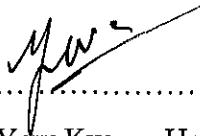# CERTIFICATION OF APPROVAL

Flash Helper System

by

Mohd Rezan Bin Abd Hamid

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

Approved by,

....

(Mr. Yew Kwang Hooi)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
JULY 2007
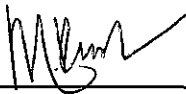
CERTIFICATION OF ORIGINALITY

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

MOHD REZAN BIN ABD HAMID

ABSTRACT

The Visit Malaysia Year (VMY 2007) campaign will coincide with the golden anniversary of Malaysia's independence of 50th independence and will showcase Malaysia as a unique destination place with many things to offers to travelers from other countries. With VMY 2007, tourism becomes important to our country and e-tourism will play a great role in the industry. An interactive system within e-tourism would be one way of attracting users and possible tourist to Malaysia. Animated avatar or virtual character made out from Flash system will greatly improve the attractiveness and products of a tourism website thus enrich user's experiences which is the main thing in attracting the users to continue browsing and using the webpage. The objective of this system is to improve the interactivity between users and the websites through the system which is represented with animated character build from Flash system. This system will help users find what they search and give a quotation of their finding and choosing. With prototyping methodology used in its development, it is hoped that the system can provide more to the tourism industry by giving good results and interactivity that users really need.

ACKNOWLEDGEMENT

First and foremost, I would like to express my gratitude to the Lord Almighty, Allah S.W.T because of His mercy and blessings enables me to withstand the pressure and challenges in completing this Final Year Project course.

I would also like to express my deepest appreciation and many thanks to my supervisor Mr Yew Kwang Hooi. Without his assistance, guidance and support, this project completion would not be possible. His advice and motivations really help and gave strength for me to complete this project.

Not forgotten to both my parents, family and peer members and friends for their morale support, encouragement and teachings which lead to the success of the project. My gratitude to all lectures in ICT and BIS department who in someway help me during project implementation and seminars.

Last but not least, to all personnel not mentioned that involves directly or indirectly with this project, thank you for your cooperation and commitment throughout the project development. What I have gained from this project all comes from them and their sincere help and support.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 **Background**

With the significances of our 50ᵗʰ year of Independence, Malaysia is promoting its tourism campaign which is Visit Malaysia 2007. Tourism becomes important to Malaysia as it can generate foreign exchanges, business opportunities and local employments. With such events, e-tourism will play a great role in the industry to attract tourist and companies to Malaysia. The nation surely got many interesting place, traditions, and heritage but tourist may know little of it. They also don't how to get around in the country and there were not many websites or portals that were interactive enough and really helpful in providing information to these possible tourists of Malaysia.

As a result, many websites promotes tourism in Malaysia but lack the main thing on how to attract this people. An interactive animated system will be something new within this field and it would surely enrich the user's experiences and improve the productivity of the websites. This project is intended to create new user's experience of navigating and browsing the websites through an animated character which can be an intelligent system that can attract users to keep navigating the webpage.

1.2 **Problem Statement**

Visit Malaysia 2007

This event and campaign requires great promotions and good designed websites to attract users or the tourist to Malaysia. And what they lacks now is a new and good websites that have elements such as good web design or system, advertisement, promotions and other alternative ways to gain the attention of these users the possible tourist going to spend their vacation in Malaysia.

Lack of Interactive System

For the moment, there were few Malaysian tourism websites that can really gain the attention or attract users to its webpage. Current websites lack systems or programs that can give interactions to users. Some websites have simple flash advertisement or banners to promote events and festivals but no such system that act independently from the websites exists.

Simple Flash

With many websites of Malaysian tourism, they were only few uses flashes program and simple flash to advertise and promote the websites. They were no flash program that can give interaction to users while they were navigating the webpage. What they lack is a flash system with good functions that can get input from users and have Action Scripts that act as a tool for this task.

1.3 <u>Objectives and Scope of Study</u>

Set up a flash avatar system in a Malaysian tourism website.

Build a flash system what can connect with tourism websites. The system will play a great role in attracting users and giving new users experiences that other websites can't give, such as interactivity and great user's experiences.

Advice, plan vacation and out with a quotation of users choosing.

The flash system will gradually help users plan their holiday in Malaysia by providing search option from the websites and come out with a list of their choosing in a quotation. Imagine a flash system's that can help users to carefully plan their holiday and give them information on what they want.

Flash program that was different than others.

This system can provide users the information as they are searching for contents and interesting places in the websites. It also has different animation when doing task as searching or giving the users quotations. This system can receive users input, search it from the websites and provide links to possible result of the search. It also will play different animation occasionally when idle or giving quotations and popping out a new menu of promotions from the websites based on its algorithm.

CHAPTER 2

LITERATURE REVIEW

2.1 <u>Understanding Flash applications</u>

Since its introduction in 1996, Flash technology has become a popular method for adding animation and interactivity to web pages; several software products, systems, and devices are able to create or display Flash. Flash is commonly used to create animation, advertisements, various web-page components, to integrate video into web pages, and more recently, to develop rich Internet applications. Macromedia Flash is a suitable program that can be used in developing the system within a website. A Flash application is a Web application that uses Flash to collect user information, send that information to a server to process, and display the results. [1]

Having defined the architecture of the purchaser and seller agents, as well as their cooperation and communication protocols, this section focuses on the multiple criteria decision making process per se. This process takes place in the decision making module of the purchaser agent. The tool implemented for the automation of this process is an extension of the work presented in [2].

From the literature review, the agent can make a decision making based on what it learns from the user's preferences and previous actions. This concept can be implemented in the flash agent where the agent learns and make decisions to give users a menu of questions and alternatives such as other animations based on the users input and preferences in previous options selected.

2.2 **Action Scripts**

Actionscript is a script language used in Macromedia Flash@ programming [1], [2]. Actionscript can be used in order to control objects from Flash movies, to create navigation tools with interactive elements, and to extend the abilities offered by Flash in creating high quality interactive movies and web applications. ActionScript provides new features for the design of interactive Internet sites, integrating multimedia and real time facilities (games, completion forms, reviews, chat, etc.) [3]

2.2.1 **Connecting Flash to the Internet**

XML can be used to convey information from Flash to the internet. Its simple syntax resembles HTML in that it employs tags, attributes and values. Without XML, one could send chunks of data to a server, but the server probably wouldn't know what to do with the chunks. [4]

```
myString = "<MyFriends><Name Gender=\"female"\>Kelly</Name></MyFriends>";
myXML = new XML (mystring);
```

The above ActionScript creates the XML document as a string and then converts it ato an actual XML object called myXML. This object can then be sent to the server using the send-related methods.

2.3 Intelligent Agents

In computer science, an **intelligent agent (IA)** is a software agent that assists users and will act on their behalf, in performing non-repetitive computer-related tasks. An agent in the sense of the word is like an insurance agent or travel agent. While the working of software agents used for operator assistance or data mining (sometimes referred to as bots) is often based on fixed pre-programmed rules, "intelligent" in this context is often taken to imply the ability to adapt and learn. [5]
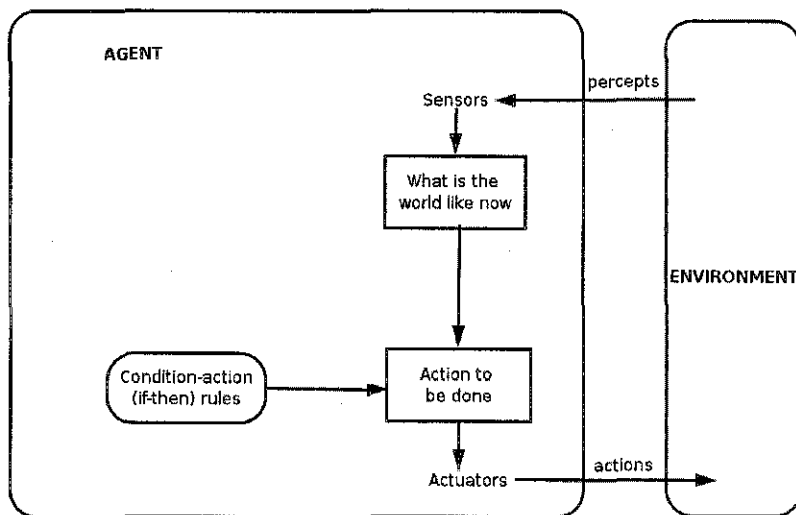
Simple Reflex Agent
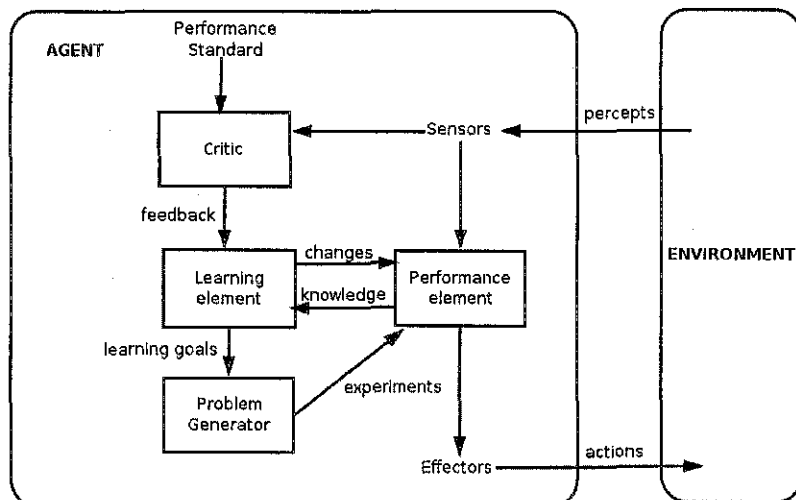
Learning Agent

2.3.1 **Software Agents**

In the end, the system may and will be using a software agent or intelligent agents in it. This agent will act as another system in the main planner system where this is the main goals and functionality of the system. A single agent will be implemented first and later multi agents when the system itself is bigger and getting complicated. This agent will act as a planner in the system and it will adapt and learn from the user's feedback.

In computer science, a **software agent** is a piece of software that acts for a user or other program in a relationship of agency [1]. Such "action on behalf of" implies the authority to decide when (and if) action is appropriate. The idea is that agents are not strictly invoked for a task, but activate themselves.

Capabilities of intelligent agents include:

* ability to **adapt**

    Adaptation implies sensing the environment and reconfiguring in response. This can be achieved through the choice of alternative problem-solving-rules or algorithms, or through the discovery of problem solving strategies. Adaptation may also include other aspects of an agent's internal construction, such as recruiting processor or storage resources.

* ability to **learn**

    Learning may proceed through trial-and-error, and then it implies a capability of introspection and analysis of behavior and success. Alternatively, learning may proceed by example and generalization, and then it implies a capacity to abstract and generalize.

Agent: An agent records data from user behaviors on clicking and reading on the web sites. Then it analyzes users' interests, and gives rewards and/or penalties. A **software agent** acts on behalf of a user or other program.

The user interface agent provides a Web interface for the user to interact with the agent and to help him or her perform data analysis and data mining activities. The user can provide a general description of the problem at hand in terms of high-level goals and objectives, or provide specific details about the data analysis or mining task to be performed. The user interface agent is responsible for receiving user specifications and delivering results. It also keeps track of user preferences. [2]

The data mining agent may contain specific algorithms implemented in-house, or it can be the conduit to special purpose data mining products that may have been installed within the environment. The interface module provides the interagent interface as well as mechanisms to tie into other applications that perform data mining operations. In this situation, the data mining agent provides an intelligent front end to the commercial data mining products. [2]

As an alternative to the procedural program, a control system can be implemented using agents. In an agent-based control system, each control loop or control rung is implemented as a separate agent. Because the system now contains many small programs, the code for each control can be assigned to the node that is closest to the trigger event or feedback device. If the control sequence is distributed across several nodes, it may make sense to allow the agent to move around the network, thus the solution becomes a mobile agent based control system.

2.3.2 WebMate
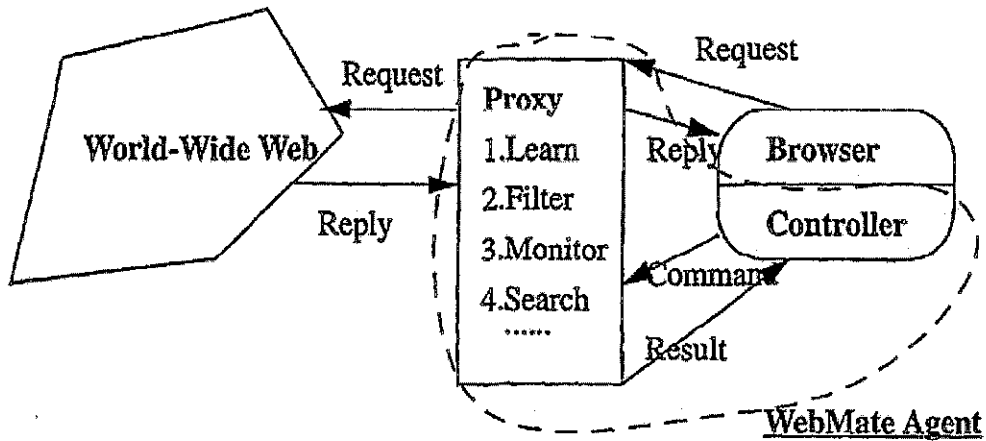
Figure 2.3.2.1 WebMate system architecture

WebMate utilizes TF-IDF method [7] with multiple vectors representation. The basic idea of the algorithm is to represent each document as a vector in a vector space so that documents with similar content have similar vector. Each dimension of the vector space represents a word and its weight. The values of the vector elements for a document are calculated as a combination of the statistics term frequency TF(ur, d) (the number of times word zu occurs in document d> and document frequency DF(zu) (the number of documents the word UJ occurs in at least once). From the document frequency the inverse document frequency ZDF(w) can be calculated. [6]

$$IDF(w) = \log \frac{|D|}{DF(w)}$$

$|D|$ is the total number of documents. The value $d^{(i)}$ of an element in the vector is then calculated as the product

$$d^{(i)} = TF(w_i, d) \times IDF(w_i)$$

Figure 2.3.2.2

We assume that a user has at most N domains of interest. Assume the initial profile set is V, IV1 = 0; the predefined number of TF-IDF vectors in the profile set is N, the preset number of elements of a vector is M. For each positive example (i.e. an HTML, documents that the user has marked "I like It"), do:

$$Sim(V_j, V_k) = \frac{V_j \bullet V_k}{|V_j| \times |V_k|} \quad j, k \in \{1, 2, \ldots, n, i\}$$

5. Combine the two vectors V_l and V_m with the greatest similarity.

$$V_l = V_l + V_m \quad (l, m) = \arg\max_{(x,y)}(Sim(V_x, V_y))$$

6. Sort the weights in the new vector V_k in decreasing order and keep the highest M elements.

Figure 2.3.2.3

We use the Trigger Pairs Model [13, 14]. If a word S is significantly correlated with another word T, then (S, T) is considered a "trigger pair", with S being the trigger and T the triggered word. When S occurs in the document, it triggers T, causing its probability estimate to change. That is, when we see the word S appearing at some point in a text, we expect the word T to appear somewhere after S with some confidence[8]. The mutual information (MI) that considers the words order is a measure of the correlation and used to extract trigger pairs from large corpus. The mutual information is given by the following formula:

$$\mathcal{MI}(s,t) = \mathcal{P}(s,t)\log\frac{\mathcal{P}(s,t)}{\mathcal{P}(s)\mathcal{P}(t)}$$

Figure 2.3.2.4

Single keywords are usually ambiguous, or too general. Moreover, they can occur in vast quantities of documents, thus making the search return hundreds of hits, most of which arc irrelevant to the intended user query. Giving additional keywords can refine search providing considerable improvement in the retrieval results. Good refinement words must have meanings that help disambiguate or make more specific the original search word. [6]

2.3.3 <u>Intelligent Agent</u>
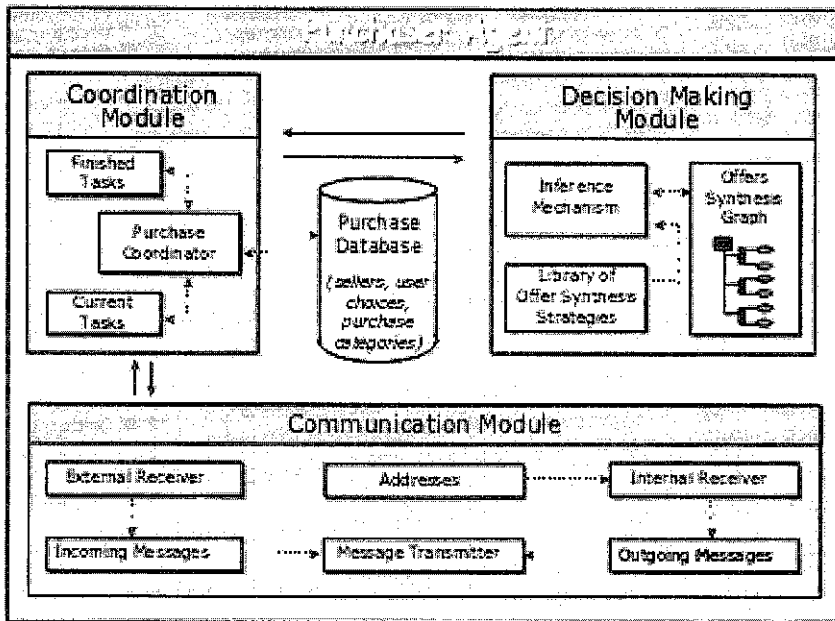
Figure 1: Architecture of a purchaser agent.

The communication module of a purchaser agent is responsible for the agent's interaction with its environment, that is the seller agents and the human user it assists. It sends and receives messages, while internally interacts with the coordination module. Its functionality is as follows: an internal receiver transforms each internally queued message (produced by the coordination module) to an inter-agent message and, in the sequel, stores it to the outgoing messages queue. In the case of selective communication, it also adds the receiving agents' addresses. An external receiver handles the opposite case, by transforming each external message received to an internal one and adding it to the incoming messages queue. Finally, a message transmitter monitors the incoming and outgoing queues, sending the queued messages to its coordination module, to another agent or to its human user, accordingly. [7]

The architecture of a seller agent is similar to that of a purchaser (Figure 2). The communication module has exactly the same functionality with the homonymous module of the purchaser agent. The coordination module is responsible for the cooperation between (i) the seller and the purchaser agents, and (ii) the seller and its merchant. A selling coordinator manages the exchange of the related messages (see next section). The selling database keeps records of the products specification, and potential or regular customers (based on the history of previous transactions) to be informed about the release of a new or promoted product. Merchants may update the related databases of their software agents at any time. [7]
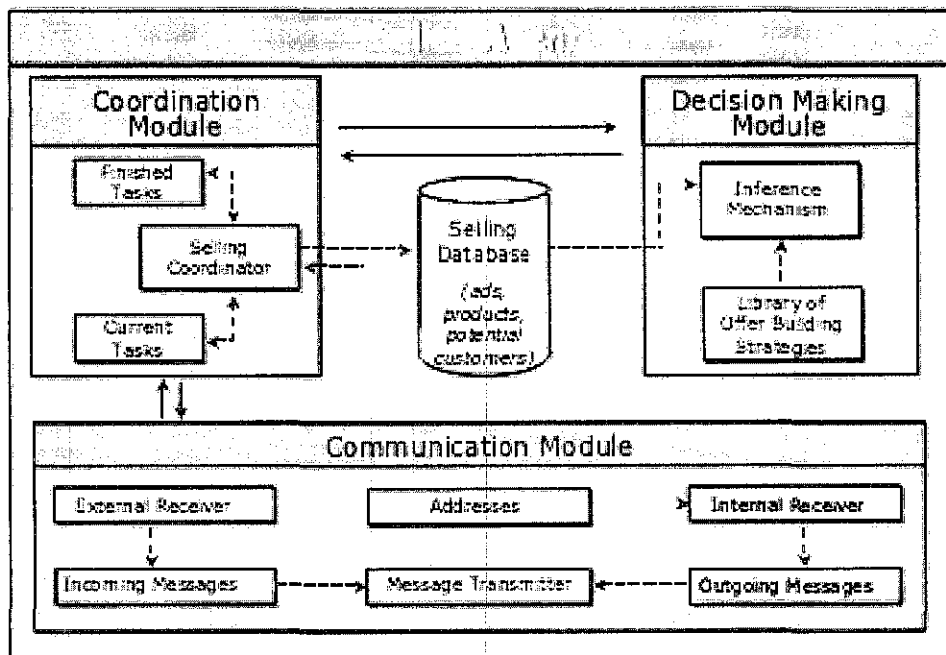
Figure 2: Architecture of a seller agent.

Having defined the architecture of the purchaser and seller agents, as well as their cooperation and communication protocols, this section focuses on the multiple criteria decision making process per se. This process takes place in the decision making module of the purchaser agent. The tool implemented for the automation of this process is an extension of the work presented in.

Concerning the building of an offer by a seller agent, its inference mechanism gets as input an *offerReqMsg* message, consults the *selling database* and the library of strategies to retrieve the appropriate data and algorithms, respectively, and produces an *offerPropMsg* message as output. The offer building strategy used in our scenario is sketched in Figure 2.3.3.5.

begin
1. get `offerReqMsg`
2. search the selling db for `objectType = purchaseObj`
3. *if* not exists then *exit*,
    else for each `objectType.objToBeSold` found *do*
        if all `offerReqMsg.constraints` are satisfied
        3.1. create a new `offerPropMsg`
        3.2. `offerPropMsg.sender ← agentID;`
            `offerPropMsg.receiver ← offerReqMsg.sender;`
            `offerPropMsg.msgID ← offerPropMsg.msgID + 1;`
            `offerPropMsg.purchaseID ← offerReqMsg.purchaseID;`
            `offerPropMsg.purchaseObj ← objectType.objToBeSold`
        3.3. search in `objectType.objToBeSold` for criteria and
            associated features appear in `offerReqMsg.description;`
            for each match (`criterion` & associated `feature`) found *do*
                add in `offerPropMsg.description` a new item `(cr, f, fv, imp)` where, `cr` and `f` correspond to the match found, while `fv` and `imp` the values of `featureValue` and `impact`, respectively, of `objToBeSold`
        end for

Figure 2.3.3.5 List of algorithm in offerReqMsg

14
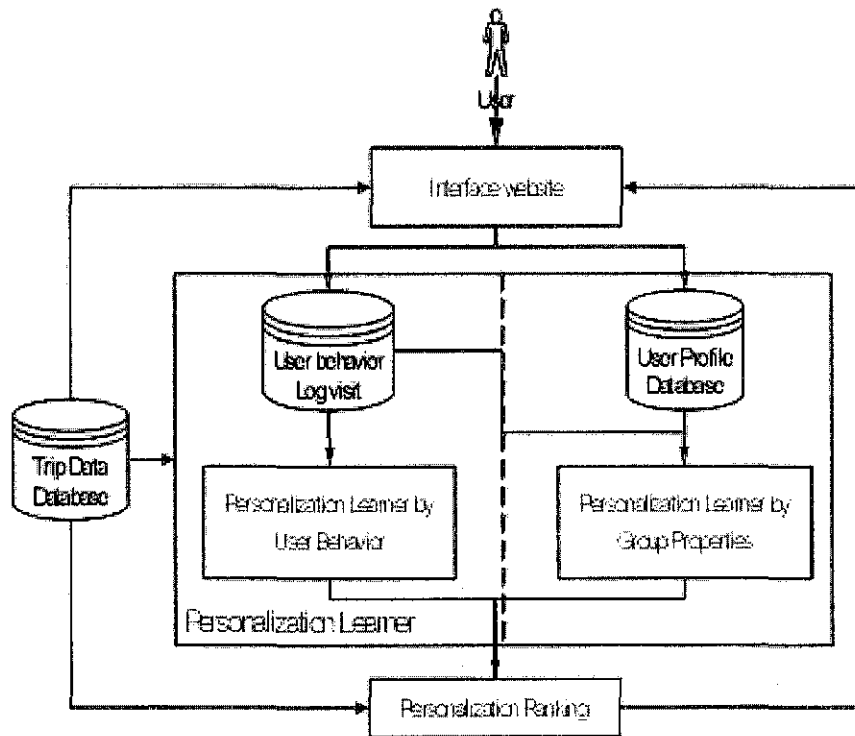
Figure 2.3.3.6 Personalization Travel Support System Structure

To perceive individual user's interests, one has to study user's behaviors by means of the information from the Interface Web Site that records two categories of data. [8]

1. Web user profile includes user name, age, and sex.

2. Traveling Information includes identification number, duration, categories, trip lowest price, trip highest price and destination country.

Personalization Learner by Group Properties: System learns from all users in one group to find the group interests of travel information by using given data on user ages and genders.

Personalization Learner by User Behavior: Recorded data is analyzed with user behaviors and the travel information in order to find the unique interest of each web user. Reinforcement learning algorithm, called Q Learning is applied at this stage.

Q Learning is used to maximize a reward to the item on the list which is clicked and award a penalty to the item that is not clicked, as shown below.

$$Q\ (s_t, a_t) \leftarrow \alpha \left[r + \gamma \max_{a_{t+1}} Q\ (s_{t+1}, a_{t+1}) \right] \quad (1)$$

Whereas max Q is defined as:

1	if user clicks the provided trip information
-1/n	if user doesn't click the trip information on the web site, where n is total number of trips per page
1/p	trips information on the database which are not recommended by the system, where p is the total number of trips in the system

given α is the learning rate valued at 0.2, and it is the discount rate valued at 0.8

Figure 2.3.3.7 Learning Algorithm

The ranking score is evaluated from the equation:

$$Qr = WtQt + WxpQxp + WmpQmp + WcQc + WdQd$$

The first approach is learning by user behavior. The Qt, Qxp, Qmp, Qc and Qd are calculated by using input data from user transactions on surfing PTS web sites and Q learning equation. Wt, Wxp, Wmp, Wc, and Wd are weights of each feature obtained from learning. After that the total score (Qr) is the summation of Qt, Qxp, Qmp, Qc and Qd multiply their corresponded weights. Next Qr score from each trip is ranked in descending order. The five maximum Qr scores are selected and recommended for trips to the users on PTS web sites.

For the second approach is learning by group property or clustering users by ages and sex. The ranking of trip provided to users is depended on user profile and user behaviors or web surfing transactions. In this approach users are clustered into group by using age and gender. Then, the value of interesting trip in each group is calculated by using user behavior or transaction on PTS web site. The process of trip ranking in this approach is the same as the above paragraph. The recommended trips are shown in Figure 3. Area number 1 which is in the middle of web page is the main box. Area number 2 which is in the right hand sight is the recommended box. [8]

2.4 E-tourism

Tourism can generates income and foreign exchange for our country Malaysia. With this year Malaysia's 50th Independence Day, Malaysia will be promoting its country throughout the year with Visit Malaysia 2007. It promotes the country and attract tourist to visit Malaysia. One way of doing this is promoting Malaysia through the internet.

To perceive individual user's interests, one has to study user's behaviors by means of the information from the Interface Web Site that records two categories of data.

1. Web user profile includes user name, age, and sex.

2. Traveling Information includes identification number, duration, categories, trip lowest price, trip highest price and destination country. [8]

WWF Malaysia estimates that Malaysia gains RM655 million per year from ecotourism. This figure is rather small compared to the estimated RM24 billion tourism brings into the economy (in 2001). As indicated in Table 1, the number of international arrivals to Malaysia showed a significant growth before reaching its first climax in 1990. Malaysia, in 1990 also recorded 54% increase in tourist arrivals from the previous year due to intense Visit Malaysia Year promotional blitz throughout the world. [9]

Year	Tourist Arrivals	% Growth
1997	6210821	-12.9948
1998	5200000	-16.2752
1999	7931149	52.5221
2000	10221582	28.87896
2001	12775073	24.98137

Source: Tourism Malaysia, 2002

Figure2.4.1: Table of Tourist Arrival in Malaysia

18

CHAPTER 3

PROJECT METHODOLOGY

Prototyping Methodology:

In this chapter, the author going to discuss on what are the procedure and processes involved during the development of Flash Helper System. Below, the author will be discussing on how the project progress in various phases that each of it have its own roles in shaping the project itself. Before that, the author has chosen to use the Prototyping Methodology as the model for this projects development. By applying this methodology, the author believes that this project will be progressing continuously and there will be no long delays until it can be completed right on the scheduled time.
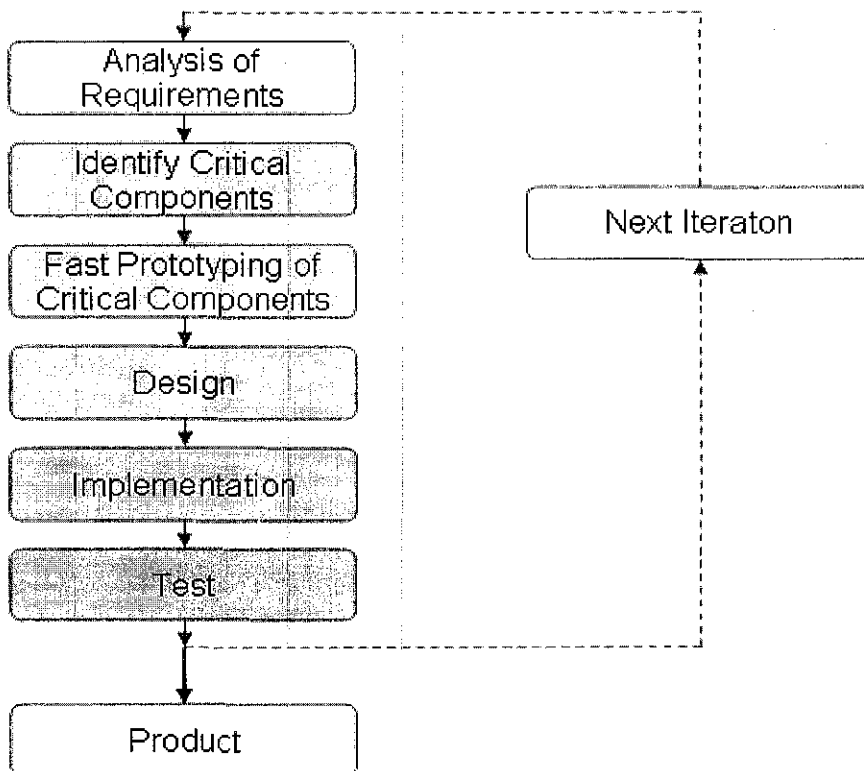
Figure 3.1: The Prototyping Methodology

3.1 **Analysis**

In this stage, the problem is defined and important criteria are generated for use in the design stage. Research and analysis will be done to find the needs, desires and capabilities of the end users, the customers or tourists. This was done through research from the internet itself, reference from previous FYP projects, research from surveys on websites and tourism.

Research and comparison are made with various tourism website in the internet. Some of the websites are user friendly and some otherwise. Not many website have a flash player or interface that can really interact with the user. A system that uses flash and can interact with user, receive feedback from user and even plan their holiday is something new that many tourism website lack of.

With the system implemented on the website, flash are highly used as a medium to get the users feedback to the system. The following is the flow of information through Flash:

- Flash receives user input through a custom Flash user interface.
- ActionScript formats the user input into data.
- The formatted data is sent to a Web server.

The following is the flow of information through the Web server:

- The Web server receives the data and passes it to an application server (for example, JSP, Perl, ColdFusion, ASP, PHP).
- The application server splits up and processes the data.
- The application server submits its results to the Web server.
- The Web server sends its results to the Flash movie in the browser.

20

The following is the flow of information back in Flash:

- Flash receives the formatted data.
- ActionScript reads the data and changes the movie based on the results.

1. Submits formatted request via browser

2. Processes request & returns formatted data to web server

3. Transmits formatted data to client browser

4. Processes response via ActionScript modifying Flash display
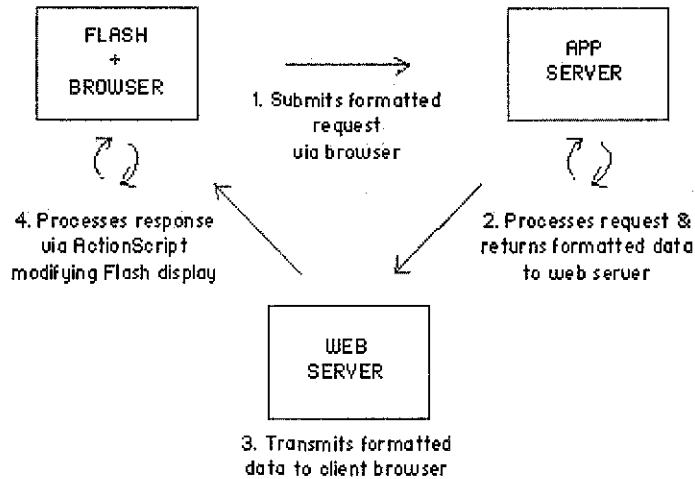
3.1. 1 <u>Gathering User Input</u>

In applications with HTML interfaces, you use pre-built text fields and other form elements to create forms. Form input is sent to the server page by page by pressing a submit button. Flash provides pre-built text fields for you to use to create forms; other elements, such as check boxes and radio buttons, must be built, taken from a common library, or downloaded from the Flash Exchange. Although this can make building standard forms more time consuming, it gives you greater flexibility over layout design and content flow. It also allows users to reuse common elements without re-downloading them. In Flash you use ActionScript to develop form logic and process the data that will be submitted to the server.

21

3.2 **Design**

In the design phase, first task needs to be completed is the system architectural design. This includes developing the infrastructural design such as the system, hardware, software specifications and the system architecture of the whole processes involve in the system. The next thing that the author needs to work on during design phase is on the system flow. System architecture is on how the system is going to function inside, components that need to be included in the system and what methods will the system work with. The interface design mainly will researched on how the user navigate and use the system, how will it react with animations from users action and what are the procedure and processes that the user would need to comply with. Sample of interface design and system architecture will be discussed in chapter 4.

3.3 **Implementation and Test**

The proposal from the design stage was implemented into a low fidelity prototype, a non-functional but representative model. The system will be made and implemented on a single computer as a prototype. Here, a simple website related to tourism is made and the flash helper system will be implemented into it. In this phase, testing will be conducted to test the reliability of the system prototype. The testing will be focuses on the coding error detection, animation scripts, layers of frames in flash, and implementing functions into the flash system with actionscript coding. When the system has been evaluated thoroughly and free from error, it will then be tested by selected users to experience the system itself. This testing is vital to ensure that user satisfaction can be achieved and any bug or error later can be fixed so that further improvement can be made.

3.4 **Development Tools**

Hardware:

- Personal Computer
- Processor : Pentium III or higher
- Memory : 285 MB of RAM or higher
- Speaker

Software:

- Microsoft Windows XP
- Macromedia Dreamweaver MX
- Macromedia Flash MX 2004
- Macromedia Flash Player
- Adobe Photoshop CS2
- MySQL
- PHP

Macromedia Flash MX 2004 has been the major tools in developing the system. Macromedia Flash is a suitable program that can be used in developing the system within a website. A Flash application is a Web application that uses Flash to collect user information, send that information to a server to process, and display the results.

CHAPTER 4

RESULT AND DISCUSSION

4.1 System Architecture
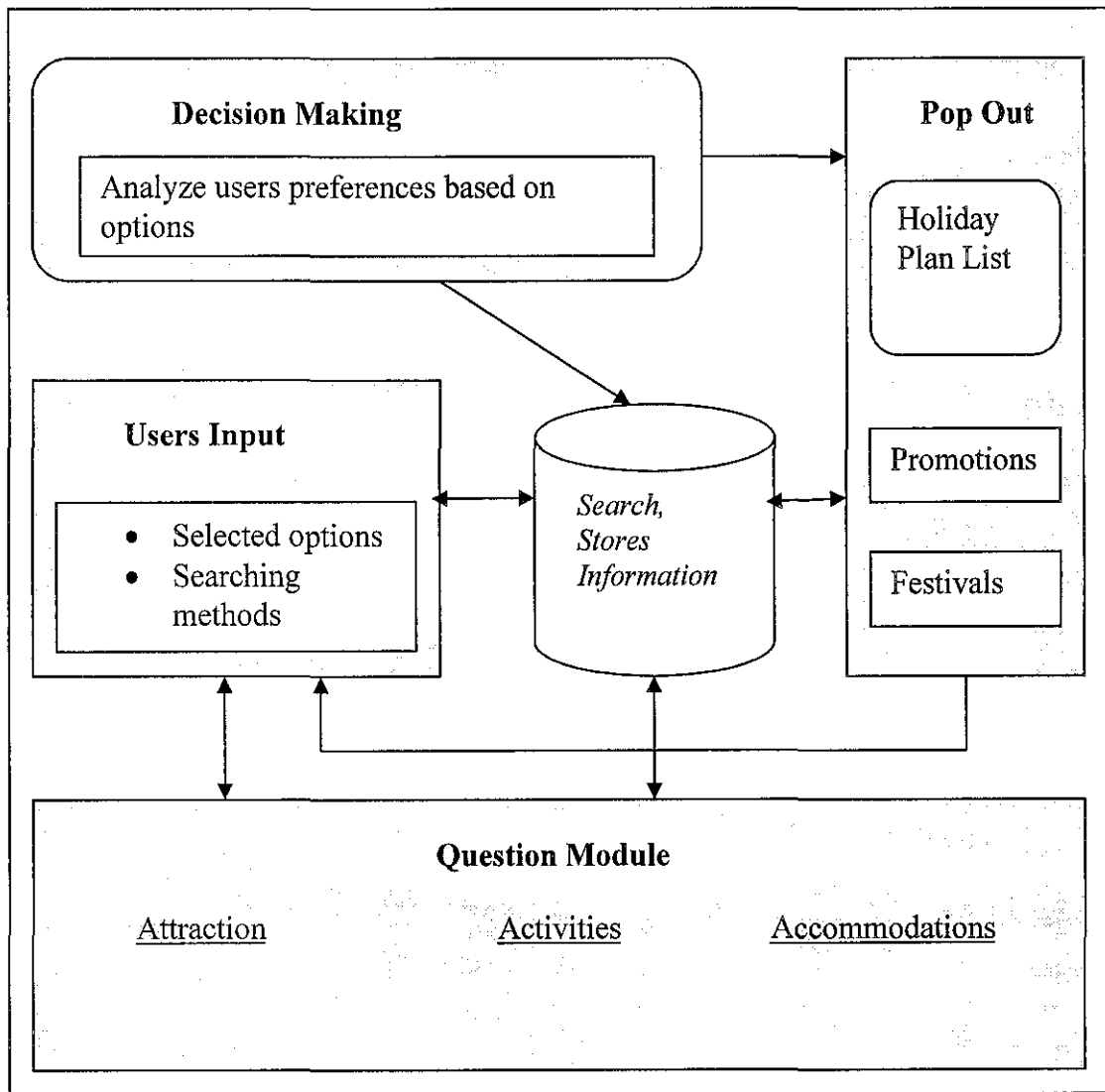
Flash Helper System Architecture

Based on system architecture, the system will consist of 3 main functions. First task is getting the users feedback. This function will get the users input from set of menus with available options. From these choices, the system will search for user's preferences and gave come out with results. Meanwhile, the system will analyze user's action and pop up a new menu containing list of promotions or festivals available from the websites. Throughout the entire process, the flash system will do different animations based on users input. Lastly, the system will produce quotation or list of user's previous choice so that it cans advice or help them plan their holiday carefully.
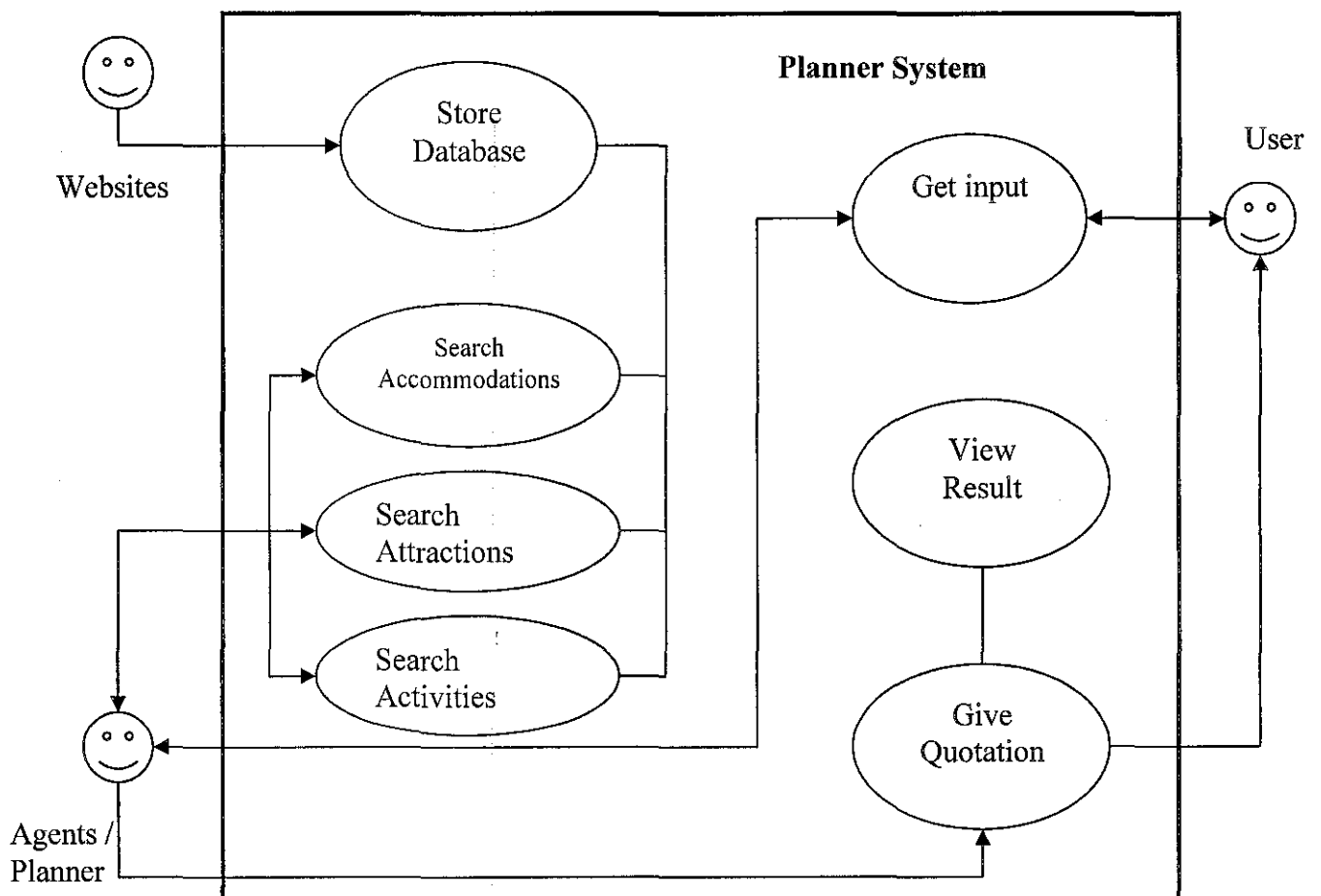
Use-Case Diagram

4.2 **Flow Chart**

```
                              ┌─────────────┐
                              │    Begin    │
                              └─────────────┘
              ┌──────────────────────┼──────────────────────┐
              ▼                       │                      ▼
       ┌─────────────┐                │               ┌─────────────┐
       │ Attraction  │                │               │ Activities  │
       └─────────────┘                │               └─────────────┘
        ┌─────┼─────┐                 │              ┌──────┼──────┐
     ┌──────┐┌──────┐┌──────┐       │          ┌──────┐┌──────┐┌────────┐
     │ K.L  ││ Food ││Theme │       │          │ Eco  ││Beach ││Shopping│
     └──────┘└──────┘│Park  │       │          │Touri.│└──────┘└────────┘
                     └──────┘       │          └──────┘
```

Festivals		Promotions

```
              ┌─────────────┐
              │Accommodation│
              └─────────────┘
             ┌──────┼──────┐
        ┌────────┐┌────────┐┌────────┐
        │ 5 star ││ 4 star ││ 3 star │
        └────────┘└────────┘└────────┘
```

List of Plan, Quotation	—	End
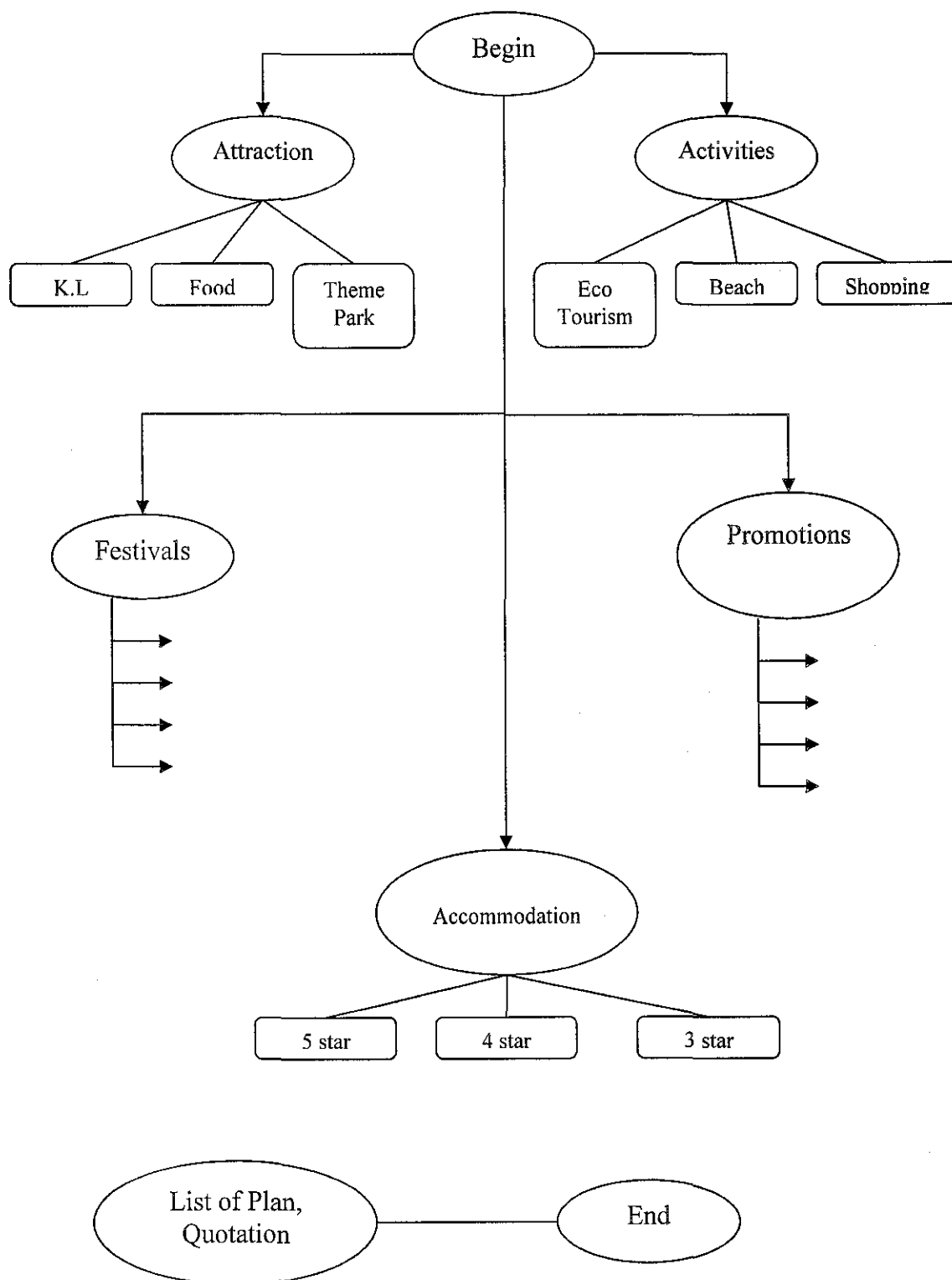
4.3 Project / Outcome
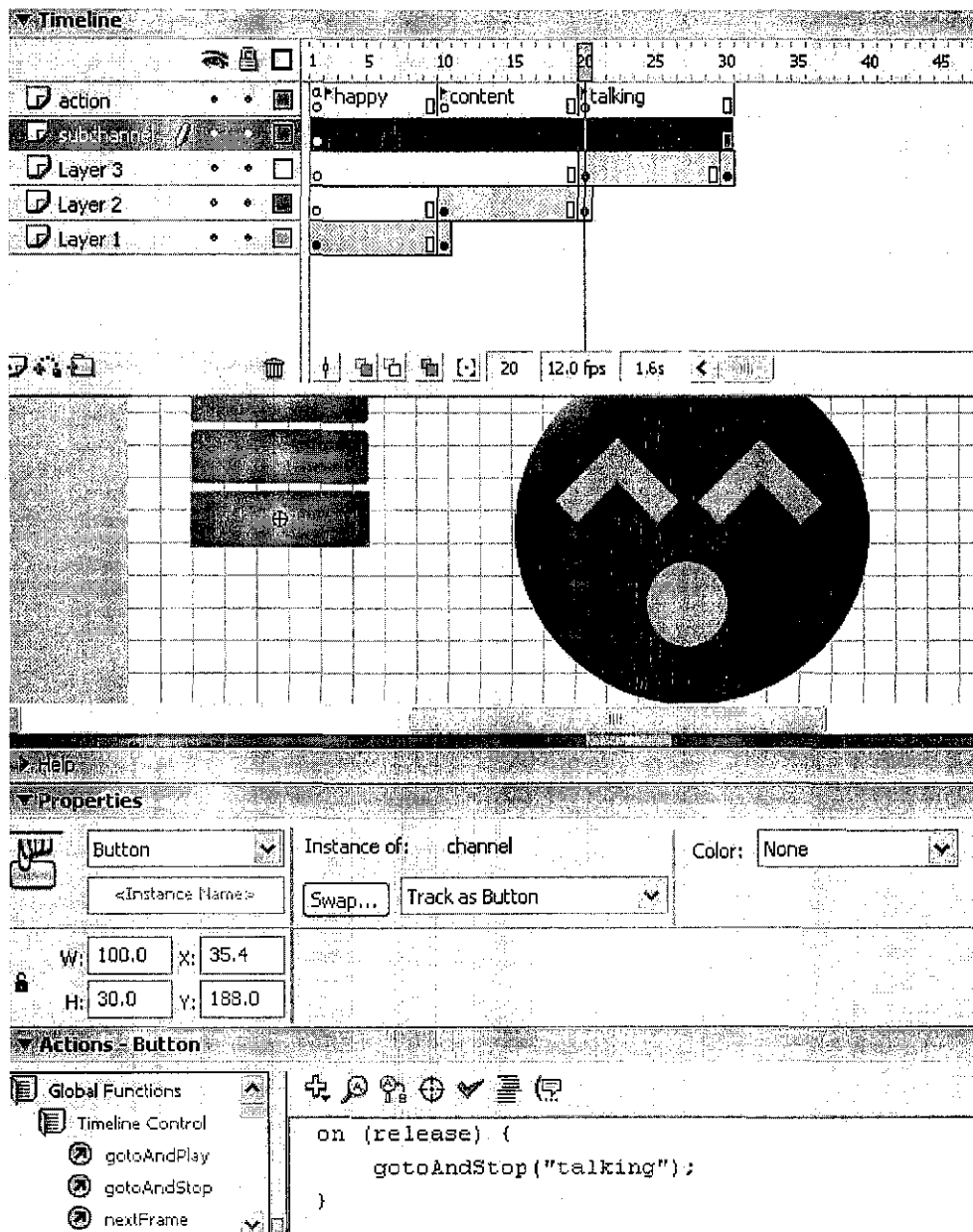
Figure 4.3.1 Sample ActionScripts and Frames of animations.

Figure 4.2.1 shows early systems development and show layers of frames and actionscript to do simple animation from the character.

Symbol Definition(s)

- smiley hehe
- hoho2
- hehe2
- control:menu
- control:scrollbar2
- control:scrollbarcor
- control:find_style
- menu:view:weblink
- button:scroll up
- button:scroll down
- control:search wind
- allbuttons
  - actions : Frame
  - actions : Fram
  - actions : Frame
  - actions : Frame
  - actions : Frame
- button

```
n = Number(n)+1;
// ------------------------------------------------
// wait for enough characters
// ------------------------------------------------
if (Number(length(findit))>=Number(/:minimum_search)) {
    text = eval("/:text" add n);
    if (not (/:case_sensitive)) {
        // ------------------------------------------------
        // convert our search list to l.c.
        // ------------------------------------------------
        input = text;
        call("convert_to_lowercase");
        text = output;
    }
    if (Number(length(findit))<=Number(length(text))) {
        m = 0;
        while (Number(m)<=Number((length(text)-length(findit)))) {
            m = Number(m)+1;
            if (substring(text, m, length(findit)) eq findit) {
                p = Number(p)+1;
```

Figure 4.3.2: ActionScripts coding

Figure 4.3.3 is showing an ActionScript codes that will show the number of links in the menu platform in the system. The system then will calculated on how many links that can be shown and sort it out.

28

4.4 Interface Design

Active search

Manual search

Figure 4.4.1: Main Interface of the Flash Helper System

In early development of the system, the above figure is the interface design for the system. Consist of Menu platform, action button, and the animated character that represent a smiley faces. It also has 2 methods of searching links, active and manual search.

Figure 4.4.2: Websites with Flash Helper System

Figure 4.4.2 shows the final product for Flash Helper System where it was implemented on tourism websites. The system will act as a helper for users to search for keywords from the list and will direct users to the webpage when users click on the results of the search.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

For the conclusion, this project requires more careful approach starting from designing the interface until the program design phase.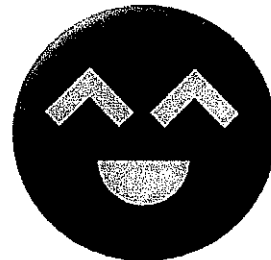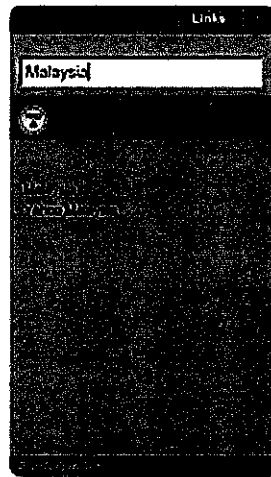 Issues, predicament and problems that arise need to be overcome as soon as possible so that this project can progress continuously. By giving more attention to the aspect of interaction between user and the system, the author can make some improvement towards it so that the system can have better interactivity and better animation in the future. With more research and time given, more ideas and features can be included in Flash Helper System for future advancement and development of the system.

In developing this system, Flash software proves that it can be a good tool in developing flash systems, movies clips, animations and advertisement. ActionScript is a foundation and super tool to create interactive movies, animation for Flash. In today's world of e-tourism, animated or virtual character is a secret way on giving users the new experiences they need and improve the productivity thus usability of the websites.

5.2 Recommendation

For future development, an AI features or artificial intelligent features can be implemented in the system. With it, the system can give suggestion to users based on their input on how to spend holidays in Malaysia. It can also suggest them interesting place or spots that can attract tourist. With research and studies from the literature review, an artificial intelligent can be made possible with the system, all it requires is more time and dedication in finding and analysis on it.

This project has meet majors of its objectives and in the same time, if giving more time surpasses the objectives itself. With more research and development time, more animations and interactive ness can be added to the Flash system thus making it a Flash system that can really attract users and give what they need, a new experiences that no Flash systems and clips can give.

REFERENCES

[1]   Exploring Flash MX 2004
        by James L. Mohler
        Thomson Delmar Learning

[2]   Application of Intelligent Agent Technology for Managerial Data Analysis
        and Mining
        by Ranjit Bose
        University of New Mexico

[3]   Building Great Flash MX Games
        by Matthew David
        Wiley Publishing, Inc.

[4]   Macromedia Flash MX ActionScripting
        by Derek Franklin / Jobe Makar
        Macromedia Press

[5]   websites of http://en.wikipedia.org/wiki/Intelligent_agent

[6]   WebMate : A Personal Agent for Browsing and Searching*
        Liren Chen and Katia Sycara
        The Robotics Institute
        Carnegie Mellon University
        Pittsburgh, PA. 15213

[7] Intelligent Agents for an Artificial Market System
Collaboration of Nikos Karacapilidis and Pavlos Moraïtis

[8] Intelligent Agent for e-Tourism: Personalization Travel Support Agent using Reinforcement Learning
By Anongnart Srivihok and Pisit Sukonmanee

[9] THE DEVELOPMENT OF ECOTOURISM IN MALAYSIA—Is It Really Sustainable?
By Dr. Badaruddin Mohamed
School of Housing, Building & Planning, Universiti Sains Malaysia, Penang

APPENDIX

LIST OF FIGURES

Table 1: Tourist arrivals in Malaysia 1981-2001

Year	Tourist Arrivals	% Growth
1981	2533104	
1982	2774698	9.537469
1983	2926550	5.47274
1984	2947314	0.709504
1985	3109106	5.489473
1986	3217462	3.485118
1987	3358983	4.398529
1988	3623636	7.878962
1989	4846320	33.74191
1990	7445908	53.64045
1991	5847213	-21.4708
1992	6016209	2.890197
1993	6503860	8.105619
1994	7197229	10.66088
1995	7468749	3.772563
1996	7138452	-4.42239
1997	6210821	-12.9948
1998	5200000	-16.2752
1999	7931149	52.5221
2000	10221582	28.87896
2001	12775073	24.98137

Source: Tourism Malaysia, 2002

Table of Tourist Arrivals in Malaysia

Rank	Trip Name	Qt	Qmp	Qxp	Qc	Qd	Qr
1	Thai Gulf-Koh Tao-Koh Nang Yuan-Chumphon	0.410	0.100	0.522	0.001	0.410	1.421
2	Rafting Kheg River-Kang Song Waterfall-Pitsanulok	0.001	0.410	0.522	0.100	0.410	1.398
3	Mo Koh Surin	0.190	0.100	0.522	0.100	0.410	1.300
4	Discovery Pattaya Package (3D2N)	0.001	0.410	0.522	0.001	0.410	1.299
5	Wonderful Thai Similan Island	0.190	0.100	0.522	0.001	0.410	1.201
6	Mae Sot Package 3 days 2 nights	0.001	0.100	0.522	0.001	0.410	1.001
7	Loei Package 3 days 2 nights	0.001	0.100	0.522	0.001	0.410	1.001
8	Kanchanaburi Night Safari Tour 2 days	0.001	0.100	0.522	0.001	0.410	1.001
9	Kanchanaburi Good Health 2days	0.001	0.100	0.522	0.001	0.410	1.001
10	Rafting Hin Peang, Winery, Water fall	0.001	0.001	0.522	0.100	0.410	0.990

Above are table of the result using algorithm of a personalization learner by user behavior with the equation of Figure 2.3.3.7 Learning Algorithm as the formula
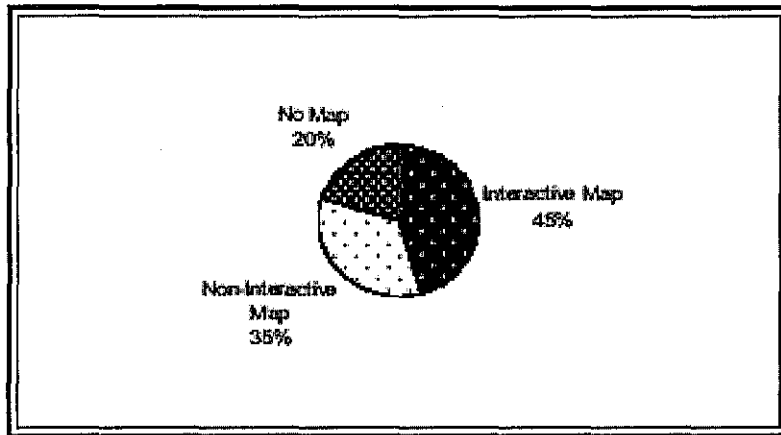
Figure 1. The distribution of sampled websites with interactive maps, non-interactive maps or no maps.

	EU	Asia	US	CA	AZ	AF
■ Sitemap	20	60	40	40		
□ Search	50	60	80			
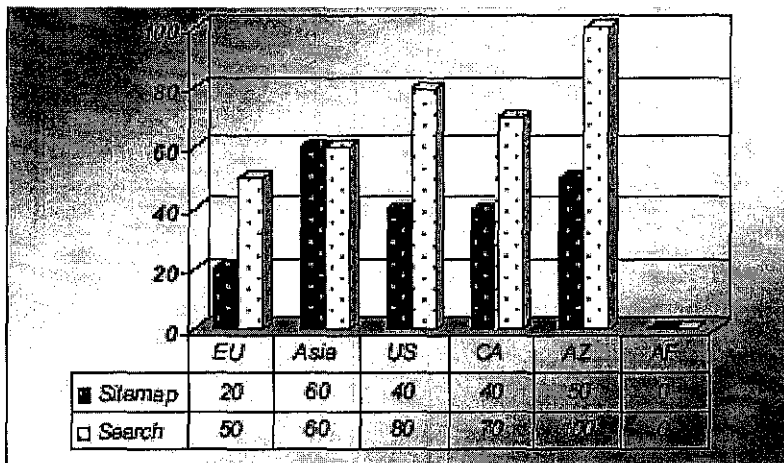
EU-Europe, CA-Canada, AZ-Australia/New Zealand, AF-Africa

Figure 3. Percentages of websites offering sitemap and search.

Figure 1 and 3 shows research of websites that offers a search and sitemap functions and have an percents of interactive and non-interactive maps.

begin

1. get `offerReqMsg`
2. search the selling db for `objectType` = `purchaseObj`
3. *if* not exists then *exit,*

     else for each `objectType.objToBeSold` found *do*

       if all `offerReqMsg.constraints` are satisfied

       3.1. create a new `offerPropMsg`

       3.2. `offerPropMsg.sender` ← `agentID`;

           `offerPropMsg.receiver` ← `offerReqMsg.sender`;

           `offerPropMsg.msgID` ← `offerPropMsg.msgID` + 1;

           `offerPropMsg.purchaseID` ← `offerReqMsg.purchaseID`;

           `offerPropMsg.purchaseObj` ← `objectType.objToBeSold`

       3.3. search in `objectType.objToBeSold` for criteria and

           associated features appear in `offerReqMsg.description`;

           for each match (`criterion` & associated `feature`) found *do*

               add in `offerPropMsg.description` a new item (`cr`, `f`,

               `fv`, `imp`) where, `cr` and `f` correspond to the match found,

               while `fv` and `imp` the values of `featureValue` and `impact`,

               respectively, of `objToBeSold`

           end for

       3.4. *for* each additional pair of `criterion` and `feature` (that not

           exists in `offerReqMsg.description`)

           if the respective `impact` is pro (positive)

               add in `offerPropMsg.description` a new item (`cr`, `f`,

               `fv`, `imp`) where, `cr` and `f` correspond to the additional

               match, while `fv` and `imp` the corresponding values of

               `featureValue` and `impact`, respectively, of `objToBeSold`

           end if

       end for

3.5. search in `objectType.objSold` for
    `offerReqMsg.preferences`;
     for each found *do*
       if there exist `arguments` referring to it
         add in `offerPropMsg.preferences` a new item (`cr1`,
         `impRel`, `cr2`, `listOfArg`) where, `cr1`, `impRel` and `cr2`
         correspond to the match found, while `listOfArg` are links
         referring to this preference of `objToBeSold`;
         add in `offerPropMsg.arguments` a new item
         (`argID`,`pref`,`just`,`imp`) where, `argID`, `pref`, `just`
         and `imp` the corresponding values of `objToBeSold`
       end if
     end for
3.6. *for* each additional preference of `objToBeSold` (that not exists
in
    `offerReqMsg.preferences`) *do*
     if there exist `arguments` referring to this preference
       add in `offerPropMsg.preferences` a new item (`cr1`,
       `impRel`, `cr2`, `listOfArg`) where, `cr1`, `impRel`, `cr2`
       and `listOfArg` the corresponding values of `objToBeSold`;
       add in `offerPropMsg.arguments` a new item
       (`argID`,`pref`,`just`,`imp`) where, `argID`, `pref`, `just`
       and `imp` the corresponding values of `objToBeSold`
     end if
    end for
  end if
end for
end

Figure 6: An offer building strategy

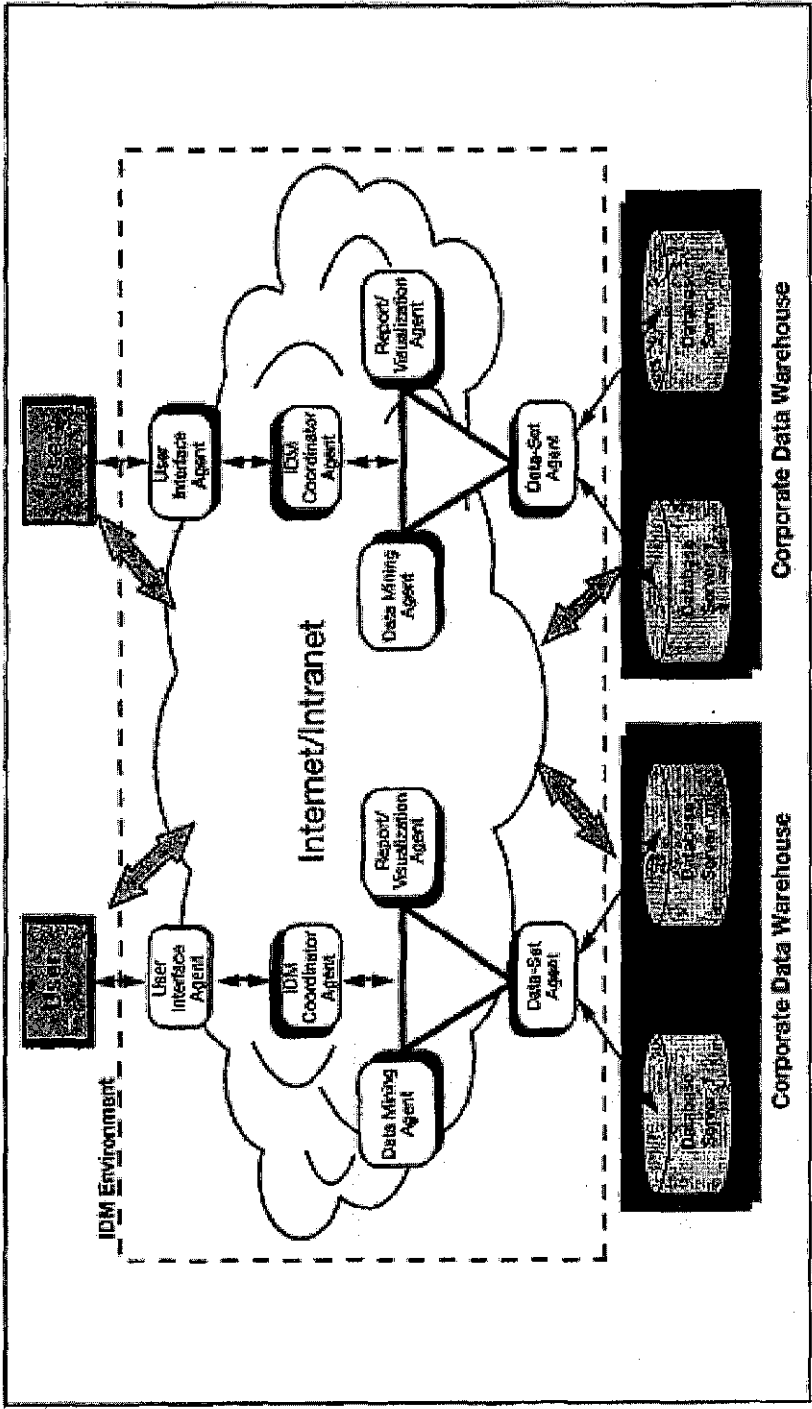Intelligent Agents Algorithm in offerReqMsg coding in Figure 2.3.3.5

Figure 1. IDM Architecture

IDM Architecture, an intelligent agents system architecture.

	🛈	Task Name	Duration	Start	Finish	Predecessors
1	▦	Progress Report	15 days	Mon 7/30/07	Fri 8/17/07	
2						
3		⊟ Research And Analysis	21 days	Fri 8/10/07	Fri 9/7/07	
4	▦	Research on System Requirement	17 days	Fri 8/10/07	Mon 9/3/07	
5	▦	Finding of article journal and surveys	15 days	Mon 8/13/07	Fri 8/31/07	
6	▦	Research Flash and Action Script	9 days	Tue 8/28/07	Fri 9/7/07	
7						
8						
9		⊟ Design	12 days	Mon 9/10/07	Tue 9/25/07	
10	▦	Designing early interface	9 days	Mon 9/10/07	Thu 9/20/07	
11	▦	Designing systems	9 days	Thu 9/13/07	Tue 9/25/07	
12						
13						
14		⊟ Implementation	10 days	Mon 9/24/07	Fri 10/5/07	
15	▦	Design and implementation of system	4 days	Mon 9/24/07	Thu 9/27/07	
16						
17						
18		⊟ Testing	6 days	Fri 9/28/07	Fri 10/5/07	15
19	▦	Testing of system	6 days	Fri 9/28/07	Fri 10/5/07	

Layers of flash sequence and coding


```
n = Number(n)+1;
// ----------------------------------------------
// wait for enough characters
// ----------------------------------------------
if (Number(length(findit))>=Number(/:minimum_search)) {
    text = eval("/:text" add n);
    if (not (/:case_sensitive)) {
        // ----------------------------------------------
        // convert our search list to l.c.
        // ----------------------------------------------
        input = text;
        call("convert_to_lowercase");
        text = output;
    }
    if (Number(length(findit))<=Number(length(text))) {
        m = 0;
        while (Number(m)<=Number((length(text)-length(findit)))) {
            m = Number(m)+1;
            if (substring(text, m, length(findit)) eq findit) {
                p = Number(p)+1;
```
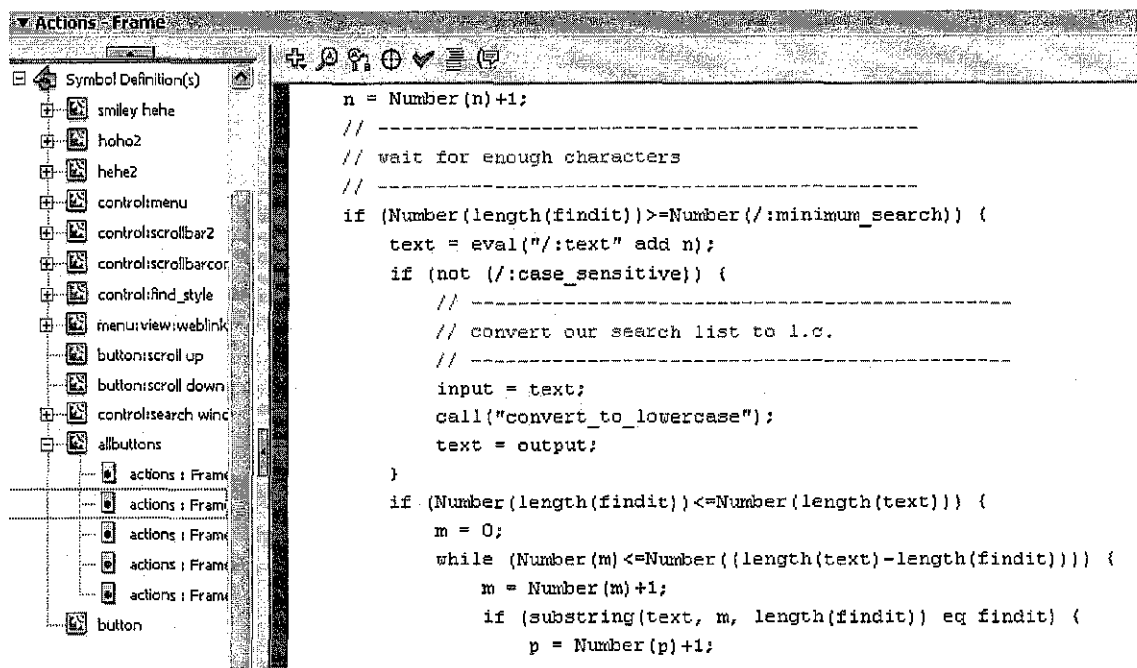
ActionScript coding that will shows list of links in the menu box