Student Record Retrieval System Using knowledge Sharing

Ibrahim A.S.Muhamadi, M.A Zaidan, A.A Zaidan, B.B Zaidan

Faculty of Computer Science & Information Technology
University Of Malaya
Kuala Lumpure, Malaysia

Abstract
Web development to become a global knowledge web development , it has taken numerous steps to improve its information systems, strengthen internally and externally focused knowledge-sharing activities, and foster broader global knowledge-sharing initiatives, all in support of enhancing the web development and its partners' and clients' access to and sharing of ideas .Most of universities have websites ,these websites have links which its content of information of a student such as examination results which is never functional and does not show or provide all the data required . Unfortunately, what is practiced now is not efficient and not modern and some universities like (FSKTM) for University of Malay hang papers on the notice board or attached them with nails or clips, one paper over another, some flopped, or some partially hidden and not in order but in a chaos way, is ridiculous. This system was developed due to the non-existence of a standard website that keeps a record for student's .for the above mentioned reason, the task for this paper became difficult. A new system of web development using knowledges sharing is presented. The proposed system aim to keep all students' recoreds or data, to distribute results to students online. A new system, Student Record Retrieval system (SRRS) has been developed for two main functions, firstly the student can access the system to see his recoreds and data, this process can be executed by login with ID and password. Secondly a staff (admin or lecturer) can access as well by login with ID and password. The proposed system has been developed using ASP.Net.

Keyword
Student Record Retrieval System (SRRS), Knowledge Sharing, web development.

I. KNOWLEDGE SHARING

As background to an assessment of the web development knowledge-sharing activities, this paper presents an exploration of the literature on the factors that can affect knowledge sharing success [1],[2]. Knowledge management involves the panoply of procedures and techniques used to get the most from an organization's tacit and codified know-how [3]. While defined in many different ways, knowledge management generally refers to how organizations create, retain, and share knowledge [3],[4]. The study of knowledge sharing, which is the means by which an organization obtains access to its own and other organizations' knowledge, has emerged as a key research area from a broad and deep field of study on technology transfer and innovation, and more recently from the field of strategic management. Increasingly, knowledge-sharing research has moved to an organizational learning perspective. Indeed, experience and research suggest that successful knowledge sharing involves extended learning processes rather than simple communication processes, as ideas related to development and innovation need to be made locally applicable with the adaptation being done by the 'incumbent firms' [4], [5] or 'the local doers of development' [3] for the ideas to be successfully implemented.

II. MOTIVATION

Normal paper or sheet in the wall was one of the oldest ways to show people the event, before the information technology light the way to have the benefit of the technologies, nowadays even in the Underdeveloped countries they use a high standard of technology in their daily lives. Now big companies want to make life easy, easy access and more security to the information. FSKTM is the faculty that deals with the Technology of Information; hence all information related must be dealt with using available technology. As far as web development tools and platforms are concerned, there are many systems available to the public free of charge to aid in development. The FSKTM, on its (http://umisisweb.um.edu.my) website has a link named examination results which is never functional and does not show anything. The SIS or Student Information Service (Used by FSKTM) provided in this website is not sufficient to keep all Student's Records or data, for all his period in the faculty as a student and the results shown is for one semester only and not for his years of study. To overcome this problem the researcher suggested Student Record Retrieval System (SRRS) as solution to the way the results being sent to students which they can access anytime, anywhere while they have a web connection or Internet as well as keeping a record of their information[6],[7].

Manuscript received June 5, 2009
Manuscript revised June 20, 2009

III. METHODOLOGY

A good software development practice starts with a good understanding on the user requirements .A requirement is a feature of the system or the description of the something that the system is capable of doing in order to fulfill the system's purpose . Requirements are to be defined according to two categories which are the functional requirement and non-functional requirement (Kendall, 1996).The functional and non-functional requirements of SRRS are gathered by way of:-

a) Software Process Model

The system development methodology is a method to create a system with a series of steps or operations or can be defined as system life cycle model. Every system development process model include system requirement such as user, needs and resource as input and a finished product as output as show in figure 1.

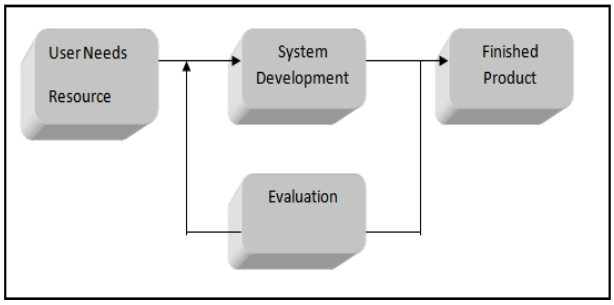

Figure 1 System Process Mode**.**

A software process model is an abstract representation of a software process. Each process model represents a process from a particular perspective, and thus provides only partial information about that process. The process model that is used to develop the system is the incremental model .This model uses the waterfall model in an iteration fashion. It focuses on the delivery of an operation product with each increment. It works well to address technical risk management and ramp staffing with the complexity of the work..Figure2 show the incremental model.

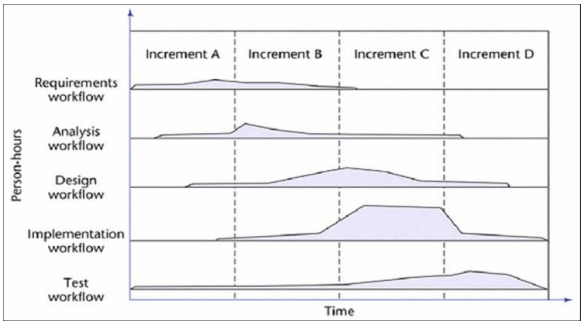

Figure 2 Incremental Model

b) Choice of Incremental Model

The incremental build model is a method of software development where the model is designed, implemented and tested incrementally until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements.

This approach is favored by many object-oriented practitioners. It basically divides the overall project into a number of increments, then it applies the waterfall model to each increment. The system is put into production when the first increment is delivered. As time passes additional increments are completed and added to the working system. The incremental development model for object-oriented project comprises the following phases:

A. Requirements Workflow

The first phase of this development involves the Requirements specification, which is the usual starting phase of all software process models. The requirement needs to be determined at the beginning phase of the project. It includes users of the system, the system's services, constraints and goals. These requirements are described and defined in details, serving as the system specification. Those analyses involved the functional requirements and the non-functional requirements. These sets of requirements are based upon extensive use and development of SRRS. Some of the requirements were gathered through user feedback and user requests conducted in an informal manner. The main goal of these requirements is to develop a system that enables student to check their recorded data. In this phase, the information about the user's requirements shall be gathered and documented. This phase can last from a few days to few weeks and is usually carried out on-site or through customer communication. Discussions shall be held to understand the user's requirements. The software requirements shall be documented. The documentation shall use the terms familiar to the customer and shall encompass the entire functionality of the product as foreseen by the customer (Student).

A review of the user requirement document is to be done. User's requirements will then be converted to implementation specific statements. The user-interface shall also be detailed. The user's software requirements shall be thoroughly understood to convert to system specifications. The software requirements specifications shall be documented using ***Use Case diagrams.*** The documentation shall use the implementation specific terms for the ease of understanding for the designers and developers.

B. System Analysis and Design Workflow

The next stage involves identifying and prioritizing user requirements. User requirements are prioritized and the highest priority requirements are included in the early increments. In this phase, the use cases are identified, the deployment and component diagrams are designed. The system design document shall be prepared. As we are developing web-base application framework and subsequently build SRRS based on this framework. The framework components that need to be developed are identified as follows:-

- Design of framework architecture
- Setting up application server
- Design of security model

Once the base (framework) has been developed, we need to identify the components required for setting up SRRS system. The components that need to be developed are identified as follows.

- Design and implementation of the new database management system
- Design of the user client interfaces
- Design of user manager module
- Design of security module

C. Implementation Workflow

In this phase, the actual coding shall be done according to the programming standard. The code shall be unit tested. The programming standards to be used shall be identified. This is one of the most critical phases as failure in coding will result in collapse of the whole project. New training skills are required, which have to be learned or developed. While system analysis and design workflow identifies the components that need to be developed, this phase develops and implements all the design requirements identified in the earlier phase.

The development involves the implementation of SRRS. This phase involves the installation of .NET framework 2.0, the Microsoft Access 2000, and the programming environment, which is ASP.NET. In order to implement this phase, new skills have to be developed. Specifically, this means learning the ASP.NET programming language, the .NET Framework, and how to implement Microsoft Access 2000 functionality.

D. Testing Workflow

In the system-testing phase, the product shall be tested module-wise and the interdependencies among the modules shall be validated. The functionality of the product shall be tested as a whole. The product needed to be tested for conformance with the system requirements, that is, the overall goal of the project. At this phase, user involvement is vital to the design, redesign and validation of the user interfaces. During the user acceptance-testing phase, SRRS system shall validate against the user requirements, acceptance criteria and acceptance data. Figure 3 shows the incremental development system.

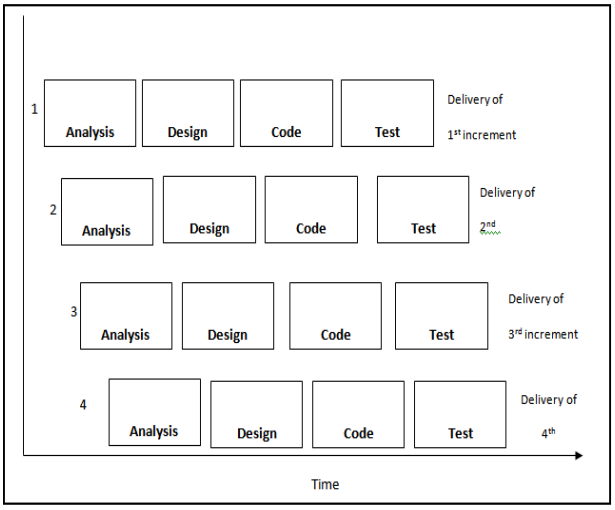

Figure 3 Increment Deployment Cycle.

E. Advantages 0f Incremental Development Model

- There is a lower risk of overall project failure. Although problems may be encountered in some increment, it is likely that some will be successfully delivered to the customer.
- Customers do not have to wait until the entire system is delivered until they can gain value from it. The first increment satisfies their most critical requirement so the software can be immediately used.
- It provides an opportunity to explore alternative strategies and revisions.
- Early feedback is generated because implementation occurs rapidly for a small subset of the system.
- It ensures that the developers build the right system according to the specification and verification of the system.
- More flexible on requirements change.
- More parallelism saves lots of time! However, there are still some weaknesses in this model
- Extra time spent on testing, documenting and maintaining a "temporary" product.

IV. SYSTEM ANALYSIS OF SRRS

A. System Requirement Analysis

The purpose of system requirements analysis is to get a thorough and detailed understanding of the business need as defined in Project Origination and captured in the Business case. Generally, Requirement analysis always classified as functional requirements and non-requirement

Table 1: Glossary

Term	Definition
User A	This abbreviation represents student
User B	This abbreviation represents admin
User C	This abbreviation represents lecturer

i. Functional Requirement

1. Log-In

Use case: Login

Figure 4. Login

USER login SRRS database
Brief description:
This use case is available on the interface of the system; users with authority (username and password) can enter SRRS.
Initial step-by-step description:

- The system displays the 'login' link.
- The user enters the username and password.
- He system authenticates the user by comparing the username and password with the database.
- The system displays the suitable models to the user to sign in.

2. Log-Out

Use case: Logout

Figure5. Logout

USER logout of SRRS
Brief description:
This use case is there for user to log out of SRRS
Initial step-by-step description:

- User press 'logout' button.
- The system closes itself.

3. Manage

Use case: manage profile

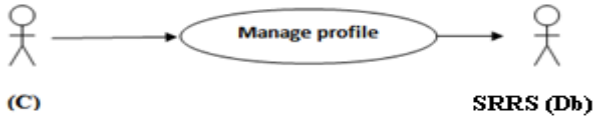

Figure 6. Manage Profile

Brief Description:
This use case is used to manage user profile
Initial Step-By-Step Description:

- User press link.
- Sir page shown and user can edit.
- User must save changes to profile.

4. View

Use case: View

Figure 7. View

Brief Description:
This use case is used to view results or profiles
Initial Step-By-Step Description:

- User clicks view.
- SRRS will show the required data.

5. Print

Use case: print

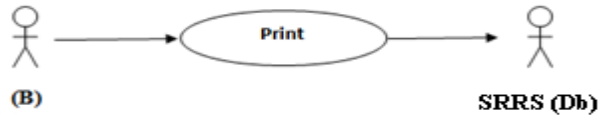

Figure 8. Print

Brief Description:
The use case print is used by Admin to print records for requesting students.
Initial step by step description:

- Admin chooses the record.
- Admin press print, printer prompt page appear.
- Admin clicks print.

6. Save

Use Case: Save or upload marks.

Figure 9. Save

Brief Description:
This use case is initiated by the Lecturer to upload student's marks.
Initial step by step description:

- Lecturer selects subject and upload marks.
- Lecturer press saves.

7. Create Account

Figure 10.Create Account

Brief Description:
This use case used by admin to create a new account
Initial Step-By-Step Description:

- Admin press button to create new profile.
- Admin writes new info.
- And saves changes.

8. Delete
Use case: Delete

Figure 11. Delete

Brief Description
This use case is used by Admin to remove any data or profile from SRRS system.
Initial Step-By-Step Description:

- Admin selects manage profile.
- Selects delete.
- System will prompt (are you sure you want to delete this item?)
- Info or data will be deleted from SRRS database when selecting a (yes).

ii. Non-Functional Requirements

A. Integrity
Integrity is the quality of correctness, completeness, wholeness, soundness and compliance with the intention of the creators of the data. It is achieved by preventing accidental or deliberate but unauthorized insertion, modification or destruction of data in a database. Data integrity is one of the six fundamental components of information security. Integrity it's very important process not just for SRRS, it's important for any system to prevent any mistake or errors may happens from the users or operators during using the system. The most important points in an integrity process in SRRS are verified and checked are records or profiles. After a new account is uploaded to SRRS database, admin staff must check and verify the new data or profile file is in SRRS database.

B. Flexibility
The design and built of the system regarding to customer requirements. When one of university requests from me built SRRS for the keeping of student records. The interaction process (download, upload, modify, delete, verify……..Etc). After 2-3 years or less or more Database, SRRS will be an instrument to keep tracking for student records. The interface can be extended to DB capacity. So it is flexible for any request regarding to the customer requirements.

C. Usability
In general usability refers to how well users can learn and use a product to achieve their goals and how satisfied they are with that process. Also declare to users' how to use the product quickly and easily to accomplish their tasks. In my design i toke some considerations that:

- Who are the users, what do they know?
- What do users want or need to know?
- What is the general background of the users?

So the SRRS is considered as one of the easiest system in use. As researchers know the most users in the faculty are familiar with windows operating system, so this project has been used windows programs in the designing because of its popularity and also facilitate to user's access to his/her requirement direct by clicking buttons. And also the interface has designed clearly to give the users easy view.

D. Security
Security is process to prevent and detect unauthorized use of your system. Prevention measures help you to stop unauthorized users (also known as "intruders") from accessing any part of your system. Detection helps you to determine whether or not someone attempted to break into your system, if they were successful, and what they may have done. So In SRRS we have user name and password the users cannot access the system if they do not have username and password to give them authorization, this process to avoid any problem may be happen from illegal users. In addition, if user has username and password cannot do any things there is a

limits for the authorization, the normal user can do search, download and print only, but the administrator/lecturer have authority to create delete edit and upload. This Restrictions process to make a system more secure and avoid any illegal use for the system.

E. Maintainability

Depending on the definition of Institute of Electrical and Electronics Engineers (IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. New York, NY: 1990.), Maintainability is the ease with which a software system or component can be modified to correct faults, improve performance, or other attributes, or adapt to a changed environment. SRRS system is supposed to be corresponding to maintainability standard definition. This function includes SRRS Software Re-Engineering as changing the form of existing software without changing its functionality. Normally such changes are carried out in order to improve its maintainability, portability, or the efficiency of software. SRRS Software Re-engineering can include any or all of the following:-

- Improving the structure of the text;
- Control Flow Restructuring;
- Re-modularization;
- Language Conversion;
- Redistribution of functionality over a network;
- Face-lifting (improving the front end without changing the functionality);
- Migration between platforms or environments;
- Re-documentation
- Improving methods of carrying out software maintenance;
- Improving methods of visualization, assessment, and comprehension;
- Development of tools to support such improved methods;

Re-modularization of the objective of this project is to develop methods and tools which can do some or all of the following:

- Identify suitable pieces of code (or modules), within an application can gets good interface and easy to use.
- Identify the data and procedural objects which are controlled by the Admin Staff for ease use with adequate capability to discover and correct errors.
- Identify suitable pieces of code, which are supporting the system for assumption any new uploading data.

V. SYSTEM ENVIRONMENT OF SRRS

- The system programmed by ASP.net language to build a content of it and Microsoft Access 2000 to manage the database.
- The SRRS System will be uploaded to internet server.
- The users must have ISP (Internet service provider) to access into the system.
- The Lecturer user can access, choose, load, update, and save
- The Administrator can manage (view and print) for official use only.
- A Student user of the SRRS system may access, view and manage profile.
- All users can access the system using any internet browser such as: Internet Explorer, Mozilla Firefox, and Opera.
- Many users can access the system (simultaneously).

VI. REVIEWS ON LATEST TECHNOLOGIES

A. Client –Server Architecture

Client-server is a network architecture which separates the client (often a graphical user interface) from the server. Each instance of the client software can send requests to a server or application server. There are many different types of servers; some examples include: a file server, terminal server, or mail server. While their purpose varies somewhat, the basic architecture remains the same. Client-server architecture is intended to provide a scalable architecture, whereby each computer or process on the network is either a client or a server.

a) Properties of a SERVER
   - Passive (Slave)
   - Waiting for requests
   - On requests serves them and send a reply

b) Properties of a CLIENT
   - Active (Master)
   - Sending requests
   - Waits until reply arrives .

B. Tiered Architecture

A generic Client/Server architecture has two types of nodes on the network: clients and servers. As a result, these generic architectures are sometimes referred to as "two-tier" architectures. Some networks will consist of three different kinds of nodes: clients, application servers which process data for the clients and database servers which store data for the application servers. This is called three-tier architecture. The advantage of an n-tier architecture compared with a two-tier architecture (or a three-tier with a two-tier) is that it separates out the processing that occurs to better balance the load on the different servers; it is more scalable. The disadvantages of n-tier architectures are:

- It puts a greater load on the network.

- It is much more difficult to program and test software than in two-tier architecture because more devices have to communicate to complete a user's transaction.
- It is good architecture too. Figure 12 show three tire architecture.

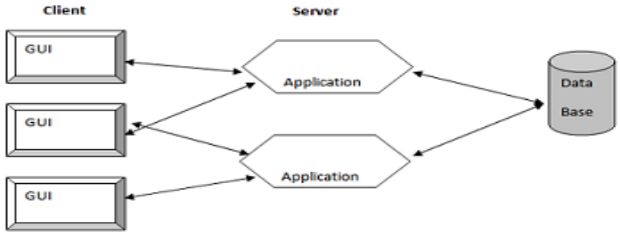

Figure 12: Three tire architecture.

VII. TESTING OF SRRS

There are two fundamental approaches to identifying test cases, these are known as functional and structure testing, each of these approaches has several distinct test case identification methods, more commonly called testing methods, functional testing is based on the view that any program can be considered to be a function that maps values from its input domain to values in its output range. (Function, domain and range) this notion is commonly used in engineering. There are two distinct advantages to functional test cases: they are independent of how the software is implemented, so if the implementation changes, the test cases are still useful and test case development can occur in parallel with the implementation, there by reducing overall research development interval, on other side, functional test cases frequently suffer from two problems: there can be significant redundancies among test cases, and this is compounded by the possibility of gaps of untested software. As shown in figure 13.

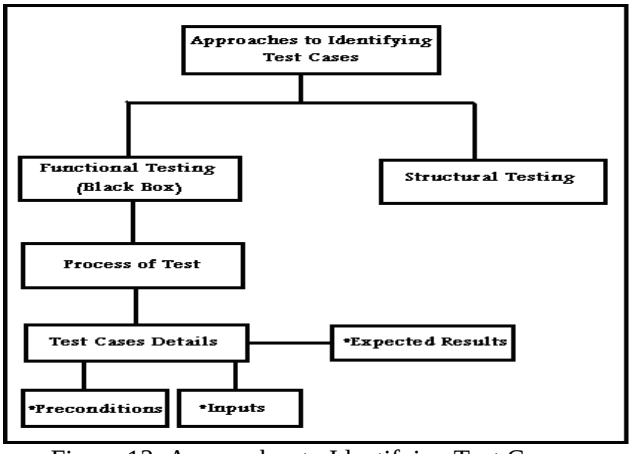

Figure 13: Approaches to Identifying Test Cases

When systems are considered to be "black boxes" test cases are generated and executed from the specification of the required functionality at defined interfaces, this leads to the function of the black box is understood completely in terms of its inputs and outputs, as shown in figure 14. Black-box testing has some important advantages:

a) It does not require that the code is seen, it is testing. Sometimes code will not be available in source code form, yet it can still construct useful test cases without it. The person writing the test cases does not need to understand the implementation.
b) The test cases do not depend on the implementation. They can be written in parallel with or before the implementation. Further, good black-box test cases do not need to be changed. Even if the implementation is completely rewritten.
c) Constructing black-box test cases causes the programmer to think carefully about the specification and its implications. Many specification errors are caught this way.

The disadvantage of black box testing is that its coverage may not be as high as like, because it has to work without the implementation. But it is a good place to start when writing test cases, with the functional approach to test case identification; the only information that is used is the specification of the software.

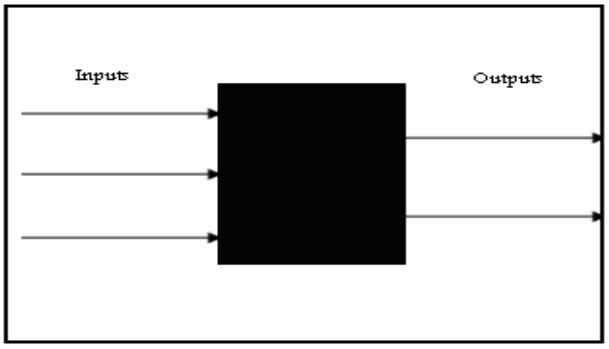

Figure 14: Black box.

A. Process of the Test

i. Test Case One:
- 9 pictures about admin and lecturer actions.

ii. Test Case Two:
- 2 pictures about student actions.

B. Test Cases Details

Test cases are known preconditions, inputs and expected results, which is worked out before the test is executed. The definition of software installation needed for test an (Preconditions) and the definition inputs should needed

for test an (Inputs) and the definition predictable results for outputs an (Except Results).

a) Preconditions:

- Installation (Microsoft Windows XP for Any Version or Vista).
- Installation (Microsoft Visual Studio 2008/ASP.Net).
- System application for this research.
- Installation (Microsoft access 2003 / 2007).

b) Inputs

The system has two types of inputs:

- Inputs for lecture /admin information's as shown in table 2.
  - One sample.
- Inputs for student information's as shown in table 3 :
  - One sample.

Table 2:
Inputs for lecture /admin information's

Lecturer ID	User Name	Password	Full Name
1	admin	0000	Administrator
2	Omar	1111	Dr Omar Zakaria

Table 3:
Inputs for student information's

Matric Number	Full Name	Password	Address
111111	IBRAHIM MUHAMADI	1212	Bukit Oug Condo

c) Expected Results:

The system will function as it is excepted:

- Student records kept.
- Result distributed.

i. Test Case One:

- Admin is already exists in database.

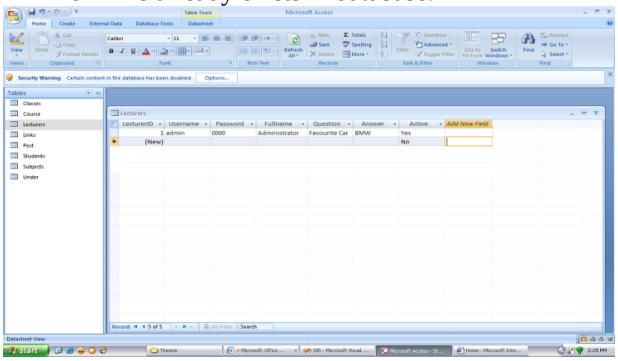

Figure 15. Show the Admin in Database.

- Admin login.

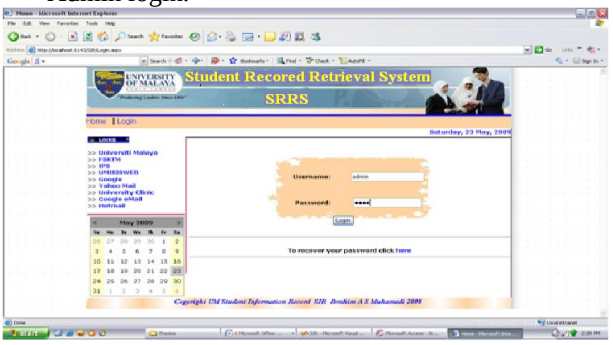

Figure 16. Shows the Admin Login.

- Admin adds a new profile (lecturer)

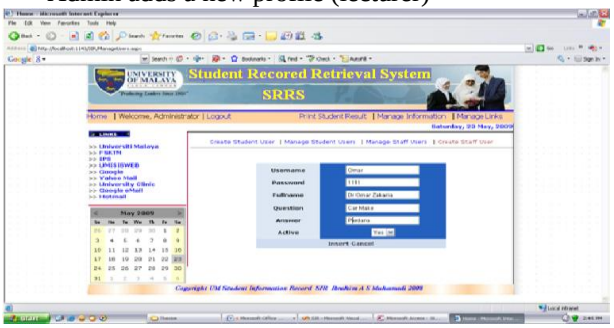

Figure 17. Shows the admin add new profile.

- The new lecturer profile appears in database.

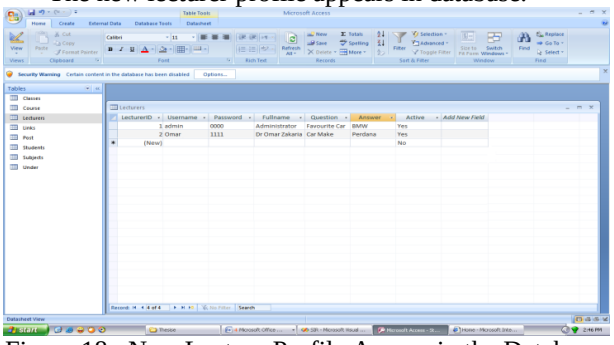

Figure 18. New Lecture Profile Appear in the Database.

- Admin adds a new profile (Student)

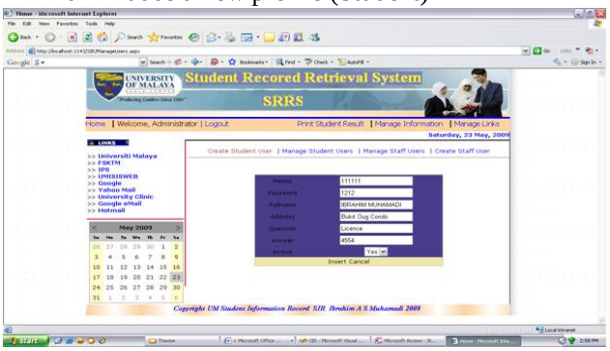

Figure 19. The Admin Add New Profile

-

- The new student profile appears in database.

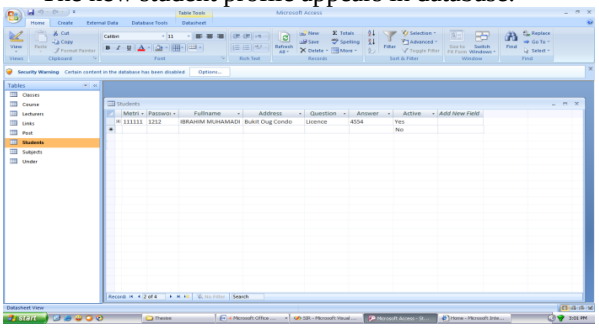

Figure 20. New Student Profile Appear in the Database.

- Lecturer login.

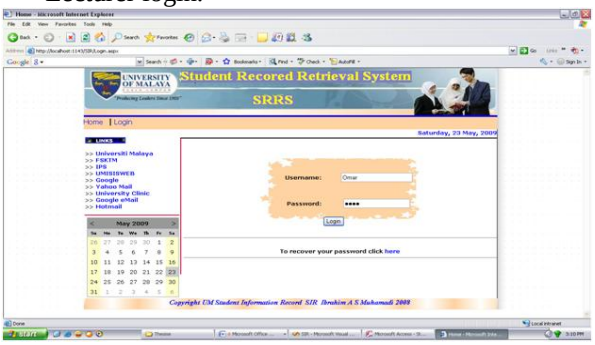

Figure 21. Shows the Lecturer Login.

- Lecturer Choosing a Subject.

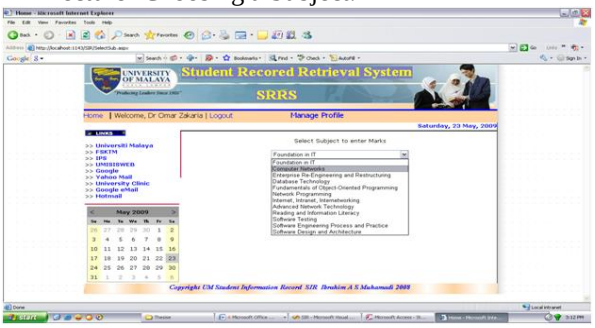

Figure 22. Shows the Lecturer Choosing a Subject.

- Lecturer Uploads Mark to Student.

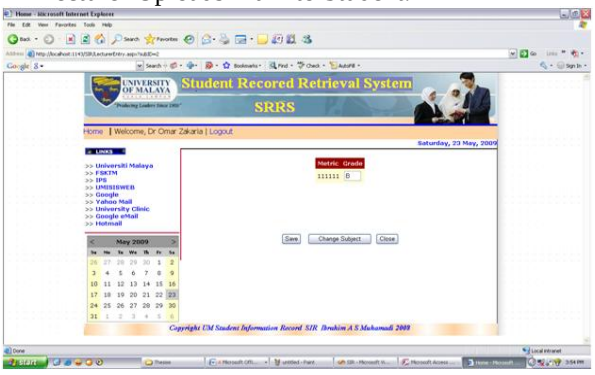

Figure 23. Shows the Lecturer Uploads Mark to Student.

ii. Test Case Two

- Student login.

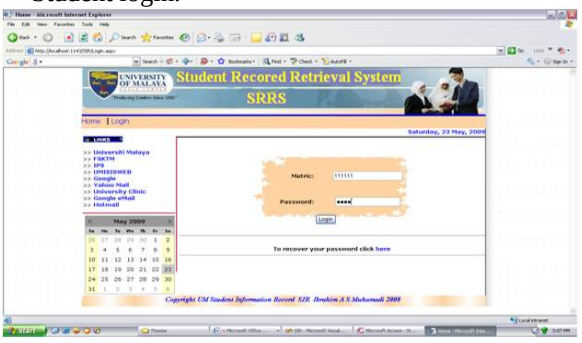

Figure 24. Shows Student login.

- Student Interface with Information.

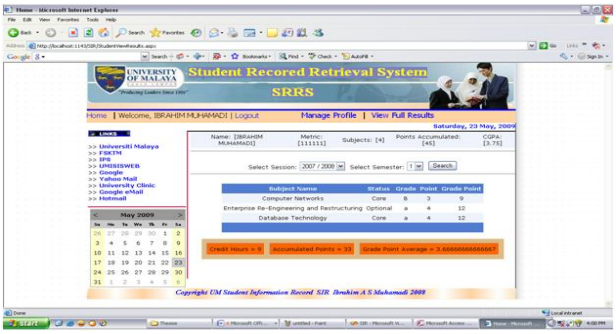

Figure 25. Shows Student Interface with Information.

VIII. CONCLUSION

This SRRS system is the solution to the way the results being sent to students which they can access anytime, anywhere while they have a web connection or Internet as well as keeping a record of their information. We get the following conclusions:

- One of the important conclusions in implementation of the proposed system is the solving of the problems that are related SRRS system that has to be deployed and accessible from anywhere through the Internet easily. This system can be assessed easily using the web browser such as Internet Explorer and Netscape Navigator by means of ID & password.
- User friendly interface which the web page is designed to be simple and systematic to ease the users when they navigate and explore the web site. The screen design maintains its consistency throughout the system. All the styles and fonts type are the same throughout the system.
- Supporting High Volume of Users, SRRS is deployed using the Microsoft Access 2000.This makes it ready to cope with large amount of users in the future.

ACKNOWLEDGEMENT

Thanks in advance for the entire worker in this project, and the people who support in any way, also I want to thank University of Malay for the support they offered, also I would like to extend our deep apparitions and thanks Dr.S.Raviraja for his support.

REFERENCES

[1] Abelson, R. P. (2008). "Script processing in attitude formation and decision making," in J. Carroll and J. Payne, editors, Cognition and social behavior, Hillsdale, NJ: Lawrence Erlbaum.

[2] Abernathy, W. J. and Utterback, J. M. (2008). "Patterns of industrial innovation," Research Policy, 14: 3-22.

[3] Allen, T. J. (2008). Managing the flow of technology: Technology transfer and the dissemination of technological information within the R&D organization, Cambridge, MA: MIT Press.

[4] Almeida, P. (2007). "Knowledge sourcing by foreign multinationals: Patent citation analysis in the U.S. semiconductor industry," Strategic Management Journal, 17 (winter special issue): 155-165.

[5] Almeida, P. and Kogut, B. (2006). "Localization of knowledge and the mobility of engineers in regional networks," Management Science, 45: 905-917.

[6] Amburgey, T. L. and Miner, A. S. (2005). "Strategic momentum: The effects of repetitive, positional, and contextual momentum on merger activity," Strategic Management Journal, 13: 335-348.

[7] Ancona, D. and Caldwell, D. (2005). "Bridging the boundary: External activity and performance in," Administrative Science Quarterly, 37(4): 634- 656.

Authers Information

Ibrahim A.S.Muhamadi- he is master student in Department of Information Technology/ Faculty of Computer Science and Information Technology/University of Malaya /Kuala Lumpur/Malaysia, He is a late comer to IT filed after a carrier of 30 years in the airlines business as a fling captain. He has contributed to many papers some of them still under reviewer.

Mussab Alaa Zaidan - he is master student in Department of Information Technology/ Faculty of Computer Science and Information Technology/University of Malaya /Kuala Lumpur/Malaysia, He has published many papers at international conferences and journal.

Aos Alaa Zaidan - He obtained his 1st Class Bachelor degree in Computer Engineering from university of Technology / Baghdad followed by master in data communication and computer network from University of Malaya. He led or member for many funded research projects and He has published more than 40 papers at various international and national conferences and journals, he has done many projects on Steganography for data hidden through different multimedia carriers image, video, audio, text, and non multimedia carrier unused area within exe.file, Quantum Cryptography and Stego-Analysis systems, currently he is working on the multi module for Steganography. He is PhD candidate on the Department of Computer System & Technology / Faculty of Computer Science and Information Technology/University of Malaya /Kuala Lumpur/Malaysia. He is members IAENG, CSTA, WASET, and IACSIT .

Bilal Bahaa Zaidan - has received his master from Department of Computer System & Technology Department Faculty of Computer Science and Information Technology/University of Malaya /Kuala Lumpur/Malaysia, He led or member for many funded research projects and He has published more than 40 papers at various international and national conferences and journals. His research interest on Steganography & Cryptography with his group he has published many papers on data hidden through different multimedia carriers such as image, video, audio, text, and non multimedia careers such as unused area within exe.file, he has done projects on Stego-Analysis systems, currently he is working on Quantum Key Distribution QKD and multi module for Steganography, he is PhD candidate on the Department of Computer System & Technology / Faculty of Computer Science and Information Technology/University of Malaya /Kuala Lumpur/Malaysia. He is members IAENG, CSTA, WASET, and IACSIT.