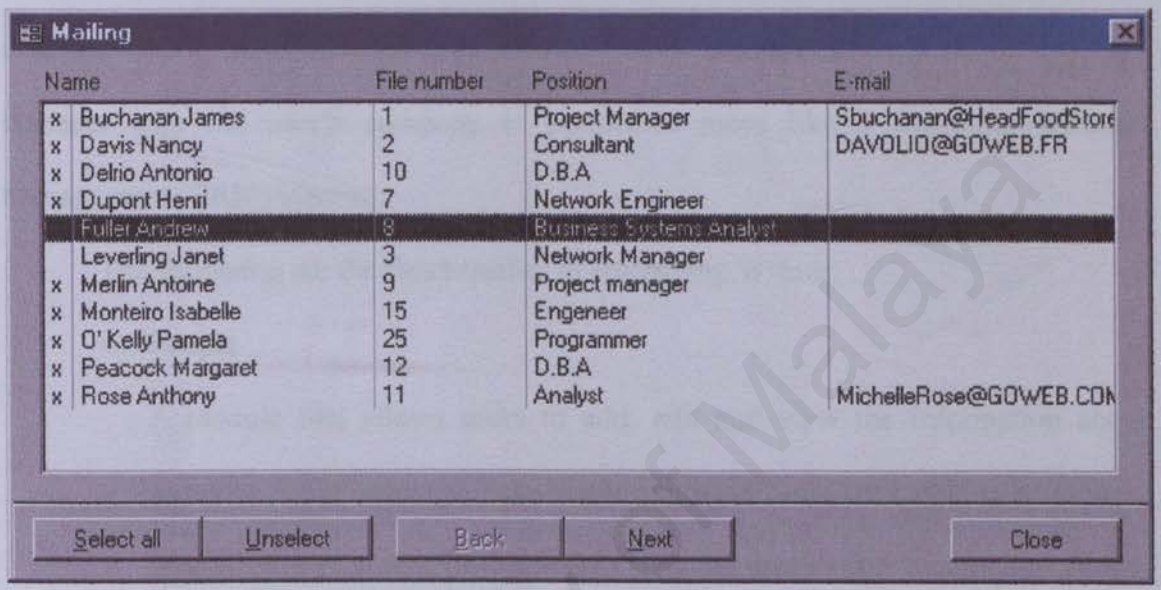


Figure 2.12: Organize the mailings to pre-selected candidates

2.5.3 Case Study 3

Recruiting Wizard version 2.42 – Walling Info System

(http://www.recruitingwizard.com)

Although this system is named as Recruiting Wizard, the functionalities provided are more than recruitment process. The Companies module (will be further describe later), which can be used to store other companies' information that has contacts with the user's company is performed more like a Customer Relation Management (CRM) system.

The following are the functionality of Recruiting Wizard:

- **People**

  A module that allows users to add, edit and view the information about someone. This "someone" can be the applicant/candidates, clients, references, friends or anyone that have contacts with the user or user's company. Multi-tab function is also included, which will store someone's contact information, personal detail, education level, skill, employment, activity, etc.

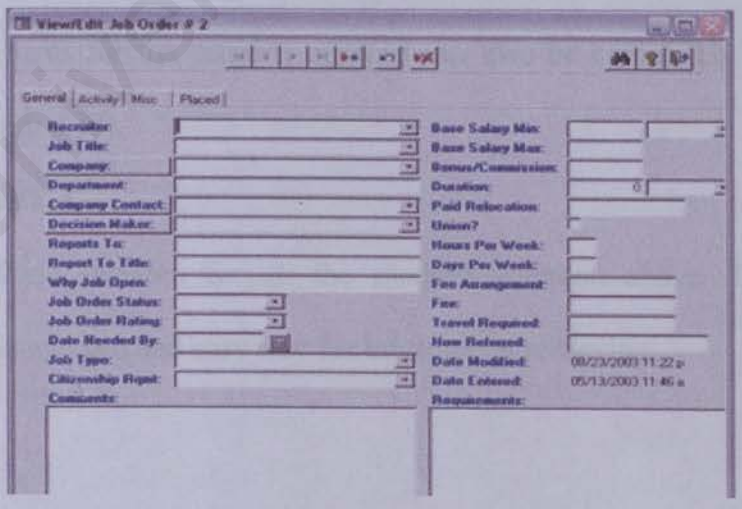


Figure 2.13: People module

- **Companies**

  As mention above, this module store the information of other companies such as clients, prospects, other recruiting firm, etc. Multi-tab function is also included to store the address of the company, contact list, benefits, job orders, and comments.

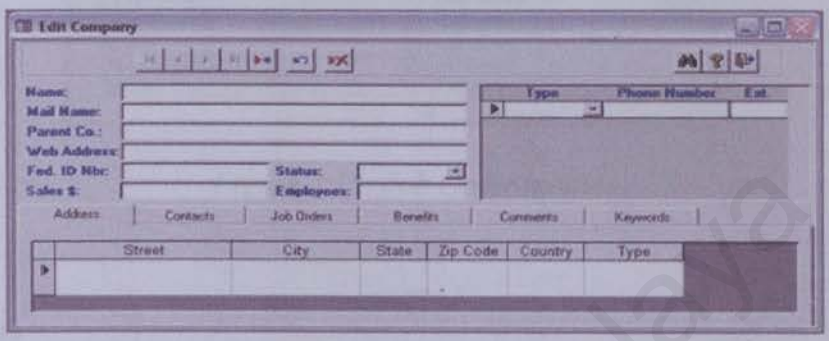


Figure 2.14: Companies Module

- **Job Orders**

  This module is for tracking job orders for both permanent and temporary. Multi-tab is included to store miscellaneous projects, activities, venue and time. Job Order records are instantly updated with resume, interview, and other activities or follow-up calls. Skill, education, and compensation requirements for the position opening can also be completely documented (see figure 2.15).

- **Search/Find**

  Search/Find function for all the modules mention above. This features provided and fast and easy way for information retrieving.

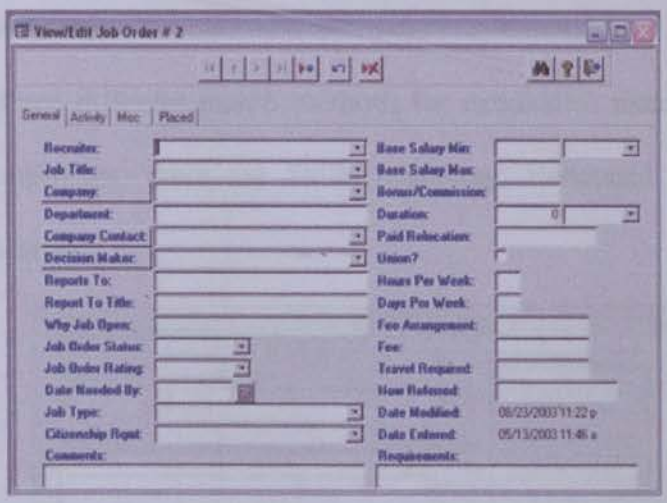


Figure 2.15: Job Order module

- **Planner**

  To-do list for the user to organize his/her work, sorts and prioritizes all planned calls and activities. Incomplete items are carried forward to the next day and shown as overdue.

- **Report Generating**

  - Generate the report according to the following:
  - Activity by recruiter
  - Company call list
  - Contact by company
  - Job Order by status
  - Skill list
  - Task list
  - Employees at client sites

- **Match Candidates**

There are three different match methods for candidates matching with Job Order requirements; there are Skills Matching, Enhanced Matching and Keyword Search.

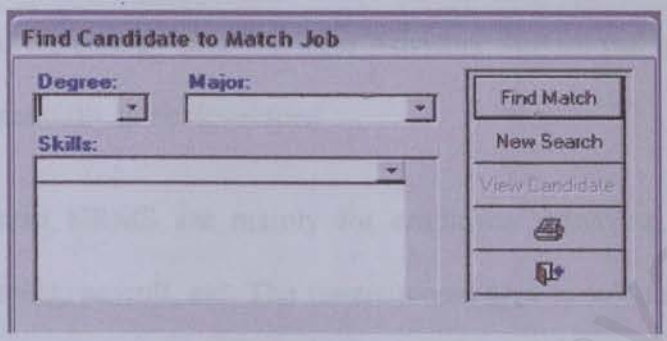


Figure 2.16: Skills Matching

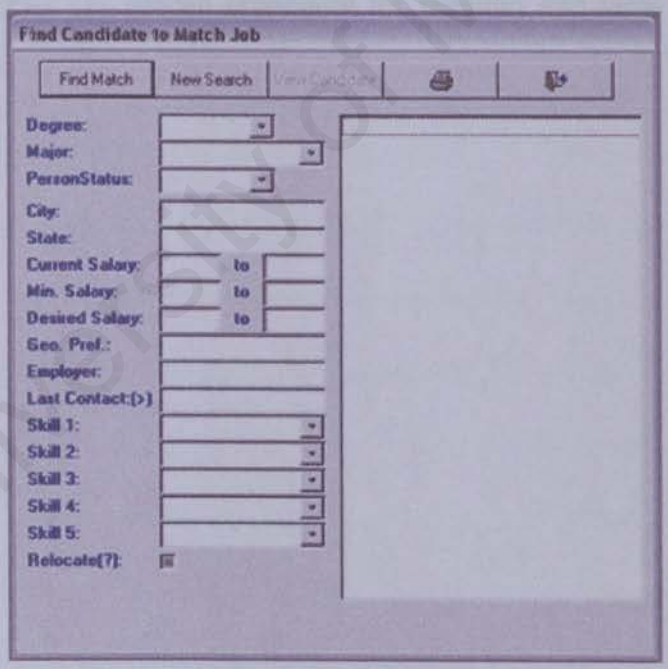


Figure 2.17: Enhanced Matching

2.5.4 Conclusion

From the research, it is found that most Human Resources Management Systems (HRMS) are built using the non-AI technology. In addition, the Existing System Review session above showed that 2 out of 3 of the reviewed system is the Microsoft Access base system. Thus, an AI technology base Business Advisory System can be considered as a new concept and potentially to be developed.

Further more, most HRMS are mainly for employee managing, recruitment process, customer managing, payroll, etc. The retrenchment area is seldom considered. In fact, both recruit and retrench process happen every single day in this world. Again, an AI base Business Advisory System that provides both the recruitment and retrenchment functionalities are needed especially for medium and large corporation.

2.6 Technology Review

2.6.1 System Architecture

Web Services

The term web services is fairly self-explanatory, it refers to accessing services over the web. But, there's more to it than that, the current use of the term refers to the architecture, standards, technology and business models that make web services possible. Web Services are self-contained, modular applications that can be described, published, located, and invoked over a network, generally, the Web. Web services perform functions, which can be anything from simple requests to complicated business processes. In other words, web services are interoperable building blocks for constructing applications.

The Web Services architecture is the logical evolution of object-oriented analysis and design, and the logical evolution of components geared towards the architecture, design, implementation, and deployment of e-business solutions. Both approaches have been proven in dealing with the complexity of large systems. As in object-oriented systems, some of the fundamental concepts in Web Services are encapsulation, message passing, dynamic binding, and service description and querying. Fundamental to Web Services, then, is the notion that everything is a service, publishing an application programming interface (API) for use by other services on the network and encapsulating implementation details.

IBM Web Services

IBM has published its web services architecture which captures the infrastructure required to support web services in terms of three roles - service provider, service requestor and service registry - and the verbs describing the interactions between them: *publish, find* and *bind*. Service providers *publish* services to a service broker. Service requesters *find* required services using a service broker and *bind* to them. Bind is the step that allows an application to connect to a web service at a particular web location and start interacting with it.

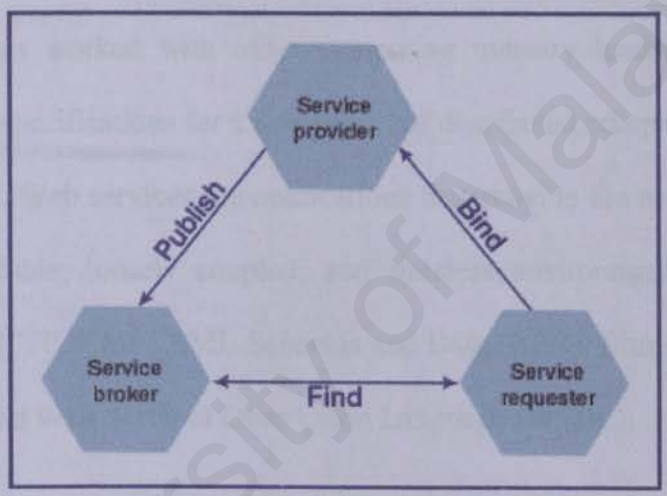


Figure 2.18: IBM Web Services architecture

The full description of a service for IBM's architecture is found in two separate documents, a Network Accessible Service Specification Language (NASSL) document and a Well-Defined Service (WDS) document. NASSL is an XML-based Interface Definition Language (IDL) for network-based services, and is used to specify the operational information for a Web Service, such as service interface, implementation details, access protocol, and contact endpoints. A WDS document is used to specify the

non-operational information for a service, such as service category, service description, and expiration date, as well as business information about the service provider, such as company name, address, and contact information. A WDS document is complementary to a corresponding NASSL document. Together these two documents are used to specify a full service description that allows service requesters to locate and invoke a service.

XML Web Services

Microsoft has worked with other computing industry leaders to create and standardize a set of specifications for a new model of distributed computing called XML Web services. XML Web services are applications that provide the ability to exchange messages in a scalable, loosely coupled, and stateless environment using standard protocols such as HTTP, XML, XML Schemas and Data (XSD), Simple Object Access Protocol (SOAP), and Web Services Description Language (WSDL).

XML Web services make possible the building of modular applications within and across companies in heterogeneous environments making them interoperable with a broad variety of implementations, platforms and devices. The SOAP-based XML messages of these applications can have explicit (structured and typed), or loosely defined parts (using arbitrary XML). The ability of the messages to evolve over time without breaking the protocol is fundamental to the flexibility and robustness of XML Web services as a building block for the future of the Web.

XML Web services enable the exchange of data and the remote invocation of application logic using XML messaging to move data through firewalls and between heterogeneous systems. Although remote access of data and application logic is not a new concept, doing so in a loosely coupled fashion is. The only assumption between the XML Web service client and the XML Web service provider is that recipients will understand the messages they receive. This is meant that, both the XML Web service client and the XML Web service provider are freed from needing any knowledge of each other beyond inputs, outputs and location. As a result, programs written in any language, using any component model, and running on any operating system can access XML Web services.

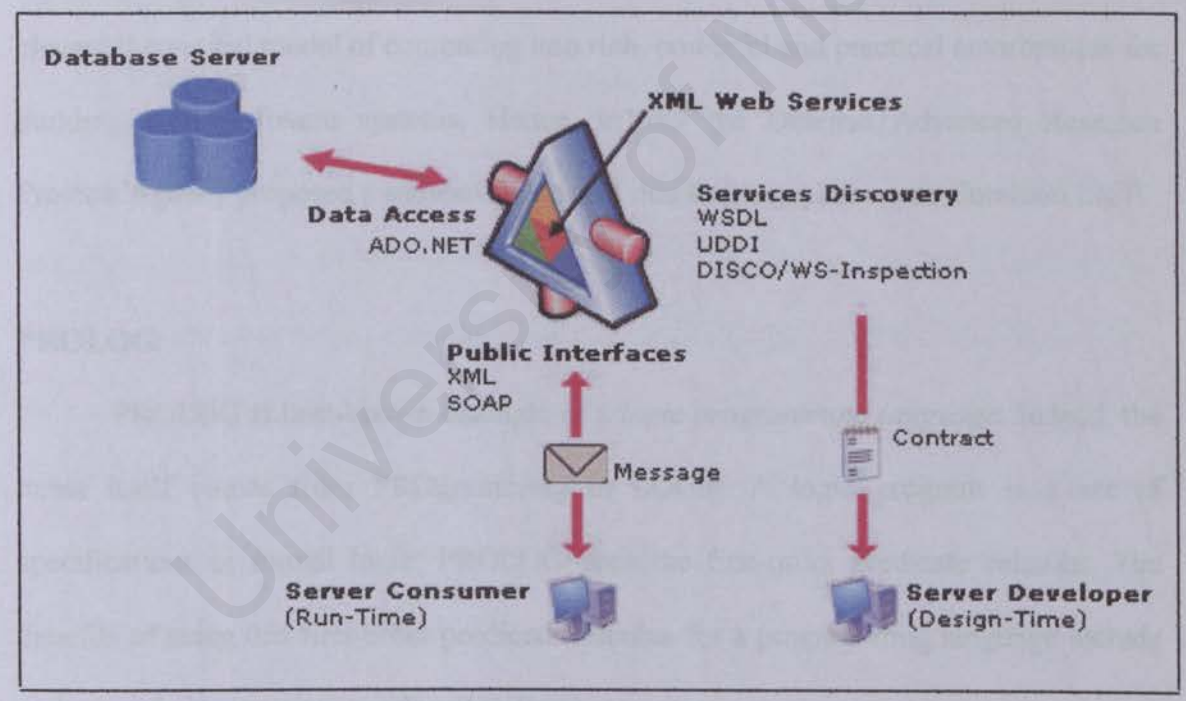


Figure 2.19: Microsoft .NET XML Web Services architecture

2.6.2 Programming Language

LISP

LISP was first proposed in the late 1950s. LISP was originally intended as an alternative model of computation based on the theory of recursive functions. The list is the basis of both programs and data structures in LISP: LISP is an acronym for LISt Processing. Thus, LISP provides a powerful set of list-handling functions implemented internally as linked pointer structures.

Through time, LISP have been adding hundred of specialized functions for data structures, programming control, real and integer arithmetic, input/output (I/O), editing LISP functions and tracing program execution. LISP has evolved from a simple and elegant theoretical model of computing into rich, powerful and practical environment for building large software systems. Hence, in1983 the Defense Advanced Research Projects Agency proposed a standard dialect of this language, known as Common LISP.

PROLOG

PROLOG is best-known example of a *logic programming language.* Indeed, the name itself comes from PROgramming in LOGic. A logic program is a set of specifications in formal logic; PROLOG uses the first-order predicate calculus. The benefits of using this first-order predicate calculus for a programming language include a clean and elegant syntax and well-defined semantics.

PROLOG supports a declarative programming style (constructing a program in term of high-level description of a problem's constrains) rather than a procedural

programming style (writing programs as a sequence of instructions for performing an algorithm). This mode of programming essentially tells the computer "what is true" and "what needs to be done" rather than "how to do it".

Java 2 Platform, Enterprise Edition (J2EE)

Java 2 Platform, Enterprise Edition (J2EE) was designed to support applications that implement enterprise services, for example the Enterprise Information System (EIS). Such applications are inherently complex, potentially accessing data from a variety of sources and distributing applications to a variety of clients. Thus, J2EE application model defines architecture for implementing services as multi-tier applications to simplify the complexity.

To better control and manage these applications, J2EE application model partitions the work needed to implement a multi-tier service into two parts: the business and presentation logic to be implemented by the developer, and the standard system services provided by the J2EE platform. The developer can rely on the platform to provide the solutions for the hard systems-level problems of developing a middle-tier service. The middle tier is typically run on dedicated server hardware and has access to the full services of the enterprise.

J2EE application model starts with the famous programming language – Java, Java Server Pages (JSP) and the Java Virtual Machine. J2EE also include the JavaBeans Component Model to make it easy to componentize the Java code.

Microsoft .NET

Microsoft .NET or .NET provide a multi-language platform or much more known as the .NET Framework with integrates features and capabilities from Microsoft product teams, from partners and even from competitors. .NET re-invents how the programmers think about software development based on the best modern programming ideas.

Beside the .NET Framework, the programming language likes Visual Basic (VB) have been enhanced to a truly object-oriented language known as VB.NET. A brand new language, C#, combines with the power and flexibility of C++ has been born. In addition, a new plug-in interface that allows third-party developers to create new languages which fit into the .NET world also being implemented.

For the web development, .NET has brought in a whole new concept and technology about Microsoft's Active Server Page (ASP) name as ASP.NET. Unlike ASP (uses only limited scripting language), ASP.NET are able to use any kind of .NET compatible language to accomplish its goals. ASP.NET also boasts a whole new approach to dynamic, interactive applications called Web Forms. The unique part of ASP.NET is separating the user interface (HTML), and server site coding (server control) called the "code behind".

Furthermore, in order to increase the portability, .NET has implemented the XML Web Services, which uses the common protocol such as HTTP, SOAP, to communicate with the non-Windows platform like Linux, UNIX, Macintosh, and so on.

Microsoft .NET versus Java 2 Platform, Enterprise Edition (J2EE)

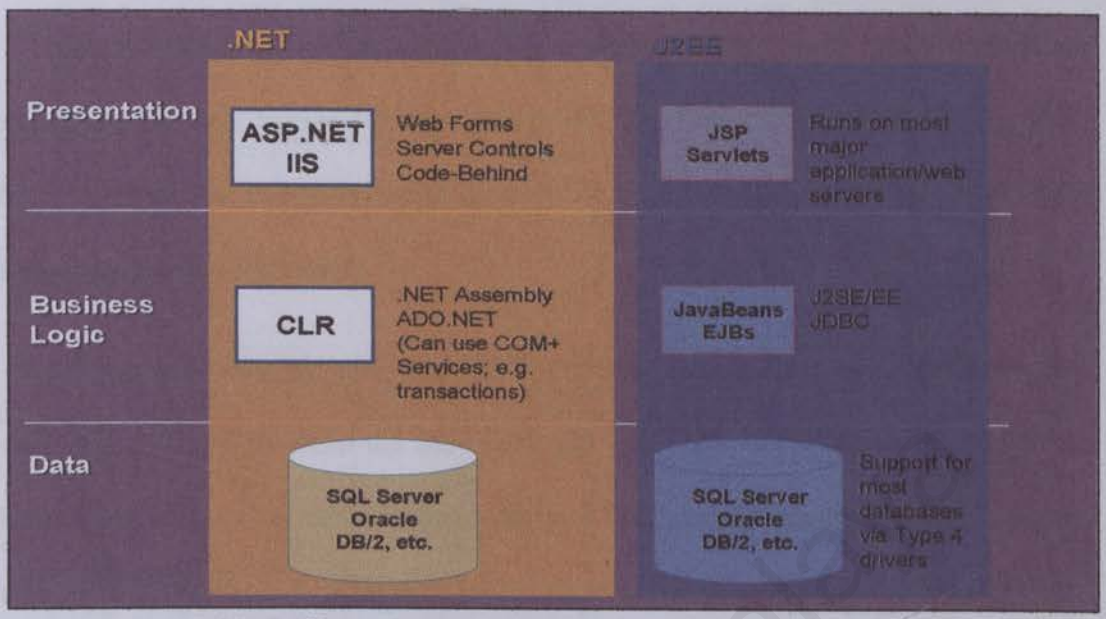


Figure 2.20: Different between .Net and J2EE Web Architecture

Table 2.2: Comparison between .Net and J2EE

	Microsoft.NET	J2EE
Programming Language	Based on C# programming language	Based on Java programming language
Pre-written codes	.NET common components	Java core API (application program interface)
Internet	ASP.NET	JSP/Servlets
Common language for runtime	Internal Language (IL)	Java Virtual Machine and CORBA (common object ORB architecture), IDL (interface definition language) and ORB (object request broker)
User interface component	Win Forms and Web Forms	Java Swing
Database connectivity	ADO.NET (.NET ActiveX Database Object) SOAP-based Web Services	JDBC (Java database connectivity) EJB(Enterprise Java Beans) JMS (Java Message Service) Java XML Libraries (XML4J, JAXP)

2.7 Chapter Summary

This literature review is done mainly for gaining information in order to develop this project. At the very beginning, the concept of AI is being discussed including two popular techniques that normally use to develop AI application; Expert Systems (rules-based) and Case-based Reasoning.

In Existing System Review session, 3 existing system have been chosen to be reviewed. Although all of them are the client site applications (windows application), web base applications also exist. Unfortunately, AI technology base's systems are still very rare, as well as the functionalities of the retrenchment area, which is also seldom found.

As for the current Technology Review, it has been separated into 2 parts: System Architecture and Programming Language. Web Services is the only system architecture that has been reviewed in this project, which includes the IBM Web Services and Microsoft .NET XML Web Services. For the later part, the review is on 2 fifth generation programming languages: PROLOG and LISP, and 2 popular language platforms: J2EE and Microsoft .NET.

In chapter 3, Methodology, the main focus in on the software process model that is being used and the reason why it is being used together with the Information Gathering Methods.

Chapter 3: Methodology

Chapter 3: Methodology

3.1 Software Process Models

Software process model is an abstract representation of a software process (Sommerville, 2001). Each process model contains a coherent set of activities for specifying, designing, implementation, software system testing and presented from a specified perspective. Thus, it only provides partial information about that process.

There are lots of generic process models including the *waterfall model, evolutionary development, formal systems development, reuse-based development, incremental development, spiral development,* etc. Each model will have its pros and cons, and different project will use different model according to the project sizes and project needs. The Business Advisory System will use the waterfall model as its process model.

3.1.1 Waterfall Model

Waterfall model, also known as software life cycle, is the first model of the software development process was derived from other engineering processes (Sommerville, 2001).

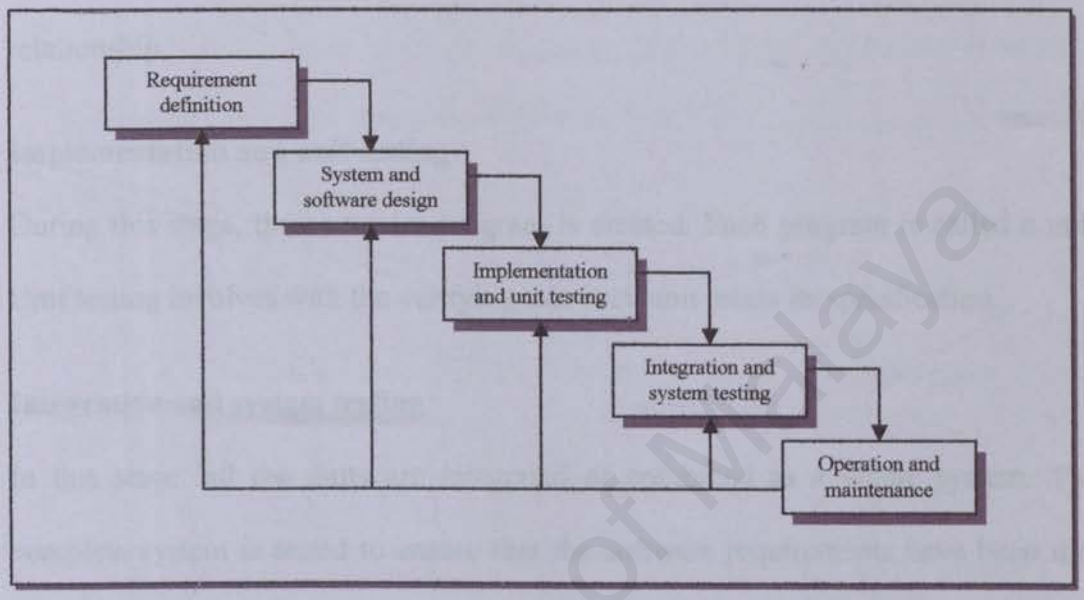


*Figure 3.1: The waterfall model (*Sommerville, 2001*)*

There are 5 stages in the waterfall model. Each development stage should be completed before the next begins. The 5 stages are:

1. **Requirement analysis and definition**

   This stage is the requirements gathering stage including understanding and determining users need. These are usually the system's services, constraints and the goals of the software. Once these are established, it will be further defined and serve as a system specification.

2. **System and software design**

   In this stage, systems design process will partition the requirement that have been defined to either hardware or software system by the followed by establishing an overall system architecture. Whereas, the software design process involves identifying and describing the fundamental software abstraction and their relationship.

3. **Implementation and unit testing**

   During this stage, the computer program is created. Each program is called a unit. Unit testing involves with the verifying that each unit meets its specification.

4. **Integration and system testing**

   In this stage, all the units are integrated or combined as a whole system. This complete system is tested to ensure that the software requirements have been met. After testing, the software system is ready for delivery to the customer.

5. **Operation and maintenance**

   Normally this is the longest life-cycle phase. The system is installed and put into practical operation and the maintenance process will take place right after this.

3.1.2 Why Waterfall Model?

The reasons of choosing waterfall model as development model for this project are:

- **Simplicity to developer**

  Simplicity in the sense of the development methodology is simple and easier to understand. Thus, the developer will have better and clearer guideline on what shall be done during development process.

- **Clearly divide the stages**

  Just need to emphasize one stage at a time and focus on the current stage without over burdening with the upcoming stage.

- **Emphasize on planning rather than rapid development**

  This allows estimation of the completion of each stage so that the system can be developed within the time frame.

- **Simplicity to user**

  When presenting or explaining the development progress to users especially those who are not familiar with software development cycle, the users will have better understanding on what is going on if compared with adopting a more complex methodology.

3.2 Information Gathering Methods

Information gathering is a very important process in order to understand the problems, user requirements and provide the groundwork for the system design. Although there is no underlying procedure or guideline to help in the information retrieving process, there are some popular methods, which commonly used such as collecting hard data like written documents or reports, interviewing, questionnaires, observation and sampling.

As for this project, some information gathering methods are impractical to carry out especially interviewing and questionnaires. This is due to the cost and time constraints as well as the difficulties in finding and getting domain experts whom are willing to help. Hence, the methods that have been carried out are the following:

- **Internet Surfing**

  Internet surfing is the most widely used method. Through internet, lots of information can be found such as the existing system, information about the domain, getting to know about the current technology, system architecture, database server, application server, various platform, etc.

- **Books and references**

  Although there are billions of information in the internet, books and references are equally important especially the seniors' thesis, which is located in the FSKTM's document room, are the best references for developing this project. Besides, books are good especially to support the information or facts from internet to avoid biases.

- **Existing System Review**

  Existing System Review is the most important method for developing this project. Through this method, the strengths and weaknesses can be easily defined. In addition, the policies and procedures or operation and management can be clearly understood. The needs of the user towards a better service can be gathered through own experiences by using the existing system.

- **Discussion with supervisor**

  Discussion with supervisor has been practiced from time to time in order to get help and advices during the development of the project. Without the supervisor's guidance, this project would never succeed.

3.3 Chapter Summary

In order to deliver an efficient system within a certain time frame, this project is adapted with the waterfall model, which emphasized more on the step by step project planning instead of rapid development. Thus, this project will be separated to 2 phases and 5 major stages.

In phase 1, the first two stages that include the requirement analysis as well as system and software design will be done together with the presentation of this project. Whereas in phase 2, the other 3 stages, which are system implementation, system testing and system operation will be done later.

Beside the software process model, four major information gathering methods have been used in developing this project. There are internet surfing, referring to the books and references, existing system review and discussion with project supervisor.

With the methodology, both waterfall model and information gathering methods, this project will be able to be completed on time as well as achieve the objective and meet the requirements.

For the next chapter, System Analysis will be carried out whereby the functional, non-functional and system requirements will be listed out in conjunction with the development technologies including the platform, web server, database server and programming tools.

Chapter 4: System Analysis

Chapter 4: System Analysis

System Analysis is the first stage of software development process according to the waterfall model. It normally answers the question *"what must be done"*. System Analysis is essential and important to determine and clarify the system requirements. Thus, System Analysis also known as *Requirement Analysis*. The IEEE Standard 610 (IEEE Standard Glossary of Computer Languages, 1993) defines a requirement as a condition or capacity needed by the system developer to solve a problem or to achieve an objective. According to the standard, requirements are partitioned into functional requirements and non-functional requirements.

4.1 Functional Requirements

Functional Requirements are statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situation (Sommerville, 2001). In some cases, it also stated what the system should not do.

The functional requirements for a system describe the functionality or services that the system is expected to provide. It differs from the user requirement because it describes the system functional in detail including its input, output, exception, and so on instead of what user wants in a fairly general way.

The functional requirements of Business Advisory System are:

- **Recruitment**

  Generate a set of minimum requirement when hiring new employee(s).

  This system should come out with a set of minimum requirement according to the manger's desire when the company needs to recruit some new employees.

- **Selecting**

  Produce a list of best quality applicant.

  By comparing the information of each applicant with a set of minimum requirement, the system should produce a list of quality applicant for further recruitment process such as interviews.

- **Retrenchment**

  Give advice on how many employees will need to be retrenched.

  According to the company's situation, the system should suggest how many employees will need to be retrenched in order to reduce the company's expenses.

  Suggest on which employee will be retrenched.

  Based on the employee's performance and other factors such as years of service, etc, the system should suggest which employee will need to be retrenched.

- **Cut Down**

  Give advice on salary cut down.

  According to the company's financial situation, the system should suggest how many percent of employees' salary will need to be cut down in order to recover the company's financial problem.

4.2 Non-Functional Requirements

Non-functional requirements, as the name suggest are those requirements which are not directly concerned with the specific functions provided by the system (Sommerville, 2001). They normally relate to the constraints on the services or functionality offered by the system such as timing constraints, constraints on the development process, standard, reliability, etc.

Most non-functional requirements are related to the system as a whole rather than to an individual system features. This means that they are often more critical than individual functional requirements. While failure to meet an individual functional requirement may degrade the system, failure to meet a non-functional requirement may make the whole system unusable.

The non-functional requirements for Business Advisory System are:

User-Friendliness

The system shall have an attractive, user-friendly and relatively easy fashion interface without having to know specifically how the system will work, which is capable of reducing learning curves. User interface design principles such as user familiarity and consistency shall be taken into consideration. The usage of intuitive and meaningful menus and icons is required too.

Robustness

The system also delivers together with the error handling and validation procedure. The system should be able to check the input validation before it continues for further data processing to avoid necessary disaster. When there is any mistake detected, the system will prompt an error message to indicate the mistake and the system is able to restart after an unexpected error of failure.

Reliability

This system should be reliable and should not cause unnecessary downtime of the overall environment. It should set up the acceptable failure rate and does not produce dangerous or costly failures when it is used in a reasonable manner.

Efficiency

Efficiency means a process or a procedure that can be called or accessed in an unlimited number of times to produce similar output at a creditable speed.

Portability

The system should be highly portable, which means that it should not only operate at a specific machine or environment. As long as the machine or environment meets it's minimum requirement (both software and hardware), it should operate well.

Expandability

The system should be ready for future enhancement including adding new features, functionality, and so on.

4.3 System Requirements

4.3.1 Server Site

Hardware Requirements

- A server with at least 1.5GHz processor (recommended 1.8GHz and above)
- At least 256Mb RAM (recommended 512Mb and above)
- 10GB hard disk (recommended 15GB and above)
- Network connection with recommended bandwidth at 10Mbps
- Other standard computer peripherals

Software Requirements

- Microsoft Windows 2000 Server with service pack 4 or equivalent
- Internet Information Services 5.0 (IIS)
- Microsoft .NET Framework 1.1
- Microsoft SQL Server 2000
- Microsoft Internet Explorer 6.0

4.3.2 Client Site

Hardware Requirement

- A computer with a least 800MHz processor
- At least 128Mb RAM (recommended 256Mb and above)
- 500MB hard disk free space
- Internet connection with 56.6kbps bandwidth (recommended 300kbps and above)
- Other standard computer peripherals

Software Requirement

- Any Microsoft Windows operation system
- Microsoft .Net Framework 1.1

4.4 Development Technologies

4.4.1 Operation System - Microsoft Windows 2000 Server

Microsoft Windows 2000 Server is the operation system (OS) that is used in the environment of developing Business Advisory System. Windows 2000 Server (originally NT 5.0) is the multipurpose network operating system that offers multi-processors up to 4 CPUs. Windows 2000 Server is built for small to medium-sized enterprise application deployments, web servers, workgroups, and branch offices. Windows 2000 Server provides an integrated, comprehensive and easy-to-use solution.

Windows 2000 Server, like Windows NT Server 4.0, has been designed from the ground up as an integrated multipurpose operating system. As opposed to combining un-integrated services, Windows 2000 Server provides complete integration between its services resulting in easier management and lower Total Cost of Ownership (TCO). For instance, once authenticated to the directory, users do not need to re-authenticate themselves to access other applications and services.

The Active Directory service in Windows 2000 Server is built completely around Internet-standards and offers extensibility and scalability. This makes it a solution upon which to build enterprise-level directory-enabled applications. Microsoft Management Console (MMC) provides customers with a single customizable interface for managing networking services and applications. Security support in Windows 20000 Server provides support for Kerberos, smart card authentication, fully integrated public key infrastructure, and file system encryption services.

4.4.2 Web Server - Internet Information Services (IIS)

Internet Information Services (IIS) is a Microsoft product that comes with the NT base windows such as Windows NT 4.0 (IIS 4.*), Windows 2000 series (IIS5.*), Windows XP Professional (IIS 5.*) and the latest release Windows Server 2003 (IIS 6.0).

IIS transmits information by using the Hypertext Transfer Protocol (HTTP) and it can also be configured to provide File Transfer Protocol (FTP) and gopher services. The FTP service enables users to transfer files to and from the Web site. The gopher service uses a menu-driven protocol for locating documents. The gopher protocol has been largely superseded by the HTTP protocol. Besides, IIS also include other services like Simple Mail Transfer Protocol (SMTP), Network News Transfer Protocol (NNTP), Microsoft Transaction Server, Microsoft Index Server, Microsoft Certificate Server and Microsoft Management Console (MMC).

With IIS, Microsoft includes a set of programs for building and administering Web sites, a search engine that allows users to create customized search forms with a variety of tools, including ASP, ActiveX Data Objects, and SQL database queries, reporting tools from Crystal reports. (Crystal Reports is a visual reporting tool that lets you create presentation-quality reports and integrate them into database applications.), and support for writing Web-based applications that access databases. Microsoft points out that IIS is tightly integrated with the Windows NT and 2000 Servers in a number of ways, resulting in faster Web page serving.

4.4.3 Database Server – Microsoft SQL Server 2000

Microsoft SQL Server 2000 is a modern, full-featured SQL database designed for small or midsize organizations. It contains all the user-friendly features, works more efficiently and has the ability of handling hundreds of transactions simultaneously without affecting performance.

Microsoft SQL Server 2000 provides the enterprise data management platform according to the organization needs to adapt quickly in a fast-changing environment. With the lowest implementation and maintenance costs in the industry, Microsoft SQL Server 2000 delivers rapid return on your data management investment. Microsoft SQL Server 2000 supports the rapid development of enterprise-class business applications that can give your company a critical competitive advantage.

Microsoft SQL Server 2000 is the record-holder of important benchmark awards for scalability and speed, Microsoft SQL Server 2000 is a fully Web-enabled database product, providing core support for Extensible Markup Language (XML) and the ability to query across the Internet and beyond the firewall.

4.4.4 Programming Tool

Microsoft Visual Studio .NET

Microsoft Visual Studio .NET is a programming tool that contains multiple programming languages such as C++, VB.NET, C#, ASP.NET, etc. In addition, .NET provides plug-in interface that allows communication with foreign languages such as Macromedia Flash MX.

Besides, .NET provides web services called XML Web Services, which can be easily built using ASP.NET either C# base or VB.NET base. These web services can be accessed through network connection and data will be sent in XML format. Both services client and provider are freed from needing any knowledge of each other beyond inputs, outputs and location. As a result, programs written in any language, using any component model, running on any operating system; can access XML Web services and understand the message received.

Further more, .NET communicates with database server using the ADO+ or ADO.NET. ADO.NET is the new approach of ADO, which built around n-tier development and architected with XML at its core. The key goal for ADO.NET was to implement improvements in the ADO model so that ADO.NET would have first-class support for the disconnected, n-tier programming environment for which most new applications are written. As a result, the concept of the disconnected "recordset" has been greatly enhanced and elevated to become the focal point in the programming model.

4.4.5 Designing Tools

Macromedia Flash MX

Macromedia Flash MX is not only a power full designing tool, it also a part of programming tool, which use the power of action script and capable to develop various types of application. For instance, web pages, games, window application, and etc can be created using Flash MX.

Adobe Photoshop 7.0

Adobe Photoshop is another power full designer tool that allows user to create a fantastic user interface. Photoshop is software that allows designers and photographers to create original artwork, correct color, retouch and composite scanned images, and prepare professional-quality separations and output with more flexibility than ever before. With a wealth of powerful painting and selection tools, plus multiple layers, special effects filters, and lighting effects, Adobe Photoshop is a camera for designers' mind.

4.5 Chapter Summary

In this chapter, the main focus is on the process of clarifying both the functional and non-functional requirements together with the system requirements in which both the client site and server site are considered.

The functional requirements are the services that the system should provide while the non-functional requirements are the constraint of the system. Both are equally important in order to deliver a successful system. However, the system requirements are less important, it is concerned with both the hardware and software that needs to run the system.

Besides, the development tools that are use for developing this system are also being explained in this chapter. This includes the operating system – Windows 2000 Server, web server – IIS 5.0, database server – MS SQL Server, programming tool – Microsoft Visual Studio .NET and designing tools – Macromedia Flash MX and Adobe Photoshop.

In chapter 5, System Design will be carried out by being partitioned into four parts: Architecture Design, Program Design, Case Design and User Interface Design.

Chapter 5: System Design

Chapter 5: System Design

System Design is a process through which requirements are translated into a model or representation of the software. It normally answers the question *"How things must be done"*. System Design is a very important factor in a system development as it determines the success of a system. The system specification describes the features of a system, the components or elements of a system and their appearance to users.

The System Design process for Business Advisory System can be divided into four parts:

- Architecture Design
- Program Design
- Case Design
- User Interface Design

5.1 Architecture Design

Architecture affects all aspects of software design and engineering. An inappropriate architecture design or a flawed implementation could result in horrendous response times, development times and the future flexibility and maintenance of the application (Gallaugher, 1995).

5.1.1 Client/Server Architecture

The term client/server was first used in the 1980s in reference to personal computers (PCs) on a network. The actual client/server model started gaining acceptance in the late 1980s. The client/server software architecture is a versatile, message-based and modular infrastructure that is intended to improve *usability*, *flexibility*, *interoperability*, and *scalability* as compared to centralized, mainframe, time sharing computing.

The client/server architecture reduced network traffic by providing a query response rather than total file transfer. It improves multi-user updating through a GUI front end to a shared database. In client/server architectures, Remote Procedure Calls (RPCs) or standard query language (SQL) statements are typically used to communicate between the client and server (Edelstein, 1994).

5.1.2 Type of Architecture Used

The vast majority of end user applications consist of three components: *presentation*, *processing* and *data*. There are variety of ways for dividing these resources and implementing client/server architecture (Gallaugher, 1995). For example, the type of architecture that normally uses are 1-tier, 2-tier, 3-tier, n-tier and web services architecture. For Business Advisory System, XML Web Services Architecture is chosen to be used (refer chapter 2 for more information about web services).

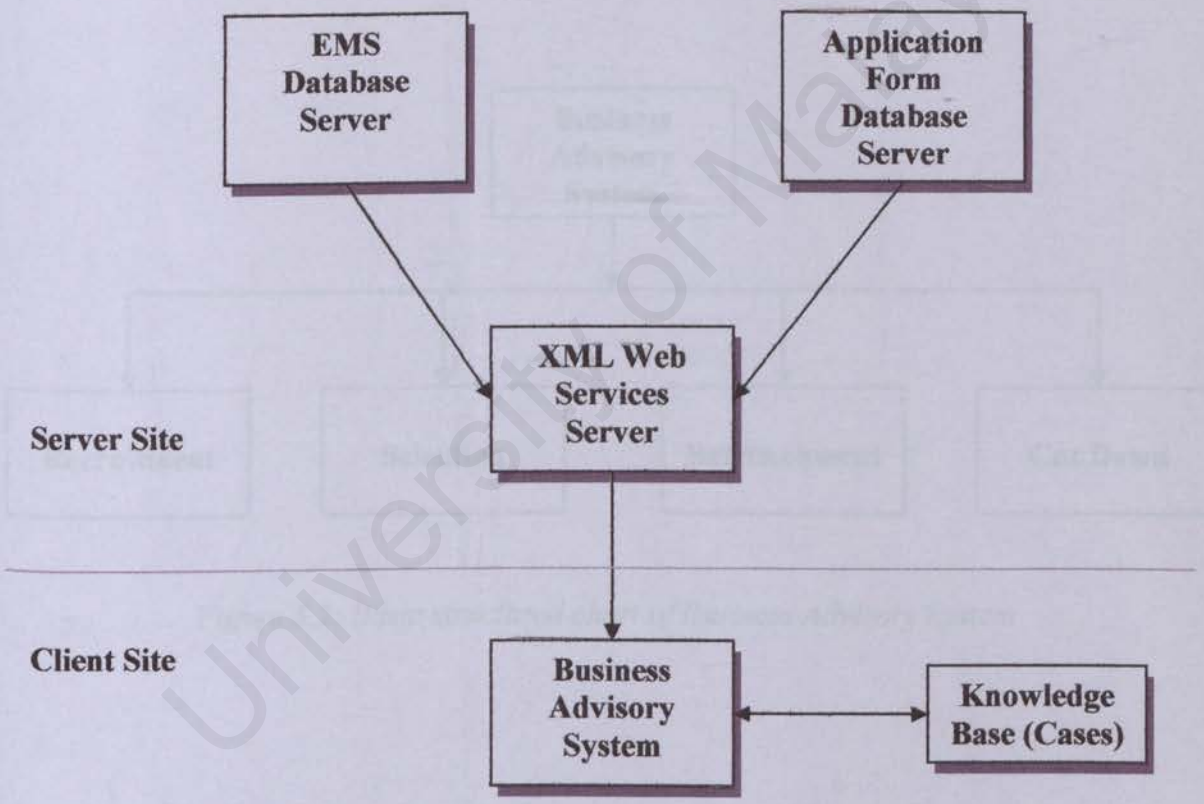


Figure 5.1: Architecture of Business Advisory System

5.2 Program Design

Apart from satisfying the user requirements, the program must be easy to read and understand. It enables other programmer to modify it later. The program should accommodate system changes that occur after the system is built. There are several concepts of program design, such as top-down design, bottom-up design, module design, etc. A variety of design tools including structured chart and data flow diagram are used in helping to develop a well structured program.

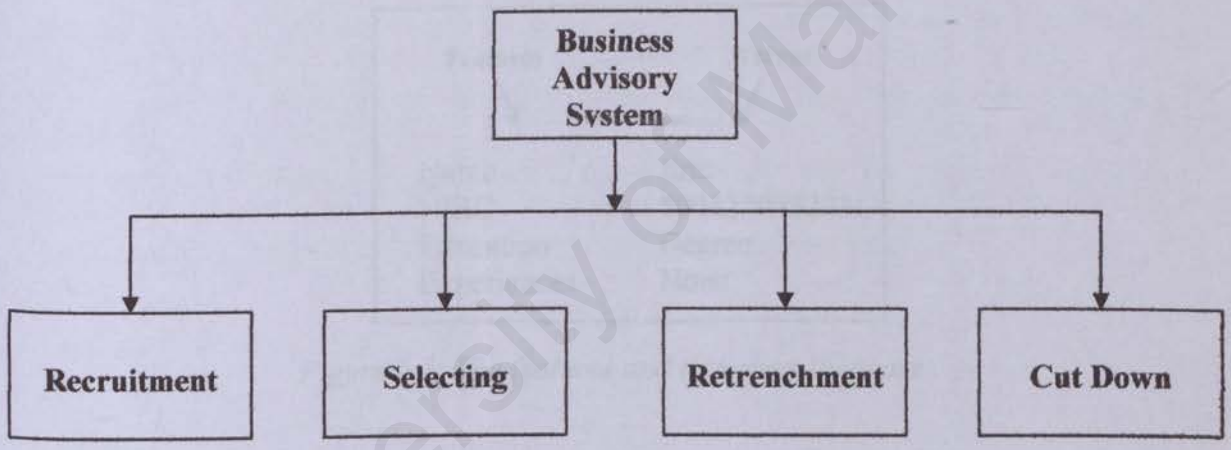


Figure 5.2: Basic structured chart of Business Advisory System

5.3 Case Design

"A case is a contextualised piece of knowledge representing on experience that teachers a lesson fundamental to achieving the goal of the reasoner."

By Kolodner, 1993

A case is a description of knowledge which can be a story, an event or some record that typically consists of features and their specific values while Case Design is a method of how to present all these cases. Figure 5.3 shows the case of an applicant.

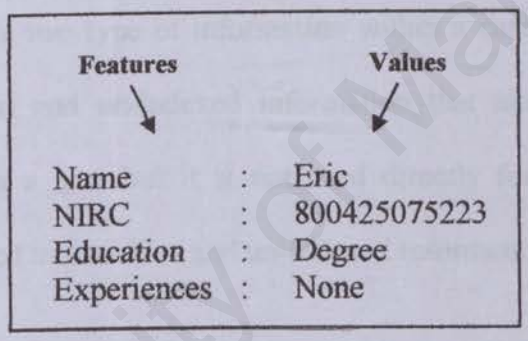


Figure 5.3: The features and its values in a case

5.3.1 Indexing

"Good indexes are predictive, and the predictions they make should be useful ones. Good indexes are abstract enough to provide coverage but concrete enough to be recognizable"

By Kolodner, 1993

An index is a computational data structure that can help in memory and searching. CBR, like most database system uses indexes to speed up the retrieval process. According to Riesbeck and Schank (1989), a good index for CBR is distinctive but not unique. There are two type of information within a case: indexed information that is used for retrieval and un-indexed information that may produce contextual information of values to a user but it is not used directly for retrieval. Figure 5.4 illustrates both the indexed information and un-indexed information.

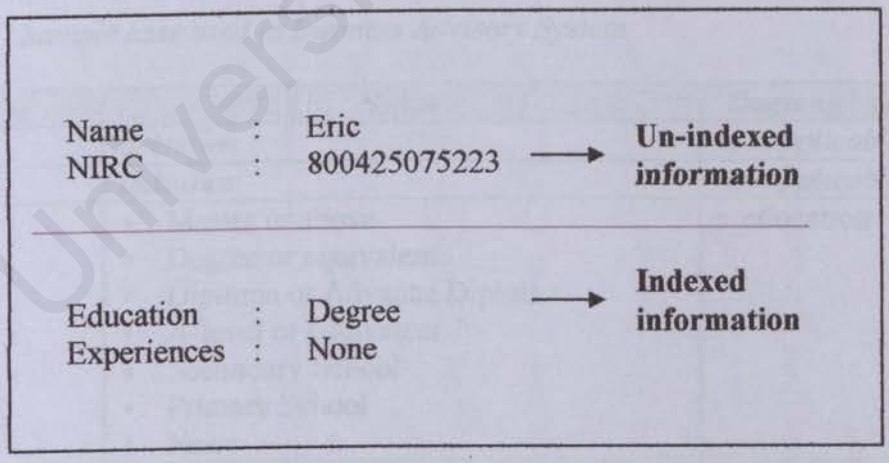


Figure 5.4: The un-indexed information and indexed information

5.3.2 Case Representation

A case can come in various sizes and shapes. It associates solutions with problems, outcomes with situations, or both. All cases are independent from each other and each case describes uniquely a particular situation. A very important key in a case is a fact but not a rule. A case is the primary element for a case-based reasoning application. It may describe a situation or problem in the form of natural language and associate each situation to a proper action (Tuban, 1995).

A case consists of many features with their values. Not all features' values are used to access the degree of similarities between the new case and the previous cases. Indexed features are use for similarities assessment but un-indexed features are not. Table 5.1 is the example of case used in BAS. The un-indexed features with unknown values, which are unpredictable, and the degree of similarity are not applicable because it is not used for the similarities assessment.

Table 5.1: Example case used in Business Advisory System

Feature	Value	Degree of similarity
Name	*unknown*	*not applicable*
NIRC	*unknown*	*not applicable*
Education	▪ Master or above ▪ Degree or equivalent ▪ Diploma or Advance Diploma ▪ A-level or equivalent ▪ Secondary School ▪ Primary School ▪ None	s_education
Experience	▪ 5 years or above ▪ 4 years ▪ 3 years ▪ 2 years ▪ 1 years ▪ None	s_experience

5.4 User Interface Design

The term "User Interface" refers to the methods and devices that are used to accommodate interaction between machines and human (user). User interface can take on many forms, but always accomplish the fundamental tasks: communicating information from the machine to the user and communicating information from the user to the machine (Brown, 1995).

However, an interface, which is difficult to use, will result in high level of user errors and cause some software system to be discarded, irrespective of its functionality. Thus, for interface design, it is important to take into consideration the user's needs and preferences.

In this project, the user interface design is based on the Graphic User Interface approach. The goals of it are to provide user-friendly, easy and faster way for the user to interact with the computer, or what is commonly known as Human-Computer Interface (HCI). These HCI general principles among others are consistency, recoverability, confirmation and verification message, reverse action and responsiveness. Figure 5.5 and figure 5.6 shows the user interface of BAS.

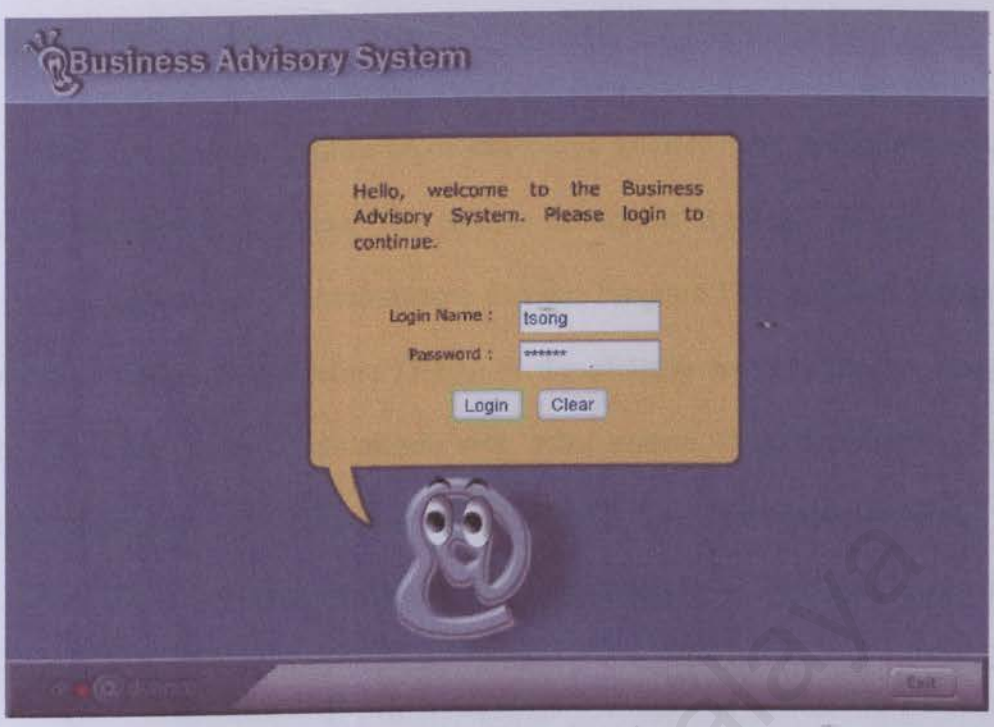


Figure 5.5: Login page of Business Advisory System

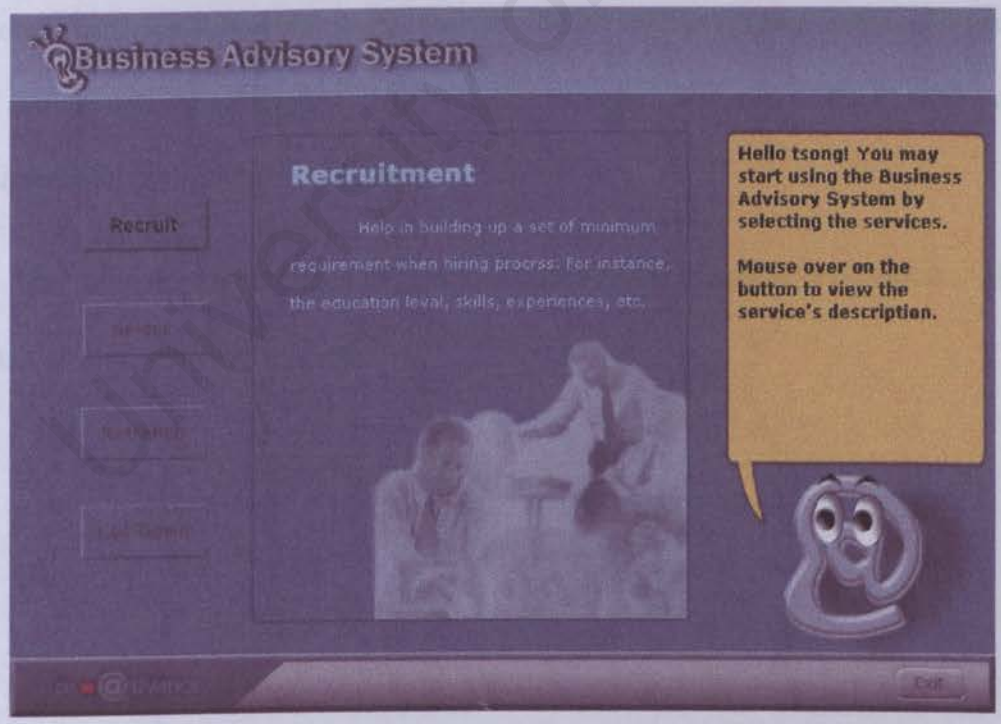


Figure 5.6: Services selection page of Business Advisory System

5.5 Chapter Summary

System design is the evaluation of alternative solutions and the specifications of detailed computer-based solution. The System Design of Business Advisory System (BAS) can be represented by Architecture Design, Program Design, Case Design and User Interface design. Architecture Design is the client/server architecture design of BAS. It shows how BAS communicate with other system to get information. While Program Design shows the general structure of BAS for further understanding, Case Design shows how all the cases are being represented and User Interface Design shows some example of BAS interface.

System design will determine the success of the application. Other features include the components and elements of a system and their appearance to the user. Functional and non-functional requirements found in the system analysis stage are turned into design specification. From this, the expected outcome of the system can be viewed through the user interface design.

The system implementation, system testing and system evaluation will be continued in the next few chapters.

Chapter 6:
System Implementation

Chapter 6: System Implementation

System implementation is a phase or process of converting a system specification into an executable system (Sommerville, 2001). It takes place right after the system design phase. This phase describes how the initial and revised process design put into the real work. Therefore, system development, coding methodology and development tools are included in this phase.

6.1 Development Environment

The development environment of BAS application consists of software and hardware configuration. Using the suitable hardware and software will help in speed up the system development. The hardware and software tools that used to develop and document the system will be discussed as below.

6.1.1 Hardware

The hardware configurations used for developing the system are:

- AMD Athlon XP 1.8
- 512 MB running on 333 MHz memory
- 40 GB of free hard disk space
- 17inch Flat Monitor
- Geforce 4 64MB DDR display adapter
- Other standard computer peripherals

6.1.2 Software

The software tools that have been used to develop BAS are:

- Microsoft Windows 2000 Server with service pack 4
- Microsoft IIS 5.0 with a SMTP virtual server
- Microsoft Internet Explorer 6.0
- Microsoft .Net Framework 1.1
- Microsoft Visual Studio .Net 2003
- Microsoft SQL Server 8.0
- Microsoft Word 2003
- Adobe Photoshop 7.0
- Adobe Illustrator 10.0
- Macromedia Flash MX 2004

6.2 System Development and Implementation

BAS is coded with the advancement of Visual Basic .Net Language with Macromedia Flash MX interfaces. The Action Script (Flash) will work with VB.Net to produce a good system in term of both graphic user interface and logic behind the system. In addition, both data and cases will be store in XML format to reduce the third parties software cost for example the Microsoft SQL Server and Microsoft Access.

6.2.1 Business Advisory System Architecture

BAS is a windows application that able to communicate with the server database to get some necessary data that manipulate by other application e.g. EMS and Online Application Form (refer chapter 5 page. 76) through a web reference call BASWebServices.

6.2.1.1 BASWebServices

BASWebServices is a web services that builds using ASP.Net. It serves as a bridge to connect the BAS with the MS SQL Server.

The connection to accessing database is opened as follows:

Web.config

```
<appSettings>
      <add key="DBConnection"
            value="server=(local);database=BAS;uid=sa;pwd="/>
</appSettings>
```

serApp.asmx.vb

```
Dim scnConnection As New
      SqlConnection(ConfigurationSettings.AppSettings("DBConnection"))
```

The functionalities that provides are:

- **LoginValidation**

  Validate the user that login to the server by using the login name and password.

```
<WebMethod()> Public Function LoginValidation(ByVal strLoginName
              As String, ByVal strPassword As String) As Boolean

    Dim sqlCmd As New SqlCommand("spLogin", scnConnection)
    scnConnection.Open()
    sqlCmd.CommandType = CommandType.StoredProcedure
    sqlCmd.Parameters.Add("@Login_Name", SqlDbType.VarChar).Value
         = clsUtils.CEmptyToNull(strLoginName)
    sqlCmd.Parameters.Add("@Password", SqlDbType.VarChar).Value
         = clsUtils.CEmptyToNull(strPassword)

    Dim blnResult As New SqlParameter("@result",
         sqlDbType.VarChar, 10)
    sqlCmd.Parameters.Add(blnResult).Direction
         = ParameterDirection.Output
    sqlCmd.ExecuteNonQuery()
    scnConnection.Close()
    Return blnResult.Value

End Function
```

- **getUserProfile**

  Get the user profile from the server database and send to BAS via a dataset.

```
<WebMethod()> Public Function getUserProfile() As DataSet

    Dim sdaCommand As New SqlDataAdapter("spGetUser",
         scnConnection)
    scnConnection.Open()

    ' Mark the Command as a SPROC
    sdaCommand.SelectCommand.CommandType =
         CommandType.StoredProcedure
    Dim dsUser As New DataSet
    Try
        sdaCommand.Fill(dsUser, "USER_PROFILE")
    Catch
        dsUser = Nothing
    Finally
        scnConnection.Close()
    End Try
    Return dsUser

End Function
```

- **getAppProfile**

Get the applicant profile from the server database and send to BAS via a dataset according to the offered vacancy's name e.g. Programmer.

```
<WebMethod()> Public Function getAppProfile(ByVal strVacancy As
              String) As DataSet

    Dim sdaCommand As New SqlDataAdapter("spGetAPPList",
         scnConnection)
    scnConnection.Open()

    ' Mark the Command as a SPROC
    sdaCommand.SelectCommand.CommandType =
         CommandType.StoredProcedure

    ' Add Parameters to SPROC
    Dim sprVacancy As New SqlParameter("@Vacancy",
         SqlDbType.VarChar, 50)
    sprVacancy.Value = strVacancy
    sdaCommand.SelectCommand.Parameters.Add(sprVacancy)

    Dim dsAPP As New DataSet
    Try
        sdaCommand.Fill(dsAPP, "APPLICANT")
    Catch
        dsAPP = Nothing
    Finally
        scnConnection.Close()
    End Try
    Return dsAPP

End Function
```

- **getEmployee**

Get the employee profile from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getPerformance**

Get the employees past 6th months' performance from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getClaiming**

  Get the employees past 6th months' claiming expenses from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getComplaint**

  Get the employees past 6th months' complaining record from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getLate**

  Get the employees past 6th months' lateness record from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getAbsent**

  Get the employees past 6th months' absenteeism record from the server database and send to BAS via a dataset according to the department e.g. Sales.

- **getProfit**

  Get the company past 6th months' net profit from the server database and send to BAS via a dataset.

- **sendEmail**

  Send an email to applicant via the SMTP server that locates in the server.

```
<WebMethod()> Public Function sendEmail(ByVal strTo As String,
              ByVal strSender As String, ByVal strSubject As
              String, ByVal strTextBody As String, ByRef
              strErrorMsg As String) As Boolean
    Try
        SmtpMail.SmtpServer = "localhost"
        SmtpMail.Send(strSender, strTo, strSubject, strTextBody)
        strErrorMsg = "Email sent successfully."
        sendEmail = True
    Catch ex As Exception
        strErrorMsg = ex.Message
        sendEmail = False
    End Try
    Return sendEmail
End Function
```

6.2.1.2 BAS

BAS is a window application that builds using VB.Net and Flash MX 2004. It is an advisory system that deals with some human resource functions.

The communication between VB.Net and Flash MX are code as below:

Inside Flash MX

- On enter frame

```
stop();
txtName.tabIndex = 0;
txtPassword.tabIndex = 1;
btnEnter.tabIndex = 2;
btnClear.tabIndex = 3;
focusManager.setFocus(txtName);

//On "Enter" key down
Key.addListener(this);
this.onKeyDown = function() {
      if (Key.isDown(13)) {
      fscommand("ButtonClick", "btnEnter");
      }
}
```

- On button click

```
on (release){
      fscommand("ButtonClick", "btnEnter");
}
```

Inside VB.Net

- On form load

```
AxFlash.Movie = Application.StartupPath & "\BAS.swf"
AxFlash.Menu = False
```

- On button click

```
Private Sub AxFlash_FSCommand(ByVal sender As Object, ByVal e As
  AxShockwaveFlashObjects._IShockwaveFlashEvents_FSCommandEvent)
  Handles AxFlash.FSCommand

   If e.command = "ButtonClick" Then
      Select Case e.args
         Case "btnExit"
              Dim msgResult As MsgBoxResult
              msgResult = MsgBox("Are you sure you want to quit
              this application?", MsgBoxStyle.YesNo)
```

```
            If msgResult = MsgBoxResult.Yes Then
                Application.Exit()
            End If
        Case "btnEnter"
            mdlGlobal.strLoginName = ""
            mdlGlobal.strLoginName =
            AxFlash.GetVariable("txtName.text")
            mdlGlobal.strPassword = ""
            mdlGlobal.strPassword =
        FormsAuthentication.HashPasswordForStoringInConfigFile(
        AxFlash.GetVariable("txtPassword.text"), "SHA1")

            If mdlGlobal.strLoginName = "" And
               mdlGlobal.strPassword  = "" Then
              MsgBox("Please enter your login and password.")
              Exit Sub
            Else
              If clsUtils.localLogin(strMessage) Then
                 AxFlash.TGotoLabel("_root", "2nd")
                 strMessage = "Hello " &
                 mdlGlobal.strLoginName &
                 "! You may start using the Business
                 Advisory System by selecting the services."
                 strMessage = strMessage + vbNewLine +
                 "Mouse over on the button to view the
                 service's description."
                 AxFlash.SetVariable("message", strMessage)
              Else
                 MsgBox(strMessage)
              End If
            End If
        Case "btnRecruit"
            AxFlash.TGotoLabel("_root", "Recruit")
            strMessage = "Please type in the job name that you
            want to offer."
            AxFlash.SetVariable("message", strMessage)
        ...
    End Select
  End If

End Sub
```

It uses a web reference called BASWebServices to get data from server database. For example, the getEmployee function will first check the connections to the server then get the employees' profile from the server database through BASWebServices. After that, the data will be writing into files in xml format.

```
Public Function getEmployee(ByRef strErrorMsg As String, ByVal
        strPath As String) As Boolean

    Dim wsBAS As New BASwebServices.serApp
    Dim ds As New DataSet

    'Check the server connection
    If clsUtils.isServerConnected(strErrorMsg) Then
        If Not mdlGlobal.blnLoginServer Then
            LoginValidation(strErrorMsg)

        If mdlGlobal.blnLoginServer Then
            'Get the employees' profile from server
            ds = wsBAS.getEmployee(mdlGlobal.strDepartment)
        Else
            Return False
            Exit Function
        End If
    Else
        Return False
        Exit Function
    End If

    If Not ds Is Nothing And Not ds.Tables(0).Rows.Count = 0 Then
        Try
            'Write the xml and xml schema files.
            ds.WriteXml(Application.StartupPath + strPath + "/Emp.xml")
            ds.WriteXmlSchema(Application.StartupPath + strPath +
                "/EmpSchema.xml")
            getEmployee = True
        Catch ex As Exception
            strErrorMsg = ex.Message
            getEmployee = False
        End Try
    Else
        strErrorMsg = "Error getting Employee List form database"
        getEmployee = False
    End If
    Return getEmployee

End Function
```

BAS consists of 7 window forms and provides 4 major functions, which is recruit, select, retrench and cut down.

Main

The main form consists of login module and services selection menu.

- **Login Module**

  The user is requires to login by giving his/her login name and password (see figure 5.5). The password will be encrypted to increase the security level. There are 2 stages of login – local login and server login.

  - Local login

    The user will first login through the local database that previously get form server.

```
<USER_PROFILE>
   <USER_ID>1</USER_ID>
   <LOGIN_NAME>tsong</LOGIN_NAME>
   <PASSWORD>356A192B7913B04C54574D18C28D46E6395428AB</PASSWORD>
   <USER_NAME>Eric</USER_NAME>
</USER_PROFILE>
```

```
Public Function localLogin(ByRef strErrorMsg) As Boolean

    If File.Exists(Application.StartupPath + "\Data\User.xml")
          And File.Exists(Application.StartupPath +
          "\Data\UserSchema.xml") Then

       Dim dsUser As New DataSet
       dsUser.ReadXmlSchema(Application.StartupPath +
          "\Data\UserSchema.xml")
       dsUser.ReadXml(Application.StartupPath +
          "\Data\User.xml")

       Dim i As Integer
       For i = 0 To dsUser.Tables(0).Rows.Count - 1
          If dsUser.Tables(0).Rows(i).Item("LOGIN_NAME") =
               mdlGlobal.strLoginName And
               sUser.Tables(0).Rows(i).Item("PASSWORD") =
               mdlGlobal.strPassword Then
               Return True
               Exit Function
          Else
               localLogin = False
```

```
            End If
        Next
    Else
        Dim msgResult As MsgBoxResult
        msgResult = MsgBox("Error occur when login through local
            database." + vbNewLine + _
            "Do you want to login through server database and
            update your local database?", MsgBoxStyle.YesNo)
        If msgResult = MsgBoxResult.Yes Then
            'Login through server
            Dim clsWS As New clsWebServices
            localLogin = clsWS.LoginValidation(strErrorMsg)
        End If
        Return localLogin
    End If

End Function
```

- Server login

  The user will only login to the server when:

  – Error occur while login to the local database

  – Get data from server database

```
Public Function LoginValidation(ByRef strErrorMsg As String) As
                Boolean

    mdlGlobal.blnLoginServer = False
    If clsUtils.isServerConnected(strErrorMsg) Then
        Try
            mdlGlobal.blnLoginServer =
            wsBAS.LoginValidation(mdlGlobal.strLoginName,
            mdlGlobal.strPassword)
            If mdlGlobal.blnLoginServer = False Then
                strErrorMsg = "Error occur when login through
                server database." + vbNewLine + _
                "Please check your user name and password or
                kindly contact the company's database admin."
            Else
                'Update the local user profile
                If Not getUserProfile(strErrorMsg) Then
                    strErrorMsg = "Error updating your local
                    database." + vbNewLine + _
                    "Please contact the system admin."
                End If
            End If
        Catch ex As Exception
            strErrorMsg = ex.Message
        End Try
    End If
    Return mdlGlobal.blnLoginServer

End Function
```

- **Services Selection Menu**

  Provides a user friendly menu and a brief explanation about the services for the user to select the functions that need to be performs (see figure 5.6).

Recruitment

The recruit form will perform the functionalities of the recruitment. It allows users to define sets of minimum requirements while recruiting process occurs. In addition, it also allows users to delete, reuse and edit the old cases that created previously. The edit function here is actually an add function where it will add a new case to the knowledge base according to the old case.

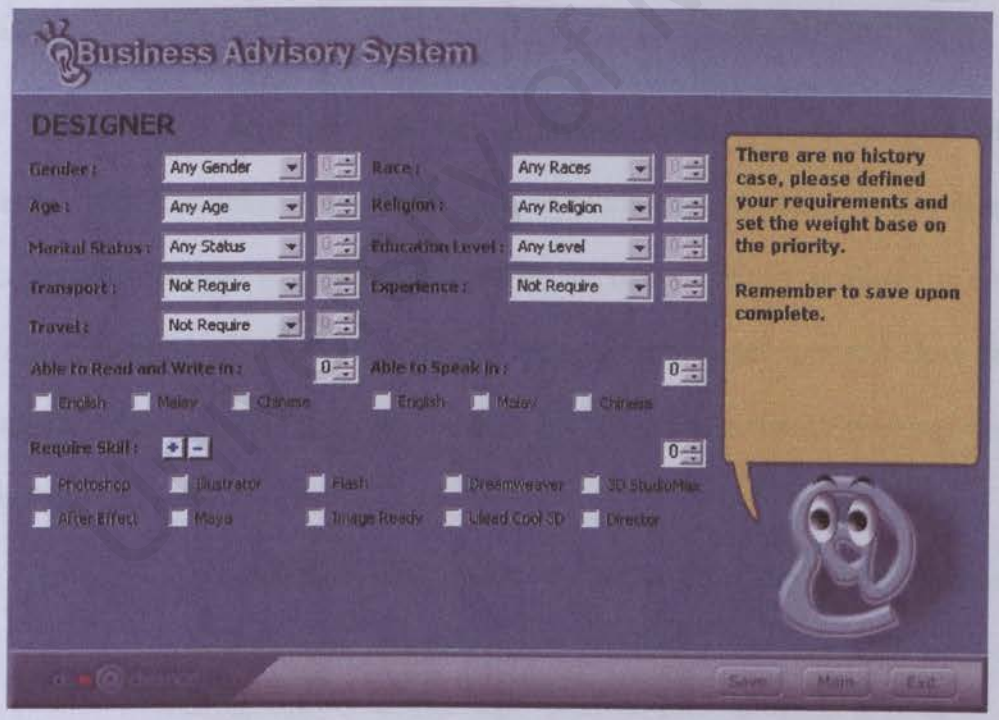


Figure 6.1: Creating a new case for the job "Designer".

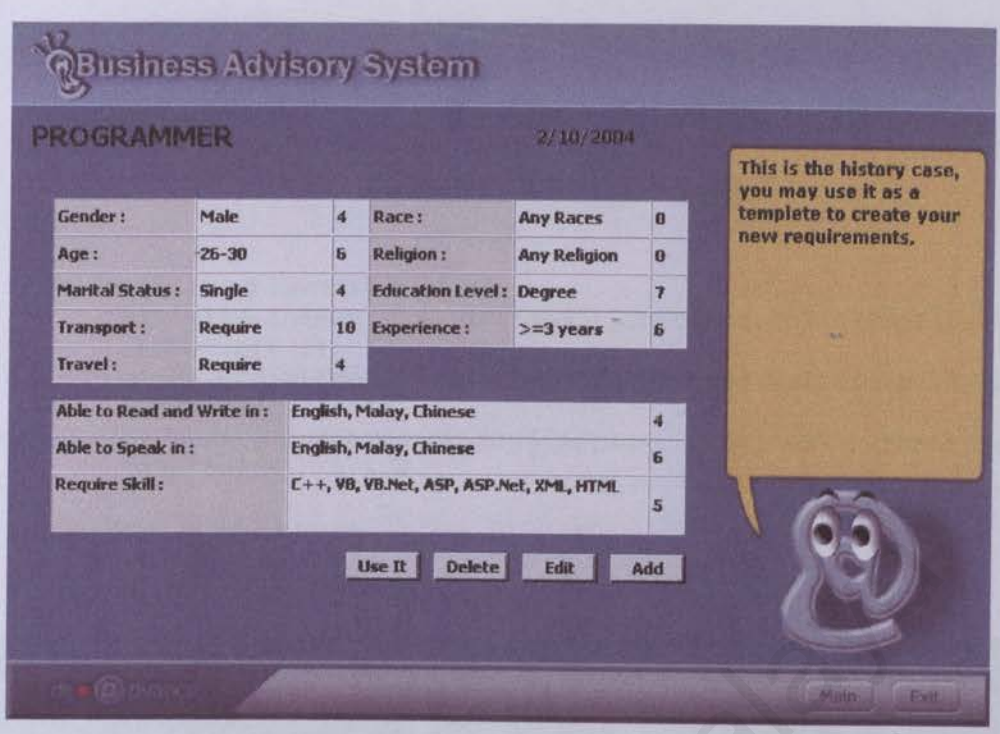


Figure 6.2: The history case for the job "Programmer".

Selection

The selection form will get the applicants' data form server database and compare it with a selected case that created in recruitment function. A list of qualify applicant will be display in a data grid. Base on the result, the user is allows to view the individual applicant's data, email to the applicant, accept the solution and close the case without accepting anyone. A new case will be created automatically when the user accepted the solution.

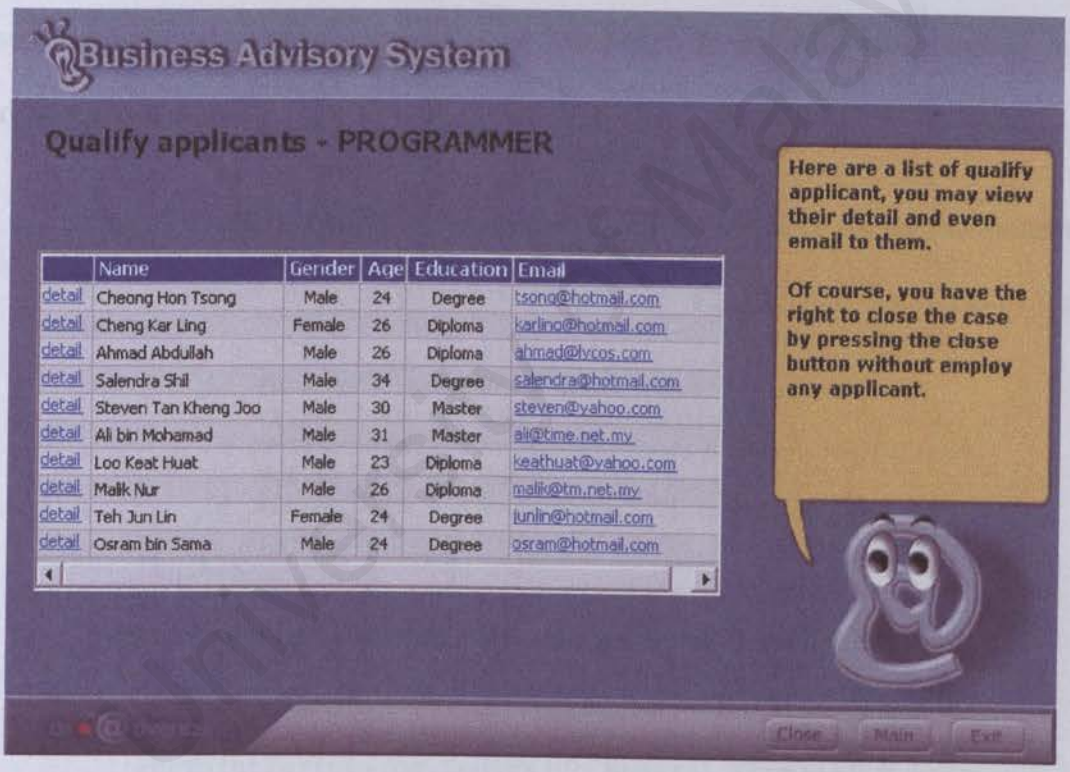
- String to integer function – give a value to the age group for the comparison.

```
Public Function strTointAge(ByRef strAge As String) As Integer

        If strAge = "below 21" Then Return 1
        If strAge = "21-25" Then Return 2
        If strAge = "26-30" Then Return 3
        If strAge = "31-35" Then Return 4
        If strAge = "above 35" Then Return 5

End Function
```

- Comparing Function – by using the nearest neighbourhood method, which is

Similarity = min/max * weight

```
'Age
dblSum(i) = dblSum(i) +
     min(strTointAge(CInt(dr("AGE").ToString.Split(cSplit)(0))),
     intRecruit(1)) /
     max(strTointAge(CInt(dr("AGE").ToString.Split(cSplit)(0))),
     intRecruit(1)) *
     CInt(dsTemp.Tables(0).Rows(intRow).Item("AGE").ToString.Split
     (cSplit)(1)) * 1.0
```



	Name	Gender	Age	Education	Email
detail	Cheong Hon Tsong	Male	24	Degree	tsong@hotmail.com
detail	Cheng Kar Ling	Female	26	Diploma	karling@hotmail.com
detail	Ahmad Abdullah	Male	26	Diploma	ahmad@lycos.com
detail	Salendra Shil	Male	34	Degree	salendra@hotmail.com
detail	Steven Tan Kheng Joo	Male	30	Master	steven@yahoo.com
detail	Ali bin Mohamad	Male	31	Master	ali@tme.net.my
detail	Loo Keat Huat	Male	23	Diploma	keathuat@yahoo.com
detail	Malik Nur	Male	26	Diploma	malik@tm.net.my
detail	Teh Jun Lin	Female	24	Degree	junlin@hotmail.com
detail	Osram bin Sama	Male	24	Degree	osram@hotmail.com

Figure 6.3: List of qualify applicant for the job "Programmer"

Retrenchment

The retrench form helps the user to do retrenchment according to the employees' performance, claiming expenses, complaint, lateness, absenteeism, basic salary and years of services. With these 7 factors, a value will be given to each and every employee, which calculates using the nearest neighbourhood method. The weight for each factor will be assign by the user before calculating.

The user will give the desire net profit, but it must be logic enough to achieve. Base on this net profit and the calculated value for each employee, the system will advise on how many employees will need to retrench and who will suppose to be retrenched.

- String to integer function – give a value to the performance for the comparison.

```
Public Function strTointPerform(ByRef strPerform As String) As
Integer

        If strPerform = "Very Bad" Then Return 1
        If strPerform = "Bad" Then Return 2
        If strPerform = "Average" Then Return 3
        If strPerform = "Good" Then Return 4
        If strPerform = "Well Done" Then Return 5

End Function
```

- Comparing Function – by using the nearest neighbourhood method.

```
Case "PERFORMANCE"
dblCondition = dblCondition +
   (strTointPerform(ds.Tables(0).Rows(i).Item(j)) / 5) *
   dsRetrench.Tables(0).Rows(intRow).Item("PERFORMANCE") * 1.0
```

- Determine how many employees will need to be retrenched and who will be retrenched.

```
Dim dblAContribute As Double

dblAContribute = dblAProfit / CDbl(dsEmp.Tables(0).Rows.Count)
```

```
For i = 0 To dsEmp.Tables(0).Rows.Count - 1
    If (dblAFigure + dblAProfit) > CDbl(strProfit) * 95 / 100
        Then Exit For
    intCounter = intCounter + 1
    For j = 0 To dsClaim.Tables(0).Rows.Count - 1
        If dsClaim.Tables(0).Rows(j).Item("EMPLOYEE_ID") =
                dr(i)("EMPLOYEE_ID") Then
            For k = 2 To dsClaim.Tables(0).Columns.Count - 1
                dblAClaim = dblAClaim +
                CDbl(dsClaim.Tables(0).Rows(j).Item(k))
            Next
            dblAClaim = dblAClaim /
            CDbl(dsClaim.Tables(0).Columns.Count - 2)
            Exit For
        End If
    Next
    dblAFigure = dblAFigure + CDbl(dr(i)("BASIC_SALARY")) +
    dblAClaim - dblAContribute
Next
```

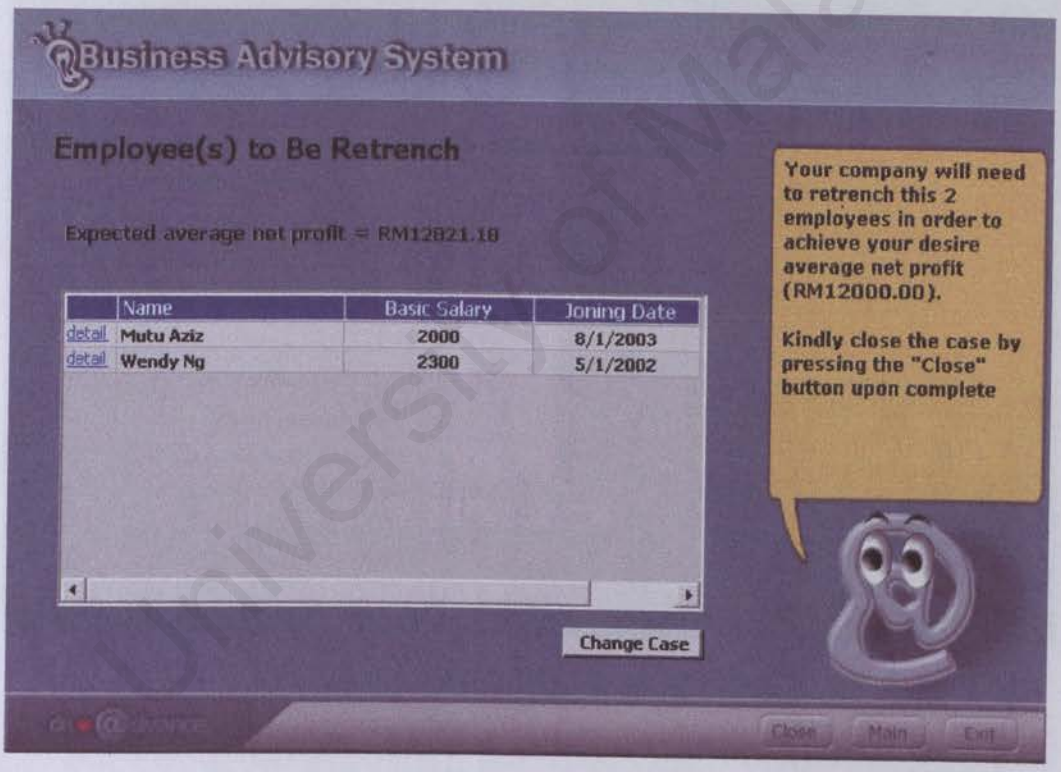


Figure 6.4: The advice for the retrenchment process.

Cut Down

The cut down form helps the user to do some adjustment on his/her employees' income to help in overcome the company's financial problems. The algorithm use here is similar to the algorithm of retrenchment. The only different is the user will need to define the maximum percent of the salary cutting and the maximum employee to be cut. For example, figure 6.5 is the case that has been used to generate the result of figure 6.6.

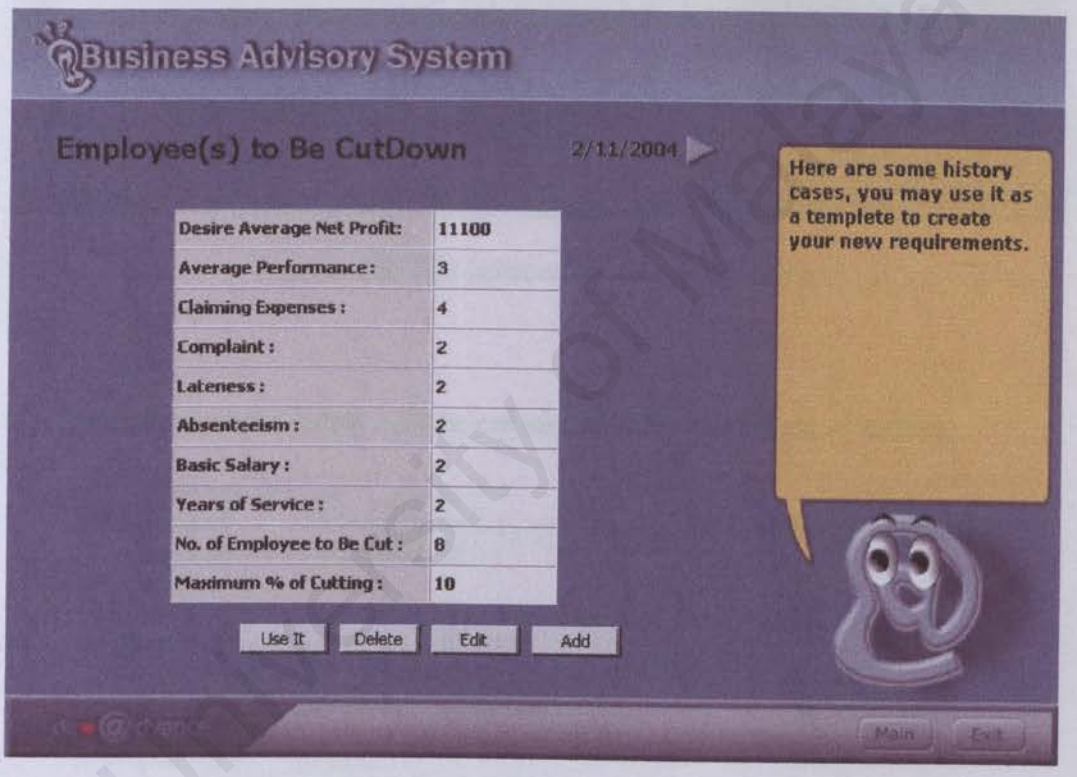


Figure 6.5: A history case for cut down.

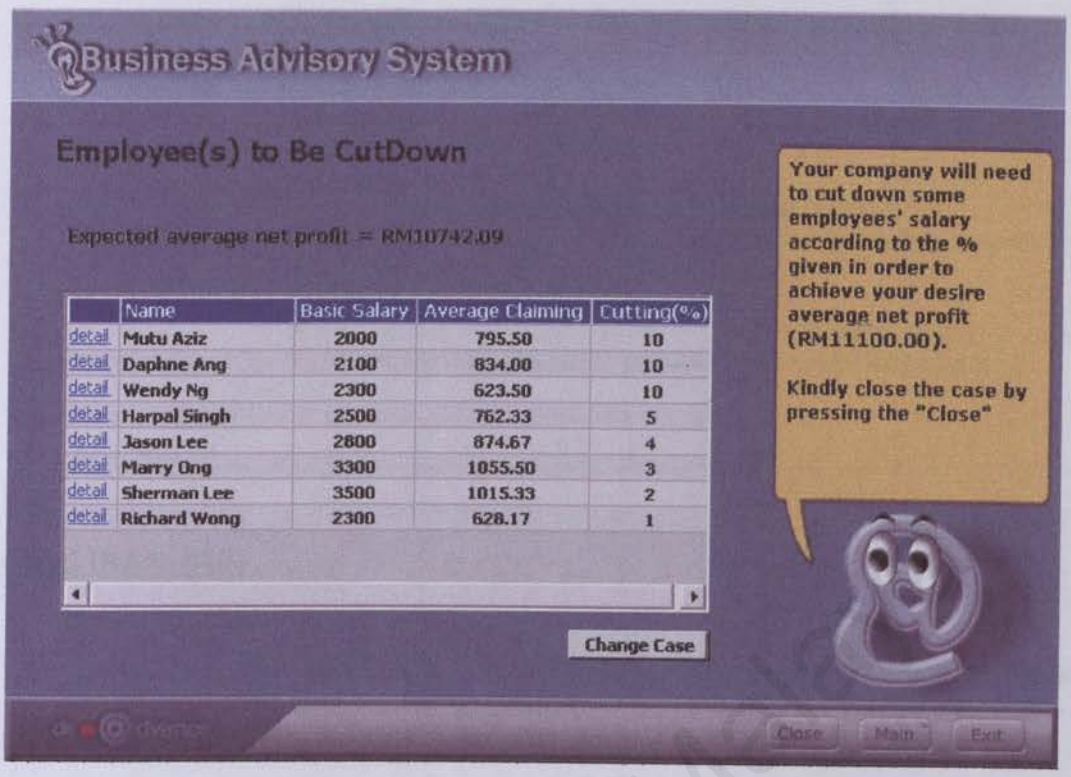


Figure 6.6: The advice for salary cut down.

- Determine whose salary will be cut and how many percent of cutting.

```
Dim intPercent As Integer
intPercent = clsSelect.min(CInt(mdlGlobal.strCutPercent),
CInt(mdlGlobal.strNoOfEmp))
Dim intRow As Integer
intRow = clsSelect.min(CInt(mdlGlobal.strNoOfEmp),
(dsEmp.Tables(0).Rows.Count)) - 1
Dim intCutPercent(intRow) As Integer
Dim dblAClaim(intRow) As Double

For i = 0 To intRow
    For j = 0 To dsClaim.Tables(0).Rows.Count - 1
        If dsClaim.Tables(0).Rows(j).Item("EMPLOYEE_ID") =
            dr(i)("EMPLOYEE_ID") Then
            For k = 2 To dsClaim.Tables(0).Columns.Count - 1
                dblAClaim(i) = dblAClaim(i) +
                CDbl(dsClaim.Tables(0).Rows(j).Item(k))
            Next
            dblAClaim(i) = (dblAClaim(i) /
            CDbl(dsClaim.Tables(0).Columns.Count - 2)) * 1.0
            Exit For
        End If
    Next
Next
```

```
While Not (dblAFigure + dblAProfit) > CDbl(strProfit) * 95 / 100
      intCounter = intCounter + 1
      For i = 0 To intRow
          If (dblAFigure + dblAProfit) > CDbl(strProfit) * 95 /
              100 Then Exit While
          If Not intCutPercent(i) =
                CInt(mdlGlobal.strCutPercent) Then
             intCutPercent(i) = clsSelect.min(intCutPercent(i) +
             (intPercent - i) * intCounter,
             CInt(mdlGlobal.strCutPercent))
             dblAFigure = dblAFigure + (intCutPercent(i) / 100)
             * (CDbl(dr(i)("BASIC_SALARY")) + dblAClaim(i))
          End If
      Next
End While
```

Add and Delete Skill

The add skill form and delete skill form are the add-on functions for recruitment. As their name suggest the add skill form provides the skill adding function whereas the delete skill form provides the skill deleting function.

- Add Skill

```
Dim strSkill As String
Dim dsSkill As New DataSet
Dim drSkill As DataRow
Dim intCount As Integer

If File.Exists(Application.StartupPath + "/Cases/Skill.xml")
       And File.Exists(Application.StartupPath +
       "/Cases/SkillSchema.xml") Then
   Try
      dsSkill.ReadXmlSchema(Application.StartupPath +
      "/Cases/SkillSchema.xml")
      dsSkill.ReadXml(Application.StartupPath +
      "/Cases/Skill.xml")
   Catch ex As Exception
      MsgBox(ex.Message, MsgBoxStyle.Critical)
   End Try
Else
   Dim dcSkill As DataColumn = New DataColumn
   Dim dtSkill As DataTable = New DataTable("Skill")

   'Create Primary Key with default value = skill_id
   dcSkill.DataType = System.Type.GetType("System.String")
   dcSkill.AllowDBNull = False
   dcSkill.Unique = True
```

```
    dcSkill.Caption = "SKILL_ID"
    dcSkill.ColumnName = "SKILL_ID"
    dcSkill.DefaultValue = "Skill_ID"

    'Add columns
    dtSkill.Columns.Add(dcSkill)
    dtSkill.Columns.Add("JobName", Type.GetType("System.String"))
    dtSkill.Columns.Add("Skill", Type.GetType("System.String"))

    'Add datatable to dataset
    dsSkill.Tables.Add(dtSkill)
    dcSkill = Nothing
    dtSkill = Nothing
End If

For intCount = 0 To 4
   Select Case intCount
      Case 0
           strSkill = txtSkill1.Text
      Case 1
           strSkill = txtSkill2.Text
      Case 2
           strSkill = txtSkill3.Text
      Case 3
           strSkill = txtSkill4.Text
      Case 4
           strSkill = txtSkill5.Text
   End Select

   If Not strSkill = "" Then
      'Add data to dataset
      drSkill = dsSkill.Tables(0).NewRow
      drSkill(0) = clsUtils.GenerateID
      drSkill(1) = mdlGlobal.strJobOffer
      drSkill(2) = strSkill
      dsSkill.Tables(0).Rows.Add(drSkill)
   End If
Next

Try
   'Write to xml file
   dsSkill.WriteXml(Application.StartupPath +
   "/Cases/Skill.xml")
   dsSkill.WriteXmlSchema(Application.StartupPath +
   "/Cases/SkillSchema.xml")
Catch ex As Exception
   MsgBox(ex.Message, MsgBoxStyle.Critical)
End Try

drSkill = Nothing
dsSkill = Nothing
```

- Delete Skill

```
Dim msgResult As MsgBoxResult
Dim i As Integer
msgResult = MsgBox("Are you sure you want to delete the skill -
" + e.RowValue("Skill") + "?", MsgBoxStyle.YesNo)

If msgResult = MsgBoxResult.Yes Then
   For i = 0 To dsSkill.Tables(0).Rows.Count - 1
       If dsSkill.Tables(0).Rows(i).Item("SKILL_ID") =
           e.RowValue("SKILL_ID") Then
          dsSkill.Tables(0).Rows(i).Delete()
          Kill(Application.StartupPath + "/Cases/Skill.xml")
          dsSkill.WriteXml(Application.StartupPath +
          "/Cases/Skill.xml")
          Exit For
       End If
    Next
    If Not BindGrid(strMessage) Then
       MsgBox(strMessage)
       Call FormRecruit()
    End If
    MsgBox("1 skill deleted")
End If
```

6.3 Chapter Summary

The development of BAS has been carefully planned and it does take a period of precious time. A lot of the functionalities as described in the previous chapter have been implemented orderly in time. Some functions that are not fully implemented would be extended in future. In short, BAS has fully utilized the Microsoft Technologies to build up the system. Whether the application is strong enough to be the security tool depends on the testing. The next chapter would draw the system testing procedures on BAS.

Chapter 7: System Testing

Chapter 7: System Testing

System testing is a significant and critical phase that ensures the system fulfills the user's requirements and assures the quality of the delivered system. While talking about the system testing, it always related with verification and validation (V & V).

"V & V is the name given to checking and analysis processes that ensure that software conforms to its specification and meets the needs of the customers who are paying for that software."

By Sommerville, 2001

Testing provides a method to discover logical error and to test the system reliability. It is done throughout system development, not just at the end. This is because system that is failed after installation will result a waste in cost, time and effort. However successful testing will result in quality software with less errors and work according to specification.

Several testing stages that involve during the development of the system are:

1. Unit testing
2. Integration testing
3. System testing

7.1 Unit Testing

In this stage, testing will be concentrated on the smallest component of the system for testing. Each individual component is tested independently without other system components, to ensure that they operate correctly. This emphasizes an important fact about testing. It demonstrates the presence, not the absence, of the program faults. For instants, this component might perform task like checking valid input value.

In BAS, some of the units that were tested independently are:

- The password is encrypted correctly.
- The double login criteria.
- The communication between VB.Net and Flash MX.
- Each and every function of the web services.
- Major functional testing (recruitment, selection, retrenchment and cut down).
- Add and delete skills.
- Unsolved case.
- Close case.
- Accept case.

BAS has integrating three types of unit testing method to test each of the unit listed above, namely

1. Ad hoc Testing
2. White Box Testing
3. Black Box Testing

7.1.1 Ad Hoc Testing

Ad Hoc or Ad Lib testing means simply play with the functioning unit, trying whatever comes to the mind, in attempt to make it fail. This type of testing was a fast and efficient way of debugging code errors during the early development stage.

The disadvantage of Ad Hoc testing is it usually finds many errors and never be sure what was or was not to be tested.

7.1.2 White Box Testing

White Box Testing, also known as structural testing, basically involved analyzes the structure of the code and use knowledge about the structure of a component to derive test data. The advantage of white box testing is that an analysis of the code can be used to find out how many test cases are needed to guarantee a given level of test coverage.

Many codex testing is conducted using the white box testing approach. This includes the basic testing on path like Windows Directory, data flow and loop testing. In short, the white box testing methodology is focused in term of coverage. And the vital part of this testing is to check for any missing function.

7.1.3 Black Box Testing

Black Box Testing, also known as functional testing, is concentrate on the functionality of code. The main objective is to uncover those wrong functions programmed correctly by feeding the input to the black box and take notes on what

output is produced. The test object's behaviour can only be determined by studying its inputs and the related outputs (Sommerville, 2001).

The advantage of this kind of testing is that a black box is free of the constraints imposed by the internal structure and logic of the test object. However the disadvantage is that it is not always possible to run a complete test in this manner due to the selection of an effective input. Black box testing is a test on boundary value analysis, error guessing and domain testing.

7.2 Integration Testing

After all components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. This step is called integration testing, also known as module testing, which verifies that the all the components work together as described in the module or system design specifications.

During the integration testing, two or more units in which either unit that use output data from or provide input data for another unit were tested in collection. The combination of many modules would test the integration of the system. Some integration may result an unexpected error. Using the integration testing approach, this kind of errors will be discovered and corrected.

The order in which components are tested affects our choice of test cases and tools. The system is viewed as a hierarchy of components, where each component belongs to a layer of the design. In this system, the Top-down Integration approach is used where testing begins from the top and works the way down. The process is continued until all the modules are tested.

7.3 System Testing

System Testing is the last testing procedure. It is performed to uncover its limitations, measure its capabilities and make certain that the entire system works according to users' specifications. Developers will join the users to perform this stage of testing where the system is checked against the users' requirements description.

System modification will be implemented if there is a need to change or do not met the users' requirements specifications. If the users are satisfied with the system's characteristics, the system is ready to be deployed for use. The testing result will show whether or not the entire system specifications and objectives are achieved.

7.3.1 System Test Considerations

In system testing, the behaviour of the individual functions and functional tests also involved:

- **The Event List**

  All the possible triggers are exercised and the expected results compared with the actual results. Every function is tested by one or more events in the event lists.

- **Error Message Testing**

  The error message, which can be generated by the system during invalid data entry are checked for spelling, appropriateness and consistence.

Acknowledgement messages also will also implement the same test. It is the message that informs the user about the state of a user request process.

- **Security testing**

  In security testing, the system is tested for improper penetration and unauthorized access, to ensure that the implementation of the user login and the valid user checking procedures especially the double login criteria.

- **Transaction Tracking**

  During transaction tracking, a list of possible transactions is tracked through the system to ascertain that they function correctly from input to output. For example, every time a screen is reached which requires input or generates input, the appropriate functions are processed and lead to subsystem for processing and then the right output is retrieve.

  This test was implementing and all the function behaves according to the requirements specification.

7.4 System Fundamental Testing

There are other tests fundamental to all software. Certain of these are difficult to measure accurately. Four of these fundamental tests are:

- **Usability**

  The usability should be based in building user friendly interfaces. The user should be able to learn how to use this software in a easy and fast way.

- **Install Ability**

  How easy is it for a novice to install the software correctly and easily without consult an expert?

- **Performance**

  Performances tests are conducted to ensure that the system response time meet user expectations and do not exceed the specified performance criteria under heavy stress or volume. During these tests, response time and the transaction rate are measured; the purpose of performance tests is to test-run the performance of various functions of the software within a specified hardware configuration. The performance tests can couple this test with stress testing.

- **Reliability**

  Reliability tests are conducted, according to mathematical models of software reliability, to ensure that the system can be probability of some function of the system failing within a specified time. Reliability testing is monitoring the mean time between failures. Reliability and consistency

testing go hand in hand where the system behavior (inputs, outputs, response time) is measured for consistency.

In conclusion, the testing steps are shown as below.

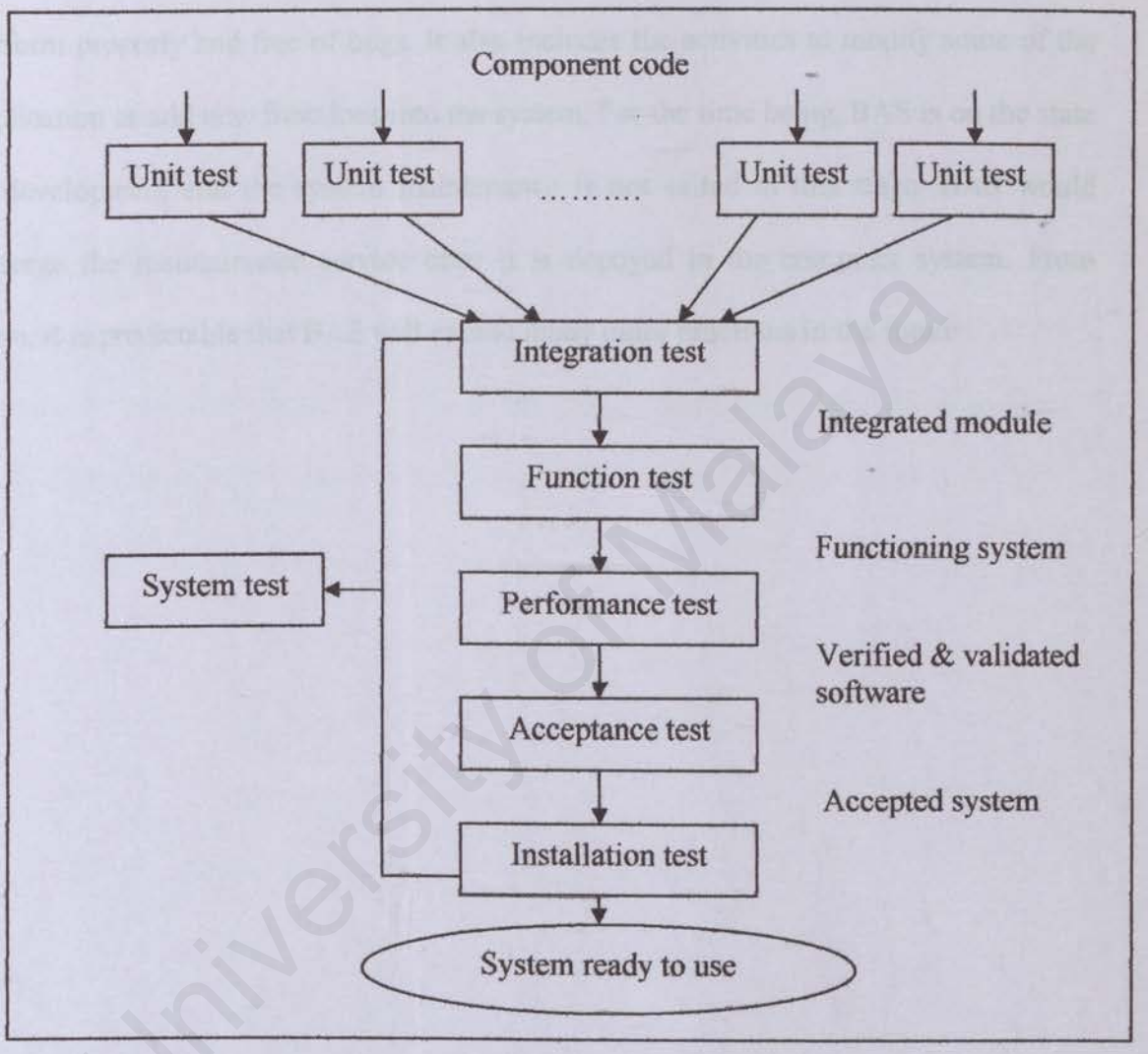


Figure 7.1: Testing Steps

7.5 System Maintenance

Usually in system development, maintenace will be conducted once the system is finished or delivered. The maintenace servises would assure the system function can perform proporly and free of bugs. It also includes the activities to modify some of the application or add new functions into the system. For the time being, BAS is on the state of development and the system maintenance is not suited in this stage. BAS would undergo the maintainance service once it is depoyed in the computer system. From there, it is predictable that BAS will extend many more functions in the future.

7.6 Chapter Summary

BAS has been undergone many testing to assure that the entire application is strong enough and no defect when user drive it in their computer. Due to the private and confidential (P&C) problem, BAS has been design with a relatively high in security to ensure that the unauthorized users would not be able to intrude into the application easily. Moreover, the double login criteria will once again ensure that the former employees, which an authorized users formerly, will turn to an unauthorized users. Consequently, all functions beyond BAS must be carefully and fully tested in all ways so that the application would likely to be the perfect application. The following chapter would deals the system evaluation on BAS and the conclusion on BAS development.

Chapter 8: System Evaluation and Conclusion

Chapter 8: System Evaluation and Conclusion

Evaluation is a process that occurs continuously at all phases of the system development. Evaluation phase was to determine the extent to which the system the expected outcomes have been realized, and the prescriptive value of the process where extraneous factors were taken consideration. Lastly, conclusion will be making for this system.

8.1 System Evaluation

System Evaluation in the BAS contains 4 different elements to be discussed. The coverage of the evaluation as below:

- Difficulty and Suggested Solution
- System Strength
- System Limitation
- Future Enhancement

8.1.1 Difficulty and Suggested Solution

In every project, problems always occur during system development. This does not vary to the development of BAS. Many problems have kept unfolding one after another as development work progressed due to many reasons. Some of the problems are unexpected and difficult to be overcome. Many of the problems can be handles rather easily.

During Analysis Phase

- **Determining Scope of the BAS**

  Due to the lack of the knowledge in business domain, is really hard to determine the scope of BAS. The scope of BAS must take consideration on time constraint so that it can be completed within the period of time. However, this is handled right after studying and analyzing several applications and their capabilities with the similar functionalities as BAS suggested. In addition, the suggestion form the domain experts also give a big hand on it. Then the whole picture of BAS is cleared and the scope is determined on the basis of moderately.

- **Determining the Artificial Intelligence Technique**

  Artificial intelligence can be considered as a new approach. Thus, with limited or even without references and existing system, it's very difficult to determine which technique to be use. For instance, the Rule-base Expert System and the Case-base Reasoning (CBR) approach. Fortunately, after studying and analyzing both approach and review the suggestion form the project supervisor, CBR has been chosen as an AI technique to be used.

During Design Phase

- **Time Constraint**

  There was not enough time to study, learn and produce the best solution of design in BAS. Mainly, this was cause by inexperience and insufficient

knowledge of designing a full extend of an application software. Furthermore, time is needed to study and explore the programming languages and the designing tools as well deeply before knowing how to integrate these technologies and languages in the process of developing and problem solving. Thus the best way is to study as many approaches used in seniors' documentation and the internet as well.

During Implementation Phase

- **No prior experience in the chosen programming tools**

  There was a learning curve in understanding how the Visual Basic.Net, ASP.Net, Action Script works since inexperience in those languages. Programming using new approach such as the communication between the .Net environments and Flash MX requires some knowledge of how the component should be used and what will need to do in order to build the required functionality of BAS. The best way of learning the coding is to study as many examples as possible. Another way to studying the programming language is by consulting the experts through internet forum such as Google Groups and Experts-Exchange.

During Testing Phases

- **Time consuming**

  It's undeniable to say that the system testing process is the most time consuming process. The iteration of testing cycle must be gone through

for quite a number of times to ensure that the system is running correctly. Fortunately, it's being done in the very last minute.

8.1.2 System Strengths

BAS has many strengths compare to the similar existing application in the market. This can be well-described as follow.

- **System Transparency**

  BAS has a high level of system transparency. In term of system transparency, the user does not need to know where the web services is, what did the web services server do, how the application is structured and how the application get necessary data from the server database. The user only need to know how to communicate with the system and give necessary inputs to the system, then BAS will do the rest for the user.

- **Systematic Error Handling**

  Input of the users will be validated and verified to prevent the errors caused by invalid input. If there is any error or invalid input occurred, an error message is generated and displayed to inform the user about the error. For example the message – "Invalid user name or password" will be prompt out when the login validation fail.

- **Outstanding Interface**

  The user interface of BAS was created using the power of Adobe Photoshop, Adobe Illustrator and Macromedia Flash MX. Unlike usual window

application, BAS is a borderless and integrated with some animation. It is an outclass work of design. As conclusion, BAS is user-friendly, nicely organized, elegantly arranged and sooth to the eyes as well.

- **Easy to use**

  It is relatively easy for the user to use this system with the guides of a help agent on the right of the programme throughout the whole route of using this system.

- **Security**

  The encrypted password and double login criteria of BAS will not only ensure that the unauthorized users from being using this system, it also disallows the former authorized users from being getting the private and confidential company data from the server database.

- **Artificial Intelligence Approach**

  With the CBR approach, this system vary from other conventional system, allows the imperfect matching method to generate a desire result or advice for the user.

- **"Amphibious" criteria**

  BAS tend to be in an "amphibious" condition, which is able to work as a client application in the client-server architecture to communicate with the server database through BASWebServices and work as a stand alone window application with local xml database.

8.1.3 System Limitations

Despite some of the system strengths mentioned previously, there are limitations, which cannot be researched and developed due to time constraint. Those limitations are:

- **The Case Features**

  All the features that use in creating cases of BAS are hard coded. It cannot be add, edit or delete by the user if they wish to modify the way they evaluate the applicants and employees.

- **Rely on Foreign Database**

  BAS will need external data from foreign database, for instance the Employee Management System's database and the Online Application Form's database, to perform the selection, retrenchment and cut down functions.

- **Not support multiple languages**

  The current developing system is only limited to one language (English) only. This is due to the time limitation. But it still can enhance to support more languages to adapt to other environments that use different languages.

- **Print Function not Integrated**

  BAS didn't integrate with the print function which will enable the user to print out the result or advice given by the system.

- **Automatically Generate an Online Application Form**

  It's better to automatically generate an Online Application Form right after the user defining the requirement case. So that, the recruitment process can be carry out immediately.

- **Automatically Generate a Retrenchment and Salary Cut Down Letter**

  It will be nice if the system could automatically generate the retrenchment and Salary Cut Down letter to the particular employee. The letter should stage why this action has been applied to the particular employee.

8.1.4 Future Enhancement

BAS at the present time does have many spaces to be evolved and extended to cover many more exiting functions. It is a customizable application to feed the need of the contemporary user. Lots of additional functions should be added to enhance the usage of the system. Here are some future enhancements suggested for the BAS:

- **Dynamic Case Features**

  The case features should be able to define by the user dynamically without hard coded into the system. Thus, this function can be integrated to enhance the usability of BAS.

- **Support multiple languages**

  BAS can be enhanced to provide multilingual especially in English, Malay and Chinese to fulfil the different users' requirements.

- **Integrated with Print Function**

  The usability of BAS can be enhanced if the results' or advices' printing function can be integrated.

- **Automatically Generate an Online Application Form**

  The effectiveness of recruitment process will be able to enhance with the automatic generating of Online Application Form function right after the recruit case defining process in BAS.

- **Automatically Generate a Retrenchment and Salary Cut Down Letter**

  The workload of the manager will be dramatically reduced with the automatic generating of retrenchment and salary cut down letter function in BAS.

8.2 System Conclusion

BAS is an advisory system that builds upon the latest technologies, which is using the CBR approach and programmed using the combination of a few latest programmer languages such as VB.Net, ASP.Net Web Services, XML and Action Script in Flash MX.

In the development of BAS, many measurements have been taken to ensure that the outcome of BAS does suit the need of many users, no matter the novice or the professional user. It is highly created on the basis of user-friendly interface, reliable and securable systems do not consume many computer resources. As a result, BAS has met all the quality stated above.

The development of BAS is using the Waterfall Model. This is a very excellent methodology for the inexperience to develop an application, as it will step by step lead the developer from the first step, requirement analysis and definition, to the very last step, operational and maintenance. Thus, the inexperience developer would not fell helpless in the whole the process of development.

Throughout this project, a lot of valuable knowledge and experience has learned and gained. During the period of system development, there are become clearer on how to establish the connection to database, how the web services works, how VB.Net communicates with Flash MX and how XML can behave as a local database. Besides the programming skill, the knowledge in business domain also has been gained throughout the studying and analyzing process to determine the scope of BAS.

Furthermore, skills in using software such as Macromedia Flash MX, Microsoft SQL Server, Adobe Photoshop and Adobe Illustrator have been acquired.

During the project development, programming skills and good practice on software engineering techniques are essential and must also be applied in efficient and efficiency way. Therefore, this project has provided the good chances to experience using the method, techniques, paradigms, and approaches that learned from System Analysis & Design and Software Engineering courses in the second year and third year study respectively.

In is very high expectation that BAS would be extended the functionality in the future. Some additional functions like print function, letter generating function, application form generating function, and etc should be included in the future version of BAS.

As a conclusion, BAS has successfully developed, able to deliver in time and achieved all the objectives and requirements as determine during analysis phase. BAS is an excellent system that must be tried and use in every company especially the well establish corporate.

Bibliography

Aamodt A., Plaza E. (1994). *AICom - Artificial Intelligence Communications, IOS Press.* Vol. 7: 1, pp. 39-59.

Bratko, I. (2001). *Prolog – Programming for Artificial Intelligence.* 3rd ed. Pearson Education Limited.

Brown, C. (1995). *Some Background on User Interfaces.* (1995, August 18).

Durkin, J. (1994). *Expert System – Design and Development.* Macmillian Publisher.

Edelstein, H. (1994). *Unraveling Client/Server Architecture.* (May 1994)

Forsyth and Richard., *Expert systems Principles and Case Studies*, Chapman and Hall

Gallaugher, J. (1995). *The Critical Server Architecture: A Comperison of Two and Three Tier Systems (1995, July 28).*

Gardner, T. (2001). *An Introduction to Web Services.* (2001, October 02).

Kolodner, J. C. (1993). *Case-based Reasoning.* Morgan Kaufmann Publisher.

Luger, G. F. (2002). *Artificial Intelligence – Structures and Strategies for Complex Problem Solving.* 4th ed. Pearson Education Limited.

Riesbeck, C. K., Schank, R. C. (1989). *Inside Case-based Reasoning.* Lawrence Erlbaum.

Sommerville, I. (2001). *Software Engineering.* 6th ed. Pearson Education Limited.

Turban, E. (1995). *Decision Support System and Expert System.* 4th ed. Prentice Hall.

Appendix A: Job Description

de-Advance Sdn. Bhd.

Vacancy

Job Title: Web Developer
Department: MIS
Reports To: Mr. Eric
FLSA Status: Department Manager
Prepared By: Mr. Cheong
Prepared Date: 21 August 2003
Approved By: Mr. Brandon
Approved Date: 23 August 2003

Summary Builds software applications to provide the technical architecture of Web sites or Web-based software products by performing the following duties.

Essential Duties and Responsibilities include the following. Other duties may be assigned.

Gathers feedback from design and technical staff on Web site development needs.

Builds user interface applications and back-end databases using various programming and scripting languages.

Authors complex Web pages.

Assists in project planning and Web site design.

Creates prototypes and functional specifications for software projects.

Creates technical methodologies for engineering solutions to Web-based development problems.

Meets with engineering personnel on specifics of projects. new technologies, and deadlines.

Determines new Web technologies to utilize, such as browsers, languages, and plug-ins. based on company's needs.

Conceptualizes long-term needs of Web development, and plans and manages related projects.

Makes suggestions on creating Web-based technical standards for specific Web sites and the company as a whole.

Supervisory Responsibilities

This job has no supervisory responsibilities.

Competencies

To perform the job successfully, an individual should demonstrate the following competencies:

Design - Generates creative solutions; Applies design principles.

Problem Solving - Identifies and resolves problems in a timely manner; Gathers and analyzes information skillfully; Develops alternative solutions; Works well in group problem solving situations; Uses reason even when dealing with emotional topics.

Project Management - Coordinates projects; Communicates changes and progress: Completes projects on time and budget.

Technical Skills - Assesses own strengths and weaknesses; Shares expertise with others.

Customer Service - Responds promptly to customer needs; Meets commitments.

Interpersonal Skills - Focuses on solving conflict, not blaming; Keeps emotions under control.

Oral Communication - Speaks clearly and persuasively in positive or negative situations: Demonstrates group presentation skills; Participates in meetings.

Written Communication - Writes clearly and informatively; Able to read and interpret written information.

Teamwork - Balances team and individual responsibilities; Gives and welcomes feedback. Able to build morale and group commitments to goals and objectives; Supports everyone's efforts to succeed.

Ethics - Treats people with respect; Works with integrity and ethically.

Organizational Support - Follows policies and procedures; Completes administrative tasks correctly and on time; Supports organization's goals and values; Benefits organization through outside activities; Supports affirmative action and respects diversity.

Strategic Thinking - Develops strategies to achieve organizational goals; Understands organization's strengths & weaknesses.

Motivation - Sets and achieves challenging goals; Measures self against standard of excellence.

Planning/Organizing - Prioritizes and plans work activities; Uses time efficiently; Plans for additional resources; Sets goals and objectives.

Professionalism - Approaches others in a tactful manner; Reacts well under pressure: Treats others with respect and consideration regardless of their status or position: Accepts responsibility for own actions; Follows through on commitments.

Safety and Security- Observes safety and security procedures.
Adaptability - Adapts to changes in the work environment; Manages competing demands.

Attendance/Punctuality - Is consistently at work and on time.

Dependability - Follows instructions, responds to management direction: Takes responsibility for own actions; Keeps commitments.

Initiative - Volunteers readily; Undertakes self-development activities.

Innovation - Displays original thinking and creativity; Meets challenges with resourcefulness: Generates suggestions for improving work.

Qualifications To perform this job successfully, an individual must be able to perform each essential duty satisfactorily. The requirements listed below are representative of the knowledge, skill, and/or ability required. Reasonable accommodations may be made to enable individuals with disabilities to perform the essential functions.

Education and/or Experience

Bachelor's degree (B. A.) from four-year college or university; or one to two years related experience and/or training; or equivalent combination of education and experience.

Language Skills

Ability to read and interpret documents such as safety rules, operating and maintenance instructions, and procedure manuals. Ability to write routine reports and correspondence. Ability to speak effectively before groups of customers or employees of organization.

Mathematical Skills

Ability to calculate figures and amounts such as discounts, interest, commissions, proportions, percentages, area, circumference, and volume. Ability to apply concepts of basic algebra and geometry.

Reasoning Ability

Ability to apply common sense understanding to carry out instructions furnished in written. oral, or diagram form. Ability to deal with problems involving several concrete variables in standardized situations.

Computer Skills

To perform this job successfully, an individual should have knowledge of MS SQL server Database software; Adobe illustrator, Adobe Photoshop, Macromedia Flash Design software: Microsoft Visual Studio.Net, Macromedia Studio MX Development software; Microsoft Project Project Management software and Microsoft Office Word Processing software.

Certificates, Licenses, Registrations
Driving Licenses (customer on site support)

Other Skills and Abilities
Ability to speak in Chinese.

Other Qualifications
Must be able to work for OT when require.

Physical Demands The physical demands described here are representative of those that must be met by an employee to successfully perform the essential functions of this job. Reasonable accommodations may be made to enable individuals with disabilities to perform the essential functions.

Work Environment The work environment characteristics described here are representative of those an employee encounters while performing the essential functions of this job. Reasonable accommodations may be made to enable individuals with disabilities to perform the essential functions.

The noise level in the work environment is usually quiet.

Appendix B: Evaluation

The system has been tested and evaluated by two 3rd year undergraduates from the Faculty of Business & Accountancy, University Malaya.

The following are the comment that made by the mentioned 2 undergraduates about BAS.

Name : Chong Chee Jean
Faculty : Business & Accountancy.
Year : 3

Comment :

This Busines Advisory System program helps the management to sort the problems and criterias of recruiting and retrenchment. Thus, save time and cost.

A good set up. Can improve in by adding more details about individual particulars, where manager has the option to add / cancel any requirements.

10/Feb/2004
5.10pm.

Name: Juan Pui Fung

Faculty: Business and Accountancy

Comment:-

It is userfriendly. Even I'm not very familiar ~~with~~ with computer but I feel easy to use.

It also very useful and pratical. A HR manager may waste a lot of time to ~~see~~ read the resume from applicants, but if he or she has this software, the manager can finish her job earlier.

I feel confident with the security system as well.

In the process of selection, I may be confucius with too many candidates. Now, the computer will advise me in selection. I will be more easier in making a decision. Moreover, the decision is more accurate.

It will be great if there is a remark collum in recruitment page.

(JUAN PUI FUNG)

10/2/2004

Appendix C: User Manual

Introduction

Business Advisory System (BAS) is a window application, but able to communicates with the server database through web services.

The purpose of this system is to comply with the company's objective, which is to maximise the profit and minimize the cost. It helps the Human Resources Department especially management level to provide useful advices in decision making during selection, recruitment, retrenchment and salary cut down.

BAS can be divided into five sections according to it major functionalities:

- Section 1 – Login Authentication
- Section 2 – Recruitment
- Section 3 – Selection
- Section 4 – Retrenchment
- Section 5 – Salary Cut Down

Login Authentication

There is a double login criteria in BAS, where the first login (local login) will be perform by the user, whereas the second login (server login) will be perform automatically by the system when the user need to get data from server database.

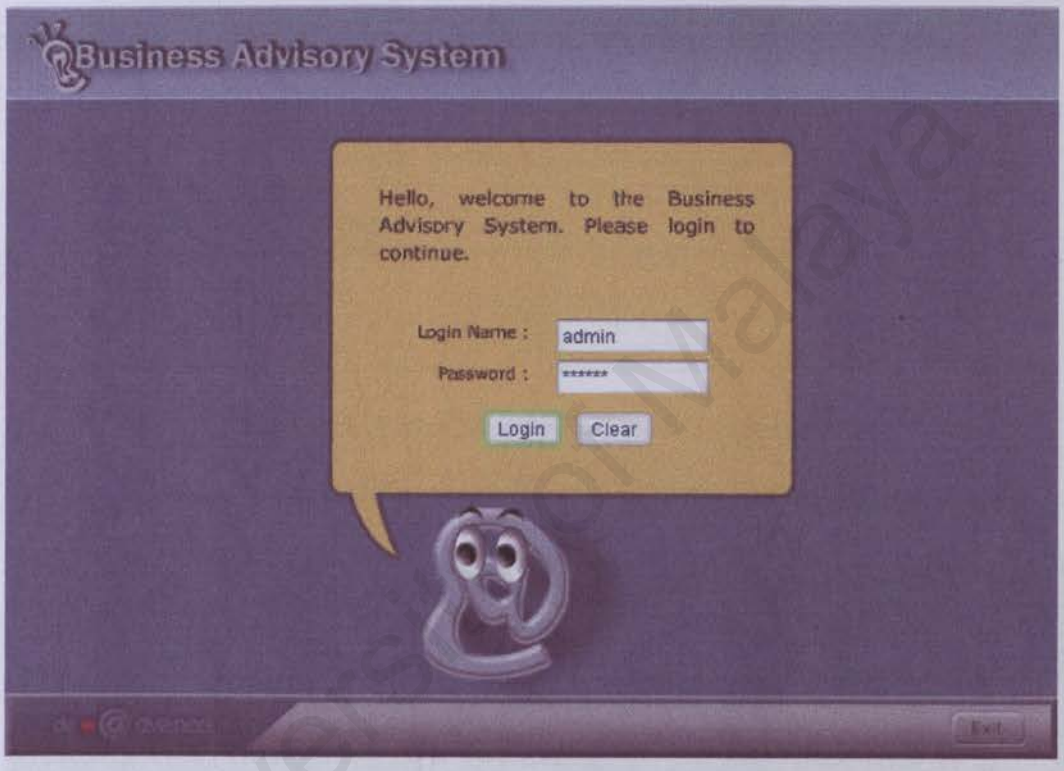


Figure 1: The login page.

While enter to the system, the user can't be avoid from login to the system by giving the login name and password. The passwords that contain in the database (both local and server) are encrypted to further increase the security of BAS.

If the local database isn't exists or login local fail, an error message will be prompt out as shown in figure 2. The system will try to login through server and update the local database.

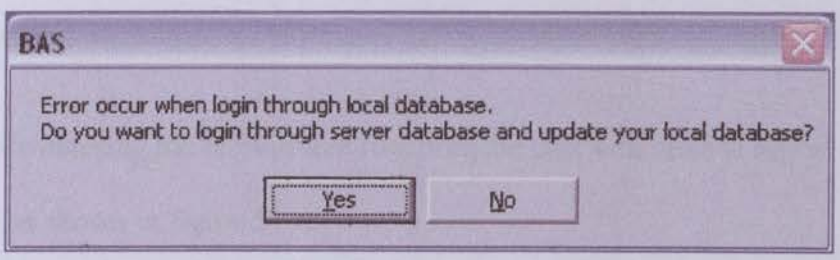


Figure 2: Error Message for local login fail.

If failure occurs for login to the server, an error message will be prompt out as shown in figure 3. These will determine that the user is an unauthorized user.

Figure 3: Error Message for server login fail.

After login, the main page will be shown to the user to select the services

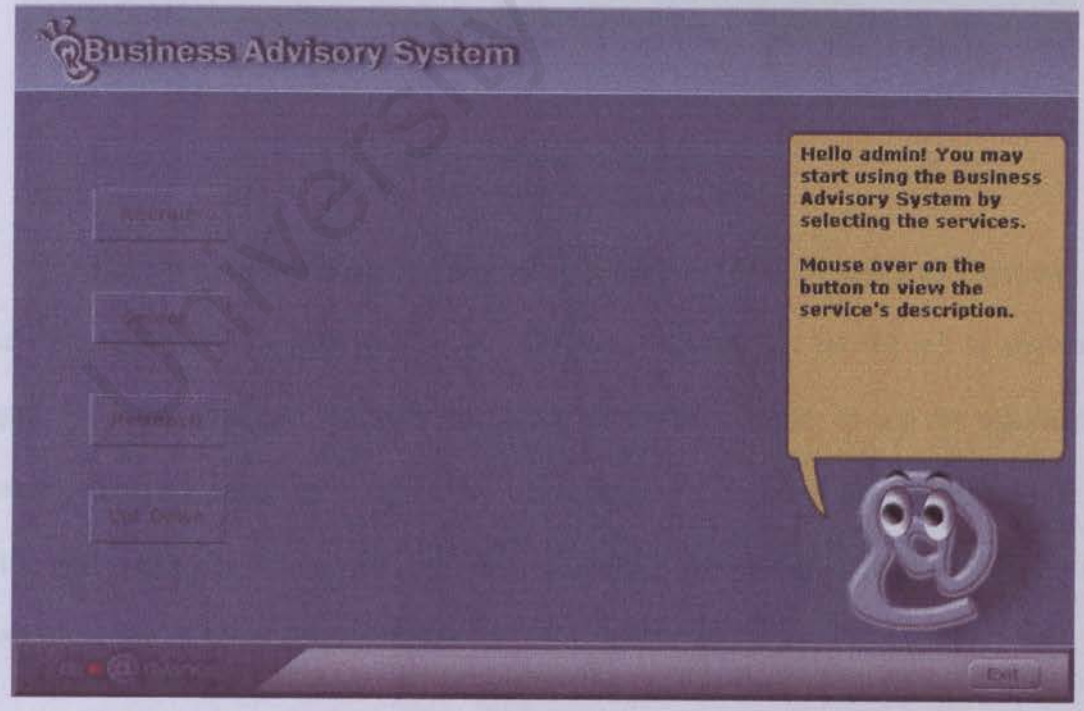


Figure 4: Main page of BAS.

Recruitment

Before entering the recruitment function, the user will need to key in the name of offering job as shown in figure 5.

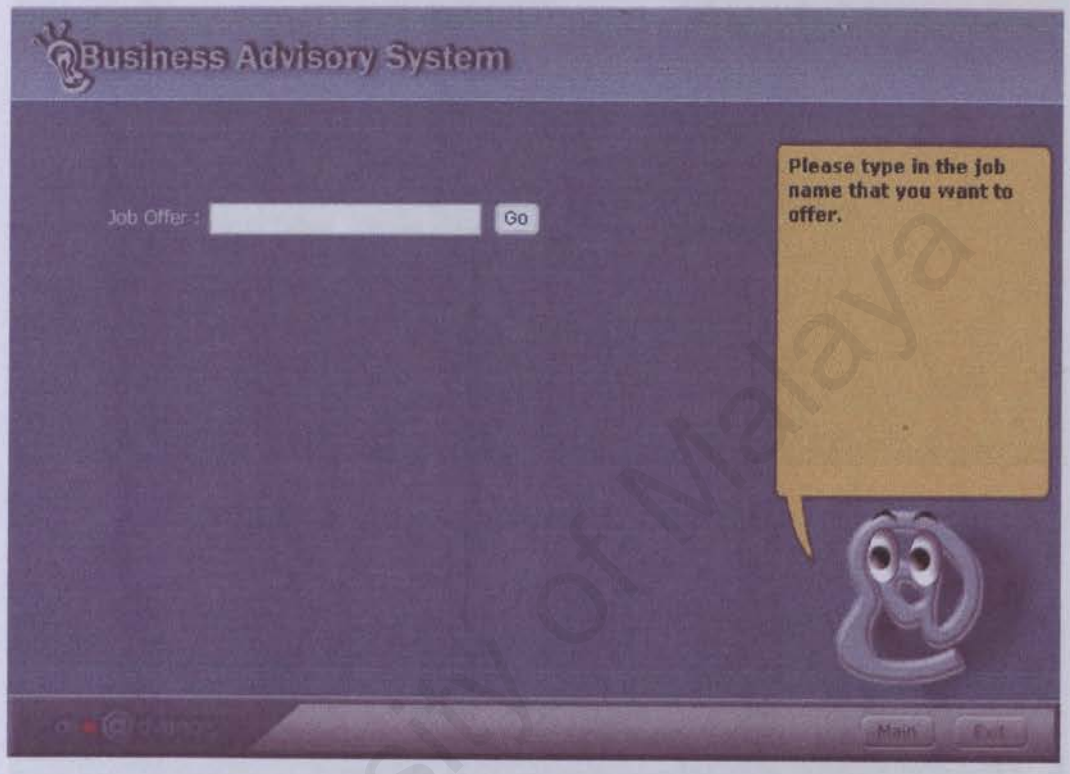


Figure 5: Pre-recruit page.

After the pre-recruit page, if the job has not been offered using this system, the user will need to add a new case or the minimum requirement for the job as shown in figure 6. Or else the user will have to choice to choose whether to use the old case or create a new one as shown in figure 7. Click "Save" if it is a new case or click "Use It" if the user wishes to the old case. Of course the user has the right to delete the case if the case is not use by other unsolved case.

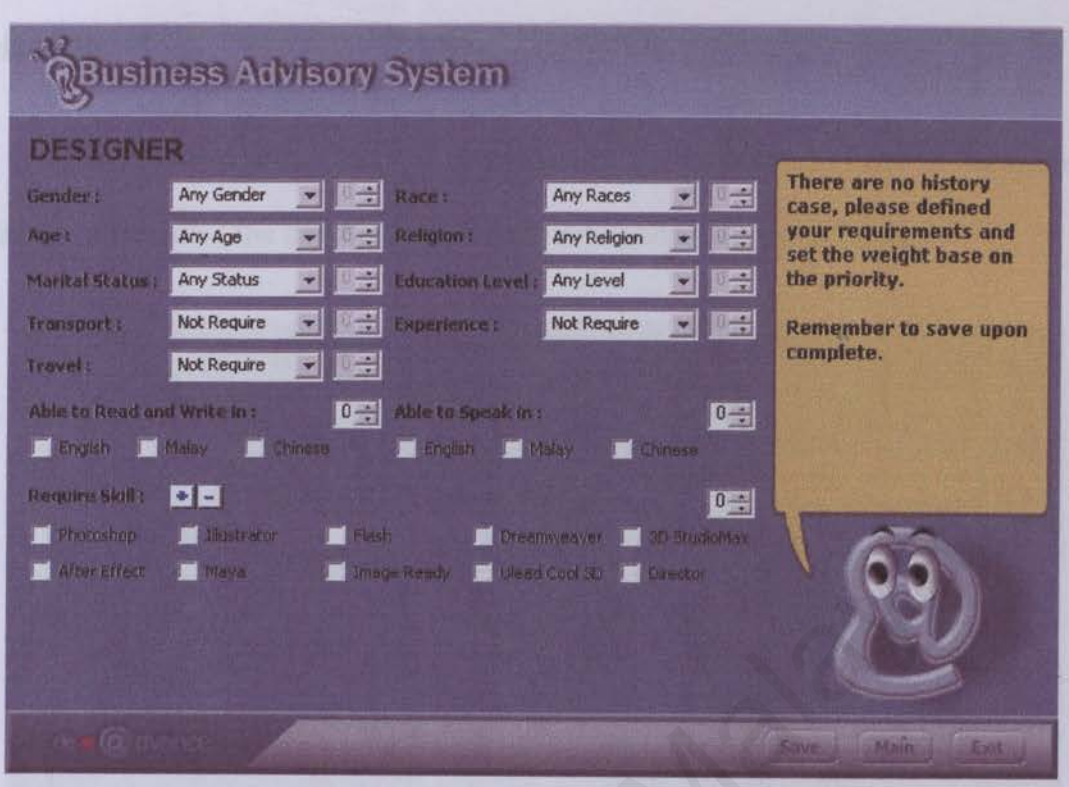


Figure 6: Creating case for the job "Designer".

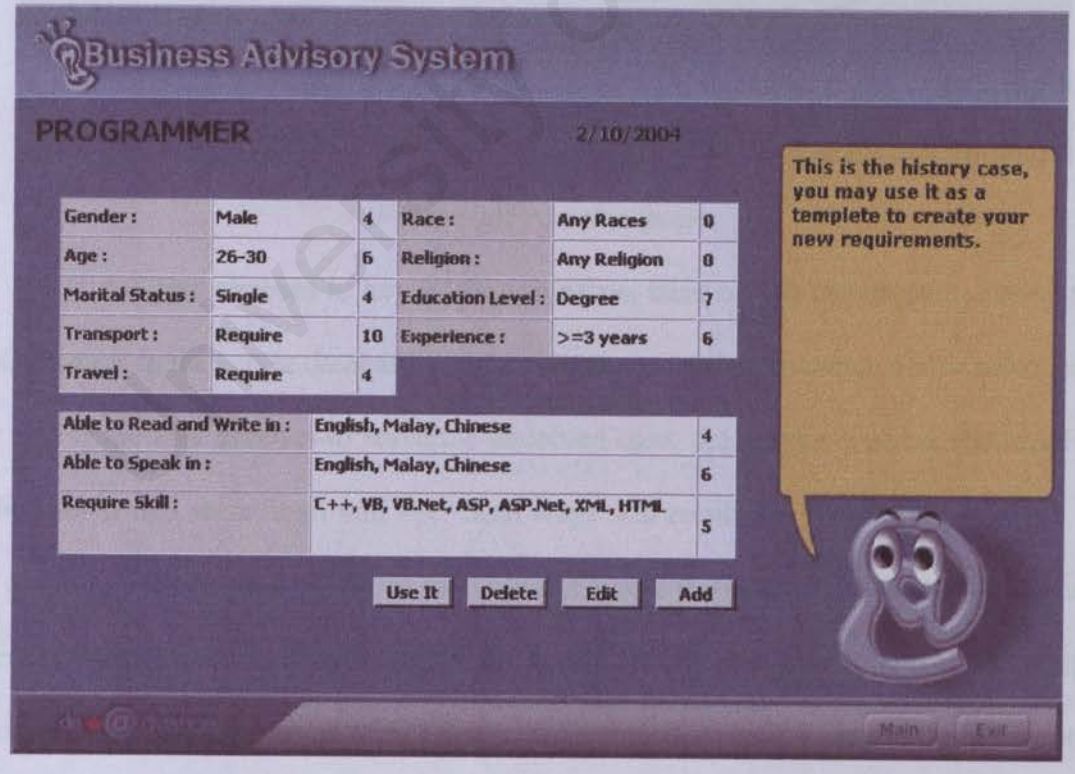


Figure 7: History case for the job "Programmer".

Selection

Before entering the selection function, the user will need to key in the name of offering job or solve the unsolved case in the data grid as shown in figure 8.

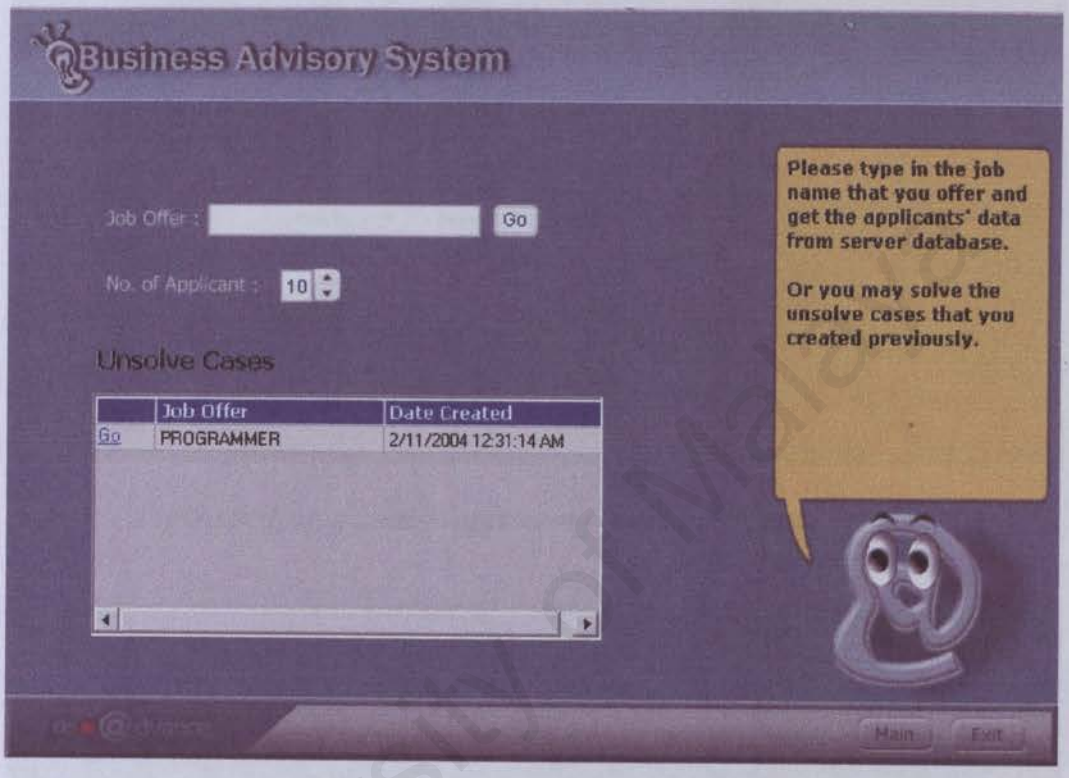


Figure 8: Pre-select page.

If the user chooses to key in the job name, base on the recruit case, the system will connect to the server database and get the necessary data to create a new select case. Or else if the user chooses to solve the unsolved case, the system will use the unsolved information that store in an xml file. Both ways will result the system to generate a list of qualify applicant in a data grid as shown in figure 9. Base on the data grid, the user can view the detail of an individual applicant as shown in figure 10 and even email to them directly through server's SMTP server. User may close the case without accepting any applicant or accept the applicant or leave it as an unsolved case.

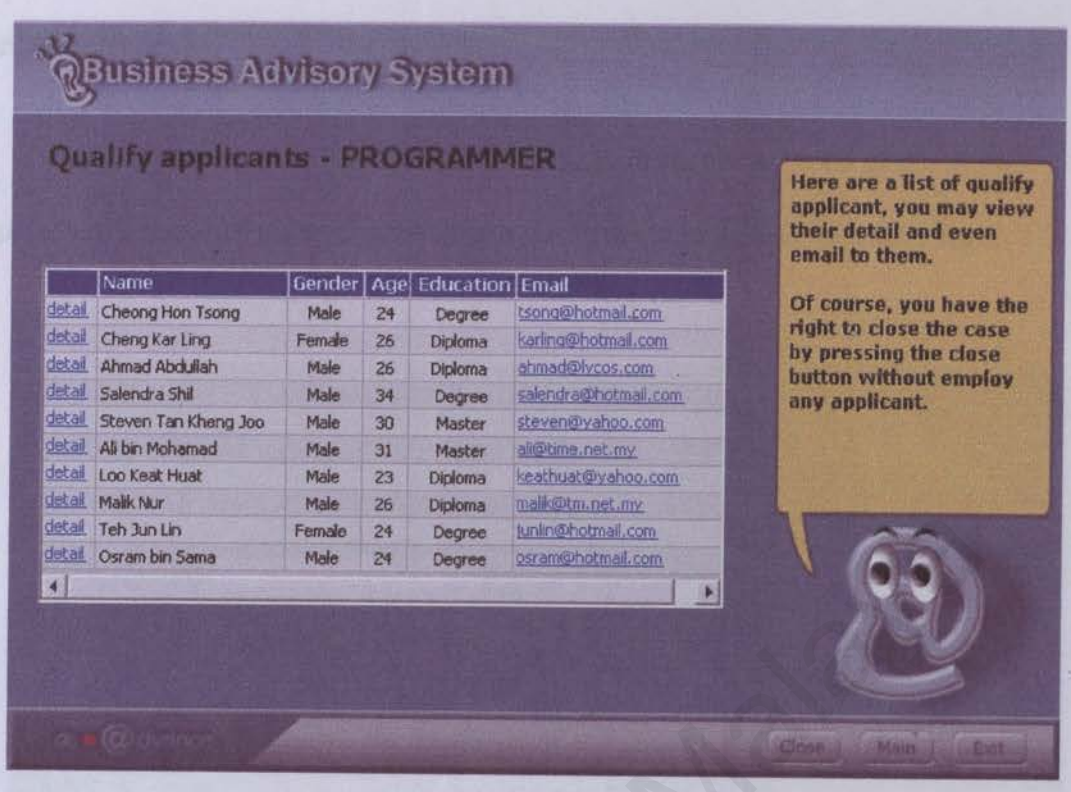


Figure 9: List of qualify applicant for the job "Programmer".

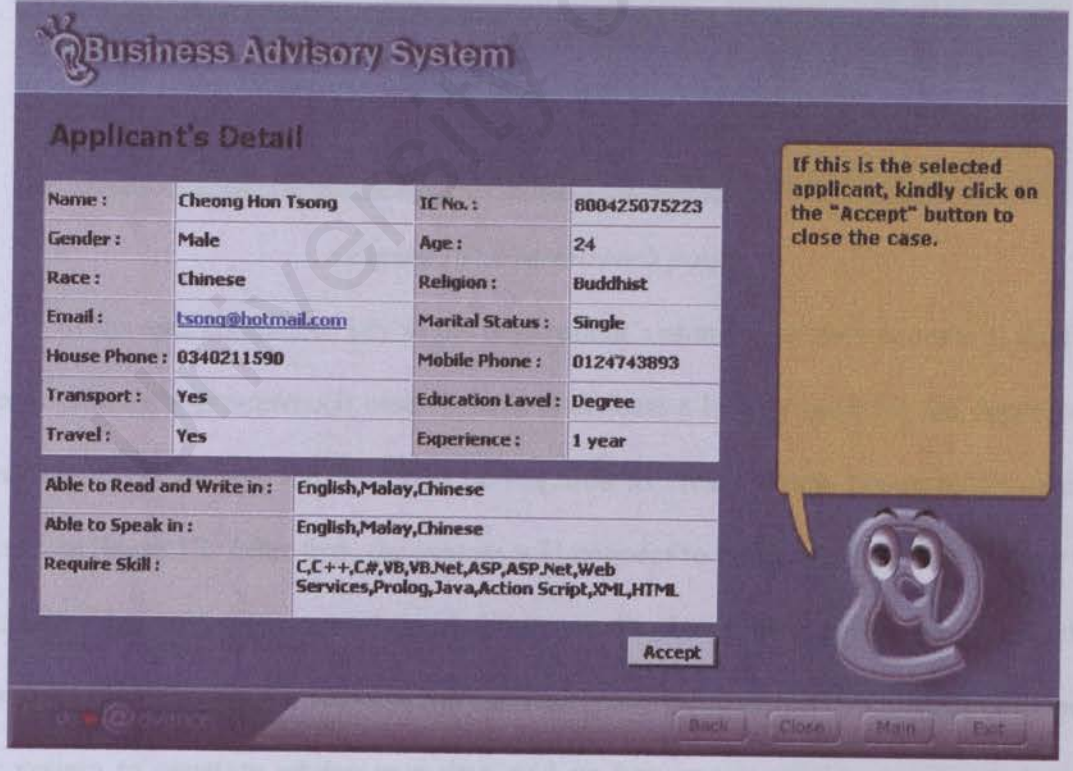


Figure 10: Applicant's Detail.

Retrenchment

Before entering the retrenchment function, the user will need to key in the department's name or solve the unsolved case in the data grid as shown in figure 11.

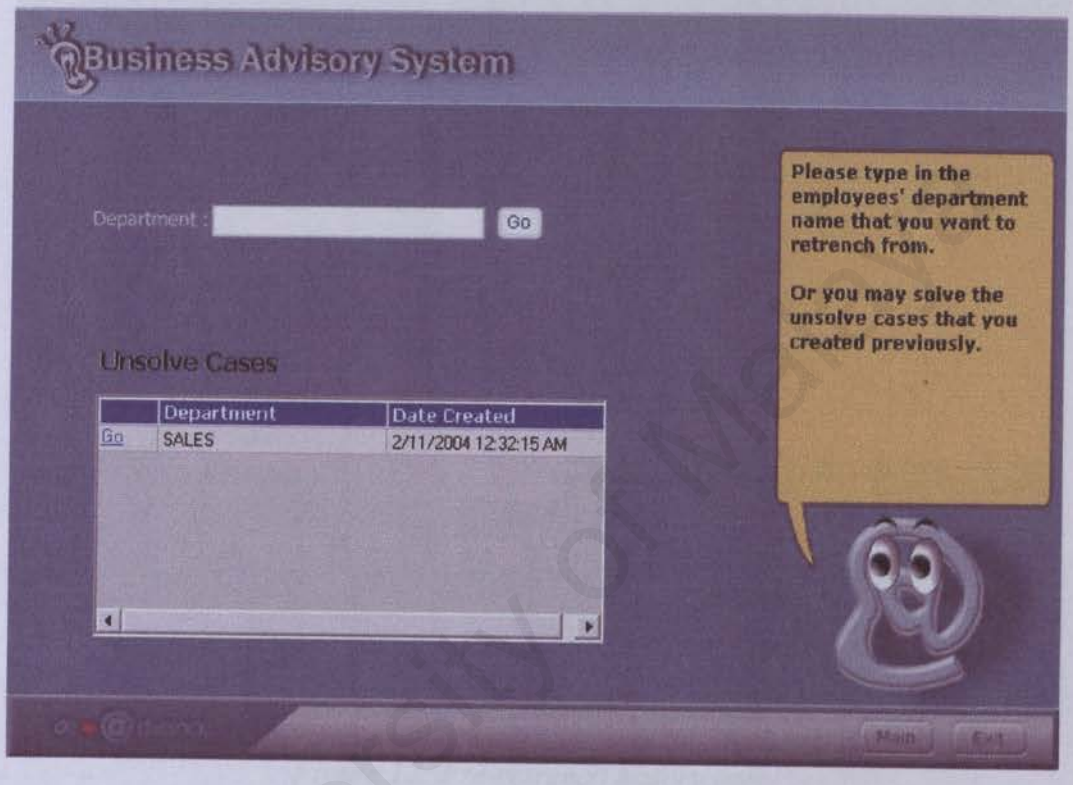


Figure 11: Pre-retrench page.

If the user chooses to key in the department's name, then the system will lead the user to choose a pre-retrench case if there is at least a history case for the department (see figure 12). If not, the user will be required to create a new pre-retrench case as shown in figure 13. After that, the system will connect to the server database and get the necessary data to create a new retrenchment case. Or else if the user chooses to solve the unsolved case, the system will use the unsolved case information. Both ways will result the system to generate advice in a data grid on how many employee will need to be

retrenched and who should be retrenched (see figure 14). Base on the data grid, the user can view the detail of the employee as shown in figure 15. User may close the case or leave it as an unsolved case.

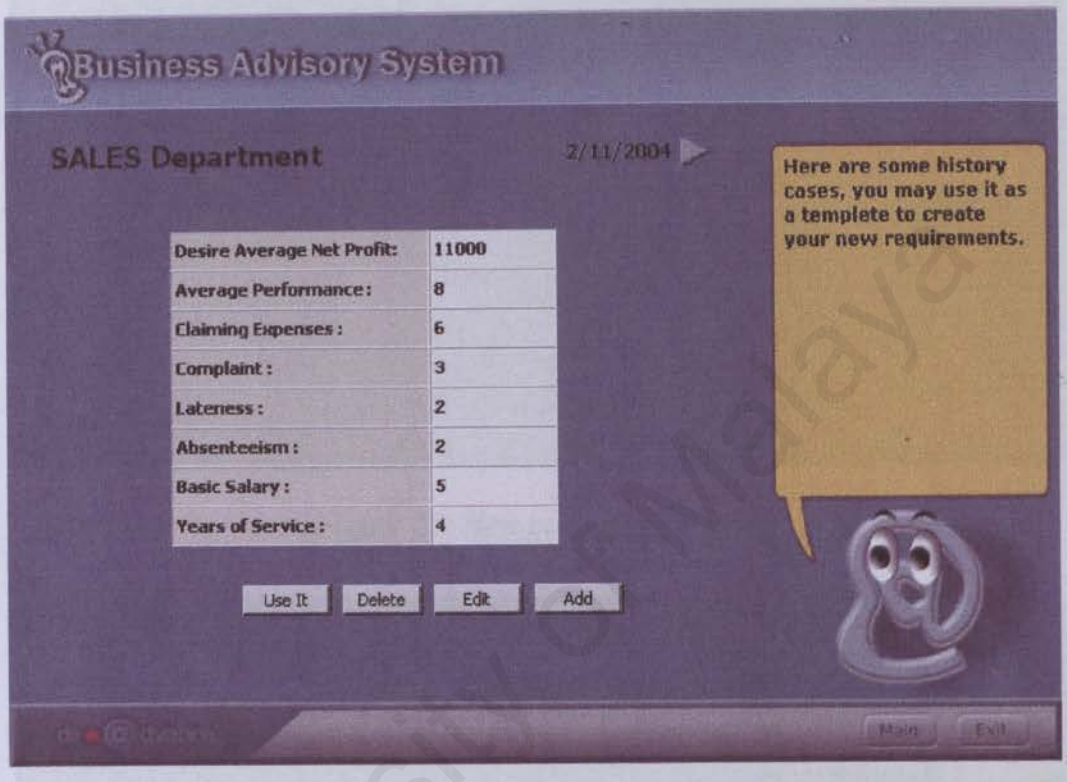


Figure 12: Pre-retrench history case.

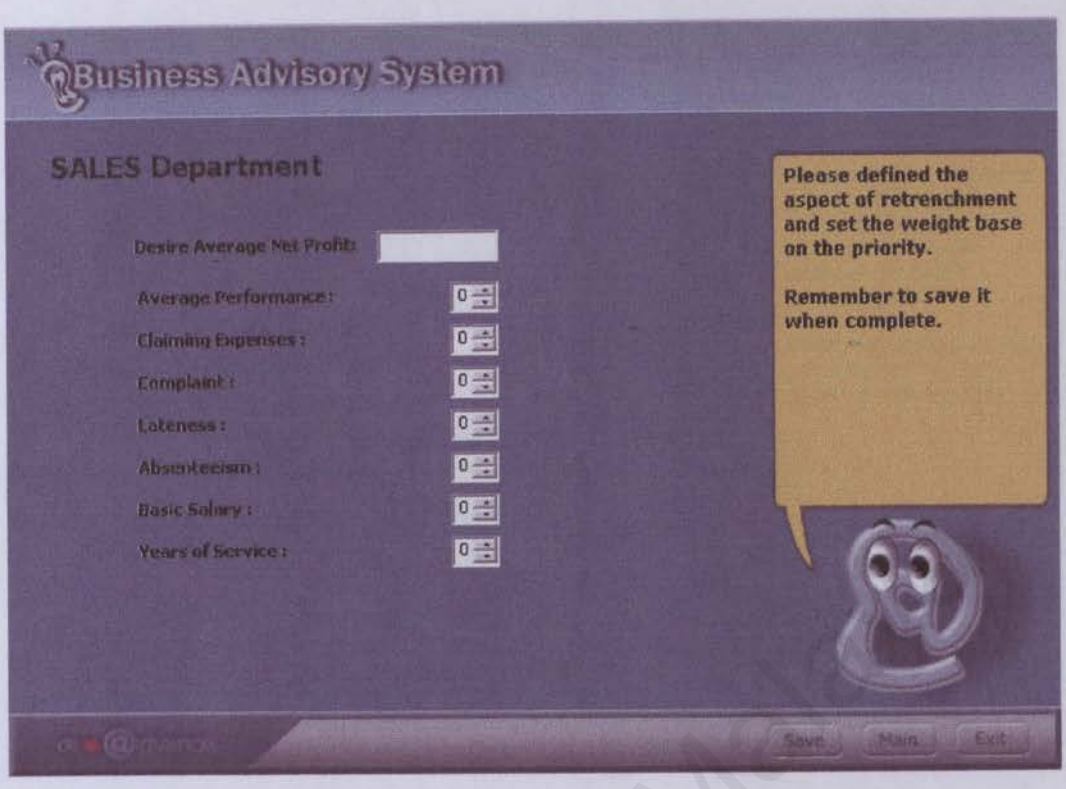


Figure 13: New pre-retrench case.

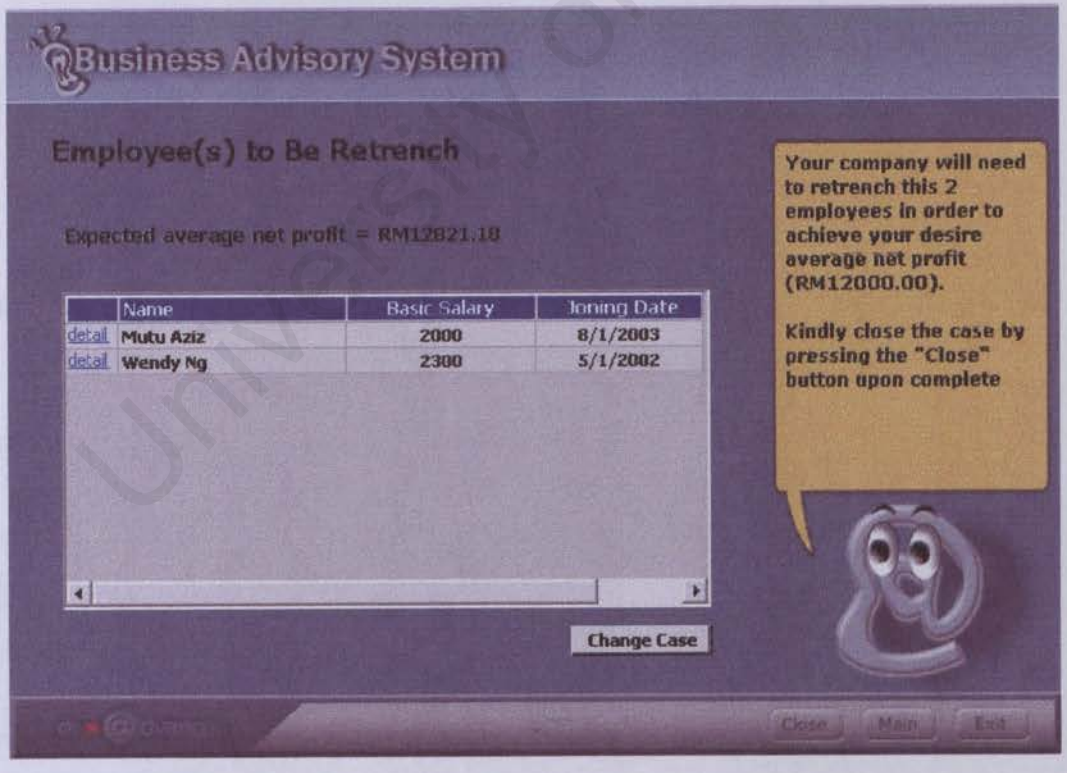


Figure 14: Advice for the retrenchment.

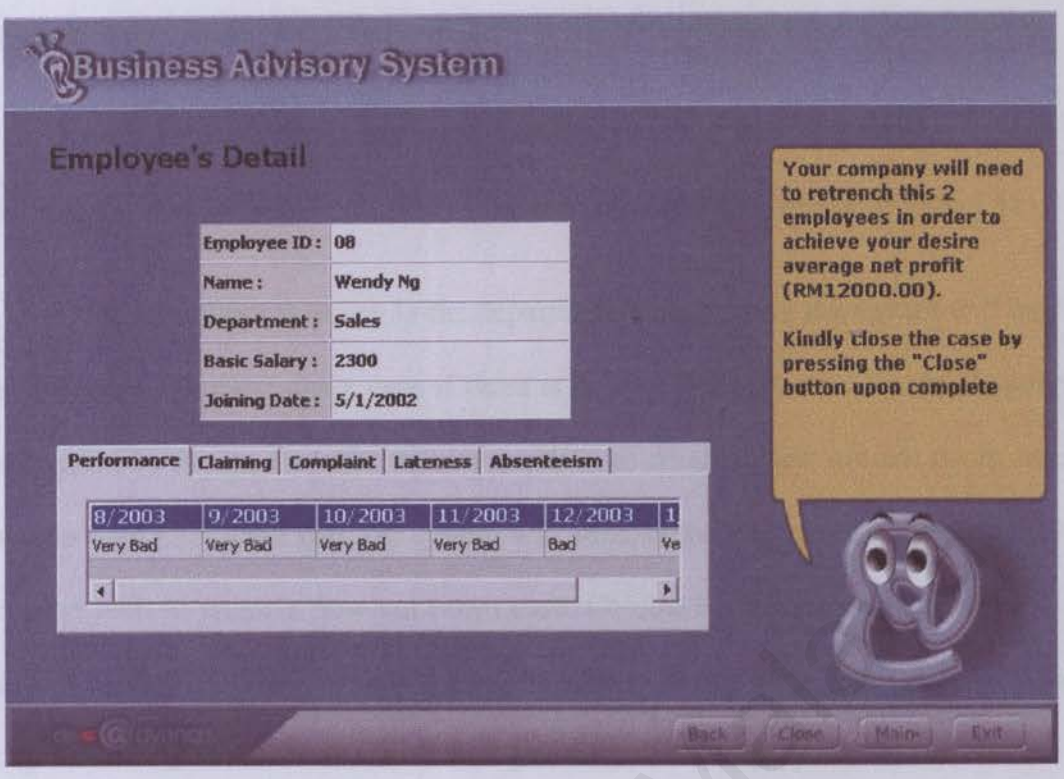


Figure 15: Employee's detail.

Salary Cut Down

Before entering the salary cut down function, the user will need to key in the department's name or solve the unsolved case in the data grid (similar to figure 11).

If the user chooses to key in the department's name, then the system will lead the user to choose a pre-cut down case if there is at least a history case for the department (see figure 16). If not, the user will be required to create a new pre-cut down case as shown in figure 17. After that, the system will connect to the server database and get the necessary data to create a new cut down case. Or else if the user chooses to solve the unsolved case, the system will use the unsolved case information. Both ways will result the system to generate advice in a data grid on which employee's salary will be cut down and how many percent of cutting will need to apply as shown in figure 18. Base on the data grid, the user can view the detail of the employee (similar to figure 15). User may close the case or leave it as an unsolved case.

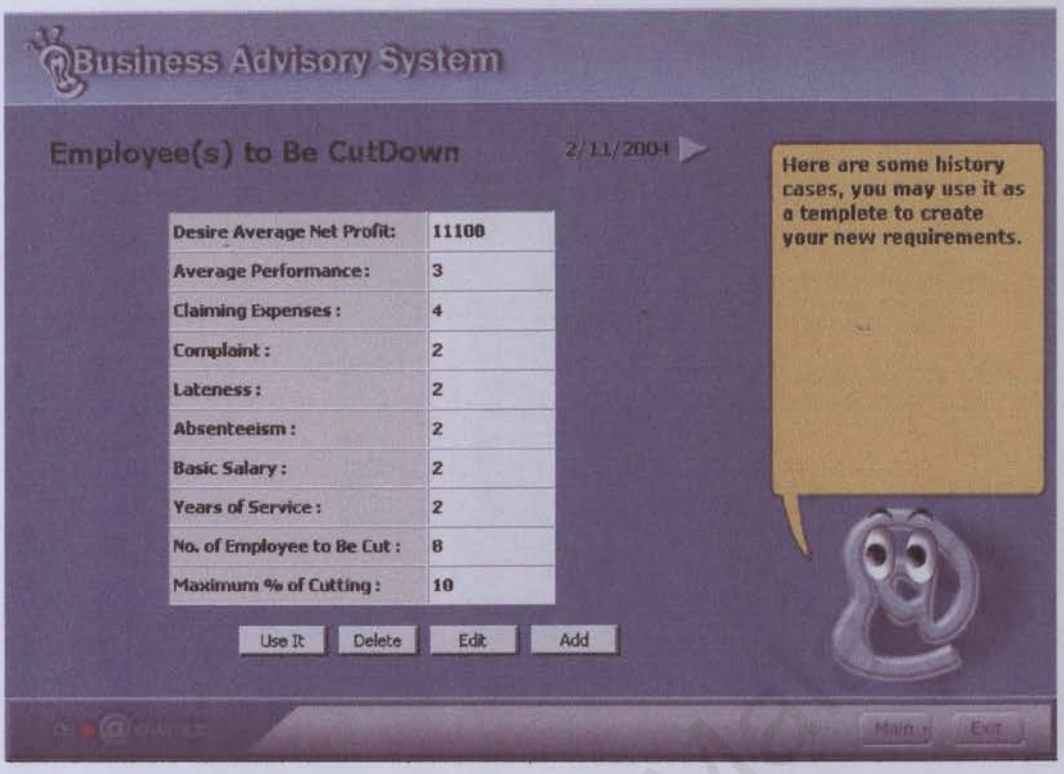


Figure 16: Pre-cut down history case.

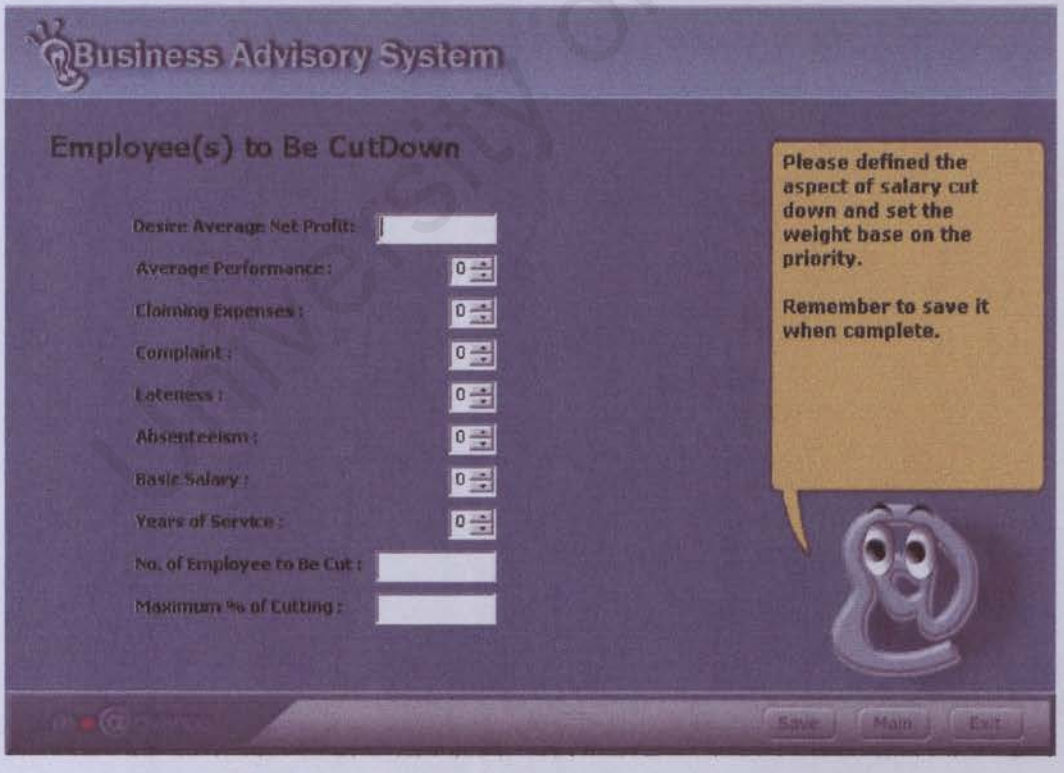


Figure 17: New pre-cut down case

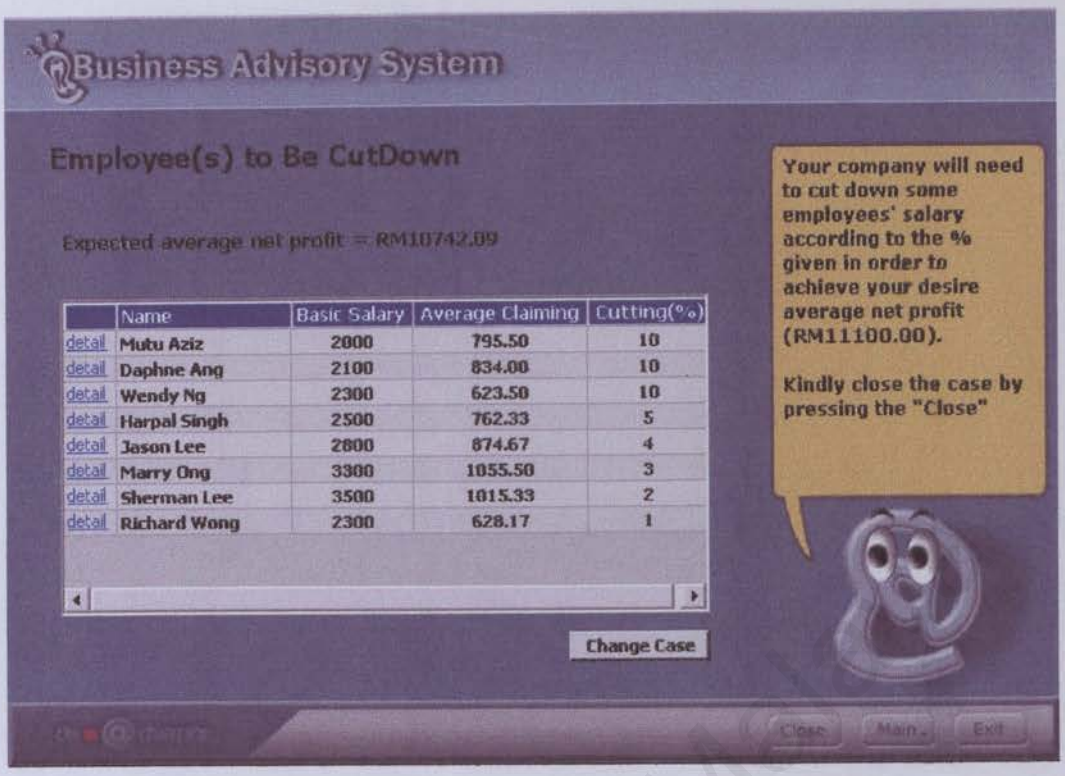


Figure 18: Advice on salary cut down.