

## CHAPTER 6

### SYSTEM DESIGN

#### 6.0 Introduction

System design is purposed to create a blueprint for a new system that satisfied all documented requirements included functional requirements and non-functional requirements. During this stage, all necessary outputs, inputs, interfaces and processes are identified. The topic discussed in this chapter include the system architecture, process flow, entity relationship diagram, context diagram, data flow diagram, database design and graphical user interface design.

#### 6.1 System Standard

At section 2.1.8, there are several learning technology standards that had been discussed, such as SCORM, AICC and IMS. LOOOP system architecture design is based on one of the described standards, SCORM. SCORM standard is selected because this standard is designed specifically for browser-based delivery of learning content. On LOOOP, the learning content is retrieved on demand. Firstly, LOOOP user chooses a specific learning content from his or her learning contents list. After receiving the command, LOOOP system sends a command line that is written in LOOOP language to LOOOP socket that is placed on the server. The command line contains the instruction on what types of data should be retrieved from the database and returns the appropriate data to LOOOP client system. Every single record from learning contents database is similar to SCO (Sharable Content Object) in SCORM. The records will then be sent back in a

package that is in LOOOP language to the LOOOP client system according to the command line or instruction that was sent to LOOOP socket earlier. LOOOP client system, which is a learning management system extracts the SCO from the package and displays the learning content to users. Figure 6.1 illustrates the LOOOP standard.

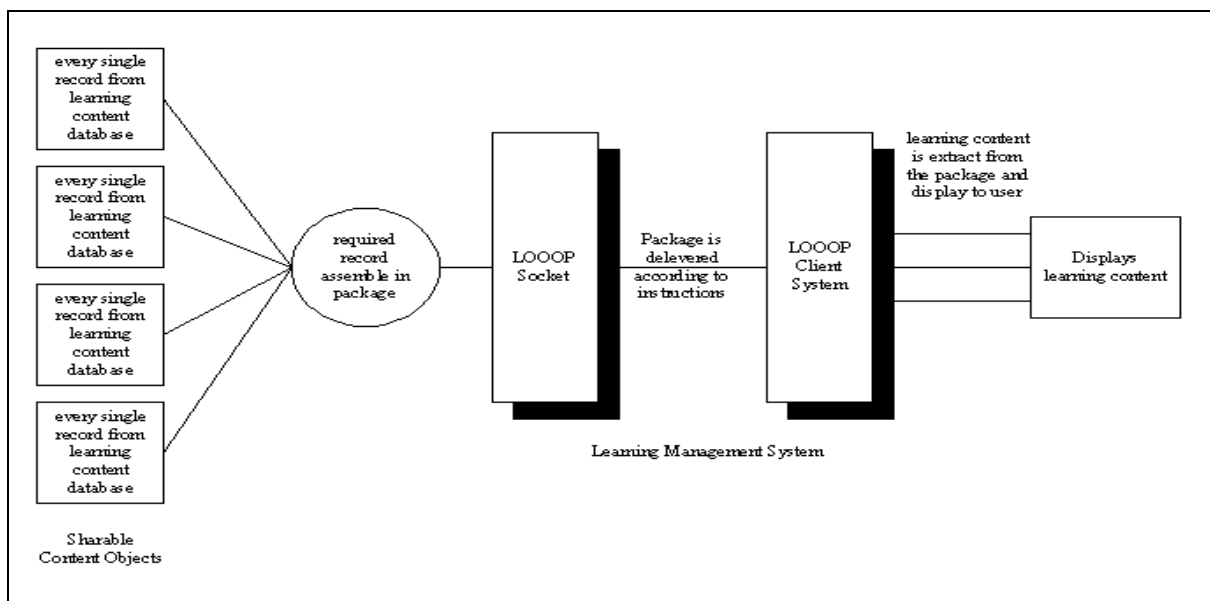


Figure 6.1 System standard

## 6.2 System Architecture

Overall LOOOP system can be divided into two parts, one the client-side and the other server-side. Client-side application is the main operation system for LOOOP. Server-side application is a socket that purposes to receive required information passing from client-side application and sent back the results to client-side application. The server-side application is only operated when LOOOP registered users choose to save their learning contents online. Figure 6.2 illustrate the overall system architecture for LOOOP system.

In order to communicate with the online database that is located on the server, the required information that passing to server-side application is SQL statement, purposed statement and related data. After the server-side application performed the SQL statement and retrieved the data from the online database, the results are sent back to client-side

application. The required information is delivered to server-side application and the results sent back to client-side application are in LOOP language, a language that is designed based on SCORM concept for used in LOOP data passing. This language is similar to Extensible Markup Language (XML). The difference between both languages is the tag. For example, XML language used <CHAPTER> and LOOP language use {{|CHAPTER|}}. Figure 6.3 illustrated the difference between XML and LOOP language.

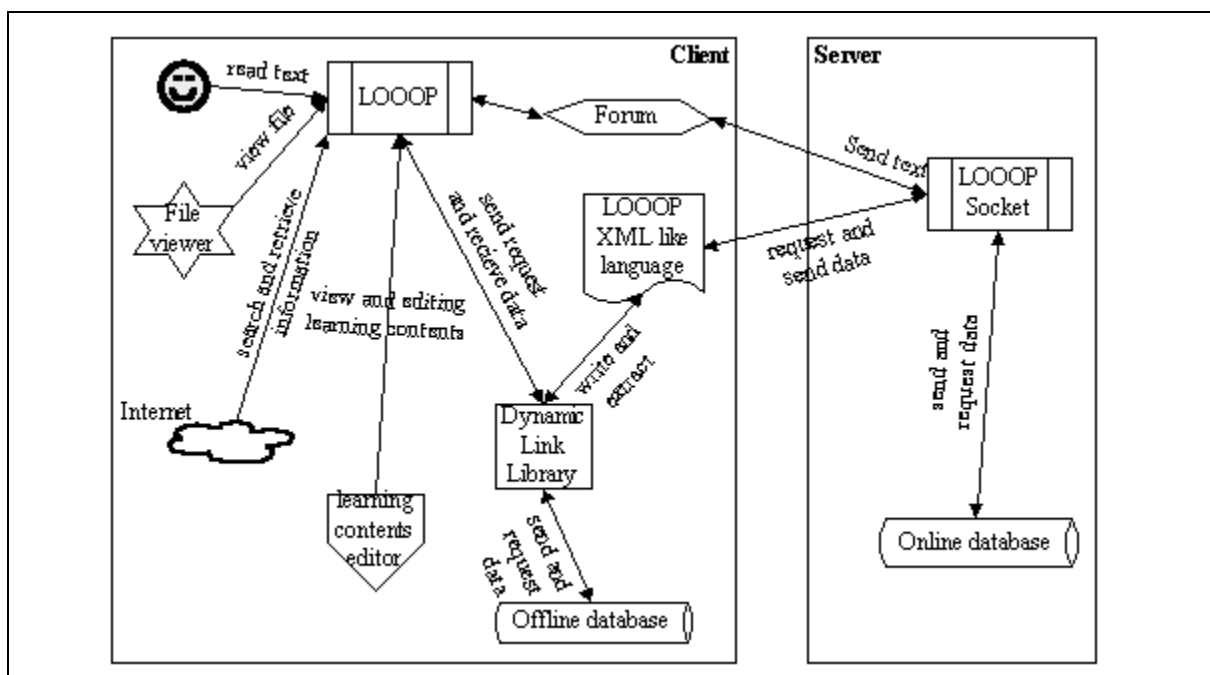


Figure 6.2 System architecture

<u>XML</u>	<u>LOOP language</u>
<pre>&lt;?xml version='1.0' encoding='utf-8'?&gt; &lt;LOOP&gt;   &lt;LTITLE&gt; Title &lt;/LTITLE&gt;   &lt;LCONTENT&gt; Learning Content &lt;/Content&gt;   &lt;LNOTE&gt; NOTE &lt;/LNOTE&gt; &lt;/LOOP&gt;</pre>	<pre>{{ LOOP }}   {{ LTITLE }} Title {{/LTITLE }}   {{ LCONTENT }} Learning content {{/Content }}   {{ LNOTE }} NOTE {{/LNOTE }} {{/LOOP }}</pre>

Figure 6.3 Differential between XML and LOOP language

As mentioned in earlier paragraph, client-side application is the main operation system for LOOP. Besides performing as a client-server application, client-side application also perform as a standalone application when LOOP registered users choose not to save their learning contents online. An offline database that had the same data

structure with the database on the server is installed in client-side machine. Client-side application communicates with the offline database through a self-created DLL file. This DLL not only created for communicated with LOOOP database only, but also for reusability and interoperability purposed. Learning Objects developers can communicate with the contents of Learning Objects placed inside the database using this DLL file.

Besides DLL file, the system is developed as an ActiveX control. This has increased the reusability, interoperability and durability for the system. The ActiveX control can be merged and used with other applications or web pages. The accessibility of the system can also increase while the control merged with the web page. In this project, the control is merged with client-side application.

There are five main features tools in client-side application. The five main features tools are learning contents editor, information searching agent, file viewing tool, reader agent and communication tool. Learning contents editor is a tool for viewing or editing the learning contents. This tool is developed using DHTML editing control. This ActiveX control is selected because this control allowed HTML page editing where the learning contents are saved in HTML format.

Information searching agent is another feature tool for LOOOP, this agent extracted the search result page from GOOGLE.com and listed out the records found on a tree view control. Besides searching information from Internet, this agent also searched keyword from the sharable contents of Learning Objects. File viewing tool is a simple tool that played files. Currently, this tool is able to open six format types of files that includes picture files (GIF, JPEG, BMP), multimedia files (WAV, AVI) and shockwave flash files (SWF). Two ActiveX control had been used to play the files, included Shockwave Flash control and Windows Media Player control.

Reader agent is an agent used together with learning contents editor. This agent is used to read the text on learning contents. This agent is developed using another ActiveX control, Microsoft Voice Text control. Communication tool is a forum that is similar to chatting application found on the market, such as ICQ, mIRC and MSN Messenger. This tool is developed using Microsoft Winsock control. This communication tool is not designed for one-to-one conversation, every online users can read the sender message.

### **6.3 Process Flow**

Although LOOOP system is a client-side application that needs a server for data processing, it can also be used without a server as a standalone application. There is a selection for user to choose either online database or offline database. Figure 6.4 illustrated the process flow for LOOOP. At the beginning, user is shown with LOOOP system login and register page. User is required to enter his or her registered user name and password for verification. If the details matched with the records found in the database, the system retrieve the learning contents from his or her personal table on database. If the data entering did not match with any records in the database, the system inform and request user to enter the correct user name and password or request user to register again to the system.

For these new users, they are required to register themselves to the system. Registering of a new user is a must because the system needs to create a new table for them to record their personal learning contents. First, the system records the new user name and password that had been entered. A same user name is not allowed in this system because this can cause a conflict when creating a new table with the same table name and when retrieving learning contents. If the user name is found in the database, the system informs the new user to enter a new user name. A list of registered user name can be viewed from the drop down list box for user name on log in frame. Second, the system guides new user

to choose the available contents of sharable Learning Objects to add into their initial personal learning contents. Sharable Learning Objects are existing Learning Objects that have been created and to be shared among users. This operation is not compulsory because user can choose to add the contents in new future.

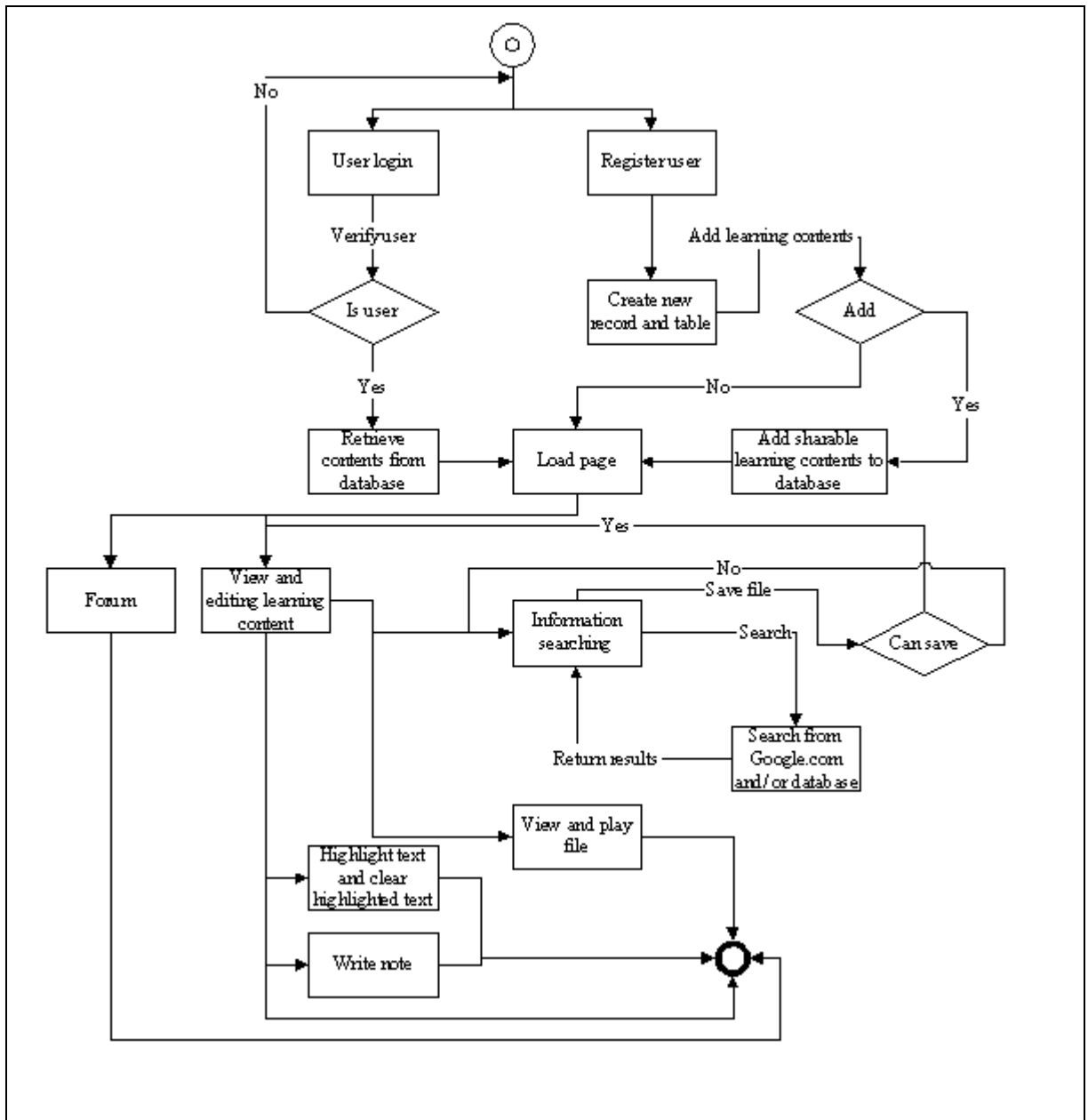


Figure 6.4 Process flow

Generally, the process flow that use Learning Objects begins after user successfully login to the system. Every time a user login to the system, the system retrieves the Learning

Objects from user personal database and display the content of Learning Objects on the screen. After the contents are successfully allocated on the screen, the Learning Objects is now ready to be modify or to be viewed. Besides, users are able to perform traditional learning styles such as highlighting the text and clear highlighted text, write note and information searching via Google.com using the features tool provided in the system. Any of the changes will change the contents of currently viewed Learning Objects when it is saved. Thus, user can also create a new Learning Object or share the existing personal Learning Objects among registered users. Most of the functions provided are simple where system can execute directly, but still there are some are complex, such as information searching, view and play file.

Once the users login themselves to the system, they are able to access to the forum for discussion. The forum will lists out the entire available online registered user. Besides having a discussion among users, users can also send file including the Learning Objects that are saved in LOOOP language file to any user in the forum. For information searching, user needs to give a term, word or phrase to the system. LOOOP retrieves the search page for the search term from Google.com and extract the first 30 results. The results are displayed to user in a list box with URL and its description. Besides, LOOOP also searched the keywords on sharable contents of Learning Objects saved in the database and display the results to user. For durability purposes, user can save the information found in the system as their personal learning contents. Furthermore, Learning Objects can be a combination of text, image, animation and sound and video. For this purpose, a 'view and play file' process includes in the system for playing multimedia file or animation file and viewing image file that associates with Learning Objects. When a file is requested, LOOOP retrieve the selected file data string from database and convert it into its original file type. The file is displayed on a provided file viewer.

### **6.3.1 How Learning Objects Applied In The System**

In LOOOP, Learning Objects is saved as a single record in database. The database of the system acts as a Learning Objects repository. The database contains several tables including a single table for sharable Learning Objects. Sharable Learning Objects is the Learning Objects that can be shared by user and the contents of Learning Objects can only be modify after another user saves it as his or her personal Learning Objects in his or her own table. Besides sharable Learning Objects, user can create a new Learning Objects using the system from blank or save from information searching. The Learning Objects in this system can be viewed and modified when it is retrieved from repository and its content showed to user. Besides, the animation file, multimedia file or image file associates with Learning Objects can also be played using the system. Thus, user can also perform some traditional learning styles such as highlighting text and jotting down notes on the currently viewed Learning Objects. The modification of the contents can be saved into the repository for future review.

### **6.4 Entity Relationship Diagram And Context Diagram**

ERD is a diagram illustrating the relationship between two or more entities. Crow's foot notation, which is one of several types of cardinality notation, had been chosen to be used in this project to show relationships between entities (Gary, 2001). Before defining the entities, the type of Learning Objects repository has been reviewed to decide which type of Learning Objects repository needs to be developed. In LOOOP, every single record in a table is a Learning Objects and every single field in a table is the metadata for the Learning Objects.

LOOOP contained nine entities including user, sharable content, personal content, content picture file, board file, animation file, multimedia file, image file and last view



chapter record. All these entities are separated as small chunks that has few attributes by table in database which means they are not arranged by chapter, but arranged by type of media or content.

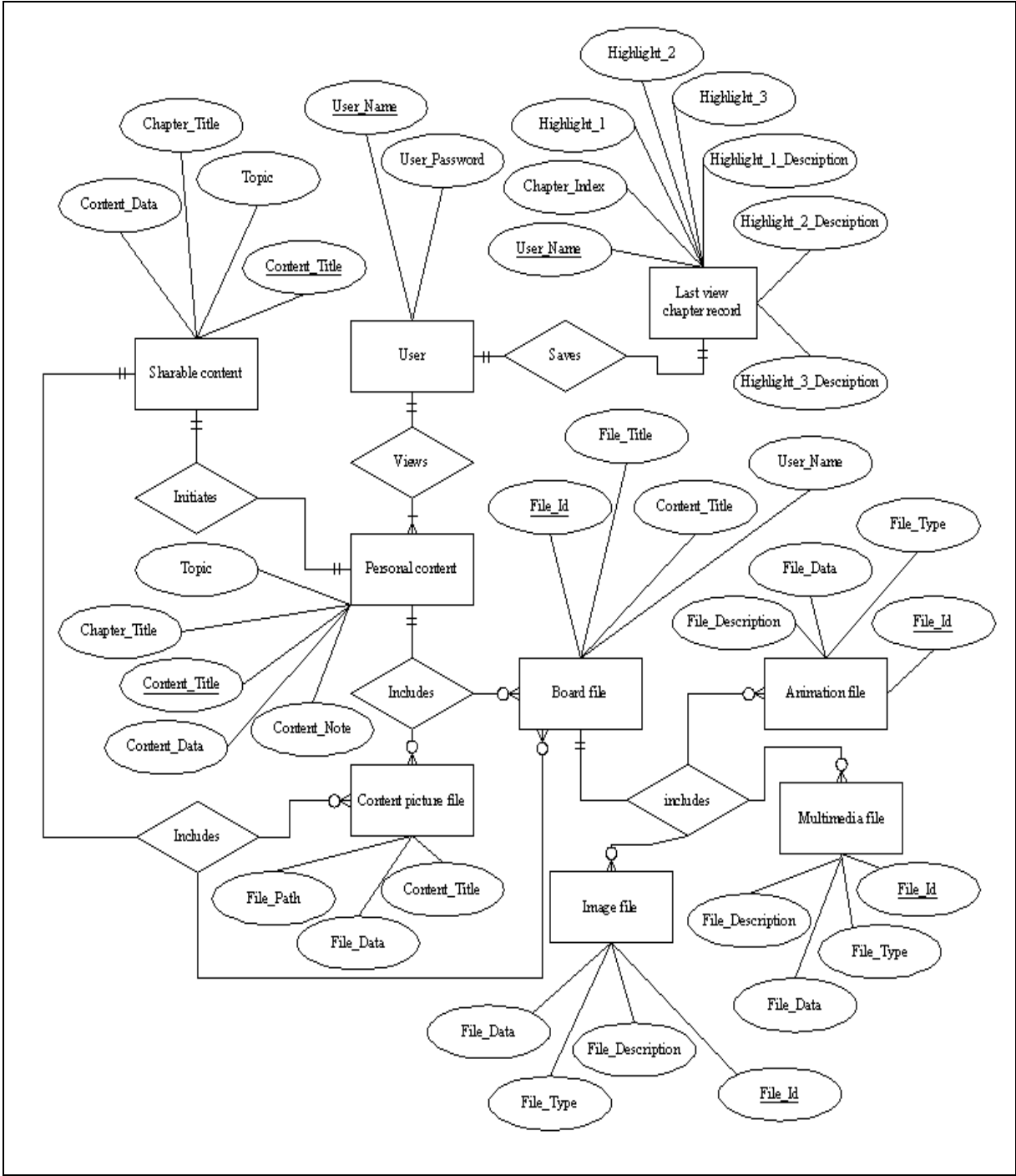


Figure 6.5 ERD for LOOP

To form a complete ERD for LOOOP, the process flow that is described in section 6.3 is reviewed. When a user log in to LOOOP, user can modify the record of the 'last view chapter record' at the same time. This is shown with the cardinality between user and last view chapter record is one to one. After user successfully log in to LOOOP, user is able to view one or more content from his or her personal content. The cardinality between user and personal content will be then one to many.

A page of content may be a plain text or has picture on it. The cardinality between personal content and content picture file is one to zero or many. The board file can also be attached with the page of content. The cardinality between personal content and board file is one to zero or many. Board file is a table that contains information for animation file, multimedia file and image file such as 'file\_id'. As the description, a single record of board file can have either one of these three types of file. The cardinality between board file and animation file, multimedia file and image file is one to zero or many. At last, the cardinality between personal content and sharable content is one to one because no duplicated title for contents are allowed to be saved. As an information, sharable content is the entities that contains Learning Objects that is shared and can only be modify after saved to personal content.

Context diagram is a simple top-level diagram that had only one process that represented the entire system and showed the relation to system environment (Gary, 2001). In order to represent the entire system, Yourdon symbol set had been chosen to be used in this project. Figure 6.6 illustrated the context diagram for LOOOP. In this diagram, there are three entities (Learner, Google.com and LOOOP Socket) and one process (LOOOP).

This context diagram was drawn with the assumption that learner is using online database to save their learning contents. For learner who is using standalone application and not connected to Internet access, entity Google.com and entity LOOOP Socket are

eliminated because LOOOP needed Internet connection to access to the entity Google.com and LOOOP Socket. For learner who is using standalone application and connected to Internet, only entity LOOOP Socket is eliminated because the learning contents is saved in learner computer.

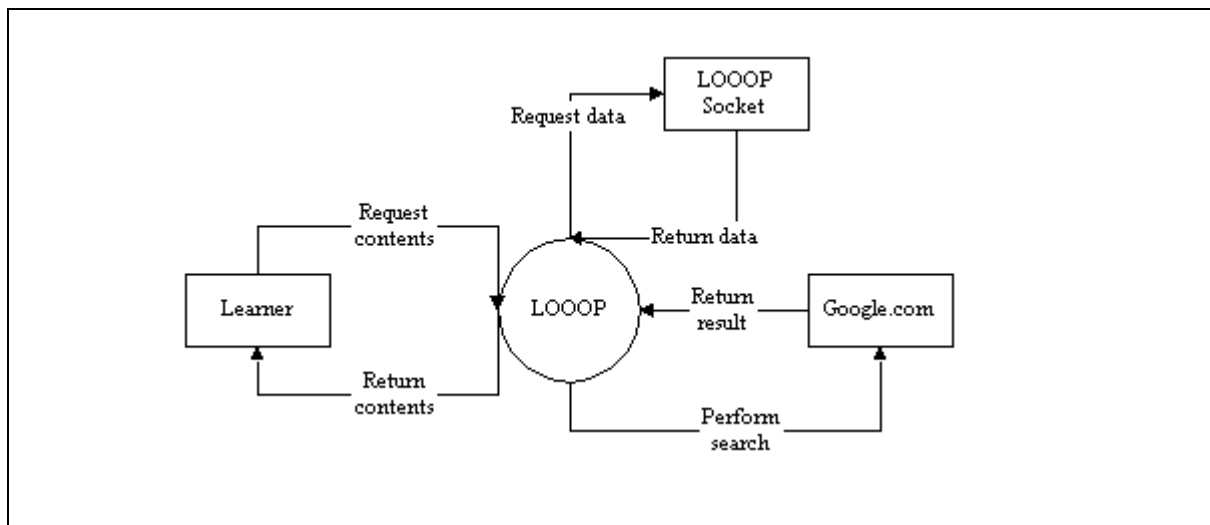


Figure 6.6 Context Diagram for LOOOP

First, learner requested learning contents from LOOOP. LOOOP received the command and sent the requested message to LOOOP Socket that is located on server in LOOOP language. LOOOP Socket retrieves the related data from database and returns the results to LOOOP in LOOOP language. The results that is in LOOOP language are extracted before being displayed to learner on LOOOP. For information searching, LOOOP sent the terms to Google.com and retrieved the result page from Google.com. The result page is extracted for getting the records URL and descriptions. The URL and descriptions are displayed to learner through LOOOP.

## 6.5 Data Flow Diagram

A data flow diagram (DFD) showed how data moved through an information system and represented a logical model that showed how the system is done. Context diagram is

one of the data flow diagrams. By zooming in context diagram, a diagram that shows major processes, data flows, and data stores was drawn. This diagram is called diagram 0 (Gary, 2001). Figure 6.7 illustrated the diagram 0 for LOOOP.

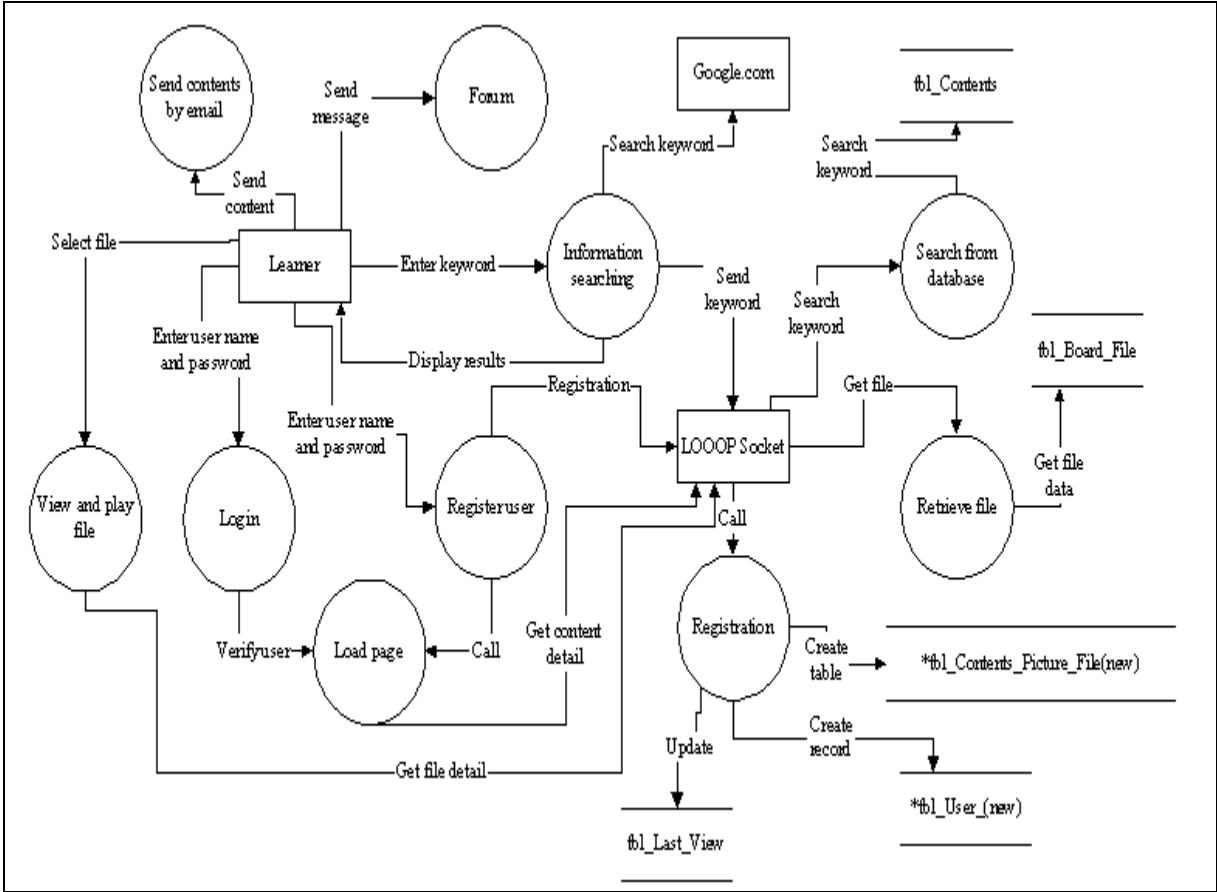


Figure 6.7 Diagram 0 for LOOOP

As in the figure, learner is a starting point. The process login and process register user are first process when the system start executing, depending on whether the learner is an existing user or new user. For existing user, learner enters the user name and password. The load page process is called after the login process verifies the user name and whether the password is valid. For new user, learner enters a new user name and password to register to the system. The register user process is executed and then sent the user name and password to entity LOOP Socket for registration. In the registration process, table `tbl_User_(New)` and `tbl_Contents_Picture_File_(New)` are created. Besides, a new record

is also created in table tbl\_Register\_User. The (New) is a changeable word depending on the new user name that had been saved. After this, the load page process begins.

For information searching, learner needs to key in a keyword. Information searching process is called to send the keyword to entity Google.com and also entity LOOOP Socket for the process search from database. Process search from database searched the keyword from the table tbl\_Contents, which is a table that saved all the sharable learning contents.

To view and play file, learner needs to select a file. The process view and play file is called to get the file data and information from the database through entity LOOOP Socket. The sent contents by email process is called when learner needed to send their personal learning content to his or her friend. Lastly is the forum process. This process is called when sending a message to the forum.

Figure 6.7 showed the diagram 1 for login. First, learner enters his or her user name and password to the system. The login process called the verify user name and password process to check the validity of the user name and password that had been entered. If the information is not valid, login process requests the learner re-enter the user name and password. If the information is valid, then load page process is called.

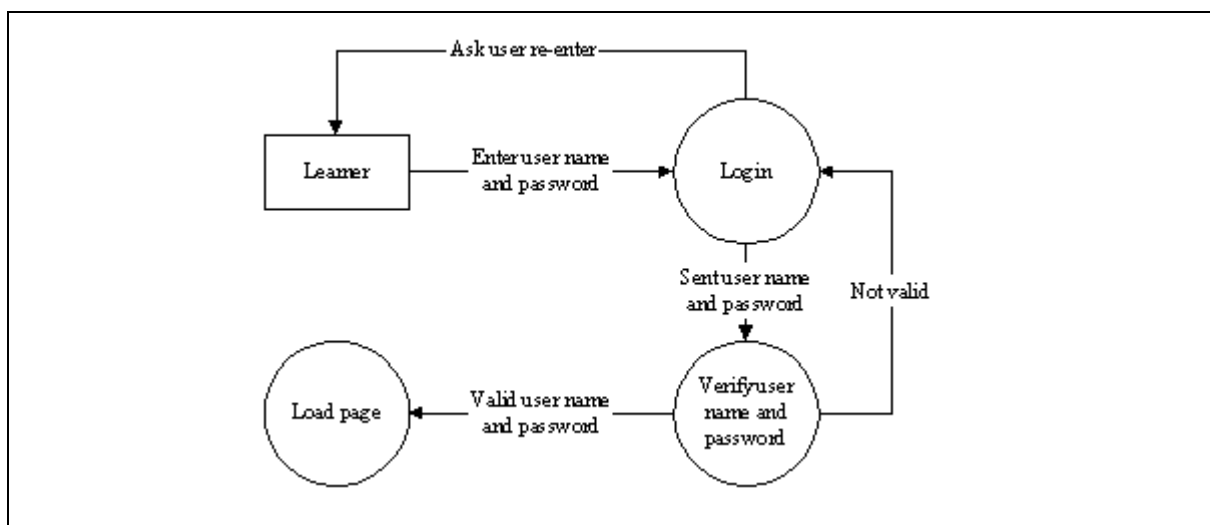


Figure 6.8 Diagram 1 DFD for login

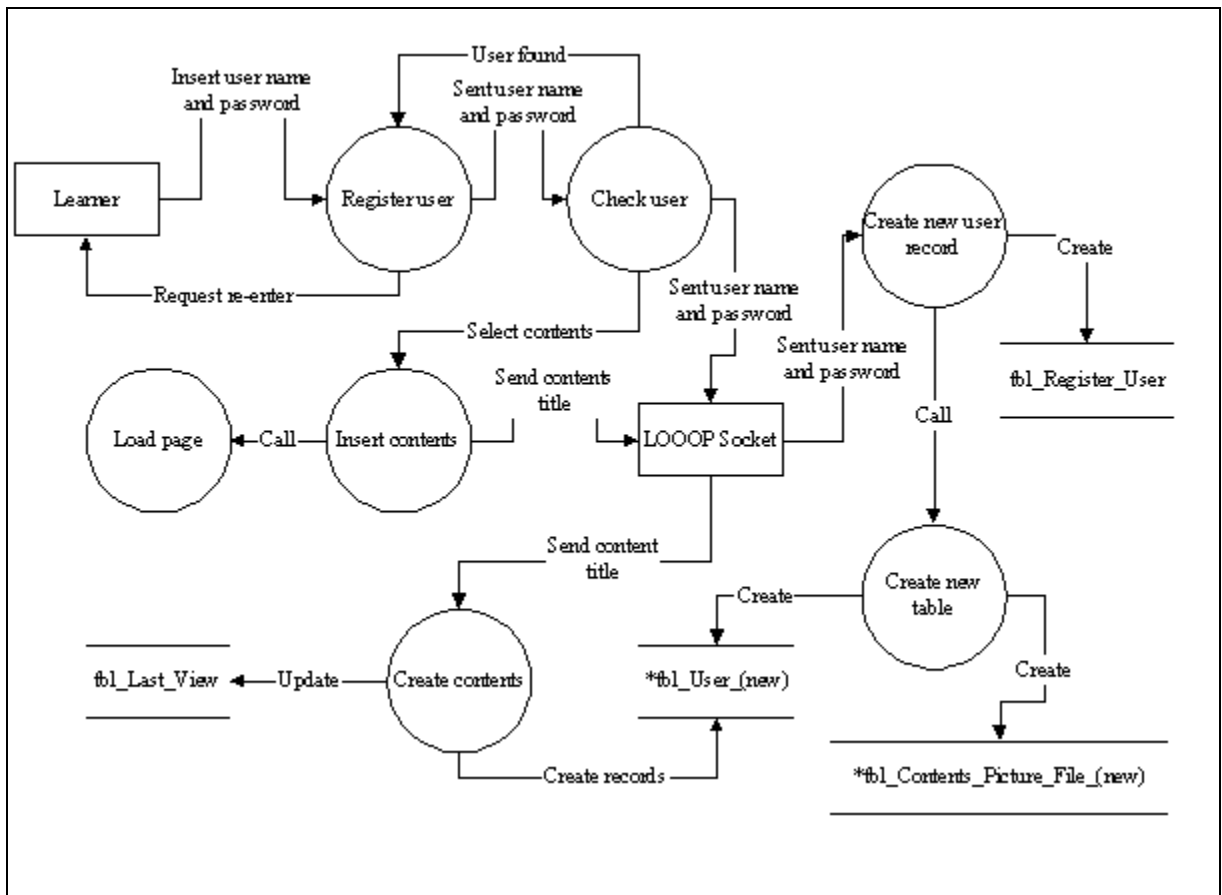


Figure 6.9 Diagram 1 DFD for register user

Figure 6.9 illustrated the diagram 1 for register user. First, learner enters user name and password to the system. The check user process is called to check whether the user name that had been entered is found in the registered user list. If record is found, the system requests learner to enter another user name. If user name cannot be found, the system sends the user name and password to entity LOOP Socket to save it in the table tbl\_Register\_User. At the same time, create new table process is called to create a tbl\_Contents\_Picture\_File\_(new) table and create a tbl\_User\_(new) table. On the system side, learner is guided to insert contents process. If learner selected to insert the available sharable learning contents as his or her personal learning contents, then insert contents process sent the selected contents title to entity LOOP Socket. The create contents process is called to create the learning contents record on table tbl\_User\_(new) and updated the tbl\_Last\_View. After that, the load page process is called.

Figure 6.10 illustrated the diagram 1 DFD for load page. Load page process is called after the login process or register user process. This process send the content title to entity LOOOP Socket for retrieving the learning content, note and files associate with this content title. The process retrieve learning content is called to retrieve the learning content from table tbl\_User\_(new). The process retrieve file is called to retrieve the file data and information from table tbl\_Board\_File and table tbl\_Contents\_Picture\_File\_(new). Besides, the update record process is also called to update the table tbl\_Last\_View.

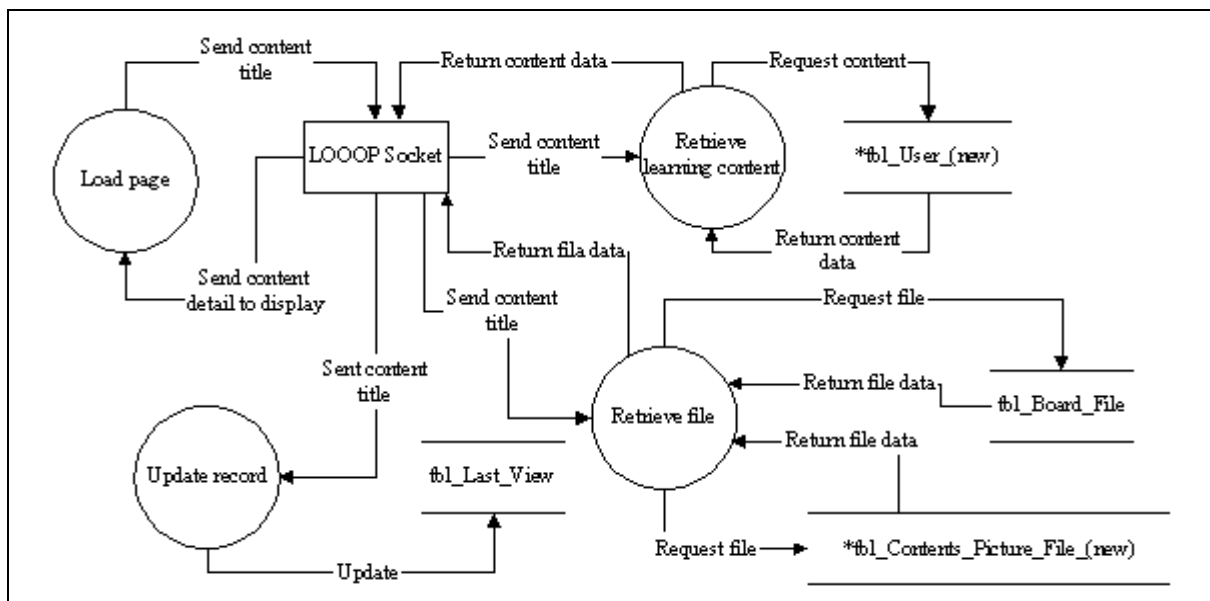


Figure 6.10 Diagram 1 DFD for load page

Figure 6.11 illustrated the diagram 1 DFD for view and play file. First, learner selects a file from the files listed box. The system sends the file information to LOOOP Socket. LOOOP Socket will then extract the LOOOP language and get the ‘file\_id’. The file detail and data are sent back to view and play file process after LOOOP Socket performed a search file process and retrieved the file details from the table ‘tbl\_Animation\_File’, ‘tbl\_Multimedia\_File’ and ‘tbl\_Image\_File’ in database. To avoid duplication of ‘file\_id’ in these three table, the ‘file\_id’ follows the ‘file\_id’ created on ‘tbl\_Board\_File’. The file data is sent to a process called convert string to file. The process

converts the file from string to its original format before displaying to learner. For your information, the file is saved in string format in database.

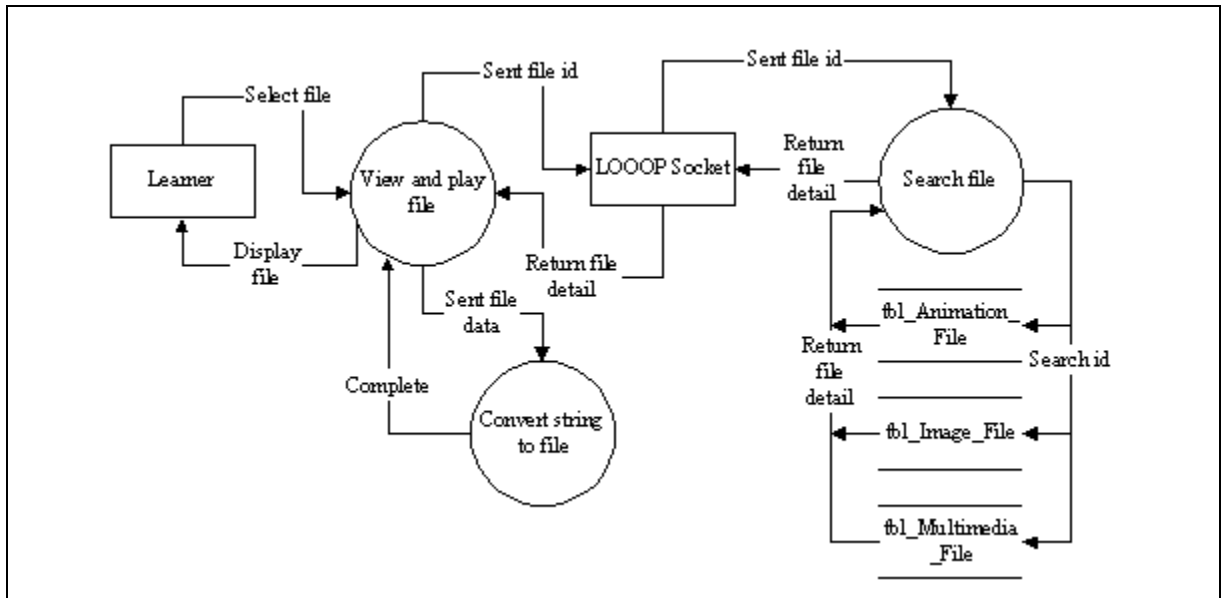


Figure 6.11 Diagram 1 DFD for view and play file

Figure 6.12 illustrated the diagram 1 DFD for forum. First, learner typed and sent the message to a process called forum. This process sent the message to LOOP Socket. After LOOP Socket searched for all available online learners' IP address, the message is sent to all online learners according to their location.

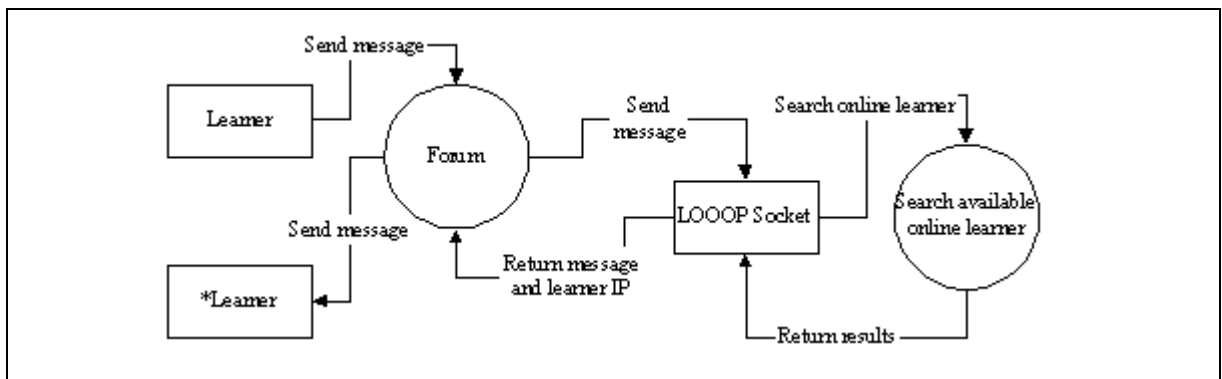


Figure 6.12 Diagram 1 DFD for forum

Figure 6.13 illustrated the diagram 1 DFD for information searching. First, learner needs to enter a keyword, word or phrase to perform a search. The system sends the keyword to Google.com and retrieves the result page. The result page is passed through a process called extract HTML page process to get the entire records URL and descriptions



and is shown to learner. At the same time, the system also sends the keyword to a process called search from database. This process sends the keyword to LOOOP Socket. A search match word process is called to search the matched word from sharable learning contents table and the results returned to LOOOP Socket. LOOOP Socket sent back the results and list out the related learning content title to learner.

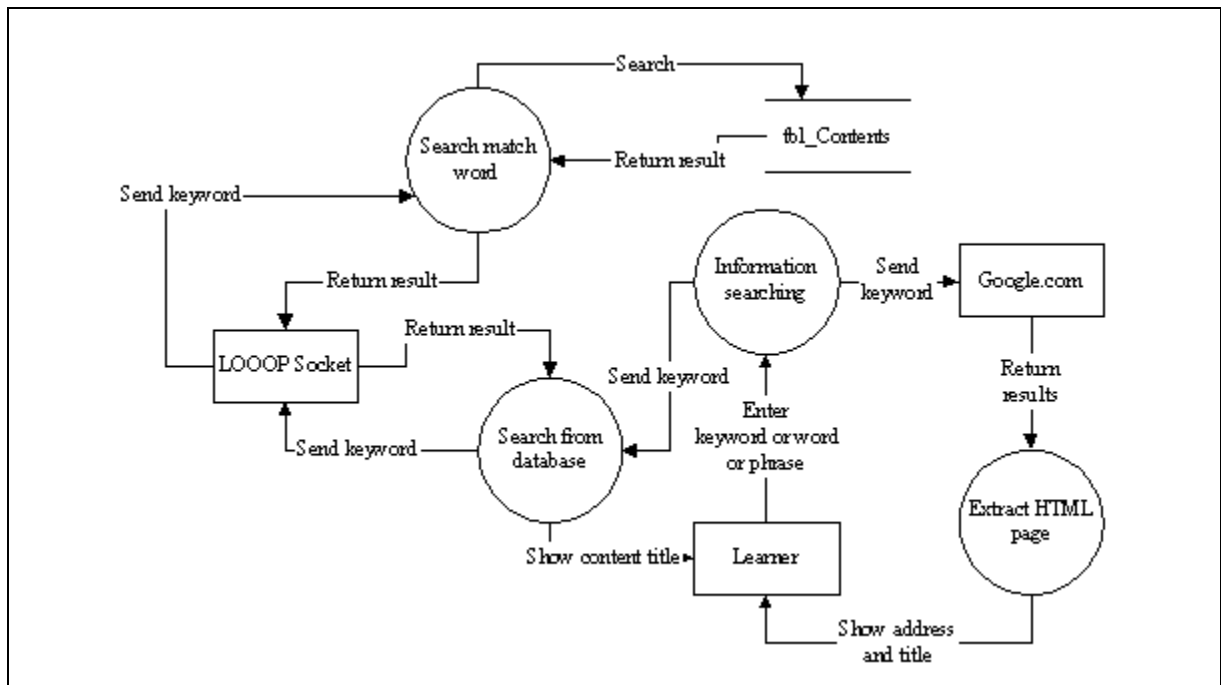


Figure 6.13 Diagram 1 DFD for information searching

Figure 6.14 showed the diagram 1 DFD for send learning contents by e-mail. This is a simple process in LOOOP. When learner clicks on the send email button, the button send a command to the system to write the current displayed learning content and note to LOOOP language file. The file is sent as attachment on the email and sent to recipient. The sending email process required user configured his or her client mailing system, such as Microsoft Outlook.

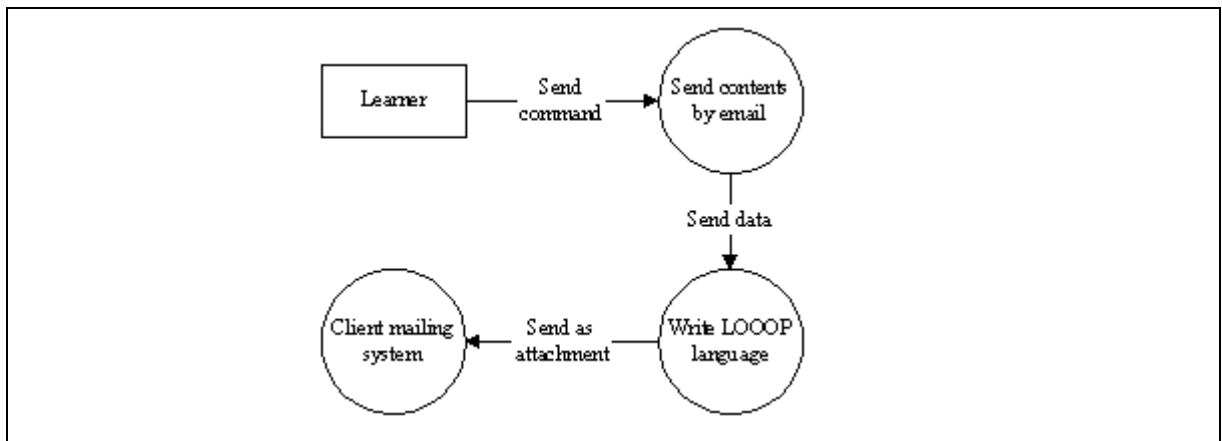


Figure 6.14 Diagram 1 DFD for sent learning contents by e-mail

## 6.6 Database Design

Database and metadata are important component in LOOOP. A database is a carefully organized set of data stored on a computer and managed by a special application called a database management system (DBMS) (Burrows, 2000). A metadata is data that provided information about, or documentation of, other data managed within an application or environment. In this project, the metadata is the field name of every table created in the database. For example, the metadata for table `tbl_Register_User` were `User_Name` and `User_Password`.

There are several tables in LOOOP database, such as `tbl_Register_User`, `tbl_Contents`, `tbl_Board_File`, `tbl_Animation_File`, `tbl_Multimedia_File`, `tbl_Image_File`, `tbl_Contents_Picture_File`, `tbl_Last_View`, user defined table `tbl_User_(new)` and `tbl_Contents_Picture_File_(new)`. Table '`tbl_Register_User`' is used to record those registered users details. Table '`tbl_Contents`' is used to record the sharable learning contents or contents of Learning Objects, such as content title, chapter title and contents. Table '`tbl_Board_File`' is used to record the external file such as animation file, multimedia file and image file id and information. Table '`tbl_Animation_File`' is used to record the file data of animation file. Table '`tbl_Multimedia_File`' is used to record the file data of

multimedia file such as video file and audio file. Table ‘tbl\_Image\_File’ is used to record the additional image file data that associates with the Learning Objects. Table ‘tbl\_Content\_Picture\_File’ is used to record the picture file that is found inside the learning content. Table ‘tbl\_Last\_View’ is used to record the chapter that user last view and other information. User defined table ‘tbl\_User\_(new)’ and ‘tbl\_Contents\_Picture\_File\_(new)’ are created while a new user registered to the system. Table ‘tbl\_User\_(new)’ is used to record user personal learning contents. This table is an instance for table ‘tbl\_Contents’. Table ‘tbl\_Contents\_Picture\_File\_(new)’ is used to record the picture file that is found in the learning content page. This table is an instance for table ‘tbl\_Contents\_Picture\_File’. The (new) is renamed with user name while a new user register to the system. The number of user defined table are increased when a new user is registered. Table 6.1 to 6.7 illustrated the details of each table in LOOOP database and figure 6.15 showed the relationship between each table.

Table 6.1 Details for table tbl\_Board\_File

<b>Table: tbl_Board_File</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Id	Primary	Autonumber	Number of file.
File_Title		Text	Title of the file.
User_Name		Text	The user name that current file belong to.
Content_Title		Text	The chapter that current file belong to.

Table 6.2 Details for table tbl\_Contents

<b>Table: tbl_Contents</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
Topic		Memo	Topic for current chapter.
Chapter_Title		Memo	Chapter title.
Content_Title		Memo	Content title.
Content_Data		Memo	Chapter contents.

Table 6.3 Details for table tbl\_User\_New

<b>Table: *tbl_User_New</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
Topic		Memo	Topic for current chapter.
Chapter_Title		Memo	Chapter title.
Content_Title		Memo	Content title.
Content_Data		Memo	Chapter contents.
Content_Note		Memo	Content note.

Table 6.4 Details for table tbl\_Default\_File

<b>Table: tbl_Default_Picture_File</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Path		Text	File path and file name.
File_Data		Memo	File data that had been converted to string.
Chapter_Title		Memo	The chapter that current file belong to.

Table 6.5 Details for table tbl\_Contents\_Picture\_File\_new

<b>Table: **tbl_Contents_Picture_File_new</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Path		Text	File path and file name.
File_Data		Memo	File data that had been converted to string.
Chapter_Title		Memo	The chapter that current file belong to.

Table 6.6 Details for table Register\_User

<b>Table: tbl_Register_User</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
User_Name	Primary	Text	User name (Only unique user name is allowed to be create).
User_Password		Text	User defined password.

Table 6.7 Details for table tbl\_Last\_View

<b>Table: tbl_Last_View</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
User_Name		Text	User name that current record belong to.
Chapter_Index		Text	Last content view index. (tree-view node index)
Highlight_1		Text	First highlight color.
Highlight_2		Text	Second highlight color.
Highlight_3		Text	Third highlight color.
Highlight_1_Description		Text	First highlight color description.
Highlight_2_Description		Text	Second highlight color description.
Highlight_3_Description		Text	Third highlight color description.

Table 6.8 Details for table tbl\_Animation\_File

<b>Table: tbl_Animation_File</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Id	Primary	Autonumber	Number of file.
File_Description		Memo	Description of the file.
File_Type		Text	File type.
File_Data		Memo	File data that had been converted to string.

Table 6.9 Details for table tbl\_Multimedia\_File

<b>Table: tbl_Multimedia_File</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Id	Primary	Autonumber	Number of file.
File_Description		Memo	Description of the file.
File_Type		Text	File type.
File_Data		Memo	File data that had been converted to string.

Table 6.10 Details for table tbl\_Image\_File

<b>Table: tbl_Image_File</b>			
<b>Field name</b>	<b>Key</b>	<b>Data type</b>	<b>Description</b>
File_Id	Primary	Autonumber	Number of file.
File_Description		Memo	Description of the file.
File_Type		Text	File type.
File_Data		Memo	File data that had been converted to string.

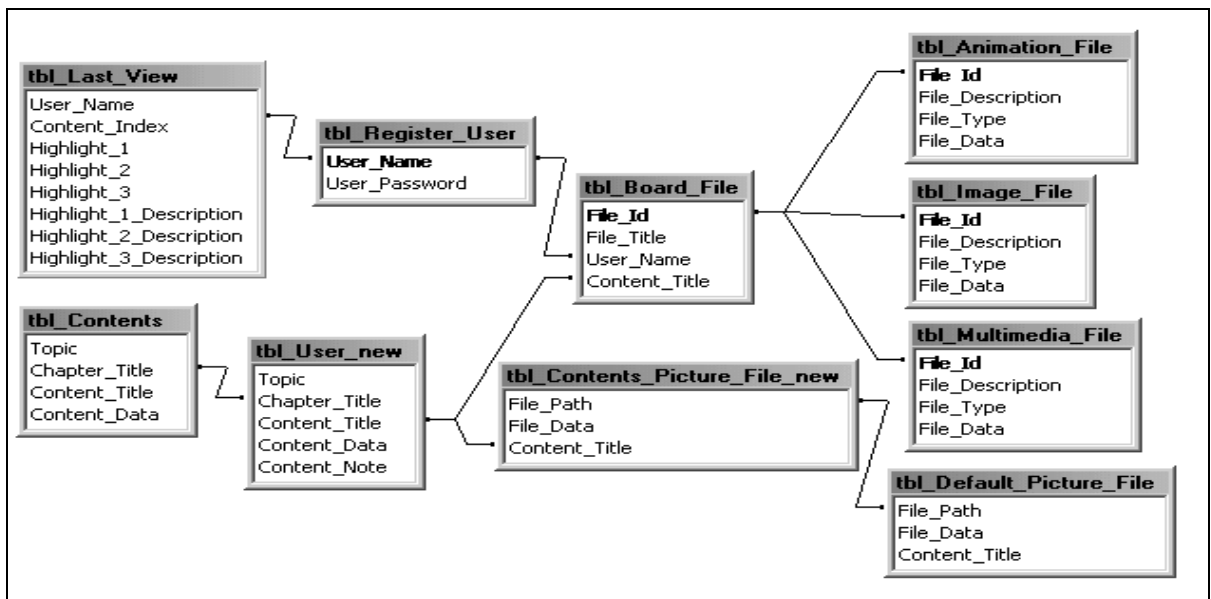


Figure 6.15 Relationship between each table

## 6.7 Graphical User Interface Design

LOOOP graphical user interface (GUI) is the place where the system interacted with learner. LOOOP GUI is designed with WIMP (Window, Icon, Menu and Pointer) interface based on the template showed in figure 6.16, 6.17 and 6.18.

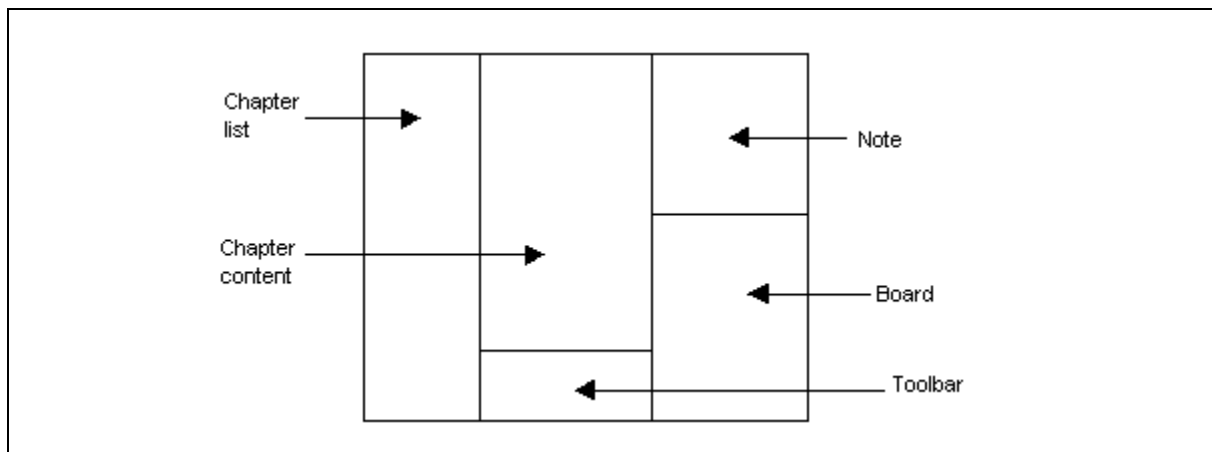


Figure 6.16 GUI template for learning contents page

Figure 6.16 illustrated the GUI template for learning contents page. There are five parts that are drawn in this template, such as chapter list, chapter content, toolbar, note and board. The chapter list is the list of all the content title in learner personal learning contents. The chapter content is the displayed content page. Toolbar is the editing tool for learning contents page. Note is an additional note that write by user. Board is the list of external file that associated with the current content.

Figure 6.17 illustrated the GUI template for search information page. There are three parts drawn in this template, such as keyword input box, result list and web page. The keyword input box is the place for user key in the term, phrase or keywords that is needed to search. The result list is the list of all records found including search from database or Internet. The web page is the place to display the selected result page.

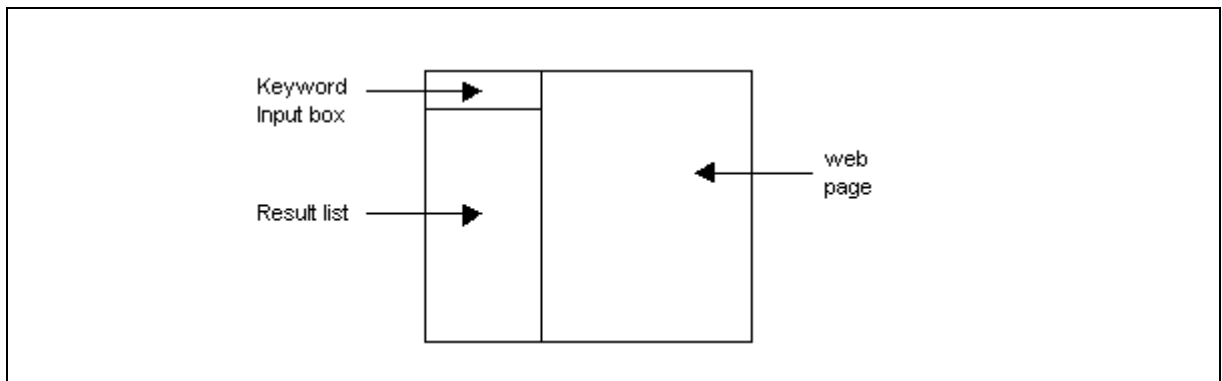


Figure 6.17 GUI template for search information page

Figure 6.18 illustrated the GUI template for view file page. There are two parts in this template, such as file information and file viewer. The file information is the place showing the details of currently displayed file. The file viewer is the place displaying or playing the file.

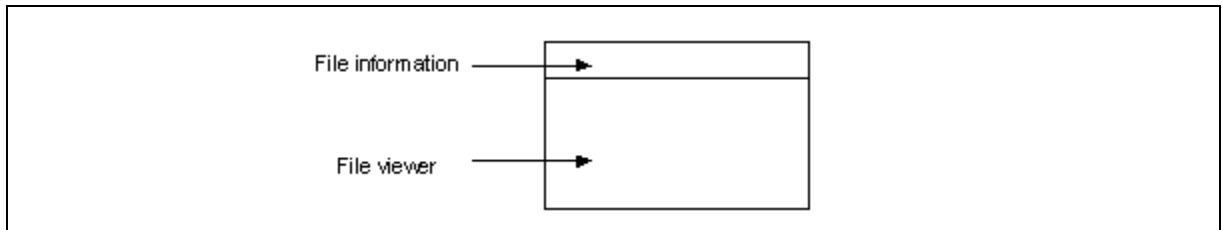


Figure 6.18 GUI template for view file page

From the templates that had been designed earlier in this project, the actual LOOOP GUI had finally designed out. The entire system had four pages and two external windows. The four pages are login page, learning contents page, search information page and view file page. The two external windows are forum window and help window. Figure 6.19 to 6.22 shows the four pages and 6.23 to 6.24 shows the two windows.

Figure 6.19 illustrated the login page for the system. This page contains two frames, such as log in frame and register frame. Log in frame is for registered user to login to the system, while register frame is for new user to register to the system. Log in frame contains one user name combo box control, one password textbox control and one log in button.

Register frame contains one user name textbox control, one password textbox control and one confirm password textbox control. For log in frame, user needs to select the user name that is already in the combo box control because the entire registered user name has appeared in the list.

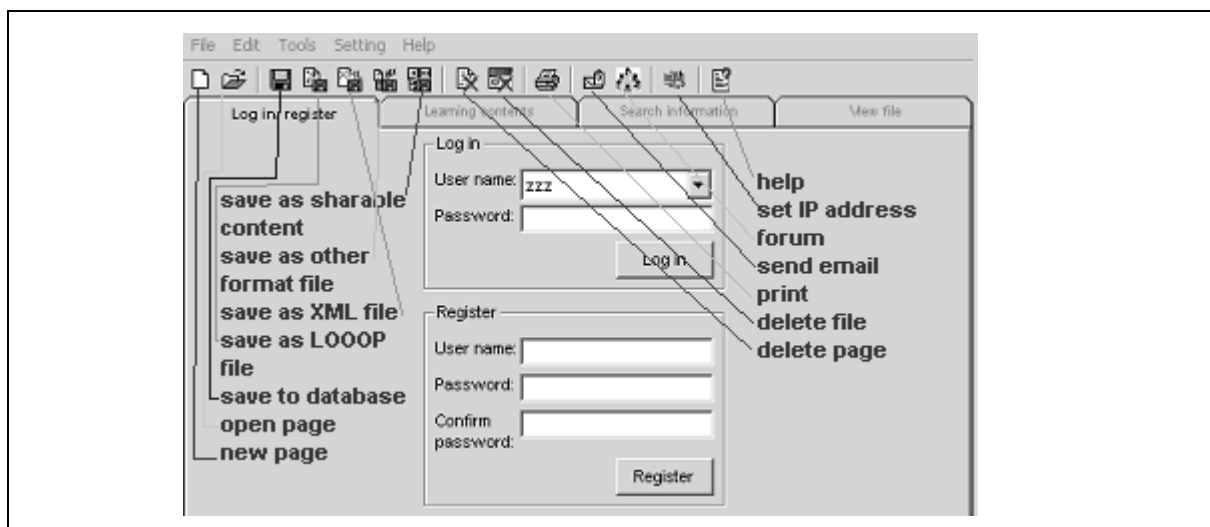


Figure 6.19 Login page

Figure 6.20 illustrated the learning contents page for the system. Based on the GUI template that had been designed in the earlier stage, the actual learning contents page is implemented. This page contained five parts, list of contents in treeview control, contents in DHTML control, editing tools in toolbar, user note in textbox control and extra file that associated with the content in treeview control. Besides, there is a list of useful tools provided in the system as shown in the figure 6.20 for content editing purposed. The tools provided included content navigation tool, editing tool, Internet search button, reader agent, highlight text and clear highlighted text button.



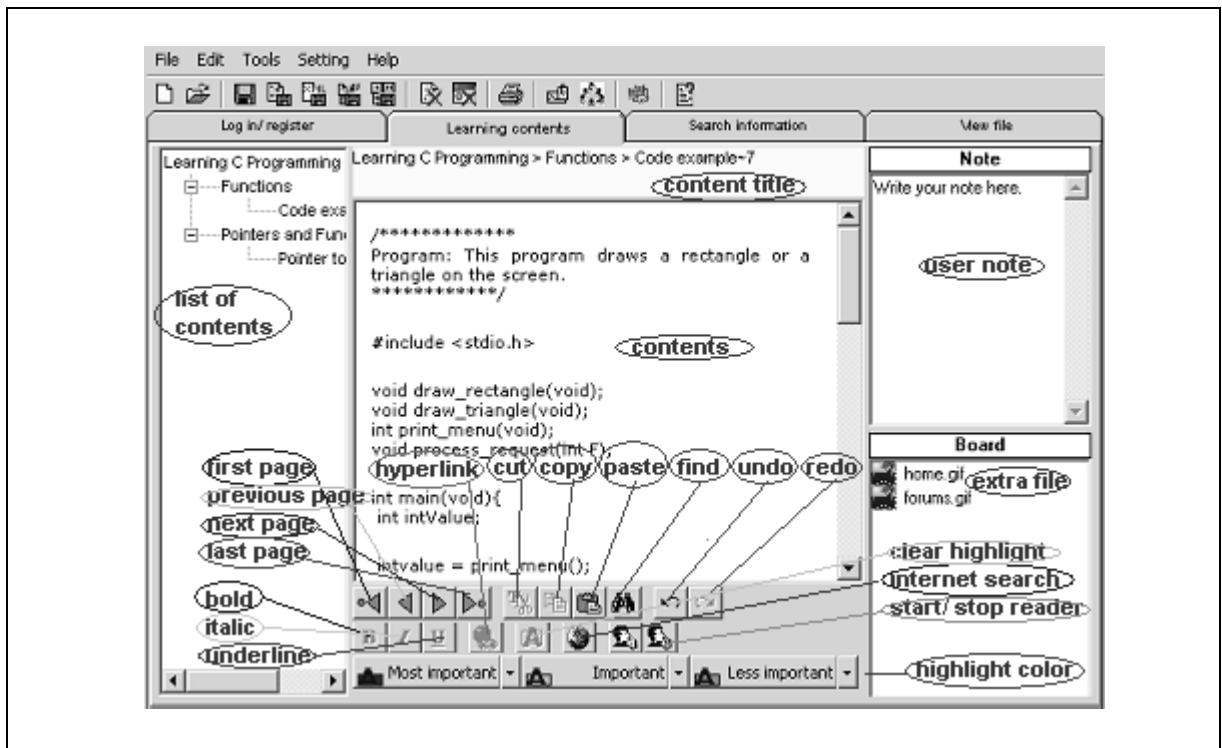


Figure 6.20 Learning contents page

Figure 6.21 illustrated the search information page for the system. This page has three main parts. The keywords input part is the place for user key in the keyword and search. In this part, there are an input keyword area in textbox control, two selections in checkbox control and a search button. User needs to select either search from contents or search from Internet or both when performing a search. The result list part is the part listed out all the records found from Internet and database in treeview control. The final part is the result page review in web browser control. This part is included with the two frames for file saving purposed and one frame for list of downloadable file. The two file saving purposed frames are located on top and bottom of the control, just as what had shown in figure 6.21.

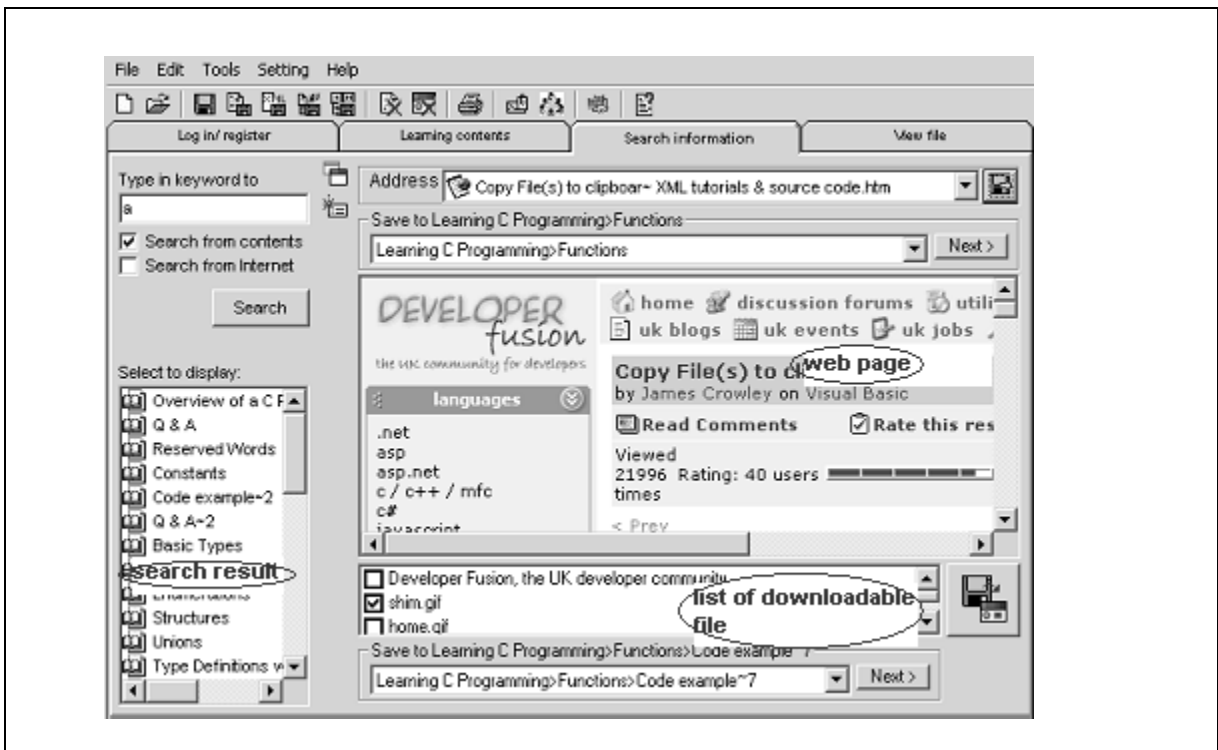


Figure 6.21 Search information page

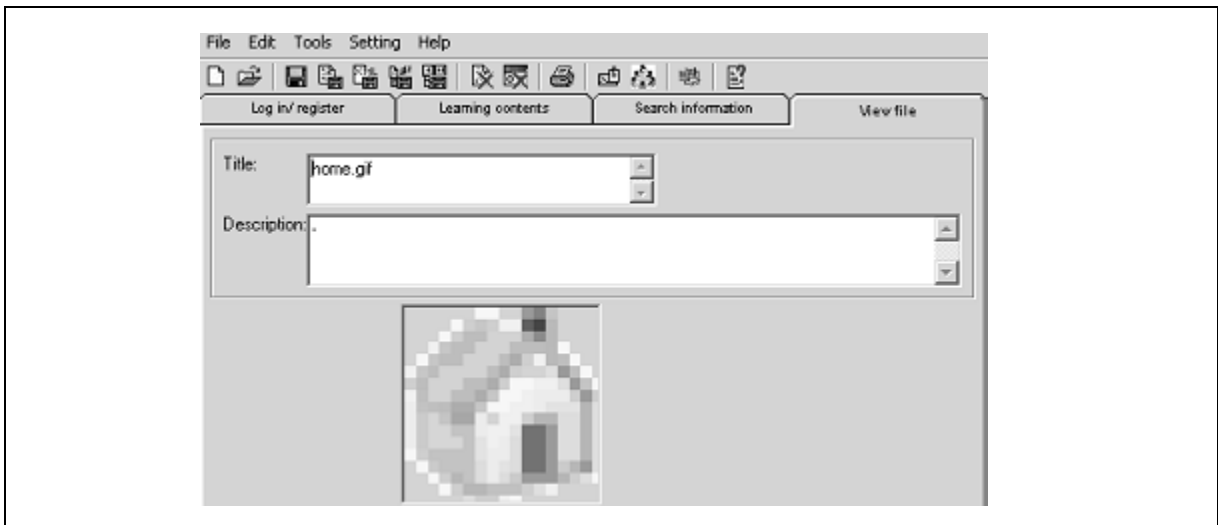


Figure 6.22 View file page

Figure 6.22 illustrated the view file page for the system. Same with what had been discussed on GUI template for view file page earlier, the actual GUI for view file page is divided into two parts. The upper part is the file information part. The file title and file

descriptions are displayed to user. User can change the file title and file description on each textbox. The lower part is for viewing and playing file purposed.

Figure 6.23 illustrated the forum window for the system. This window contains one chat screen in textbox control, one message input area in textbox control, one online user list in listbox control and two buttons. One button is for user sending file to another online user and another one button is for sending the message written in message input area.

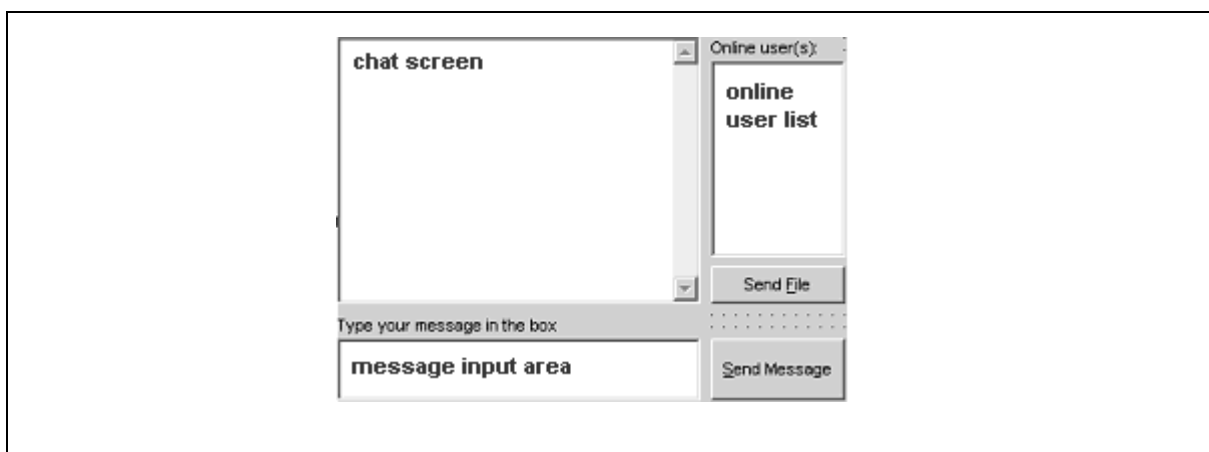


Figure 6.23 Forum window

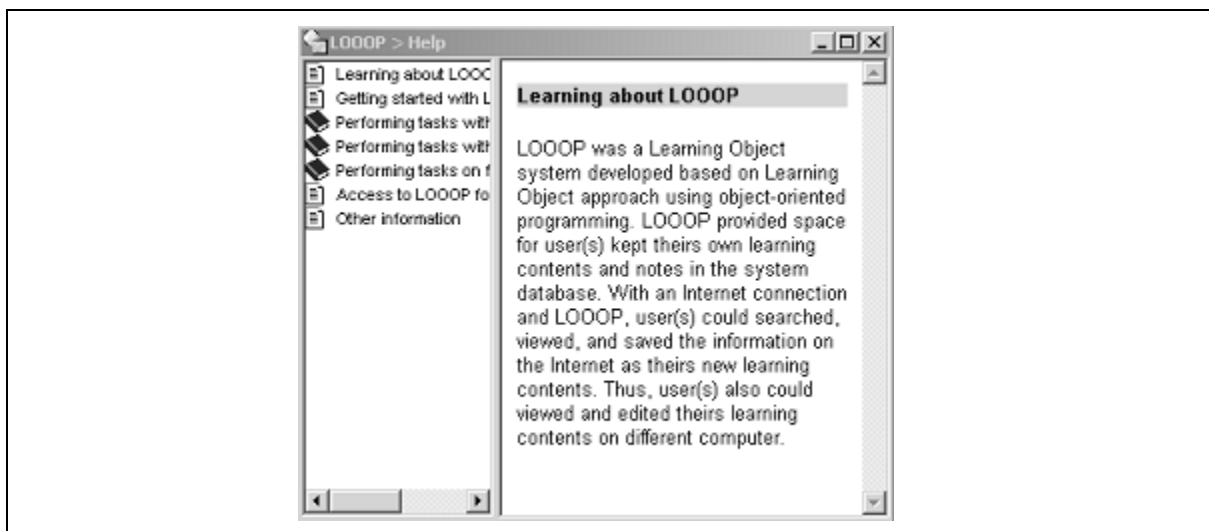


Figure 6.24 Help window

Figure 6.24 illustrated the help window for the system. This window contains one help topics list in treeview control and one web browser control to display the content of each help topic.

## **6.8 Summary**

Based on the analysis done on system analysis step, the system can design properly. First, the system architecture is drawn as the structure of the system. Second, the process flow is identified for clearly determining the flows of each process for the system. Third, the relationship between entities is identified and the data flow for each functions or features provided is drawn. Fourth, the database is drawn based on the data flow diagram. From the design, the GUI can be drawn out based on the template that is designed in the earlier stage of software development life cycle.