Journal of
Applied Sciences

Developmental Issues of Web-based Decision Support System

[1]Fadhilah Ahmad, [2]A. Noraziah, [1]Suhailan Safei, [1]Azrul Amri Bharun Abidin,
[3]N. Ahmed Abdalla and [1]Aznida Hayati Zakaria Mohamad
[1]Faculty of Informatics, University Sultan Zainal Abidin, UniSZA, 21300 Gong Badak Campus,
Kuala Terengganu, Malaysia
[2]Faculty of Computer Systems and Software Engineering,
[3]Faculty of Electrical and Electronic Engineering,
University Malaysia Pahang, Gambang 26300, Malaysia

Abstract: Decision Support Systems (DSS) are computer programs that aid users in a problem solving or decision-making environment. DSS contains decision making information and other related data that may be of interest to unscrupulous competitors and stakeholders. Therefore, a DSS that is developed for web-based environment has to be secured from unauthorized access or malicious attacks. This study describes a framework of a secured DSS suitable to support decision making activities. Security issues are discussed from various perspectives with the proposed mechanisms to ensure a secured DSS implementation in distributed environment. Issues in web-based application development that include data growth, system performance and appearance compatibility are also presented together with their proposed solutions. Considering security and development issues could contribute towards successful implementation in the decision making process.

Key words: Decision support system, security, web-based application, distributed environment, locking mechanism, SQL injection

INTRODUCTION

Advances in computer and Internet technologies as well as related software applications have great impact on organizations and society. The infrastructures provided by these technologies improve many aspects of organizational activities (Sidek *et al.*, 2010) ranging from traditional uses in payroll and book keeping functions to more complex managerial areas such as the process of evaluation and decision making (Turban *et al.*, 2005). Today, many software applications that include transaction processing, monitoring activities, problem analysis and solution applications are implemented over the web-based technologies (Geoffrion and Krishnan, 2001). The use of these technologies could reduce much of the cost of business implementations in the long term, improved communication, decision making and enable an organization to stay competitive with their rivals.

In many organizations, decision making is the task that must be done by the managers and it is one of their challenging roles. Decision making is limited to Decision Maker's (DM's) computation, foresight, analytical powers and sometimes to psychosocial biases such as negative and cost consequences and also to gains rather than loss alternatives. Additionally, decision situations are unique from one another in the sense of the clarity of goals, the type of DMs involve and the general consensus among DMs. Therefore, the system to support decision making which is called Decision Support System (DSS) should deserve special attention by researchers in this area. The system developed can assist managers especially at management control and strategic planning levels to make better decisions (Ozturan and Ayan, 2001; Rigopoulos *et al.*, 2008b; Main and Mesgari, 2009). It was applied in several areas such as banking decisions (Rigopoulos *et al.*, 2008a), human resource selection (Ahmad *et al.*, 2010; Celik *et al.*, 2009; Khosla *et al.*, 2009), land use planning (Main and Mesgari, 2009), procurement selection (Ahmad *et al.*, 2007; Rapcsak *et al.*, 2000; Mohamad, 2005; Bertolini *et al.*, 2006; Sorenson and Kanavos, 2010), technology evaluation (Lai and Tsai, 2009) and many more.

DSS which are implemented in web-based environment is proned to be attacked by automated abuse. The abusement would make the system processing unnecessary request generated by automated program. This kind of abuse would degrade the service of the given system. Therefore, the system would require protection

---

Corresponding Author: Fadhilah Ahmad, Faculty of Informatics, University Sultan Zainal Abidin, UniSZA,
21300 Gong Badak Campus, Kuala Terengganu, Malaysia

against those kind of abuse by using human-based character recognition via web security measures. Some other issues that need attention in a web-based system is the SQL injection and Cross-Site Scripting (XSS) (Ishizaka and Labib, 2011). The aim of this study is to present a framework for Group Decision Support System (GDSS) in distributed environment. Consequently, security and development issues for the system are discussed as they could influence the successful implementation of the system in the distributed environment.

WHAT IS DECISION SUPPORT SYSTEM?

DSS is computer programs that aid users in a problem solving or decision-making environment. These systems employ data models, algorithms, knowledge bases, user interfaces and control mechanisms to support a specific decision problem (Barkhi *et al.*, 2005). A DSS does not simply provide direct solutions, but it rather gives a recommendation based on the input of the Decision Makers (DMs). Input data to DSS may come from a wide range of sources either from internal or external organizations.

A decision can be defined as making choices among a set of alternatives. Decision making is a process that must be done by the managers via a set of phases. Simon (1977) has proposed three decision making phases which are intelligent, design and choice. Intelligence phase provides relevant information by finding, identifying and formulating the problem. In design phase, it involves developing alternatives by identifying or analyzing them. While in choice phase, alternatives are evaluated by giving advice about which of them to choose. These phases can be iterated by repeating any of these phases if a DM is not satisfied with the result or if the problems still persist. This decision making process can be designed as part of the DSS.

DSS is also known as group DSS (GDSS) when a group of decision makers are involved in the evaluation process (Ahmad *et al.*, 2008; Raja and Srivatsa, 2006). There is a need to consider a group of DMs for improving productivity of decision making and quality of results. There also has been an increasing trend towards distributed computing environment, in which a collection of sites each of which hosting a database and possibly multiple model bases can be dispersed within an organization (Moreira *et al.*, 2011). This type of setting may cause possible multiple updates to be executed concurrently to the shared data items in the repository when certain GDSS operations are executed. If these updates are not controlled, erroneous updates can occur.

The design of distributed GDSS therefore must include a mechanism to isolate one update from the effect of concurrently executing updates. One possible approach to deal with this issue is via the use of locking mechanism (Moreira *et al.*, 2011; Xiao and Lu, 2011; Jea *et al.*, 2011).

ISSUES IN WEB-BASED DEVELOPMENT AND APPLICATION

During the early version of web applications such as DSSs, they were run and maintained on a central system (Sharda *et al.*, 1988). Today, the system can be implemented in geographical dispersed or distributed areas in the web-based environment for better availability and reliability of data as well as for performance improvement of the system (Deris *et al.*, 2005; Noraziah *et al.*, 2007). To ensure successful deployment of the system in distributed and web-based environment, many factors have to be considered that include the design of its framework, security and also the development issues.

A framework for group DSS in distributed environment: A Web-based application requires a Web browser such as Netscape Navigator or Internet Explorer (Power and Sharda, 2005). Protocol such as TCP/IP is also needed to link the computer server that is hosting the application to the user's computer in the network. Bhargava and Krishnan (1998) have given clear classification of web enabling technologies. The technologies are divided into three which are server-side, client-side and the distribution-side environment technologies. Figure 1 describes the relationship among these technologies with GDSS in distributed environment.

Server-side technologies: Provide platform, independent and universal access to DSS. The server-side application program located in the web server can manage and implement client tasks. Some common technologies include web server, HyperText Markup Language (HTML), Common Gateway Interface (CGI), Java, PHP, Perl and possibly a database product such as Oracle or MySQL. Data storage for GDSS contains database and model base integration. A certain access to this data store is controlled by local consistency protocol such as locking. Local operations include synchronize updates of a file by possibly concurrent processes and then synchronously propagates the changes to the main processing center at remote site.
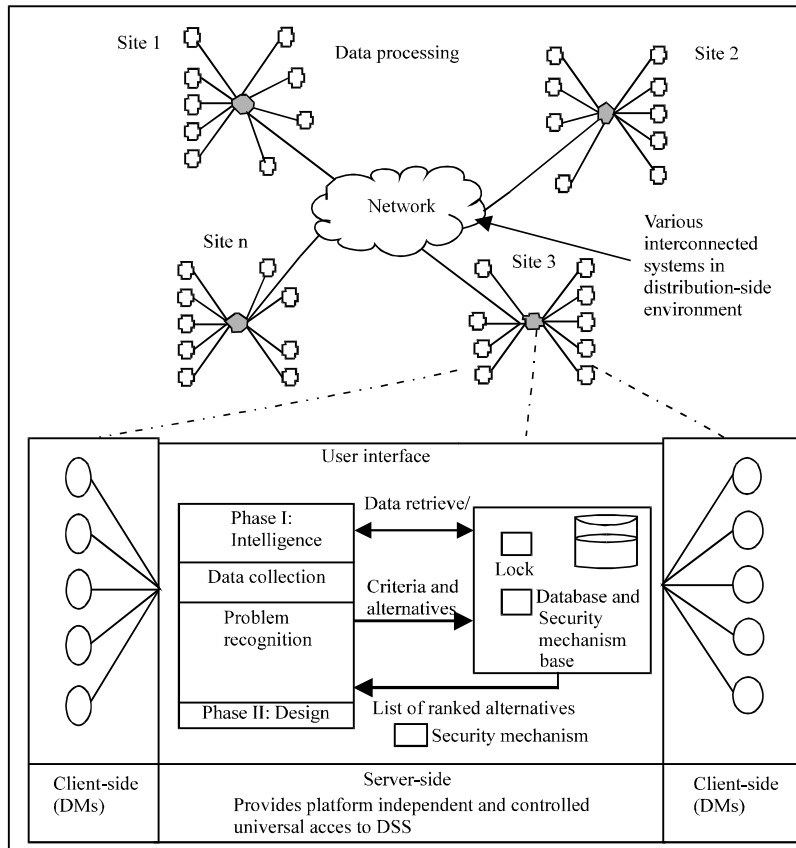
Fig. 1: Group decision support systems technologies in distributed environment

Client-side technologies: Enable incorporation of more intelligence to be embedded in the user interface. Some common technologies include client-side scripting languages, ActiveX controls and browser plug-ins. DMs in a certain client-side can only access related information belonging to this particular site only. Any attempts to access the remote sites are blocked by the security mechanism unless the internet users are given the access right.

Distribution-side technologies: Enable GDSS components to be distributed and deployed across various interconnected system. Some common technologies include Common Object Request Broker Architecture (CORBA), Distributed Component Object Model (DCOM) and Java (Fig. 1).

Security issues: There are three main security issues that must be handled for web-based applications. They are automated abuse, SQL injection and Cross-site Scripting (XSS).

Automated abuse: As the World Wide Web became more accessible to more and more people, the possibility of web-based DSS to be abused are greater. Among others is automated abuse, which acts like a human in order to abuse many Internet based application. CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart) are common security measures for preventing automated programs from abusing online services.

SQL injection: SQL injection attacks are also known as SQL insertion attacks. SQL injection is a security issue that exploits a security vulnerability that occurs in the database layer of a web-based application. It is a code injection technique upon present of vulnerability in the user input that are incorrectly filtered, or SQL statements that is not strongly typed and thereby unexpectedly executed.

There are a number of approaches that have been proposed to prevent SQL injection vulnerabilities. Microsoft (2006) proposed the use of syntax embedding that combines the syntax of the client languages into the

syntax of the host language at the serve-side. They make use of API (Application Programming Interface) and the conceptual ease of string manipulation for constructing sentences to prevent vulnerabilities independent of the language used.

Cross-site scripting: Cross-Site Scripting (XSS) is another type of computer security vulnerability that is typically found in web applications. It enables malicious attackers to inject client-side script into web pages viewed by other users. XSS vulnerability can be used by attackers to bypass access controls such as the same origin policy. The impact to the system may vary from only degradation of service up to a significant security risk.

Previous studies has suggested a number of approaches to deal with this problem. They include encoding output based on input parameters, filtering input parameters for special characters and filtering output based on input parameters for special characters. It has been suggested that a character set for the web pages such as ISO 8859-1 is specified during filtering or encoding, in order to ensure that the filter is checking for the appropriate special characters.

Development issues: Most of the web development implementation is based on server execution. Clients for

the web system can concurrently access the system via independent browsers for data viewing or posting. Based on web-based application layers shown in Fig. 2, three important issues in the development stages need to be considered; data growth in the database, performance of server execution codes and appearance compatibility in various browsers.

Data growth: A simple and small database system does not expose to major issues/problems in a web-based application. However, the capacity of database will expand and the number of users will grow periodically. Therefore, a web-based application should be designed by considering temporal aspect and validity usage of certain data in a table. Executing queries that select all data in a table should be avoided. If all data in a table need to be read, the queries have to be split using SQL limit statement. Otherwise, it is possible to limit the maximum rows of the table and create sub-sequence tables in order to store new data if the maximum limit has been reached. Sub-sequence tables can be named by embedding month or year in the table name. Via this operation, data size can be shrunk and manipulation of data on each table can become much faster.

Performance of server execution codes: The growth of database can affect system performance. Larger data
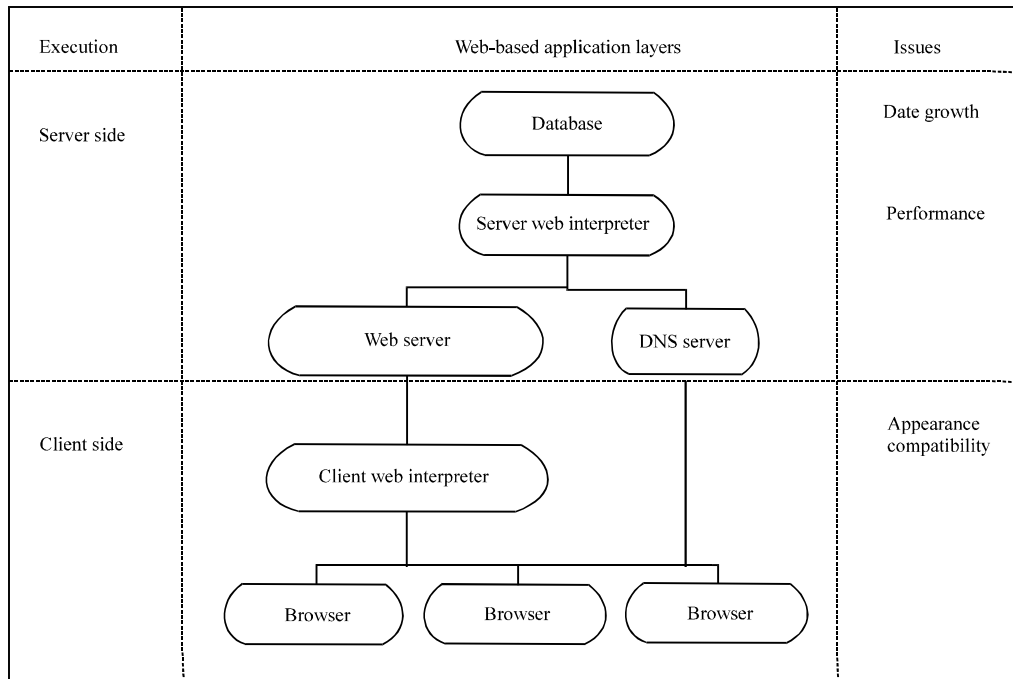

Fig. 2: Web-based application layers

normally consume longer time to be processed at server side. Usually, at the beginning of web-based application operation, posting data to the client is just a flashing process. However, there is a timeout limitation for communication between web server and its clients. Thus, when writing web-based codes, it is important to ensure that data is transferred to the client before the time expires. Although, eventually server has completed the client's request, the client may not get updated data because of being automatically terminated from server connection due to timeout limitation. Database manipulation and coding interpretation are done at server side. As all clients can access the server concurrently, some of the operation especially when updating database will be queued. Therefore, it is important for the web-based system to provide the status of current server execution processes as ignoring this notification may cause the user to assume that the system is unreliable or hang.

In terms of processing time, some of the calculation processes consume longer time to be completed. This requires the client to wait for the operation to be completed and the browser may seem to be inactive. To solve this problem, the developer needs to consider splitting some of the calculation processes into stages. Each stage will be completed and subsequently shown to the client's browser till the end of the process. Frame techniques can be used for this purpose.

Appearance: The appearance of web data especially when using styles may look differently depending on the browser setting and types. Screen resolution may also affect the format of data view. Thus, testing need to be performed for the system output using multiple resolutions and browsers. If plug-ins is used at the client side, it is important to ensure that all the required components are accessible and compatible in any browsers.

Based on our experience, it is important to consider the above mentioned factors in the development of web-based DSS as it could contribute towards its successful implementation (Fig. 2).

CONCLUSION AND RECOMMENDATIONS

The available, practical and affordable use of cyberspace technology has made the philosophy and concept of DSS organizational decision making becomes a reality. Further investigations on DSS in a wide context such as its infrastructures, group behaviors, development methodologies, processes and techniques and integration with other disciplines may lead to new research topics and

robust systems. In future work, we will adopt the DSS framework presented in this study for the application on a secured e-procurement in distributed environment. Possible problems of using locking mechanism such as deadlock and performance degradation will be studied closely in order to identify the factors that causing them and then suggest possible solutions.

REFERENCES

Ahmad, F., M.Y.M. Saman and N.M.M. Noor, 2008. A group decision support system (GDSS) through enhanced AHP for evaluation processes: Tender evaluation case study. Proceedings of the IEEE International Workshop on Digital Tainment and Visualization, June 2008, UMT Malaysia.

Ahmad, F., M.Y.M. Saman, N.M.M. Noor and A. Othman, 2007. DSS for tendering process: Integrating statistical single-criteria model with MCDM models. Proceedings of the 7th IEEE International Symposium on Signal Processing and Information Technology, Dec. 15-18, Cairo Egypt, pp: 863-863.

Ahmad, F., S. Safei, M.Y.M. Saman and H. Hassan, 2010. Integrated decision support system using instant messaging and enhanced AHP for human resource selection. Proceedings of the 1st International Symposium on Computing in Science and Engineering, June 3-5, 2010, Kusadasi, Aydin, Turkey.

Barkhi, R., E. Rolland, J. Butler and W. Fan, 2005. Decision support system induced guidance for model formulation and solution. Decision Support Syst., 40: 269-281.

Bertolini, M., M. Braglia and G. Carmignani, 2006. Application of the AHP methodology in making a proposal for a public work contract. Int. J. Project Manage., 24: 422-430.

Bhargava, H.K. and R. Krishnan, 1998. The World Wide Web: Opportunities for operations research and management science. INFORMS J. Comput. 10: 359-383.

Celik, M., A. Kandakoglu and D. Er, 2009. Structuring fuzzy integrated multi-stages evaluation model on academic personnel recruitment in MET institutions. Expert Syst. Appl., 36: 6918-6927.

Deris, M.M., M. Zarina, W.A. Suryani and Z. Aznida, 2005. Performance modelling of neighbor replica distribution technique on distributed database systems. Asia J. Inform. Technol., 5: 234-242.

Geoffrion, A.M. and R. Krishnan, 2001. Prospects for operations research in the E-business era. Interfaces, 31: 6-36.

Ishizaka, A. and A. Labib, 2011. Selection of new production facilities with the group analytic hierarchy process ordering method. Expert Syst. Appl., 38: 7317-7325.

Jea, K.F., T.P. Chang and C.W. Cheng, 2011. A generic simulation model for evaluating concurrency control protocols in native XML database systems. Comput. Stand. Interfaces, 33: 280-291.

Khosla, R., T. Goonesekera and M.T. Chu, 2009. Separating the wheat from the chaff: An intelligent sales recruitment and benchmarking system. Expert Syst. Appl., 36: 3017-3027.

Lai, W.H. and C.T. Tsai, 2009. Fuzzy rule-based analysis of firm`s technology transfer in Taiwan`s machinery industry. Expert Syst. Appl., 36: 12012-12022.

Main, H.H. and M.S. Mesgari, 2009. Developing a knowledge-based spatial decision support system for urban landuse allocation. J. Applied Sci., 9: 1758-1763.

Microsoft, 2006. How to prevent cross-site scripting security issues. Revision: 3.6, March 1, 2010. http://support.microsoft.com/kb/252985

Mohamad, N.N.M., 2005. An integrated web-base decision support system for tendering processes. Ph.D. Thesis, University of Manchester.

Moreira, L.O, F.R.C. Sousa and J.C. Machado, 2011. A distributed concurrency control mechanism for XML data. J. Comput. Syst. Sci., 10.1016/j.jcss.2011.02.006

Noraziah, A., M.M. Deris, M.Y.M. Saman, N.A. Ahmed, R. Norhayati and Z.M. Alfawaer, 2007. Preserving data consistency through neighbor replication on grid daemon. Am. J. Applied Sci., 4: 751-758.

Ozturan, M. and Z. Ayan, 2001. BUADVIS- a decision support system for student advising. J. Applied Sci., 1: 3-5.

Power, D.J. and R. Sharda, 2005. Model-driven decision support systems: Concepts and research direction. Decis. Support Syst., 43: 1044-1061.

Raja, K. and S.K. Srivatsa, 2006. A distributed data management in knowledge based group decision support systems. J. Applied Sci., 6: 27-30.

Rapcsak, T., Z. Sagi, T. Toth and L. Ketszeri, 2000. Evaluation of tenders in information technology. Decis. Support Syst., 30: 1-10.

Rigopoulos, G., J. Psarras and D.T. Askounis, 2008a. A TAM model to evaluate user`s attitude towards adoption of decision support systems. J. Applied Sci., 8: 899-902.

Rigopoulos, G., J. Psarras and D.T. Askounis, 2008b. A decision support system for supervised assignment in banking decisions. J. Applied Sci., 8: 443-452.

Sharda, R., S.H. Barr and J.C. McDonnell, 1988. Decision support system effectiveness: A review and an empirical test. Manage. Sci., 34: 139-159.

Sidek, R.M., A. Noraziah, M.F.J. Klaib and M.H.A. Wahab, 2010. Expediency heuristic in university conference webpage. Proceedings of the 2nd International Conference on Networked Digital Technologies, July 7-9, 2010, Springer-Verlag Berlin Heidelberg, pp: 566-576.

Simon, H.A., 1977. The New Science of Management Decision. 3rd Rev. Edn., Prentice-Hall, Englewood Cliffs, NJ.

Sorenson, C. and P. Kanavos, 2010. Medical technology procurement in Europe: A cross-country comparison of current practice and policy. Health Policy, 100: 43-50.

Turban, E., J.E. Aronson and T.P. Liang, 2005. Decision Support Systems and Intelligent Systems. 7th Edn., Pearson/Prentice Hall, USA., ISBN-13:9780130461063, Pages: 936.

Xiao, Y.Y. and K. Lu, 2011. WITHDRAWN: Secure concurrency control protocol with timeliness guarantees in real-time database systems. Math. Comput. Model., 10.1016/j.mcm.2011.01.009.