Reverse Engineering Approach in a Development of a Decision Support System for Forensic DNA Analysis

Noor Maizura Mohamad Noor
Department of Computer Science
Faculty of Science and Technology
Universiti Malaysia Terengganu (UMT)
Kuala Terengganu, Terengganu, Malaysia

Ahmad Faiz Ghazali
Department of Computer Science
Faculty of Science and Technology
Universiti Malaysia Terengganu (UMT)
Kuala Terengganu, Terengganu, Malaysia
ahmad.faiz14@yahoo.com

Md Yazid Mohamad Saman
Department of Computer Science
Faculty of Science and Technology
Universiti Malaysia Terengganu (UMT)
Kuala Terengganu, Terengganu, Malaysia

Zafarina Zainuddin
School of Health Science, Health Campus
Universiti Sains Malaysia (USM)
Kubang Kerian, Kelantan, Malaysia
Abstract

In getting software requirement and specification based on existing system, reverse engineering (RE) can be done by creating abstraction views of a system in order to help software engineer maintain and re-engineer it. Requirement engineering and specification in a development of a decision support system (DSS) presented in this paper is replaced by RE approach. Before that, software engineer needs to identify the pattern of design used in an existing system’s implementation, including programming details. RE can extract information of software design from source code, but the level of abstraction and documentation is yet to be discovered. RE can be done at least in three stages; RE of user interfaces, RE to understand internal process of a system and RE of database.

Keywords: Decision Support System, Forensic DNA Analysis, Reverse Engineering

1 Introduction

Generally, software is computer programs. Ian Sommerville [23] defined software not only computer programs but also all the required documentation and required configuration for computer programs to operate. According to Ian Sommerville [23], software engineering (SE) is a process of software production, beginning from software specification until software deployment and testing. As a program evolves, it becomes increasingly difficult to understand and reason about changes in the source code. Eventually, if enough changes are made, reverse engineering (RE) techniques must be used in order to understand the current behavior of a system. Roger S. Pressman [20] mentioned about the abstraction in RE;

“Ideally, the abstraction level should be as high as possible. That is, the RE process should be capable of deriving procedural design representations (a low-level abstraction), data and control flow models (a relatively high level of abstraction), and entity relationship models (a high level of abstraction). As the abstraction level increases, the software engineer is provided with information that will allow easier understanding of the program”.

Decision support system (DSS) is a concept of application that aims to help decision-making processes [29]. DSS can means different thing to different people, hence there is no universally accepted definition of DSS [29]. Turban and Aronson [29] mentioned that “DSS is also sometimes used as an umbrella terms to describe any computerized system that supports decision-making in an organization.”. Today, DSS are being applied in multiple domains, including forensic, medical, military, financial and engineering [22, 25, 29, 31].

In field SE, RE is a method of design recovery as mentioned a lot by Roger S.
Reverse engineering approach

Pressman [20, 19] but not Ian Sommerville [23]. Using concept RE, where based only on user interfaces, designs of internal and external data processing in a system flow can be recovered. Besides, from further analysis, processes, components, and its relationships among each other can be identified. After that, the representation of the system is created at a higher level of abstraction or in another form. In this research, a RE approach has been practiced to study Forensic DNA Databank of Malaysia (FDDM) used at Royal Malaysian Police (RMP). User interfaces of the software have been studied. Then, rapid prototype is built in a platform of Web to ease accessibility. What is the problem encountered during this design recovery processes? Information provided is very limited, due to national security issue. The information that initially cannot be recovered including:

i) Programming languages and technologies used to build the application.

ii) Source codes

iii) Mathematical equations used for analysis.

iv) The meaning and purposes of the available information and menus

RE can reduce certain costs. In SE, development is a risky and expensive proposition, hence improving the productivity of software development and the quality of delivered software will result in significant economic returns [1].

2 Reverse Engineering

Researchers Okolica [18] mentioned about classical definition of reverse engineering (RE) in 1990 by Cross [8] as “the process of analysing a subject system to (i) identify its system’s components and their inter-relationships and (ii) create representations of the system in another form or at a higher level of abstraction”. RE for software as rephrase by Roger S. Pressman [20] is “the process of analysing a program in an effort to create a representation of the program at a higher level of abstraction than source code. Reverse engineering is a process of design recovery”.

Software RE is a sub-discipline of SE [15]. It creates a suitable representation of the system in a higher level of abstraction or another form to support the comprehension of software systems. Figure 1 shows the schematic view of RE process [20].
The ‘abstraction level’ here refers to the sophistication of the design information that can be extracted from source code. Extract abstraction is the core activity of RE. Three phases in RE are including RE of user interfaces, RE to understand process, and RE to understand database. This concept is practiced as shown in section Results and Discussion of this paper.

3 Decision Support System

Decision support system (DSS) has been applied in a wide range of computer applications that commonly manipulate information in variety fields. This is including computer-based systems that are being used in handling data in order to achieve an optimization for decision-making in forensic science. Basic components in a DSS consist of model base, database, user interface and knowledge base. Information technology and human resources utilized in field forensic science and computer science will be able to help supporting decision-making using computerized forensic DNA analysis and paternity test.
In the early 1970s, the major concepts of DSS was first articulated by Scott Morton [11] as “interactive computer-based systems, which help decision-makers utilize data and models to solve unstructured problems”. Keen and Scott Morton [14] defined DSS as “a computer-based support system for management decision makers who deal with semi-structured problems”. DSS gather the human intellectuals and computer’s capabilities to improve the quality of decisions [14].

Scientist working in forensic field will be able to provide DNA data for analysis. Knowledge that can be interpreted from forensic will be able to help scientist to make decisions or provide a new alternative for solutions. Other recent research works in DSS come out with proposed software [13]. Vescoukis et al. [30] and Miranda et al. [17] proposed architecture in their research work. Researchers Tremblay et al. [28] doing related works to Web-based DSS concept.

4 Forensic DNA Analysis

Decision theory plays an important role in forensic science (FS) [6, 12, 16, 21, 24-27]. Deoxyribonucleic acid (DNA) profile analysis using computer programming is not a new research in biology and information technology. Projects have been done in developed countries and around the world to develop software tools to analyze DNA data. Aitken and Gammerman [4] initiated the discussion on the use of graphical probabilistic models for evaluating evidences in FS, and then, the first example to focus on Bayesian networks (BN) and DNA evidence is the work by Dawid and Evett [9] because it presents with example involving blood stains, as reviewed by Biedermann and Taroni [7]. Dawid et al. [9] discussed an example involves two kinds of scientific evidence which is more complex than the one discussed by Aitken and Gammerman [4]. A different approach has been presented by Garbolino et al. [10] where they do not primarily focus on a specific scenario. The difference between the study by Aitken and Gammerman [4] and [2, 3] is the nature of knowledge included into the graphical models of BN. His research works [4] is created based on own judgement for probability values, but models that he created later [2, 3] is come out based on data.

Forensic DNA Analysis Software (ForAS) proposed in this research is proven to be reliable and helpful for paternity test in forensic DNA analysis. The importance of this research work is it is a great start to develop more applications and extended software from ForAS for crime investigation. A new Web-based DSS to manage overall crime cases from the identification of suspects until apprehension of criminal offenders is currently under extensive development. It is another way of designing from a different perspective of overall process involved in the related systems. Technology is not the main focus, but the use and its relevancy to specific designs are included [5].
5 Results and Discussions

Figure 2 shows the screenshots of implemented software after Stage 1 of reverse engineering process for the functionality of Forensic DNA Databank of Malaysia (FDDM). User interface design, especially the frame, looks almost exactly like FDDM (it is not included here due to security issue). During this stage, basic DNA profile match can already be performed and results displayed; but using only mock data.
Reverse engineering approach

During Stage 2, almost overall user interface are changed from Stage 1 (compare Figure 2 and Figure 3). The location of menu, icons, and banners are changed, but maintain the same functionality DNA profile match in forensic DNA analysis.
Figure 3: Stage 2 - Second prototype after redesign the user interfaces

Figure 4 shows the comparison of user interfaces design during Stage 1 and Stage 4 for the core process (main engine) of forensic DNA analysis. The similarities are both can be used to perform DNA profile match in forensic DNA analysis. The difference is the software proposed during Stage 4 in this research work applied more user-friendly interfaces. The arrangements of menu are much better than the first software proposed during Stage 1 based on the results of usability study that has been conducted (not included in the scope of this paper). For each allele match, green colour is applied in the result of matching columns, or red if vice versa.
A lot of programming works has been done during implementation at Stage 3 and Stage 4 of this research work following methodology software reverse engineering and software reengineering. Figure 5 and Figure 6 shows the visualization features that has been tried to propose to be included as an enhancement in Web application for forensic DNA analysis. This is because this feature is not provided in current FDDM, hence reverse engineering is not occur for data visualization. The idea here is to find out the way to provide enhancement compared to the existing forensic DNA analysis tools.
Figure 6 shows the final version of visualization using concept visualization tree. In the first version as in Figure 5, the frequency is tried to be visualized using plot graph but it does not really helpful to domain expert compared to visualization tree.

The Web application proposed is then extended into part specifically built to handle forensic DNA analysis. It is implemented during the final stage for enhancement. This paper has presented stages performed for design recovery to identify the behaviour of user interface, nature of processing data and structure of system flow. There are several issues that have been identified:

i) The design of architecture for proposed Web-based DSS was first initialised by process reverse engineering of existing software. This research has shown that an existing software development can be used as a starting point to implement new software from the recovered design.
ii) The unavailability of source code and information in the current documentation is not an excuse for not being able to recover some of the designs from the main system flow. An understanding of the application domain such as software for domain experts like forensic scientists, customers, or any other specific users can help the recovery of information about the purpose of a function and its significance.

iii) Sometimes re-engineering has to be done after reverse engineering because too little or outdated information is provided. There might be a need to rewrite and recode the system to improve its usability, ease of maintenance, or further enhancement with more functionality. It is easier to change a design than source code. Thus, software re-engineering after software reverse engineering will be useful in terms of providing better design and implementation after design recovery.

For AS presented results of analysis with more detail information of DNA allele frequencies. Allele numbers, allele frequencies, locus frequencies, estimated DNA profile frequency, allele matching, locus matching, fractions of locus match and its percentages are very helpful for forensic scientists. Allele and locus match in columns of decision table with systematic and detail data representation is applied with green colour for match and red colour for not match. Systematic and detail data representation of statistical DNA analysis is very important for the ease and quality decision-making process of a DSS.

7 Conclusion

In this research, reverse engineering has been done in a few stages. Stage 1 involving the reverse engineered of user interface. Then, reverse engineering of data structures and algorithms is the crucial part. Statistical forensic DNA analysis for paternity test is applied for main engine of the software. Contribution on the concepts and operational basis for DSS that have been done in this research work is implementing software development process in a Web-based DSS using reverse engineering and re-engineering methods. Applying software RE approach to DSS is one of the methods in developing Web-based DSS, besides just following traditional Software Development Life Cycles (SDLC). The contribution is architecture using RE approach for developing a Web-based DSS. Through RE of software, the user interfaces, the data structures, the algorithms and how the analysis being performed can be recovered. If some basic information of the software is obtained, then further research can be done to obtain more information.

Anyway, reverse engineering approach is successfully applied in this research
The procedure steps in design recovery can be described as;

i) Collect available information: All possible information that is related is obtained as much as possible, including source codes, designs and its documentation if possible. Personnel experience may be included.

ii) Study provided information: Study and extensive reviews must be done on the collected information so that more information can be recovered and further plan can be initiated. The structure of internal and external data can be identified as well.

iii) Sketch-out the windows of system flow: Once understood and identified, the system flow of the observed software needs to be written or drawn. Then, the system flow for the proposed software with enhancement can be provided.

iv) Review again the recovered design: Review and recheck the design to verify that it represents the structure of the software as desired and planned. There is also a need to review the recovered design for consistency with available information and to find out where it can be improved from time to time.

In summary, this research was successfully using approach reverse engineering to build a decision support system for forensic DNA analysis from scratch. The original aim of DSS to use computerized system to support decision makers is achieved. Hopefully, further investigation based on this research methodology for software reverse engineering to build new software using concept decision support system can be practiced. This will contribute to develop and enhance the concept of DSS and its role to help decision-makers utilize information technology for problem solving in future.

References


Received: May, 2012