INCOME TAX PREPARATION SYSTEM

LEE HONG YIN

WEK98108

A graduation exercise submitted to the Faculty of Computer Science and Information Technology, University Malaya, in partial fulfillment of the requirement for the Bachelor of Computer Science Degree.

UNIVERSITY OF MALAYA

FACULTY OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY

INCOME TAX PREPARATION SYSTEM

LEE HONG YIN

WEK98108

LATIHAN ILMIAH II (WXES3182)

UNDER SUPERVISION OF

PROFESSOR MADYA RAJA NOOR AINON ZABARIAH RAJA ZAINAL ABIDIN

ABSTRACT

The current income tax assessment system whereby income tax is assessed on the income earned in the preceding year and is assessed by the Inland Revenue Board (Jabatan Hasil Dalam Negeri) will be changed to a Self Assessment System.

Self Assessment is the manner by which a taxpayer works out and pays his own income tax. It is not a new tax but a system whereby the taxpayer is given the responsibility to compute his own tax liability.

Self Assessment is a total process change from the previous formal assessment system. Under the formal system, taxpayers are required to declare their income in the Return Form, submit the Return Form to the Inland Revenue Board (IRB) and IRB will then raise the assessment. The notice of assessment is sent to the taxpayer and based on the tax raised in the Notice of Assessment, payment must be made accordingly.

Therefore this project will produced an online Income Tax Preparation System for the tax payers to ease their burden and duty to the country.

ACKNOWLEDGEMENTS

I would like to convey my special thanks to Puan Raja Noor Ainon, my supervisor at the Faculty of Computer Science & Information Technology, University Malaya, for her helpful guidance and comments at various stages of this project development.

I would also like to take this opportunity to thank Mr. Sellepan, the moderator for this project, for his suggestions and comments.

Not forgetting , my sister who help in my report and Mr. Ku Seng Lip for his support and help.

To my parents, lecturers, and friends who have supported me through the whole thing.

Table of Contents

Chapter 1
Introduction

INCOME TAX PREPARATION SYSTEM

1.1 Project Definition

The current income tax assessment system whereby income tax is assessed on the income earned in the preceding year and is assessed by the Inland Revenue Board (Jabatan Hasil Dalam Negeri) will be changed to a Self Assessment System.

Self Assessment is the manner by which a taxpayer works out and pays his own income tax. It is not a new tax but a system whereby the taxpayer is given the responsibility to compute his own tax liability.

Self Assessment is a total process change from the previous formal assessment system. Under the formal system, taxpayers are required to declare their income in the Return Form, submit the Return Form to the Inland Revenue Board (IRB) and IRB will then raise the assessment. The notice of assessment is sent to the taxpayer and based on the tax raised in the Notice of Assessment, payment must be made accordingly.

However, several problems have been noticed when the Self Assessment System be implemented later. The problems are stated below :-

- Taxpayers are having difficulty in calculating their payable tax
- Learning how to calculate income tax is not easy and also a complicated process.

- It will be a time consuming process.
- Inaccurate and wrong income tax calculation might happens.

The purpose of this project is to solve the current Self Assessment System problems faced by taxpayers. This project will design a system that assist them to prepare the return forms and furnish an estimate of the tax payable for a year of assessment and pay the tax by monthly payments.

1.2 Objective

The aim of the project is to develop an on-line Income Tax Preparation System using World Wide Web (WWW) to help and assist taxpayers.

To achieve this aim, objectives are set out as below :

- To simplify the calculation process for taxpayers
- To reduce processing error due to lower level of human intervention
- To have well-managed information
- To reduce the time needed to preform calculation
- To help taxpayers prepare the return forms

It is hoped that the Income Tax Preparation System will be able to assist taxpayers and ease their duty to the country.

1.3 Scope

This project will develop a system that help the taxpayers to estimate their payable tax and assist them to prepare the return form. The system will be developed mainly for individuals (married and single). It will also stored all the tax calculation rules in a database. This rules will be used by the system when it is performing tax calculation. The Income Tax Preparation System will be a web-based system which can be accessed by the taxpayers through the Internet or the World Wide Web.

Chapter 2
Literature Review

CHAPTER 2 : LITERATURE REVIEW

In the process of developing the required Income Tax Preparation System, the part of research is important. Research has been done to understand various concepts, which especially focus on the information and information system. Study of field such as Internet, World Wide Web (WWW), web-based application component, client / server concepts and architecture are also important.

2.1 What is Data?

Data is raw material that collected from findings and researchs. It is useless until it has been processed and transformed into useful material called information.

2.2 What is Information?

Information is data that have been put into a meaningful and useful context and communicated to a recipient who uses it to make decisions. Information involves the communication and reception of intelligence or knowlegde. It apprises and notifies, surprises and stimulates, reduces uncertainty, reveals additional alternatives or helps eliminate irrelevant or poor ones, and influences individuals and stimulates them into action.

2.3 What is Computer Network?

A computer network is a collection of computers that communicate with one another using a standardized protocol. Some network are public; anyone can utilize the network by paying a fee to a vendor who will provide access (or like students, by joining an organization that has already paid to have access). Other network are private. With these, only users who are preauthorized to connect to the network can gain access.

The most widely used public network is the Internet that will be discussed in the next section.

2.4 What is Internet?

The Internet is an international computer network that connects and joins computers in every area in the world. The Internat is a network formed by cooperative interconnection of computing networks.

In fact, the word "Internet" means interconnection and network. Wht this means is that many connecting networks, usually made up of differing kinds of computers and different technologies, are interfaced together so smoothly that the individual parts appear to be one network. This is accomplished by connecting networks using the same protocol which is TCP/IP or Transmission Control

Protocol / Internet Protocol. TCP/IP is a common set of rules that allow the variety of systems to communicate. However, computer on non-TCP/IP netwotks can still access the Internet through gateways that perform the necessary protocol translations and allow appropriate communication.

The Internet began in 1969 when the US Department of Defence (DoD) created an experimental computer network originally called the ARPANet. The ARPANet was designed to support military research and in particular, research about how to build computer networks that could withstand partial outages, meaning it would continue to function if one or more computer on the system were eliminated due to boom attack, etc.

There is no central authority, no governing body nor overall organizational scheme to the vast amounts of information available. The Internet transfers data in five different ways : gopher, telnet, FTP (File Transfer Protocol), HTTP (HyperText Transfer Protocol) and E-mail. Each computer that is connected to the Internet is provided a unique address which is an IP address.

2.5 **What is World Wide Web?**

The World Wide Web or WWW is an Internet navigator tool through which the Internet users can access the other front ends, navigators, information, services

and resources. One of the major problems with the existing Internet is that it was very unorganized and unconnected. It was a vast library that had no central index. It was a treasure of information, but unfortunately no way of accessing the information. The solution to this problem that emerged is known as World Wide Web or WWW.

Englishman Tim Berners-Lee invented the WWW in 1990 while working at CERN, the European Particle Physics Laboratory. At first, the Web was set up as a way for scientists to share informations with each other. It has since become a worldwide success because it makes it very easy for computer novices to browse through text, graphics and multimedia.

WWW is an information system based on hypertext, which offers a means of moving from document to document within a network of information. WWW uses the concept of a page for viewing information. Each page is actually a single text file written in something called HyperText Markup Language or HTML. This HTML file is retrieved form a remote computer, known as the HTTP server, by a WWW browser and is used to determine the appearance of that particular WWW page. An HTML document can contains pointers to other HTML documents, graphics, files, sounds and even description for buttons and other on-screen elements for displaying data. This interconnection of HTML documents on computers all over the internet, each containing pointers to other HTML documents on other computers on the Internet, is where the term "web" came from.

2.6 Web Browser

A web browser is a software program that acts as an interface between the user and the innerworkings of the Internet, or the World Wide Web. A browser is also referred to as a web client that acts in conjunction with a web server. The browser acts on behalf of the user by contacting a web server and requesting information and receiving information and then displaying it on a screen.

They are many different browsers. All perform the same basic function (transferring hypertext document) but many have specific features that are unique. Some commonly used browsers are Microsoft Internet Explorer, Nescape Navigator and NCSA Mosaic.

Browsers can be text-based or graphical. A graphical browers allows the user to see more of what the World Wide Web has to offer and can make the Internet more easier to use. A text-based browers only allows a user to see only text whereby graphics elements are not displayed.

2.7 Web Server

A web server is a software program running on a computer connected to the Internet. The term "web server" is also used sometimes to refer to the computer on which the software is running. More often, the computer is called a server and is

running more software than just web server software. Some examples of web servers are Microsoft Internet Information Server, Personal Web Server for MS FrontPage, Nescape Enterprise Server and Apache.

The purpose of a web server is to respond to requests for WWW files. When a computer user surf the WWW, he or she is actually sendings requests to web servers. The servers are sending back various files that are used to construct the web pages that the surfer requested.

When a web browsers try to access information stored in a database, web server acts as the client to the database server. The web server accepts the query from the browser and passes the qeury to the databae server. Moreover, the web server also formats the results into HTML, and sends the results back to the browser.

2.8 **Database Server**

The database server is very important in the Internet application development. It can be used to store and retrieve the information that is stored in a database. The database connects to the web server. The ability of web-based application to access database is the key to successfully corporate application development on the web.

2.9 Concepts of Client / Server

Client / server system allow many people to share centralized data. This requires a network of computers which are the clients, to be linked to a central computer, known as the server. Any kind of data can be stored in a database on the server. The database is accessed with a software application that runs on client computers.

The client is a process (program) that sends a message to a server process, requesting that the server perform a task (service). Client programs usually manage the user-interface portion of the application, validate data entered by user, dispatch requests to server programs and execute business logic. The client-based process is the front-end of the application that the user sees and interacts with. The client process contains solution-specific logic and provides the interface between the user and the rest of the application system. The client process also manages the local resources that the user interacts with such as monitor, keyboard and peripherals. One of the key elements of a client workstation is the graphical user interface or GUI. Normally a part of operating system, the window manager detects user actions, manages the windows on the display and displays the data in the windows.

A server process (program) fulfills the client request by performing the task requested. Server programs generally receive requests from client programs, execute database retrieval and updates and manage data integrity and dispatch responses to

client requests. Sometimes server programs execute common or complex business logic. The server-based process may run on another machine on the network. This server could be the host operating system or network file server; the server is then provided both file system services and application services. In some cases, another desktop machine provides the application services. The server process acts as a software engine that manages shared resources such as databases, printers, communication links or high powered-processors. The server process performs the back-end tasks that are common to similar application.

2.10 **Internet Standards and Characteristics**

Three standards emerged from all of these activity that are important to network database applications. The first is HTML itself. Since, today, amost every computer has one or more browsers that can process HTML, it makes sense for Internet-oriented database application to create forms and print reports in HTML format.

Another important standard for database applications is the syntax by which t6he document is referenced on the Internet. The Universal Resource Locator (URL) provides a standardized means to locate any document. A URL consists of three parts : a service name, a domain name and a path name with a file name. Thus, in the URL :

http://www.companyx.com/dir1/myfile.abc

http:// is the name of the service (in this case hypertext transfer protocol), www.companyx.com is the name of a domain and dir1/myfile.abc is the name of the path and file.

A third important standard that emerged from the Internet concerns the means by which URL files are to be processed. Not all files are HTML; some are graphics, sound, movie or animation. When a server sends a file to a client, it also sends a code indicating the file's type. The client interprets the code to determined which program can process the file. The system of codes used is called Multipurpose Internet Mail Code or MIME. The code was initially developed for mail, but today it is used more frequently in the processing of HTML documents.

Figure 2.1 lists several types of files, their normal file extensions and their MIME types. Three-letter file extensions are used on Windows and Macintosh systems, and four-letter extensions are used on Unix.

MIME Type	Name	File Extension
Text / HTML	HyperText Markup	.htm, .html
Text / plain	Text	.txt
Application / postscript	Postscript	.pdf, .ps, .ai
Image / gif	Graphics Interchange Format	.gif
Image / jpeg	Joint Photographic Experts Group	.jpg, .jpeg

Audio / basic	mu-law	.au
Audio / x-wav	Microsoft Wave File	.wav
Audio / basic	Macintosh Sound File	.snd
Video / quicktime	Macintosh Quick Time	.qt, .mov
Video / x-msvideo	Microsoft Video	.avi

Figure 2.1 Sample MIME Types

2.11 What is Intranet?

The term Intranet has several different interpretations. Most common the term means a private, local or wide-area network (LAN or WAN) that uses TCP/IP, HTML and related browser technology on client computers and web server technology on servers. Less commonly, the term is used to mean any private LAN or WAN that involves clients and servers.

Two characteristics of Intranets differentiate them from the Internet. First, Intranets are private. Either they are not connected to a public network at all, or they are connected to a public network via a firewall, which is a computer that serves as a security gateway. Firewall computers monitor the source and destination of traffic between the Intranet and the Internet and filter it. Some firewalls operate so as to allow only certain traffic through; others operate so as to prohibit certain traffic; still others operate in both modes.

Because intranets are private, security is less of a concern. This does not mean that security is not a concern at all but rather that less elaborate security measures need to be taken. In most cases, computers on an intranet are known and supported by organization that owns and maintains the intranet. Unauthorized activity and security problems are easily identified and managed than on a public network.

The second major differentiating characteristic of an intranet is speed. Most users connect to the Internet via a modem. Since today's modems operate in the range of 56 kbs (thousands of bits per second), anyone planning an Internet application needs to size the application's transmission requirements accordingly. Large files cannot be transferred in a reasonable amount of time.

On the other hand, the transmission speed of a local area network can be in the range of 100000 kbs. Hence, files downloaded on an intranet can be more than 1000 to 10000 times larger than those downloaded from the Internet.

Because of the speed difference, large bitmaps, sound files and animation can be included in intranet applications. More important in the database world, large query responses can be transmitted to client computers. In addition, because of the increased speed, it is possible to download large program files to the client computer. This means that more of the application processing can be conducted on the client in an intranet than can be conducted on a client in the Internet. However, the Internet is the most potential of the two because it can reach millions of computer that are unreachable through the intranet.

2.12 **HTML 1.0 and 2.0**

HTML 1.0 refers to the original set of markup tags. HTML 1.0 is so limited that a browser that restricted HTML documents to HTML 1.0 would be a museum piece.

HTML 2.0 includes a more generous set of markup tags than HTML 1.0; in particular, it allows markup tags that define user input fields. HTML 2.0 defines the de facto common core of markup tags. You can create a relatively sophisticated Web page with HTML 2.0 markup tags.

2.13 HTML 3.0 and NHTML

HTML 3.0, still in the process of standardization, adds additional markup tags to those defined in HTML 2.0, such as tags to define tables, figures, and mathematical equations. HTML 3.0 expands some tags to include more functionality, such as centering text or images in the browser, and adding background colors and images.

NHTML, a nickname for Netscape's extension of HTML 2.0, is another set of markup tags that goes beyond those defined in HTML 2.0. Netscape, like other developers of cutting edge Web browsers, is trying to influence the development of the HTML 3.0 standard, and has developed extensions of its own. At the same time, Netscape is making an effort to conform to the evolving HTML 3.0 specification. Furthermore, Netscape continues to support markup tags that the draft HTML 3.0 specification has declared obsolete.

Netscape's browser, Netscape Navigator, is not precisely HTML 3.0-compliant. The best way to find out whether Netscape Navigator supports a

particular markup tag is to get the latest version and try a document containing the tag.

2.14 Why NHTML isn't proper SGML

Formally, a Web page is the third part of an SGML DTD, and as such, should conform to the SGML DTD specification. A few features of NHTML do not conform to the rules of the SGML DTD specification. If browsers actually treated a Web page as the third part of an SGML DTD, this would be a problem.

However, browsers typically accept a certain hard-coded level of HTML-typically HTML 2.0 with some HTML 3.0 extensions and some NHTML extensions-and ignore markup tags that they do not recognize. Where this nonconformity does present a problem is in writing tools that validate Web pages. These tools typically use an SGML parser, and they require a page to be part of a properly conforming SGML DTD for the level of HTML they check.

So far, no one has come up with a validation tool that recognizes and validates the syntax of JavaScript.

2.15 LiveScript

Netscape began working on a scripting language called LiveScript, which quickly evolved into what is now JavaScript. Although JavaScript and Java are not

the same thing, Netscape intends JavaScript to tie into Java; hence the name change. Netscape and Sun Microsystems (the developers of Java) are working closely on the development of the two languages. There are few other major differences between LiveScript and JavaScript, the biggest being that LiveScript was case-insensitive and JavaScript is case-sensitive.

2.16 A Brief History of SQL

The history of SQL begins in an IBM laboratory in San Jose, California, where SQL was developed in the late 1970s. The initials stand for Structured Query Language, and the language itself is often referred to as "sequel." It was originally developed for IBM's DB2 product (a relational database management system, or RDBMS, that can still be bought today for various platforms and environments). In fact, SQL makes an RDBMS possible. SQL is a nonprocedural language, in contrast to the procedural or third-generation languages (3GLs) such as COBOL and C that had been created up to that time.

The characteristic that differentiates a DBMS from an RDBMS is that the RDBMS provides a set-oriented database language. For most RDBMSs, this set-oriented database language is SQL. *Set oriented* means that SQL processes sets of data in groups.

Two standards organizations, the American National Standards Institute (ANSI) and the International Standards Organization (ISO), currently promote SQL

standards to industry. Although these standard-making bodies prepare standards for database system designers to follow, all database products differ from the ANSI standard to some degree. In addition, most systems provide some proprietary extensions to SQL that extend the language into a true procedural language.

2.17 A Brief History of Databases

Database systems store information in every conceivable business environment. From large tracking databases such as airline reservation systems to a child's baseball card collection, database systems store and distribute the data that we depend on.

Until the last few years, large database systems could be run only on large mainframe computers. These machines have traditionally been expensive to design, purchase, and maintain. However, today's generation of powerful, inexpensive workstation computers enables programmers to design software that maintains and distributes data quickly and inexpensively.

2.18 Dr. Codd's 12 Rules for a Relational Database Model

The most popular data storage model is the relational database, which grew from the seminal paper "A Relational Model of Data for Large Shared Data Banks," written by Dr. E. F. Codd in 1970. SQL evolved to service the concepts of the

relational database model. Dr. Codd defined 13 rules, oddly enough referred to as Codd's 12 Rules, for the relational model:

0. A relational DBMS must be able to manage databases entirely through its relational capabilities.

1. Information rule-- All information in a relational database (including table and column names) is represented explicitly as values in tables.

2. Guaranteed access--Every value in a relational database is guaranteed to be accessible by using a combination of the table name, primary key value, and column name.

3. Systematic null value support--The DBMS provides systematic support for the treatment of null values (unknown or inapplicable data), distinct from default values, and independent of any domain.

4. Active, online relational catalog--The description of the database and its contents is represented at the logical level as tables and can therefore be queried using the database language.

5. Comprehensive data sublanguage--At least one supported language must have a well-defined syntax and be comprehensive. It must support data definition, manipulation, integrity rules, authorization, and transactions.

6. View updating rule--All views that are theoretically updatable can be updated through the system.

7. Set-level insertion, update, and deletion--The DBMS supports not only set-level retrievals but also set-level inserts, updates, and deletes.

8. Physical data independence--Application programs and ad hoc programs are

logically unaffected when physical access methods or storage structures are altered.

9. Logical data independence--Application programs and ad hoc programs are logically unaffected, to the extent possible, when changes are made to the table structures.

10. Integrity independence--The database language must be capable of defining integrity rules. They must be stored in the online catalog, and they cannot be bypassed.

11. Distribution independence--Application programs and ad hoc requests are logically unaffected when data is first distributed or when it is redistributed.

12. Nonsubversion--It must not be possible to bypass the integrity rules defined through the database language by using lower-level languages.

2.19 **Today's Database Landscape**

Computing technology has made a permanent change in the ways businesses work around the world. Information that was at one time stored in warehouses full of filing cabinets can now be accessed instantaneously at the click of a mouse button. Orders placed by customers in foreign countries can now be instantly processed on the floor of a manufacturing facility. Although 20 years ago much of this information had been transported onto corporate mainframe databases, offices still operated in a batch-processing environment. If a query needed to be performed,

someone notified the management information systems (MIS) department; the requested data was delivered as soon as possible (though often not soon enough).

In addition to the development of the relational database model, two technologies led to the rapid growth of what are now called client/server database systems. The first important technology was the personal computer. Inexpensive, easy-to-use applications such as Lotus 1-2-3 and Word Perfect enabled employees (and home computer users) to create documents and manage data quickly and accurately. Users became accustomed to continually upgrading systems because the rate of change was so rapid, even as the price of the more advanced systems continued to fall.

The second important technology was the local area network (LAN) and its integration into offices across the world. Although users were accustomed to terminal connections to a corporate mainframe, now word processing files could be stored locally within an office and accessed from any computer attached to the network. After the Apple Macintosh introduced a friendly graphical user interface, computers were not only inexpensive and powerful but also easy to use. In addition, they could be accessed from remote sites, and large amounts of data could be off-loaded to departmental data servers.

During this time of rapid change and advancement, a new type of system appeared. Called *client/server development* because processing is split between client computers and a database server, this new breed of application was a radical

change from mainframe-based application programming. Among the many advantages of this type of architecture are

- Reduced maintenance costs.
- Reduced network load (processing occurs on database server or client computer)
- Multiple operating systems that can interoperate as long as they share a common network protocol
- Improved data integrity owing to centralized data location

In *Implementing Client/Server Computing,* Bernard H. Boar defines client/server computing as follows:

> *Client/server computing* is a processing model in which a single application is partitioned between multiple processors (front-end and back-end) and the processors cooperate (transparent to the end user) to complete the processing as a single unified task. Implementing Client/Server Computing A client/server bond product ties the processors together to provide a single system image (illusion). Shareable resources are positioned as requestor clients that access authorized services. The architecture is endlessly recursive; in turn, servers can become clients and request services of other servers on the network, and so on and so on.

This type of application development requires an entirely new set of programming skills. User interface programming is now written for graphical user interfaces, whether it be MS Windows, IBM OS/2, Apple Macintosh, or the UNIX X-Window system. Using SQL and a network connection, the application can interface to a database residing on a remote server. The increased power of personal

computer hardware enables critical database information to be stored on a relatively inexpensive standalone server. In addition, this server can be replaced later with little or no change to the client applications.

2.15 Early Implementations

Oracle Corporation released the first commercial RDBMS that used SQL. Although the original versions were developed for VAX/VMS systems, Oracle was one of the first vendors to release a DOS version of its RDBMS. (Oracle is now available on more than 70 platforms.) In the mid-1980s Sybase released its RDBMS, SQL Server. With client libraries for database access, support for stored procedures (discussed on Day 14, "Dynamic Uses of SQL"), and interoperability with various networks, SQL Server became a successful product, particularly in client/server environments.

One of the strongest points for both of these SQL Server powerful database systems is their scalability across platforms. C language code (combined with SQL) written for Oracle on a PC is virtually identical to its counterpart written for an Oracle database running on a VAX system.

2.16 SQL and Client/Server Application Development

The common thread that runs throughout client/server application development is the use client/server computing of SQL and relational databases.

Also, using this database technology in a single-user business application positions the application for future growth.

2.17 An Overview of SQL

SQL is the de facto standard language used to manipulate and retrieve data from these relational databases. SQL enables a programmer or database administrator to do the following:

- Modify a database's structure
- Change system security settings
- Add user permissions on databases or tables
- Query a database for information
- Update the contents of a database

2.18 Open Database Connectivity (ODBC)

ODBC is a functional library designed to provide a common Application Programming Interface (API) to underlying database systems. It communicates with the database through a library driver, just as Windows communicates with a printer via a printer driver. Depending on the database being used, a networking driver may be required to connect to a remote database.

The unique feature of ODBC (as compared to the Oracle or Sybase libraries) is that none of its functions are database-vendor specific. For instance, you can use

the same code to perform queries against a Microsoft Access table or an Informix database with little or no modification. Once again, it should be noted that most vendors add some proprietary extensions to the SQL standard, such as Microsoft's and Sybase's Transact-SQL and Oracle's PL/SQL.

ODBC has developed into a standard adopted into many products, including Visual Basic, Visual C++, FoxPro, Borland Delphi, and PowerBuilder. As always, application developers need to weigh the benefit of using the emerging ODBC standard, which enables you to design code without regard for a specific database, versus the speed gained by using a database specific function library. In other words, using ODBC will be more portable but slower than using the Oracle7 or Sybase libraries.

2.19 SQL in Application Programming

SQL was originally made an ANSI standard in 1986. The ANSI 1989 standard (often called SQL-89) defines three types of interfacing to SQL within an application program:

- Module Language-- Uses procedures within programs. These procedures can be called by the application program and can return values to the program via parameter passing.
- Embedded SQL--Uses SQL statements embedded with actual program code. This method often requires the use of a precompiler to process the SQL

statements. The standard defines statements for Pascal, FORTRAN, COBOL, and PL/1.

- Direct Invocation--Left up to the implementor.

Before the concept of dynamic SQL evolved, embedded SQL was the most popular way to use SQL within a program. Embedded SQL, which is still used, uses *static* SQL--meaning that the SQL statement is compiled into the application and cannot be changed at runtime. The principle is much the same as a compiler versus an interpreter. The performance for this type of SQL is good; however, it is not flexible--and cannot always meet the needs of today's changing business environments.

The ANSI 1992 standard (SQL-92) extended the language and became an international standard. It defines three levels of SQL compliance: entry, intermediate, and full. The new features introduced include the following:

- Connections to databases
- Scrollable cursors
- Dynamic SQL
- Outer joins

Dynamic SQL allows you to prepare the SQL statement at runtime. Although the performance for this type of SQL is not as good as that of embedded SQL, it provides the application developer (and user) with a great degree of flexibility. A

call-level interface, such as ODBC or Sybase's DB-Library, is an example of dynamic SQL.

Call-level interfaces should not be a new concept to application programmers. When using ODBC, for instance, you simply fill a variable with your SQL statement and call the function to send the SQL statement to the database. Errors or results can be returned to the program through the use of other function calls designed for those purposes. Results are returned through a process known as the *binding of variables*.

Chapter 3
System Analysis

CHAPTER 3 : SYSTEM ANALYSIS

This phase involves all the activities necessary to determine the system requirements. A requirement is a feature of the system or a description of something the system is capable of doing in order to fulfil the system's purpose. Requirements are divided into two categories : functional and nonfunctional requirements.

There are a variety of techniques to determine what the users really want. Among the information gathering techniques are questionnaires and observation of the environment in which the system is to be installed.

3.1 Questionnaire

Questionnaire had been distributed to potential users of this system to gain their opinions about the implementation of online income tax preparation system.

There are several reason to use a questionnaire as tool to gather information :

- The people that needed to be questioned are widely dispersed.
- It is an exploratory study.
- To gauge overall opinion before the systems project is given any specific direction.
- Any problem with the current system that needed to be identified and addressed.

3.1.1 Questionnaire Design

The combination of open-ended questions and closed questions are chosen in the questionnaire. Open-ended question is chosen to gather all possible responses to the questions from the respondents. Closed question is chosen to limit the response and eventually ease the analysis and interpretation of their responses. The design of the questionnaire is attached in Appendix.

3.1.2 Respondents

The respondents of this questionnaire is the potential income tax payers that is either single or married. 50 questionnaires were distributed to the potential system users.

3.2 Questionnaire Results

The results of the questionnaire is displayed and interpreted in the next section.

3.2.1 How Frequently Respondents Surf the Internet?

	Total	Percentages (%)
Always	8	16
Usually	20	40
Sometimes	15	30

Seldom	5	10
Never	2	4

Table 3.1 How Frequent Respondents Surf The Internet

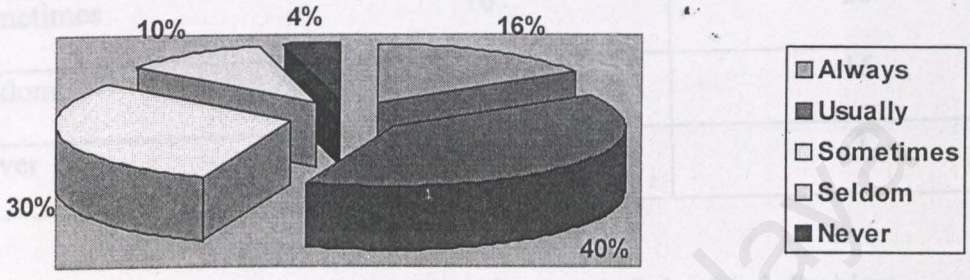


Figure 3.1 Percentage of Internet Surfing

The chart shows that majority of the respondents always and usually surf the Internet. Figure 3.1 shows that 16% of the respondents always surf the Internet and 40% of them usually surf it. However there are about 4% respondents that do not surf the Internet at all.This may be because they do not have access to the Internet.

Because of the fact that most of the respondents are always exposed to the Internet, there is no problem to put the income tax preparation system online.

3.2.2 **Internet as Tool to Solve Problem**

	Total	Percentages (%)
Always	5	10
Usually	25	50
Sometimes	10	20
Seldom	8	16
Never	2	4

Table 3.2 How Frequent Respondents Use The Internet to Solve Their Problem

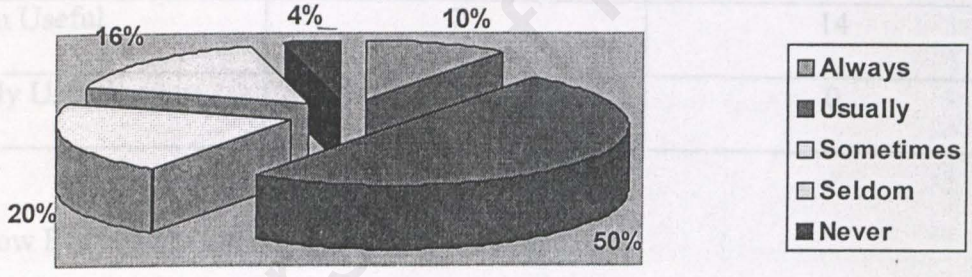


Figure 3.2 Percentage showing Internet as Problem-Solving Tool

The pie chart in Figure 3.2 clearly show that most of the respondent use the Internet as a tool to solve their problem. This is a good sign that there will be requests for the online income tax preparation system as they needed to solve their income tax problem as quickly as possible.

3.2.3 Does Internet Is An Effective Tool to Solve Problem ?

	Total	Percentages (%)
Extremely Useful	9	18
Very Useful	24	48
Useful	10	20
Often Useful	7	14
Rarely Useful	0	0

Table 3.3 How Effective is Internet for Solving Problem

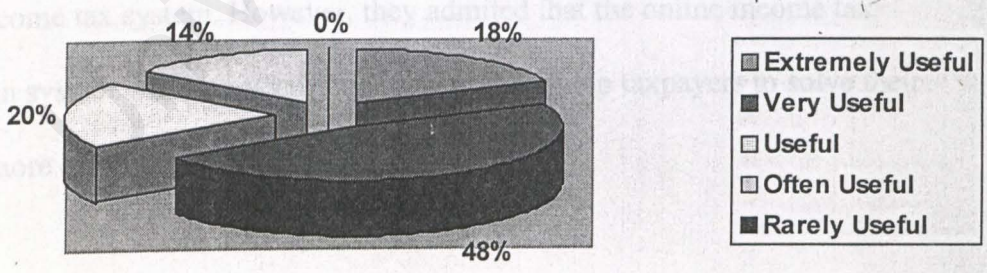


Figure 3.3 Percentage Effectiveness of Internet

Table 3.3 and Figure 3.3 shows that the responents admit that the Internet is a very effective to solve any problems that they had. These figures indicated that the Internet will be a primary tool for the income tax payers to solve their problem.

3.2.4 **Current Income Tax System**

Below are some of the comments relating the current income tax system that are given by the respondents :

- Taxpayers are having difficulty in calculating their payable tax
- Learning how to calculate income tax is not easy and also a complicated process.
- It will be a time consuming process.
- Inaccurate and wrong income tax calculation might happens.
- Taxpayers needed to follow many procedures.
- Too many forms to fill in and they do not know what to do with each form.

There are still some respondents that are satisfied with the current income tax system because they have a very solid understanding of the innerworking of the current income tax system. However, they admited that the online income tax preparation system will be a very good idea as it can help taxpayers to solve their problem more quickly and effectively.

3.2.5 **Opinions on the Online Income Tax Preparation System**

	Total	Percentage (%)
Yes	45	90
No	5	10

Table 3.4 Opinions on an Online System

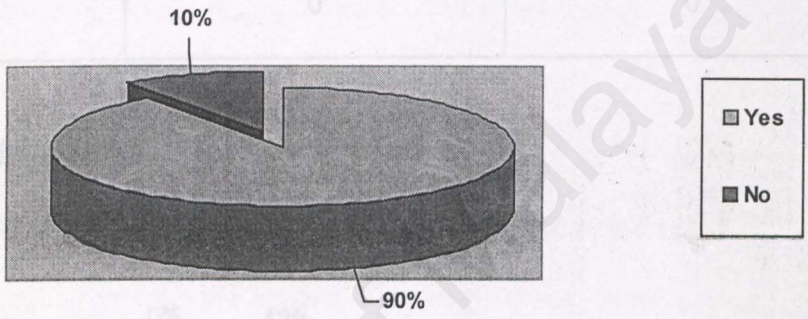


Figure 3.4 Respondents' views on Online Income Tax System

The statistics show that 90% of the respondents agree to have an online income tax preparation system. The reasons given by the respondents are as below :

- Save time and very convenient.
- It is easier, faster, more efficient and simplify the process.
- No more long queue.

3.2.6 Online Income Tax Preparation System

	Total	Percentages (%)
Extremely Useful	6	12
Very Useful	20	40
Useful	15	30
Often Useful	9	18
Rarely Useful	0	0

Table 3.5 How Necessity is Online Income Tax System

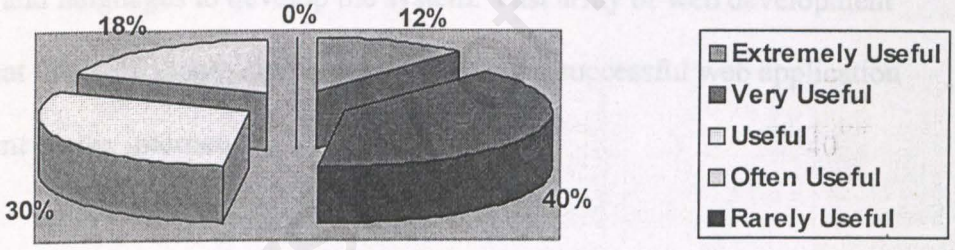


Figure 3.5 Importance of an Online Income Tax System

A total of 82% of the respondents strongly support the online income tax system. They find that the online system is useful, very useful and extremely useful.

Below are some of the features, which are suggested by the respondents to be included in the online income tax system :

- Guides and FAQ
- Help on what forms or documents that needed to verify their payable tax
- View their last payable tax
- Print out of payable tax

3.3 Consideration on the programming technologies and languages

3.3.1 Introduction

The main task in this section is to identify suitable programming technologies and languages to develop the system. Vast array of web development tools show that there are many approaches in building a successful web application for deployment on the Internet.

Analysis was done to choose the most suitable programming technologies and languages. Consideration on the availability of the related development tools was also an important factor. The ideal solution for an Internet application is one that is easy to build and deploy, and also possesses the scalability of traditional client/server systems. A good Internet application should allow for easy integration with the latest emerging technologies.

3.3.2 **Active Server Pages (ASP)**

Active Server Pages is an open, compile-free application environment in which you can combine HTML, scripts and reusable ActiveX components to create dyamic and powerful web-based business solutions. Active Server Pages enables server side scripting for IIS with native support for both VBScript and Jscript.

When the scripts run on the server rather than on the client, the web server does all the work involved in generating the HyperText Markup Language (HTML) pages that can send to browsers.

Active Server Pages is a feature of and can be used with the following web servers :

- Microsoft Internet Information Server on Windows NT Server
- Microsoft Peer Web Services on Windows NT Workstation
- Microsoft Personal Web Server on Windows 95/98

ASP can create a web interface to a database which is the best part of ASP. This involves using ActiveX Data Object (ADO). SQL statements execute inside ASP pages using ADO, this is easily done with any kind of database that can be accessed via an ODBC link.

3.3.3 Scripting Languages

3.3.3.1 VBScript

VBScript has been designed to make it easier to develop client-side Web applications that run on the Web browser. Before VBScript is invented, people were discovering the virtues of providing dynamic information to users browsing a Web site. CGI applications were typically used to create Web pages that displayed dynamic information. Although this worked well in some cases, it did not work well for some people. The development of a CGI application typically meant learning a programming language such as C or C++, compiling the CGI application, transferring it to the CGI directory of a Web server, and testing the CGI application for bugs. Even the slightest change to the application meant recompiling the entire application and repeating the process of copying the application to a CGI directory and testing the application for bugs. To solve this problem, Web scripting languages such as JavaScript and VBScript were developed to aid in the development of client-side and server-side CGI applications.

3.3.3.2 Benefits of Using VBScript

There are many benefits to using VBScript. VBScript is a powerful, lightweight, easy-to-use, freely available, cross-platform, cross-language scripting language for the Internet. It is designed to leverage the skills and investments of millions of Visual Basic programmers to the Internet. This is great news for all the millions of Visual Basic programmers who want to leverage their Visual Basic skills to the

Internet. VBScript is a powerful, lightweight, easy-to-use, freely available, cross-platform, and cross-language scripting language.

- VBScript is powerful. Various capabilities of VBScript can be used to develop richly interactive Web pages that respond to user input in an intelligent manner. For example, when a user submits a form, a VBScript subroutine can be triggered to verify that the form is properly filled in with valid values. In the case of a server-side CGI application, VBScript can be used to process data submitted by users with the aid of ActiveX controls specially designed for Microsoft Active Server Pages.
- VBScript is lightweight. VBScript code is lightweight, fast, and has been optimized to be transmitted via the Internet. Because VBScript code is lightweight, it can be quickly transmitted to users browsing a Website--even via relatively slow Plain Old Telephone Service (POTS) links to the Internet.
- VBScript is easy to use. Compared to scripting languages such as JavaScript, VBScript is easier to use because it is based on the easy-to-learn BASIC (Beginner's All Purpose Symbolic Instruction Code) language.
- VBScript is freely available. VBScript is a freely available scripting language. Microsoft has made VBScript freely available to software vendors so they can add scripting capabilities to their applications with the aid of VBScript.
- VBScript is cross platform. VBScript will be functioning on UNIX as well as Macintosh computers in addition to Windows 95 and Windows NT computers.

- VBScript is cross language. VBScript supports any language (such as C++ and Java, for example) that enables objects to be compiled as ActiveX controls.

VBScript can be used to leverage the skills of millions of Visual basic programmers to the Internet. VBScript can be used to easily create active and intelligent Web pages. VBScript is a subset of Microsoft Visual Basic and is upwardly compatible with Visual Basic for Applications (VBA). VBA is shipped with Microsoft Office applications to make it easier for developers to build custom solutions using various components of Microsoft Office. The capability to provide scripting, automation, and customization capabilities for Web browsers and Web servers is a major feature of VBScript.

3.3.3.1 **JavaScript**

JavaScript is a lightweight object-based scripting language created by Netscape Communications Corporation for developing Internet applications. JavaScript is lightweight in that there isn't a great deal to learn and you can be productive with it very quickly, in contrast to much more complex languages such as Java. As a scripting language, JavaScript is meant to tell an application what to do. Unlike languages used to create applications, it cannot do anything without the application.

You can develop server applications or client applications with JavaScript. In this book, the term "server" refers to the computer where your Web pages reside. The term "client" refers to the browser application that loads and displays your Web pages. You can embed JavaScript statements in Web pages, which are written in HTML (Hypertext Markup Language). JavaScript is an extension to HTML that lets you create more sophisticated Web pages than you ever could with HTML alone.

3.3.4 **Common Gateway Interface (CGI)**

The Common Gateway Interface (CGI) provides a mechanism for a program on the server to interact with the client's browser. You can use any language to write CGI programs, and CGI programs may be interpreted (PERL scripts, for instance) or compiled (C or C++). One popular use of CGI is in hit counters-programs that modify the page to show how many times that page has been visited. Another popular use of CGI is in form handling, where a program on the server reads the data from the user input fields and does some work based on that data.

JavaScript, which does its work in the client's browser, cannot entirely replace CGI. For instance, a hit counter has to update a file on the server so it can remember how many times the page has been visited by all visitors. That's a little difficult for JavaScript, but a JavaScript Web page can keep track of how many times a given visitor has visited the page. So can CGI, but only if given an endless supply of disk space on the server.

JavaScript can do a lot of the same things CGI can do, and it can often do them much more efficiently. For example, JavaScript can do form validation more efficiently than CGI. When a non-JavaScript page has user input fields, it sends all the field values to a CGI server application. The CGI application then has to figure out whether the data in each field makes sense before doing something with the data. A JavaScript page, however, can validate the data entered before it is sent to the server. If the data is invalid, JavaScript can block transmission to the server. Because all of this work is performed on the client side, JavaScript does not waste bandwidth transmitting bad data and then receiving an error page from the server.

JavaScript can also replace some of the animation and page-reloading functionality of CGI. To perform animation and page-reloading, CGI provides mechanisms called "server push" and "client pull." With server push, the Web page server maintains a connection between the client (the browser) and server. Server push restricts the number of simultaneous connections the Web page server can maintain-a popular page using server push will frequently reward potential visits with a "sorry, not now, try later" message. Client pull, on the other hand, involves the client frequently re-establishing its connection to the server, artificially adding to the traffic at the server. You can use JavaScript to create dynamic documents that would have required either server push or client pull in CGI, but that involve no additional traffic or long, drawn-out connections between the client and the server.

JavaScript is independent of the server platform, the hardware and software that your server uses. CGI has to be written in a language that your server platform supports. No single language is supported by all server platforms.

Finally, not all Web space providers allow Web pages to use CGI. CGI requires that the program be executed on the server, but some Web space providers are nervous about the possible side effects of badly written or maliciously written CGI programs being executed on their machines. Some providers only allow the use of a limited set of applications. Many providers do not support server push CGI. JavaScript running on the client browser is perfectly safe to the server, and affords you, the creator of the JavaScript document, much greater flexibility in how your document interacts with the reader.

3.3.5 **Java**

Many people confuse JavaScript with Java, which is a programming language developed by Sun Microsystems, Inc. Each has its own Usenet newsgroup, yet people frequently post questions about Java to the JavaScript newsgroup, and vice versa.

Java is a programming language and JavaScript is a scripting language. Java programs are compiled on the server. You can write stand-alone programs in Java. Scripts written in JavaScript are interpreted by the browser. You cannot write stand-alone programs in JavaScript-you need a browser to interpret JavaScript.

Java is object-oriented. It employs classes and inheritance. It provides encapsulation of data. JavaScript is object-based. There are no classes. There is no inheritance. Data within objects is readily accessible. Java is compiled into "applets" that are accessed from HTML pages. JavaScript is embedded in HTML.

Java requires that data types be strongly typed (if a function expects one of its arguments to be a number, the function will not accept a character string). JavaScript is loosely typed. JavaScript has numbers, character strings, and Booleans (logical yes/no, true/false, on/off data) and freely interchanges them.

Java can be used to create very powerful applications. JavaScript scripts cannot really do all the neat things that Java applets can. On the other hand, it is much more difficult to write programs in Java than it is to write scripts in JavaScript.

3.3.6 **Microsoft ActiveX**

Microsoft's ActiveX technology represents a huge advance in the capabilities of Internet Explorer. ActiveX has relaxed the OLE control requirements to practically nothing. Although previous OLE controls (such as the .OCX files shipped with Visual Basic) contained a lot of baggage inappropriate to use on the Internet, new ActiveX controls (conforming to the redesigned control requirements specification) can be a lot more streamlined, facilitating the easier production of high-quality dynamic content for HTML documents. It's easier to create ActiveX controls than previous OLE controls, but it still requires a great degree of

programming knowledge. Casual HTML authors, though, will no doubt be able to take advantage of a multitude of freely available ActiveX controls in time.

Internet Explorer 3.0 allows for the use of ActiveX controls, active scripts (such as VBScript), and active documents. ActiveX can be used to encapsulate practically any application or applet for use within HTML documents.

3.4 Hardware and Software Requirements

3.4.1 Server Hardware Requirements

The server computer requirements are :

1. A server with at least Pentium 133 Mhz processor
2. 1 GB of free hard disk space
3. At least 32 MB RAM
4. Network Interface Card (NIC) and network connection with recommended bandwidth at 10 Mbps or more

3.4.2 Server Software Requirement

Software	Description
Windows NT Server 4.0	Network operating system
Internet Information Server 4.0	Web-server service
Microsoft SQL Server 7.0	Database connectivity interface driver
Microsoft Internet Explorer 5.0	Web Browser
Active Server Pages	Server scripting engine

ActiveX Data Object	Access medium to SQL Server

3.4.3 **Client Hardware Requirements**

- At least 16 MB memory
- Pentium 100 Mhz processor

3.4.4 **Client Software Requirements**

Microsoft Internet Explorer 4.0 or above.

Chapter 4
System Design

CHAPTER 4 : SYSTEM DESIGN

System Design is a process to convert the conceptual ideas from requirement specification in System Analysis into more technical specification.

A design specification describes the features of the system, the components or elements of the system and their appearance to users.

4.1 System Structure Chart

The Income Tax Preparation System is divided into 2 major component : User and Administrator. Each of the two component is further divided into many modules.

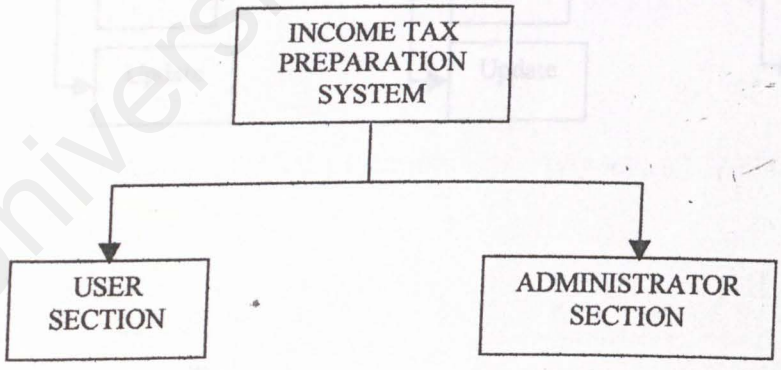


Figure 4.1 - Structure Chart for Income Tax Preparation System

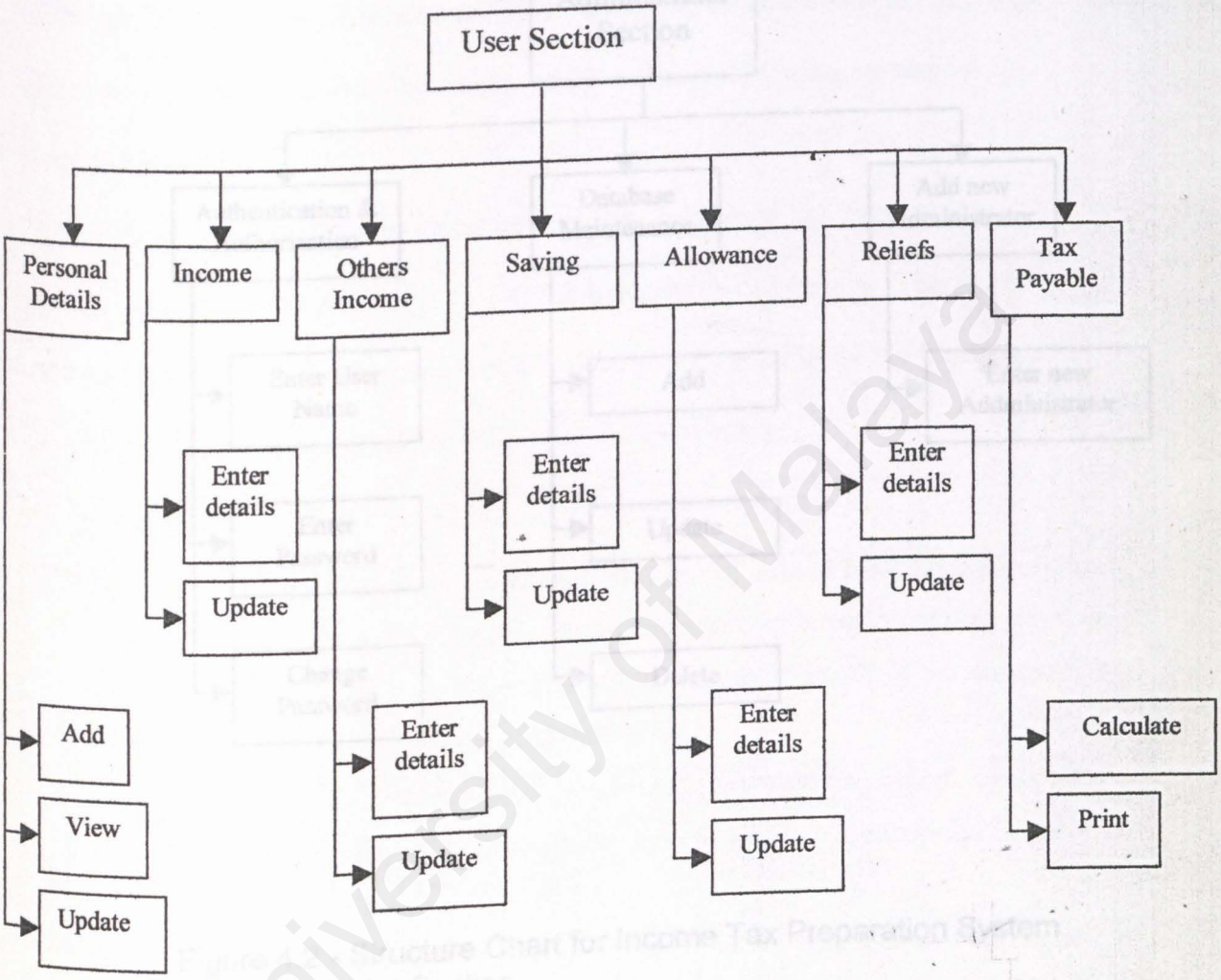


Figure 4.2 - Structure Chart for Income Tax Preparation System - Tax Payers Section

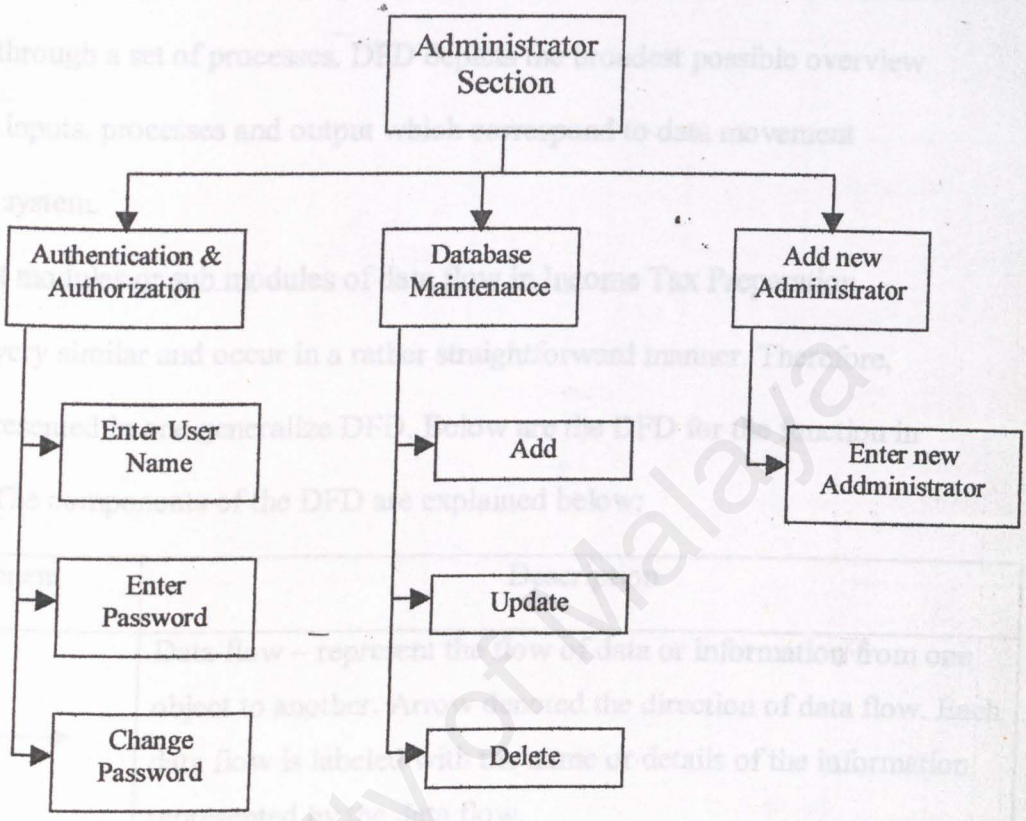
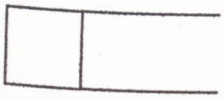
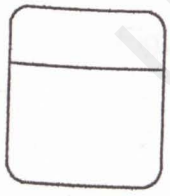
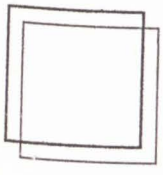


Figure 4.2 - Structure Chart for Income Tax Preparation System - Administrator Section

4.2 Data Flow Diagram (DFD)

Data Flow Diagram is the graphic model of the flow, use and transformation of the data through a set of processes. DFD depicts the broadest possible overview of a system inputs, processes and output which correspond to data movement through the system.

Most modules or sub modules of data flow in Income Tax Preparation System are very similar and occur in a rather straightforward manner. Therefore, they are represented in one generalize DFD. Below are the DFD for the function in the system. The components of the DFD are explained below:

Component	Description
	Data flow – represent the flow of data or information from one object to another. Arrow denoted the direction of data flow. Each data flow is labeled with the name or details of the information represented by the data flow.
	Data store – hold data for a time within the system. Comprise two sections : Identifier information and Description of the data stored.
	Process – transform the input data to output data. Represented by a rectangle shape. Comprise three sections : Top section contains the identifier information, Center section contains a description of the process and Lower section contains the physical or computer program information
	Entity – any object in the real world.

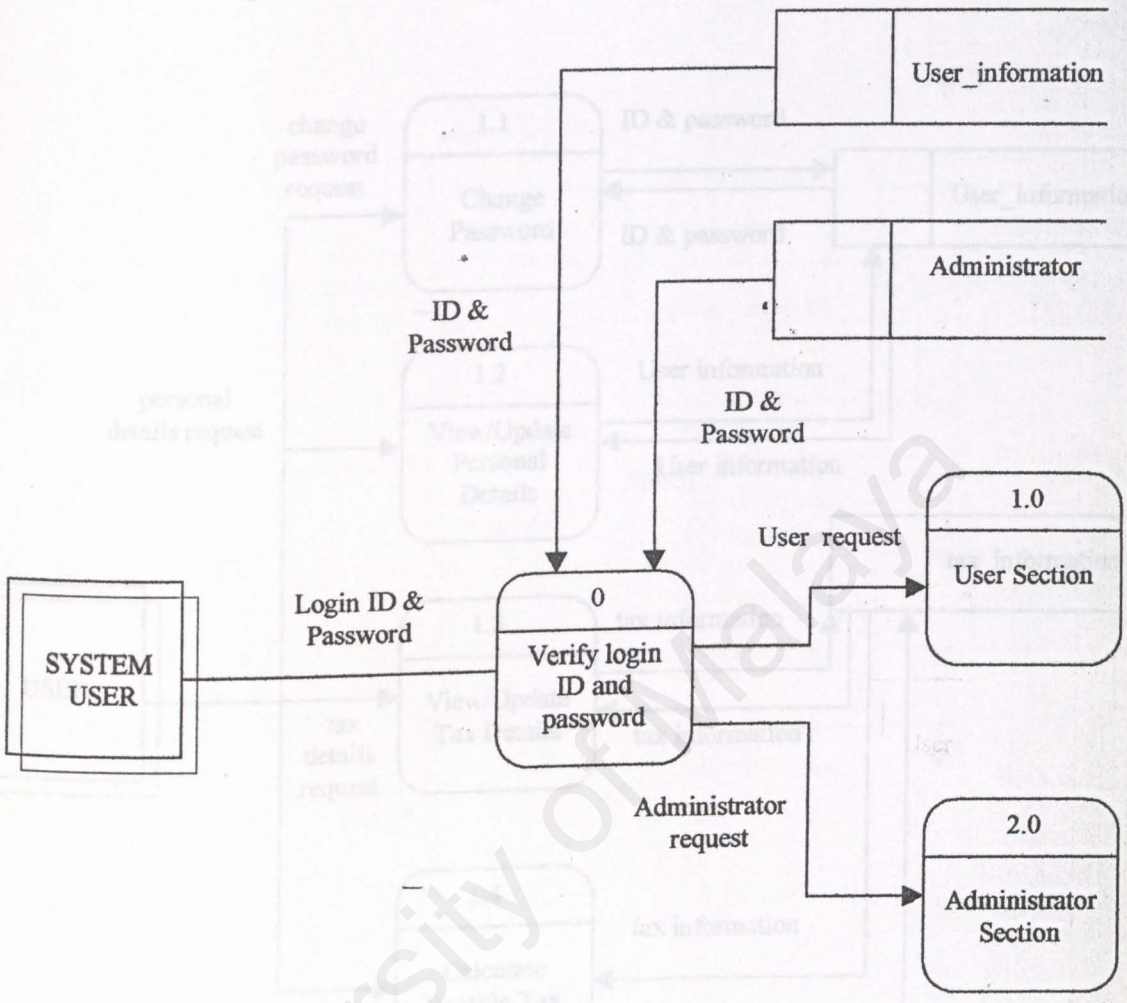


Figure 4.2 – Data Flow Diagram for System Overview

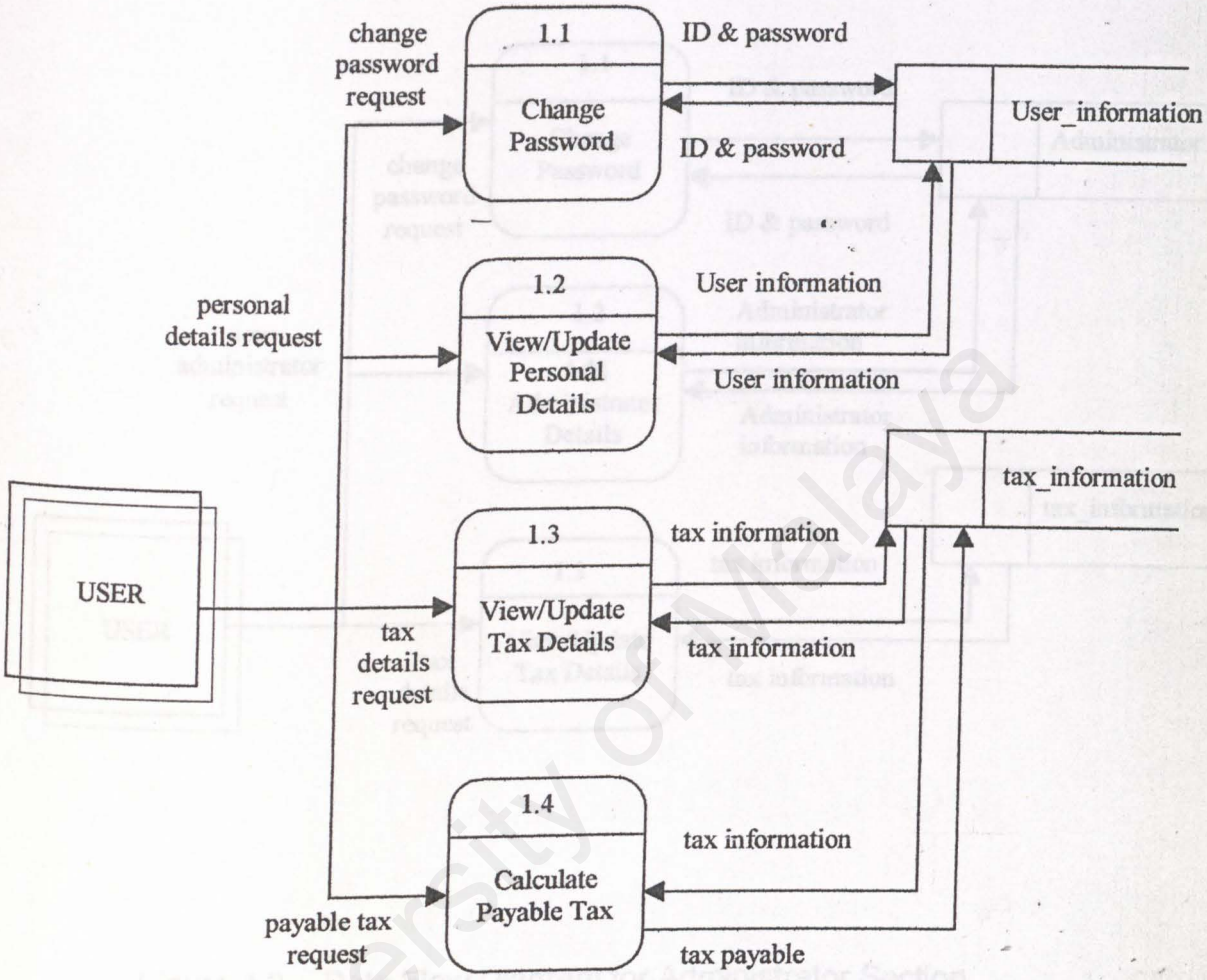


Figure 4.2 – Data Flow Diagram for Tax Payers Section

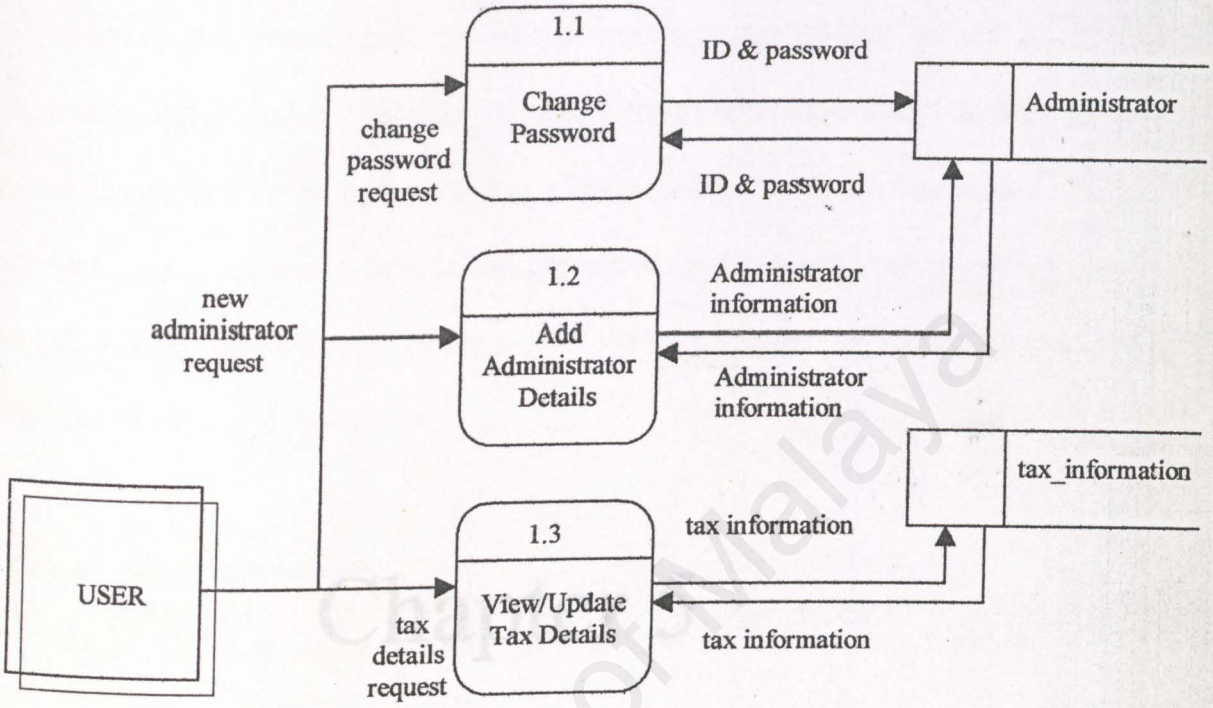


Figure 4.2 – Data Flow Diagram for Administrator Section

Chapter 5
System Implementation and Testing

CHAPTER 5 : SYSTEM IMPLEMENTATION AND TESTING

System implement is a process that converts the system requirement and designs into program codes. This phase at times involves some modification to the previous design. System testing verifies that a system solves the problem as defined by the requirement document. Income Tax Preparation is developed using top-down approach, which involves building the high-level software modules, which are refined further into function and procedures.

5.1 Development Environment

The hardware and software used have been considered carefully to facilitate the system's development.

5.1.1 Hardware Requirements

The following hare specification have been used to develop the Income Tax Preparation System :-

- Pentium II 166 Mhz CPU or above
- 32 MB RAM
- 4.3 GB hard disk
- Other standard desktop components

5.1.2 Software Requirements

In the early development's stages, software such as Microsoft Word 97, Microsoft Excel 97 and Visual Professional 5.0 are used to draw the tables, ER diagram and system structure chart as well as to capture other system requirement of Income Tax Preparation System.

The following Table 5.1 shows a list of software used during the development of the system.

Software	Module	Description
Microsoft Windows 2000	System requirement	Operating system
Internet Information Server 5.0	System requirement	Web Server host
ODBC 32-bit Driver	System requirement	Connect database to the web server
Microsoft SQL Server 7.0	System requirement	Database Server
Microsoft Access 2000	System development	Design the database
Microsoft Visual InterDev 6.0	System development	Coding the web, ASP and HTML
Microsoft Visual Basic 6.0	System development	Create new ActiveX component
Internet Explorer 5.0	System development	Viewing the system
Microsoft Image Composer 98	Interface Design	Image design
Paint Shop Pro 5.0	Interface Design	Image design

Table 5.1 - Software For Development

5.2 Database Development

The database for Income Tax Preparation System is created using SQL Server Enterprise Manager, which is Microsoft SQL Server 7.0 upsizing wizard. However, creating and modifying tables and their relationship using Microsoft Access 2000 is easier than using Microsoft SQL Server 7.0.

5.3 Program Coding

5.3.1 Methology

Income Tax Preparation System is developed using a modular approach. Each module is developed separately and later integrated into a fully functional system. For each module, it is further refined into fountains and procedures. By using a modular approach, future modification and enhancement will become easier.

5.3.2 Web Page Development

Income Tax Preparation System is a web application , which makes use of the Internet browser. As such, it is coded using HTML. Basically, for client side scripting, HTML, VbScript and JavaScript are used. HTML is used to present the web pages in interactive and attractive form. Vbscript and Javascript are used mainly for form validation and animation.

For server-side scripting, ASP technology and VbScript are used. ASP eases database retrieval and manipulation. VBScript is used for server-side validation. The web server processes ASP files before presenting to the browser. Thus, users will not

be able to view the ASP codes. The web server will interpret all these codes into HTML codes whenever user request the ASP files.

Microsoft Visual Interdev 6.0 provides a rapid development environment for building ASP pages, extensive database tools for connecting a web site to any database via Open Database Connectivity (ODBC) and a multitude of wizards to create simple database web application. With different colors in codes using InterDev development, it is easier to track error and do coding.

Graphics and animation in Income Tax Preparation System are created using Paint Shop Pro 5.0, Microsoft Image Composer and Ulead Cool 3D.

5.4 Testing

Although testing is tedious, it is essential series of that help assure the quality of the eventual system. It is far less disruptive to test beforehand than to have poorly tested system fail after installation. Testing is accomplished on subsystem or program modules as work progresses.

For each module in Income Tax Preparation System, they are tested separatedly and then integrated together. After integration, the system as whole is tested again. Each module contains functions and procedures, which must be checked and tested properly. The strategies used for testing in Income Tax Preparation System are unit testing, integration and system testing.

5.4.1 Unit Testing

Unit testing is done to uncover errors in each module. In Income Tax Preparation System, each module is coded is coded and tested separatedly. Function and procedures in each module are examined appropriately. If they are found to be error-free after manual examination, they are compiled and run with test data to search for other errors.

For Income Tax Preparation System, unit testing is done concurrently with the development phase. Each of the function are reviewed and checked separately. Then the sub-module is tested to ensure it functions as desired.

5.4.2 Integration Testing

When each individual program modules are working correctly and meet the established objectives, the modules are combined into working systems. The integration needs to be planned and coordinated in a structured way in order to capture the errors more easily.

For Income Tax Preparation System, a bottom-up approach has been used. Each module at the lowest level of the system's hierarchy is tested. This approach is repeated until all modules have been tested.

5.4.3 System Testing

System testing verifies that a system solves the problem as defined by the requirement document. After the integration test, function testing is done on the system to check that it performs its functional requirement as specified in the requirement.

Once the functions work as specified, performance test is done to compare the integrated modules with the non-functional system requirement. These requirements include security, accuracy, speed and reliability.

Database testing is also done on Income Tax Preparation System to test the accuracy and integrity os the data stored by the server. Transaction posted by Income Tax Preparation System are examined to ensure that data are properly stored, updated and retrieved.

Chapter 6
System Evaluation and Conclusion

CHAPTER 6 : SYSTEM EVALUATION AND CONCLUSION

6.1 Problem and Solution

Due to the time constraint and complexity of web programming, some technical problem were encounted. Solutions have been sought from project supervisor, friends, seniors, reference books and the Internet.

6.1.1 Problems and solution during system studies and analysis

Some problems have been encountered during the system studies and analysis phase. It should be noted that initial lack of knowledge in the development of web application has indeed been a great hindrance.

6.1.1.1 Difficulty choosing methology, programming language and tools

There are many tools and possible solutions available to develop a web-based database system. Choosing a suitable technology and tool proves to be a critical process as each tools has its strength and weakness. In order to choose a suitable programming language, studies were carried out to understand the strengths and weaknesses of the programming language.

6.1.1.2 Difficulty in determining the project scope

Like many other technologies, the area of income tax is very wide. If study further, there are a lot of possible cases regarding income tax. In order to determine

the scope for Income Tax Preparation System, advice has been sought from project supervisor and functional requirement.

6.1.2 Problems and solution during project implementation and testing

The problems faced during initial project studies and analyses were not as critical as compared to the problems faced during implementation and testing period. Listed below are problems encountered during implementation and testing.

6.1.2.1 Inexperience in web-based programming

As there is no prior knowledge in developing application in a web-based, a lot of studies have to be done. New programming languages such as HTML, VBScript, JavaScript and ASP need to be study within a short time span.

6.1.2.2 Difficulty in designing a suitable user interface

Developing a suitable and user friendly interface is a difficult task because HTML provides limited drag-drop features for placing controls. Hence a lot of time is spent in arranging the controls and graphics on the web page. The user interface became difficult more because this system is develop to all kind of categories.

6.2 Future Enchancements

As income tax self assessment for indivdual will be introduce in year 2004, future enhancement that make the system more complete should be carried out.

6.2.1 **Expanding to all categories taxpayers**

There are a lot of categories of taxpayers like individual, company and so on. Expanding this system to all categories will allow other types of taxpayers to get the benefits of this system. This will make the system more complete as an income tax preparation system.

6.2.2 **Online form processing**

It is recommended that the return form which currently need manually return to Lembaga Hasil Dalam Negeri, is processing online. This features will make thing easier for the taxpayers as they can solve their tax duty completely with the Income Tax Preparation System.

6.3 **Conclusion**

Income Tax Preparation System as a fully functional system has achieved its objectives of an income tax self assessment system. System of this kind is expected to be the trend of the future.

It is hoped that the Income Tax Preparation System will continue to benefit and contribute in the country's development.

Bibliography

Bibliography

[1] Neil Randoll and Dennis Jones, 1996, Special Edition Using Microsoft FrontPage, Que Corporation, United States of America.

[2] Ted Coombs, Jason Coombs, Donald Brewer, 1996, ActiveX SourceBook : Build an ActiveX-Based Web Site, Wiley Computer Publishing, Canada.

[3] David Siegel, 1996, Creating Killer Web Sites, Hayden Books, United States of America.

[4] Kenneth E.Kendall, Julie E.Kendall, 1999, System Analysis and Design, Prentice Hall International, Inc.

[5] Susan Weinschenk, Pamela Jamar, Sarah C. Yeo, 1997, GUI Design Essentials, Wiley Computer Publishing, Canada

[6] David M. Kroenke, 1998, Database Processing : Fundalmentals, Design and Implementation, Prentice Hall, Inc, United States of America

[7] Keith Boeckner, P. Charles Brown, 1997, Oxford English for Computing, Oxford English Press, Hong Kong

[8] Web Database Development,
http://www.zdnet.com/pcmag/features/webdatabase/

[9] Jabatan Hasil Dalam Negeri,
http://www.hasil-net.org/
Last Update : 4 August 2000

[10] What is Active Server Pages ?
http://asp-help.com/
Microsoft Active Server Pages

[11] Microsoft Jscript – What is Jscript?
http://asp-help.com/getstarted/
Microoft Corporation

Appendix

Questionnaire

Title : Income Tax Preparation System

Occupation : _____

Sex : Male / Female

Status : Married / Single / Others

Please circle the number that best suggests your opinion about each topic.

1) How often do you surf the Internet?

Never	Seldom	Sometimes	Usually	Always
1	2	3	4	5

2) How often do you use Internet to do your work?

Never	Seldom	Sometimes	Usually	Always
1	2	3	4	5

3) Using the Internet as a tool to solve problem is

Extremely Useful	Very Useful	Useful	Often Useful	Rarely Useful
1	2	3	4	5

4) Please state any comments relating to the current income tax system

5) Do you want your income tax to be calculated over the Internet using an online system?

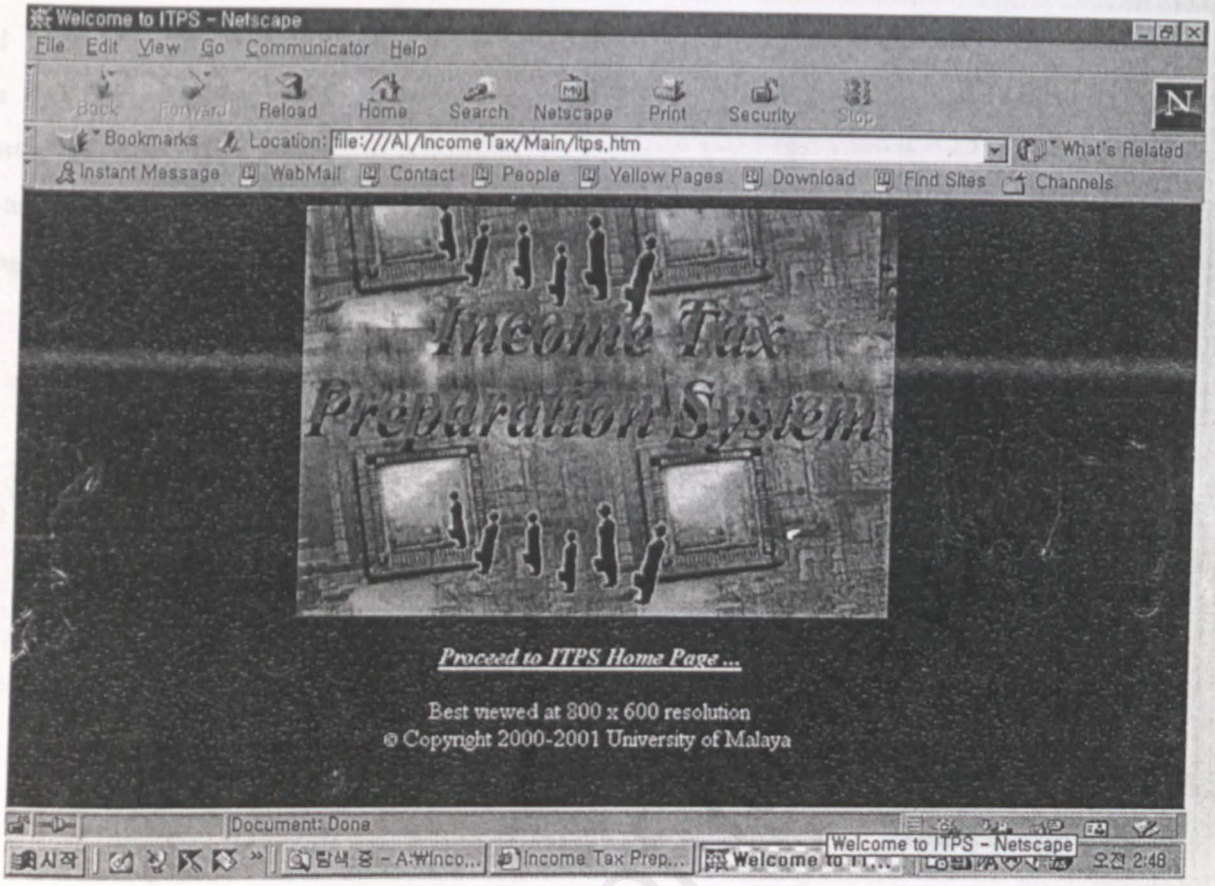
Yes	No
1	2

Please state your reason :

6) An online Income Tax Preparation System is

Extremely Useful	Very Useful	Useful	Often Useful	Rarely Useful
1	2	3	4	5

7) Any special features that you would like to include in the income tax preparation system ? (beside the common processes)



Front page of Income Tax Preparation System

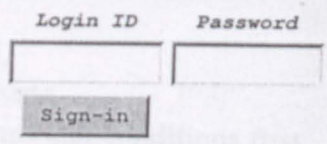
Tutorial

- Welcome to tutorial session.
- This session create for beginner, teaching them how to use Online Income Tax Preparation System.
- We provide everything about ITPS including Online Registration, Log-in, User's Category Selection, how to enter data and how to read result.
- We have another 13 pages to go, so be patient.
- Go to page by page by clicking next icon in right bottom of every page you will learn how friendly this system is.

Back Next

Home

- In order to use ITPS, user have to register as a member before they can access to our system.
- We need login ID and password because we can retrieve user record each time they login and keep their record private.
- In our home, you can see a green box as shown in *Figure 1*.
- If you already sign in as a member before, just key-in your ***Login ID*** and ***Password*** then hit the ***Sign-in button***.
- If you are non-member and you want to use ITPS, sign up as a new member by click on the word ***Sign up here!*** (as shown in *Figure 1*) or click on the icon in *Figure 2*, you will see the registration form.



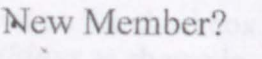


Figure 1

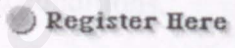


Figure 2

Back Next

Registration

- As usual when you get the registration form, fill it.
- If you face any problem while filling the form go to user manual's ***Problem in Registration*** as in left black box.
- Please remember your login ID and password.
- Your records enter will be keep secret and private.
- After finish filling the form, read the ITPS terms and conditions first before you submit the form.
- If you agree with the terms and conditions, click on the checkbox ***I have read and agree with the terms and conditions*** as shown in *Figure 3*.
- Then proceed with the registration by clicking on the ***Click here to Register*** button (as in *Figure 4*).

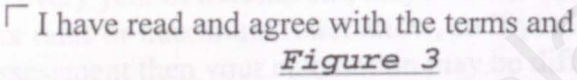


Figure 3

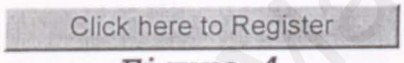


Figure 4

Back Next

User's category selection

- After you login or register as a new member you will get this page (ask for your category) before proceed with the tax calculator.
- Just answer three simple question (the answer is in the dropdown list) then your work will become easy because ITPS will filter out base on your category.
- If you select single for marital status then you will get only one column for your input.
- If you select no child, then your input will not have anything about child relief.
- The third question is more importance, it is which year of assessment do you want to process.
- In every year of assessment, maybe some changes will be made in the tax rates or maximum relief allow, so if you select wrong year of assessment then your calculation may be difference.

Back Next

Menu bar

- You will see this menu bar as in *Figure 5* in next session.
- This menu bar has three categories **Menu**, **User's Input** and **Result**.
- In **Menu**, if you need any help then click on the ***User Manual*** and if you want to change your category (describe in page 4) then click on the ***Change Category***.
- **User's Input** part are all the input from user.
- And **Result** are the calculation based on the user's input.
- You can go to any pages by clicking on the word in menu bar.

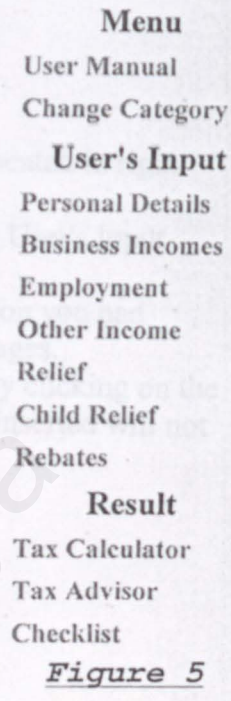


Figure 5

Back Next

Save and Continue Button

- The Save and Continue button (as in *Figure 6*) are located in right bottom of User's Input pages.
- After you has modified or inserted some value in the User's Input click on this button and continue to next page.
- If the button is press then you will save the information you had modified or inserted just now and proceed to other pages.
- If you didn't press this button and go to other pages by clicking on the menu bar, then the information you had modified or inserted will not be save in the database.

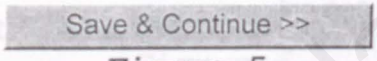


Figure 5

Input Type - Text Field

- Text field allow user to enter the value of the user in the form of number, amount or a string.
- If the text field request user to enter amount, then user are not allow to enter any alphabet.
- Some of the text field only displaying value, that field cannot be modified.
- For example in *Figure 6* below show aggregate income of user and this amount cannot be change by user because the aggregate income is calculate by system.

AGGREGATE INCOME 50,000.00

Figure 6

Input Type - Checkbox

- What user need to do for this type of input request is click to the box if relevent.
- The values of the input are TRUE or FALSE.
- If user checked the box then the value of that checkbox are true.
- For example in *Figure 7* below, check if only you are the disabled individual.

Disabled individual: ☐ *self* 0.00

Figure 7

Input Type - Dropdown List

- Users just need to select the relevant value from the dropdown list.
- The dropdown list will list all the possible value.
- For example in *Figure 8* below, if your date of birth is on 18 May 1977, then just select 18 from day, May from month and 1997 from year.

Figure 8

Back Next

Input Type - Button

- In this system most of the button are submit button.
- After you finish the page just click to the button to submit then go to next page.
- For example in *Figure 9* below, when you click to the button you will save the record you have modified to the database and you will go to the next page.

Figure 9

Result - Tax Calculator

- In the Tax Calculator you can view your chargeable income, total income tax, tax rebate and total tax charged for both combined and separate assessment.
- Beside that the system also calculate for you both the answer for child relief claim by either husband or wife.
- This session just for you to view the answer for the three possible cases then in next session (Tax Advisor) the system will suggest to you which type is more suitable.

Back Next

Result - Tax Advisor

- The Tax Advisor will suggest to you which type of assessment to use, child relief claim by husband or wife (separate assessment), and purchase personal computer rebate claim by husband or wife.
- User can also view different total amount of total tax charged between different type of assessment.
- Then the calculator will calculate user can save how much of the cost.
- If the tax charge for wife is less then purchasing PC rebate and you choose the purchasing PC rebate for wife, then Tax Advisor will advise to you choose the rebate for husband.

Back Next

Result - Checklist

- The checklist is to ensure user enclose the require documents, receipts or statements.
- This checklist is made up based on the user inputs.
- User can download this page or print-out this page and then later check whether they enclose such documents before return the form B to LHDN.

Back Next

Logout

- Finally, if a user finish their job or he/she wants to logout, they can click on the Logout word on the right top corner of the pages.
- After the user logout, it will return to home.
- If the user want to login again he/she has to key-in their Login ID and Password again.

Back Next