

Requirements Engineering for Enterprise Applications Development: Seven Challenges in Higher Education Environment

Jamaludin Sallim

Abstract—This paper describes the challenges on the requirements engineering for developing an enterprise applications in higher education environment. The development activities include software implementation, maintenance, and enhancement and support for on-line transaction processing and overnight batch processing. Generally, an enterprise application for higher education environment may include Student Information System (SIS), HR/Payroll system, Financial Systems etc. By the way, there are so many challenges in requirement engineering phases in order to provide two distinctive services that are production processing support and systems development.

Keywords—enterprise applications development, enterprise information systems, business process, requirement engineering, requirement standards, software development activities, software requirement reviews.

I INTRODUCTION

The goal of requirements engineering is to ensure that the process of developing an application is able to succeed in delivering what the client wants, with the least effort in terms of time and cost. Generally, an enterprise application for higher education environment may include Student Information System (SIS), HR/Payroll system, Financial Systems etc. By the way, there are so many challenges in requirement engineering phases in order to provide two distinctive services which are production processing support and systems development. Production processing provides support for the systems' underlying technical infrastructure, overnight batch processing, and security while systems development provides programming support for system enhancements as well as software maintenance.

"It would be a simpler world if requirements stopped arriving when the project had collected up all the requirements it could find and reviewed them with all the stakeholders. In the old days, engineers believed that this was what you did, and they ran projects with phases called User Requirements, Systems Analysis, System Architecture and so on. At the end of the first phase, the requirements were 'frozen' and everyone shut their eyes and hoped they were all correct, complete, and permanently fixed. It is rarely so easy. Even if people all agreed at a formal meeting, changes come along and can quickly increase the cost, timescale, and complexity of a project. Late changes are particularly disruptive, as they usually mean throwing away earlier work as well as adding

new elements. A good project is one that handles change well." [1]

Based on the above statement, it can be concluded that in recent years, systems developers have been adapting to developing and delivering systems and many of them accepted this challenge with reluctance and uncertainty of how they would adjust. This is also happen in higher education environment.

This study is based on author's experiences, observations and findings when studying and working in few universities and companies since four years ago. With reviewing some of respective conference and white papers, a lot of useful ideas had been taken to emerge a fundamental exercise for IT Personnel in the higher education for developing an enterprise application in their campus. The author would also like to thanks to all parties who give a generous and brilliant ideas in producing this paper.

II. SEVEN CHALLENGES IN VARIOUS ACTIVITIES

A. First Challenge: To Understand The Problem Domain

The enterprise application problem has many symptoms: systems that don't work together, too much data and not enough information; incompatible and incorrect data; and excessive maintenance costs. 70% to 90% of a university's IT budget is spent typically on current applications. There are lots of reasons for this, but the primary cause is the way the applications were developed:

- Independently as stand-alone, "stove-pipe" systems -- not designed to be interoperable.
- Not based upon a single enterprise standard
- Without a common data architecture to ensure data sharing
- In unstructured programming languages
- Poorly documented if documented at all
- Designed to reflect developers interpretation of business requirements ten years, twenty years, or even longer ago

All of these approaches have value and some will even provide at least temporary benefit. However, unless they are business-driven and model-based they are more likely to further compound the problems than provide a solution. Misunderstanding and lack of understanding of the domain or the real problem is a major challenge of this case. Any university, as such, the software developers or engineers may

make invalid assumptions and hence create poor requirements or designs. Even the users who are experts in the domain may not possess the kind of knowledge that can be easily communicated to others, and they may define their problem too broadly or too narrowly.

B. Second Challenge: To Identify the Business Process

“Business process” is a potentially wide-ranging term. For the purposes of this paper, a business process is a set of actions facilitating the transaction of business with an external or other internal entity. Although using information systems to automate business processes is much more efficient and cost-effective than employing human auditors, it brings consequences that have yet to be fully appreciated. A basic principle in the business of higher education environment is to eliminate waste from all business and educational processes. A key component in the implementation and management of an efficient environment is the ability to automate standard processes. The IT Personnel will have a tough challenge in providing the power to streamline and automate processes that are within and reach beyond the walls of educational environment using a business process automation engine. Thus to identify the business process for this application development become more challenging.

In higher education environment, Business Processes for enterprise application is identified as sequences of linked functional-level activities, which take inputs and produce outputs. The IT Personnel must emphasize the fact that the description of a business process specifically does not address the way the process transforms its inputs into its outputs - it describes what the process does, not how it does it. This is applied for all cases, in administrative, faculties and student department.

C. Third Challenge: To Emerge the Standards

Emerging the standard is also among big challenges where only the best standard can help the requirements engineering activity and workflow. For this study, IEEE Software Engineering Standard has been reviewed. Adoption of the IEEE software engineering standards is a more efficient and preferable approach. It also represents the most comprehensive and mature set of standards available. However, they may be complex and vexing at times and this is where implementing the IEEE Software Engineering Standards can help. It produces necessary information of the requirements for software systems as specified by either potential customers/users or designers/producers and constituting the substance of an agreement between them. The Software Requirements Specification (SRS) focuses on the collection and organization of all requirements surrounding the project. It provides a complete vision of the As-Is process model analysis and the To-Be process model analysis. As-Is is the requirement study and analysis of the current business processes. To-Be shall be the proposed solution. The To-Be process model analysis will have input from the As-Is analysis and will go through some form of business process. It also

defines all functionality, behavioral requirements, external interfaces, attributes, and performance of the application system. This document shall be used in designing the system. Specifying requirements is recognized as one of most difficult, yet important areas of application development. One particularly critical issue is the lack of real-life examples of requirements specifications. Thus this is the challenge that IT Personnel need consider especially who has practice so many standards at in their development activities.

D. Fourth Challenge: To Analyze and Elicit the Requirements

In principle, the requirements come from users and stakeholders of the system. Part of requirements work is to elicit the requirements from stakeholders. The assumption is that stakeholders have some demands and the role of the analyst is to elicit these demands, analyze them for consistency, feasibility, completeness and formulate them as requirements. The people who are eliciting and formulating requirements are called requirement engineers by some people, analysts by others. In practice, the analysts can be developers, expert users (preferably a team of both), independent consultant, and marketing people and so on. Stakeholders include users with various roles, the customer (who pays for the product), the customer's IT department and sometimes external parties with whom the customer co-operates. If the system is a product offered to a broader market, stakeholders may include the distributors and sometimes software houses adding special features to the product.

Elicitation is a very difficult process for many reasons:

- i. Stakeholders may have a difficulty in expressing their needs, or they may ask for a solution that does not meet their real needs.
- ii. Stakeholders can have conflicting demands.
- iii. Users find it difficult to imagine new ways of doing things or to imagine the consequences of things they ask for. When they, for instance, see the system that has been built for them, they often realize that it does not fulfill their expectations, although it fulfills the written requirements.
- iv. Sometimes there are no users because the product is completely new, and nobody has used IT for this purpose before.
- v. Demands often change over time. When users, for instance, see a smart system somewhere, they may realize that they need something similar themselves. External factors may change too, such as new operating system releases or new laws.

Even when users can express their needs, requirements engineers find it difficult to write them down in a precise way without designing the solution at the same time. The result is that the real demands and the written requirements do not

match. For this reason, it is important for stakeholders to check that requirements meet their demands.

E. Fifth Challenge: To Validate, Verify & Trace

Clients in higher education means top management, academic, staff and students. They sometimes get into knots about the difference between requirement engineering activities, and technical terms are used in different ways by different people. 'Verification' and 'Validation' is one terminology often used, and there is quite a useful distinction, between them in the following senses:

i. Validation

The customer must be able to validate the requirements to see that they correctly reflect his needs. This means that he must be able to read the specification, understand it, and say "Yes, this is what I need. This system will solve my problems." In practice it is good idea to validate intermediate work products, for instance designs of screen pictures, to see that everything still matches the customer's expectations.

ii. Verification

Verification is carried out to check that the product satisfies the requirements. As a minimum, this is done in an acceptance test where the parties go through the requirements one by one and check that the product satisfies them. It is also a good idea to verify that intermediate work products satisfy requirements. Developers as well as customers need to convince themselves that all requirements are being considered during development.

iii. Tracing

Requirements tracing is needed to compare requirements against other information. There are four types of tracing:

- **Forward**

Trace from requirements to program to see that all requirements are dealt with. This is roughly the same as verification. Trace from demands to requirements to see that all demands are reflected in the requirements. This is part of validation, but often neglected with the result that important business goals may be lost.

- **Backward**

Trace from requirements to demands to see that all requirements have a purpose. This is another part of validation. Trace from program to requirements to see that all parts of the program are required. This is neither validation nor verification, but a useful check to avoid feature creep, that is, to prevent developers from wasting time producing things that the customer did not ask for.

- **Court Cases**

If things go wrong, the requirements specification and the contract may end up in court as evidence of what was originally agreed. Many developers believe that if they satisfy the written requirements, they will win the court case. However, in most countries courts don't work that

way. If the customer had reasonable expectations that were not written in the specification, the court will rule that the supplier must fulfill this expectation. In other words, courts acknowledge reasonable tacit requirements.

F. Sixth Challenge: Reviewing Requirements

Software requirement review are an essential component which play a major role in ensuring that every deliverable produced under the IEEE 12207 Software Lifecycle Process is complete, accurate and consistent with the quality and control standards. Given their importance, project reviews cannot be omitted or overlooked without the prior consent of the Quality Reviewer Board which inclusive of XP's personnel and clients who were assigned to the project. The main aspect of requirement reviews covers:

- i. The types of project reviews.
- ii. The scheduling and conduct of project reviews.
- iii. Quality Assurance Process Flow.
- iv. Type of Requirements Review

The requirement reviews can be broadly classified into one of the following groups:

- i. Quality Reviews
- ii. Technical Reviewer
- iii. User Reviews

In general, Quality Reviewers are appointed at the commencement of the project and continue to perform their specific role throughout the project lifecycle. Quality, User and Technical Reviewers are, on the other hand, often appointed during the project according to the particular skills required for the review and to fit in with resource and scheduling priorities. They all, however, play a significant part in ensuring the quality and accuracy of the deliverable items produced and it is therefore important that the correct people are appointed.

1. Quality Reviews

Quality Reviews take place during all phases of the project lifecycle. Their overall purpose is to examine and check each of the project deliverable documents prior to delivery to the customer. For this Payroll project, the quality reviews only include documents of Project Plan, Functional Requirement Documents and Software Requirements Specification.

2. Technical Reviewer

The primary responsibility of the Technical Reviewer is ensuring that technical details within the deliverable documents are technically correct and feasible. As a result, the Technical Reviewer must have a good technical background and be able to use this knowledge to critically assess the deliverables from a technical viewpoint.

3. User Reviewer

The primary responsibility of the User Reviewer is to take an active interest in the Custom Development process and the deliverables that are produced. For this involvement to be beneficial, User Reviewers must have a good knowledge and understanding of their requirements and be able to communicate these effectively.

4. Scheduling and Conducting Review

In normal circumstances, reviews are conducted against each of the deliverables that have been produced by the project team. Hence, it is recommended that the User Reviews be conducted. The most essential review is the final one that provides the user's stamp of approval to the document concerned. Upon receipt of the deliverable document, the Reviewer should read it and identify any aspects that require changing, expansion, clarification, deletion or discussion. Whilst reviewing the document, the Reviewer should note down any issues on the Project Review Issues Forms provided by the document author or Project Manager. As soon as the issues from the review have been resolved, the document author must arrange for the document to be redistributed to the Reviewer. If issues are still not resolved, the review forms must again be collected by the document author for the whole review process to be repeated.

G. Seventh Challenge: To Manage Change and Control

Many higher education institutions have developed various applications for their community uses. This applications contains much of data that includes sources codes, inputs etc. This data will be changed time to time, either systematically or automatically. A clear understanding of how an application that supports a business process has changed is the only way to reconstruct a transaction days, weeks or even months after it has occurred. Every higher education institutions have some standard business processes involving application that contribute to the manipulation and storage of data. This application can include mainframe databases, Web-facing applications, business-to-business processes, and third-party connections, among others. Thus, all presentation, application, and data tiers of the enterprise application architecture that handle many transactions and touch a business process must have sufficient controls. This is also tough challenge for IT Personnel because it's not so easy to manage the control and change. An appropriate technique and tools must be selected in order to manage change and control in enterprise application development.

III. CONCLUSION AND FUTURE WORK

The enterprise application development is complex, especially for higher education environment because it has multiple perspectives, objectives and purposes. With this seven challenge, the IT Personnel in higher education institutions should have clear picture and strategies how to development an enterprise application because the requirement for this

environment is not similar with other enterprises that have so many requirements.

REFERENCES

- [1] Ian Alexander, "The Basics of Requirements Management: Why Projects Have Requirements", White paper, January 2002. Ian Alexander's Home Page. Available: <http://easyweb.easynet.co.uk/~iany/index.htm>
- [2] Ian Alexander, "Are There Requirements for BPS?" Proceedings of Workshop on Requirements Engineering for Business Process Support (REBPS'03) Austria, June 17, 2003
- [3] Donaldson, Scott E and Siegel, Stanley G (2001), Successful Software Development, 2nd ed. NJ, USA: Prentice Hall.
- [4] Ian Sommerville (1995). Software Engineering, 5th ed. United States of America: Addison Wesley.
- [5] Thomas J. Crowe, Krishnakant Rathi, Joseph D. Rolfes, Applying a Taxonomy of Business Processes to Identify Reengineering Opportunities: Conference paper. Department of Industrial Engineering, University of Missouri, Columbia, USA
- [6] IEEE Std 802-1981 (1981), Recommended Practice for Functional Requirement Document, Software Engineering Standards Committee of the IEEE Computer Society, New York.
- [7] IEEE Std 830-1998 (1998), Recommended Practice for Software Requirements Specifications, Software Engineering Standards Committee of the IEEE Computer Society, New York.
- [8] Lauesen, Soren (2002), Software Requirements: Style and Technique. Great Britain: Addison Wesley.
- [9] Roger S. Pressman (2001). Software Engineering A Practitioner's Approach. 5th ed. United States of America: McGraw-Hill Companies.
- [10] Peter Henderson, Yvonne Howard and Bob Walters, "A tool for evaluation of the Software Development Process", *Journal of Systems and Software*, Vol 59, No 3, pp 355-362 (2001)
- [11] Peter Henderson, "Business Processes, Legacy Systems and a Fully Flexible Future", appears in *Systems Engineering for Business Process Change*, Springer Verlag, 2000
- [12] California Software, "Application Servers and Enterprise Application Development", White Paper. Available: <http://www.californiasw.com/techcenter/javaapplicationdevelopment.html>
- [13] ClearNova, Inc. "Enterprise Application Development Challenges: An Overview for Managers" White Paper, Available: <http://itsolutions.forbes.com/forbes/>

Jamaludin Sallim. Author is with the Faculty of Computer Systems & Software Engineering, University College of Engineering & Technology Malaysia. Locked Bag 12, 25000 Kuantan, Pahang, Malaysia. Phone: +609-5492138 Fax: +609-5492000 (jamal@kuktem.edu.my).