



GLOBAL JOURNAL OF RESEARCHES IN ENGINEERING
GENERAL ENGINEERING
Volume 11 Issue 7 Version 1.0 December 2011
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Software as a Service Model

By Dr. S.R. Suresh

DEAN - Computer Applications Bharath University, Chennai

Abstract - The prime objective of various Software development methodologies is to formulate a well defined framework to develop the intended software product. The framework should address various levels in the development cycle. The intended software product should be developed with limited slack time and with high degree of quality within various levels, without draining the resources available at hand. This paper addresses a new method to develop quality oriented software and without compromising the budget parameter. Quality is a major factor defining the demand for the product, customers look for quality at a demanding cost. So designing software at a reasonable cost with a high degree of quality will always conquer the software market. The term quality in software development sector is to design a system to address the requirements of the customer and the needs of the user. Some software fails during the implementation phase due to the lack of information from the users. In usual cases the buyer might not be the user. A clear documentation must be prepared by combining the requirements of the customer with the needs of the customer. The requirements of the customer usually address the logical and business parts of the software. But the needs for the user may vary from the color of the frame to the shortcut keys.

Keywords : *Quality, Software, Cost, Framework, SaS, Service*

GJRE- J Classification : *FOR Code: 890299*



Strictly as per the compliance and regulations of:



Software as a Service Model

Dr. S.R. Suresh

Abstract -The prime objective of various Software development methodologies is to formulate a well defined framework to develop the intended software product. The framework should address various levels in the development cycle. The intended software product should be developed with limited slack time and with high degree of quality within various levels, without draining the resources available at hand. This paper addresses a new method to develop quality oriented software and without compromising the budget parameter.

Quality is a major factor defining the demand for the product, customers look for quality at a demanding cost. So designing software at a reasonable cost with a high degree of quality will always conquer the software market. The term quality in software development sector is to design a system to address the requirements of the customer and the needs of the user. Some software fails during the implementation phase due to the lack of information from the users. In usual cases the buyer might not be the user. A clear documentation must be prepared by combining the requirements of the customer with the needs of the customer. The requirements of the customer usually address the logical and business parts of the software. But the needs for the user may vary from the color of the frame to the shortcut keys.

Keywords: Quality, Software, Cost, Framework, SaS, Service

I. INTRODUCTION

This paper presents a method where we provide software as a service rather than product. This era has made more advancement in the area of information technology and communication systems. Statistics show about 80% of the world is connected through internet. In this modern age, we adapting to the new system of SaaS (Software as Service) technology is not an issue. SaaS provides a constant improvement in quality through various quality appraisal systems thus providing software with high degree of quality in a limited cost model. Here we address the model through which how this system can be implemented efficiently. The core objective of designing a software system is to customize the operations of end users in achieving their core-target in minimum usage of recourses. The core target varies based on the assignment designated to the end-user. The banker who uses a banking application may use a software system to register all the transactions between clients and the bank, an application programmer uses a compiler for creating new software applications. Well performing software uses minimal recourses to reach the target with high degree of precision.

Author : DEAN – Computer Applications Bharath University, Chennai
73 E-Mail : sur1971@gmail.com

In this paper, a new method through which we can achieve a best performing and highly reliable software through the concept of SaaS (Software as a Service) Modeling is briefed. This paper provides an insight of what a SaaS model is and the advantages of using this model in developing this kind of software. SaaS model provides a structural approach in designing a reliable software application. IEEE 610.12-1990 defines reliability as "The ability of a system or component to perform its required functions under stated conditions for a specified period of time. SaaS model provides a highly reliable system and with a longer lifespan since, the development and up gradation process is a continuous procedure through the concept of Quality through Intelligence (QTI).

II. PROPOSED SYSTEM

Various software development framework models have been framed for designing reliable software at a competitive time and with limited recourses. The paper presents here an evolving system that has been adopted in major technological giants to enhance their service to their clients. SaaS concept was published in February 2001 by the Software & Information Industry's (SIIA) eBusiness Division.

The SaaS model provides a waypoint to design a software application through the concept of software as a service. SaaS concept is widely accepted concept and various design models are used to provide a solution for the identified service. By adapting SaaS model the system become more reliable than that of other building models.

The core concept to this architecture is the "Quality Through Intelligence". This is a very special phase introduced in this model with gives a more advantageous background than that of the other models.

Since quality has a broad range of identification with respect to the end user, this system is very suited for creating software as service, since this provides an analyzing environment for continual quality improvement at various levels of usage.

III. OBJECTIVE OF SAAS MODEL

The primary objective of SaaS model is to design a feasible environment to develop software which meets the phrase "Give what the Customer Wants".

Quality: To achieve this objective a new environment is designed parallel to the implementation

and Maintenance phase, "Quality through Intelligence", this phase is a specialized zone where we design a system to learn the user. Automatic feedback is generated based on usage of the end user. Various parameters are analyzed and the system generates a feedback file which is then compared with the other files generated by other users and the resulting information is used to generate useful code to improve the operational quality of the system.

Budget : Since this an automated feedback system, no specialized research and development team is required after implementation. This reduces the budget parameter to a large extent, since almost 40% of the allotted budget is spent on research and development

IV. STAGES IN SAAS MODEL

The various stages of SaaS model are market research, problem definition, implementation plan, QA feedback and testing, customer feedback and Quality through intelligence.

This model introduces a specialized phase "Quality through Intelligence". The other phases are similar to that of the conventional software development scenario with slight modification the procedural structure. The various stages and the control flow are presented in Fig1.

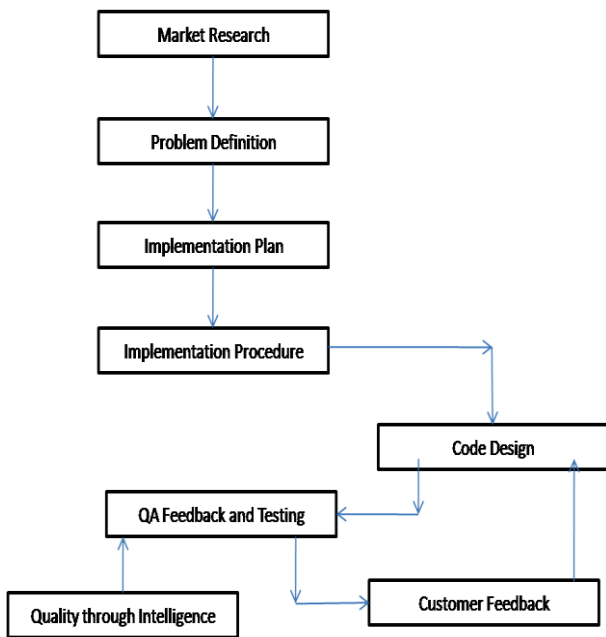


Fig 1: Schematic diagram of SaaS Model

a) Market Research

It is a very important component of business strategy. Market research involves preparing a list of services that are being utilized by various organizations and the depth of service provided is estimated through various management tools. Services involve accounting

service, logistic service, HR service, etc.. The depth of service is the degree to which the user is able to utilize the service and decide upon its output. Tabulation is prepared based upon the service's degree of intensity. Benchmarking is also done with leading service providers of similar service. Upon deriving to a service criterion the system moves to the next stage of problem definition.

b) Problem Definition

This is the stage where the system starts getting into shape. After analyzing various possibilities in the previous stage of market research, the problem is put forth for a proper and structural definition. The identified service through the market research is studied and analyzed the following components are defined.

- 1) Functionality: What the service does
- 2) People: Trained human resource required for designing the service.
- 3) Systems: higher and lower level system goals, interfaces
- 4) Components – various parts and purposes involved in formulating the service.

After structurally formulating the problem for the defined components, the system moves on to the phase of implementation plan.

c) Implementation Plan

Fig 2 : Time line of SaaS model

The time line and the hierarchy of the various stages of SaaS model is depicted in Fig.2. A road map for achieving the desired goal is laid prior to the actual implementation procedure. The road map shows the various milestones and resource utilization as per the requirement specified in the problem definition. Slack time between levels is also specified in the current zone.

The milestones denote the end of a particular phase and the beginning of a new phase. The arrow indicates a continuous process.

d) Implementation Procedure

Once the road map for implementation is clearly laid out based upon the timeline required for each phase, the control is transferred to the implementation procedure. This phase of the SaaS model defines how the total service system is to be integrated into one common service providing unit.

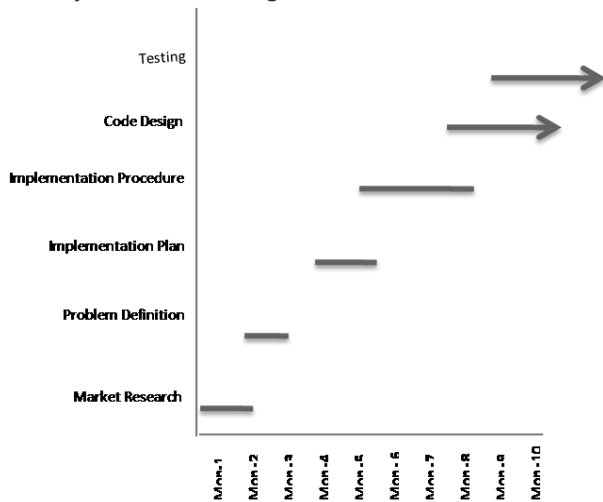
The implementation plan provides documents about the services to be provided and the resources that would be required during the implementation sequence. The implementation documents provide details about the development and integration

procedure for implementing the service. It consists of the business logic, class structures, control flows, Use Cases and other details required for the actual development of the system.

The details prepared in the implementation procedure document are utilized during the coding and testing sequences. Without proper documentation of implementation procedure further quality improvement procedures based on the inputs from the quality through intelligence will not be viable in this model. The whole system should be designed to function and learn at the same time.

e) Code Design

The coding phase encompasses the core part of designing the operational part of the system. Initially a desired language is chosen from the broad array of available programming languages. Selection of a programming language is a very essential part in designing a system. Since there is no universally superior language, each programming language has to be analyzed for its strengths and weaknesses. Selection



of the programming language has a long term implications including those of the business capability, cost and technology. Therefore selection of programming language is a management cum technological decision.

Various dimensions for selection of a programming language are capability, productivity, ramp up, extraneous factors and cost.

- 1) Capability: what the language can/cannot do
- 2) Productivity: How efficiently one can write programs using the language
- 3) Ramp Up: How easily can you get online
- 4) Extraneous factors : Factors that is outside of the programming context

5) Costs: What are the costs involved in using the language

f) QA Feedback and Testing

Parallel to the coding procedure the testing team start their testing work by preparing the test documents for the opted service. Since error free and good quality software service always takes a lead in the highly competitive software market, it is always better to have a good testing document prepared ahead of the testing procedure. To be useful, software testing procedures must encompass all aspects of the software testing process. The steps involved in testing procedures are described in Fig. 3.

It's vitally important that the procedures define the people who will be involved in the testing process, the skill set of each team member, and their availability for the duration of the testing cycle. For the software cycle to remain on track, the software testing procedure must also delineate a carved-in-stone testing schedule including dates of important milestones.

Software testing is a critical component of the software development cycle. And software testing procedures are critical to the success of the testing phase. Software remains in a perpetual state of change which is why software testing, whether manual or automated, is so vital to a software product's success. Typically, 30 – 40% of the software engineers employed at larger software development companies work on software testing and each of these individuals needs to understand his or her role in this ongoing process.

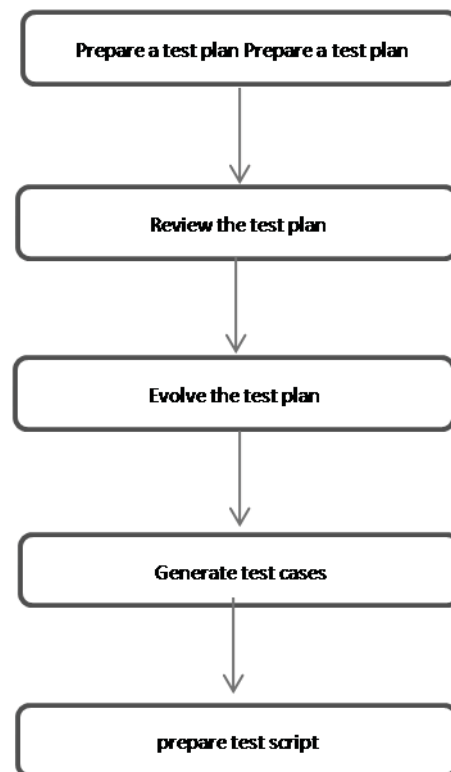


Fig 3: Testing procedure sequence

The quality assurance team keep in track the progress of the development process with accordance to the study conducted during the market research phase. Since the quality parameter varies based on the various levels of users of the service, a clear document is prepared as to monitor that the software service meets the demands without compromising the feasibility of the system.

The following are factors are measures while testing the quality of the system.

- 1) **Correctness**: extent to which a program satisfies its specification and fulfills the client's objective.
- 2) **Reliability**: extent to which a program is supposed to perform its function with the required precision.
- 3) **Efficiency**: amount of computing and code required by a program to perform its function.
- 4) **Integrity**: extent to which access to software and data is denied to unauthorized users.
- 5) **Usability**: labor required to understand, operate, prepare input and interpret output of a program
- 6) **Maintainability**: effort required to locate and fix an error in a program.
- 7) **Flexibility**: effort needed to modify an operational program.
- 8) **Testability**: effort required to test the programs for their functionality.
- 9) **Portability**: effort required to run the program from one platform to other or to different hardware.
- 10) **Reusability**: extent to which the program or its parts can be used as building blocks or as prototypes for other programs.
- 11) **Interoperability**: effort required to couple one system to another

g) Customer Feedback

The goal of designing software service through SaaS model is to achieve a high degree of customer satisfaction. This can be achieved by understanding the customer's needs which can be done through customer feedback system. The feedback from the customers allows the quality assurance team to provide the development team with new solutions to be provided by the services so as to boost the market capability of the software service.

Providing customers with free trial service packages will boost the number of feedbacks, of which the quality team can verify the quality by analyzing the inputs obtained from the customers. By fine tuning the services based upon the customer inputs, we can further enhance the market share of our service software.

Customer feedbacks increases the market potential of the service, because of the useful information provided by the customers the development team know what the customer want and the service is designed to the needs of various levels of service users.

Customer feedback is obtained initially during the release of the alpha version of the software. Since the probability of getting errors is high, usually the feedback is obtained from in-house consultants or experts. Once the beta version is released after the fine tuning of the system, it is delivered to selected community of users for feedback. These stage by stage testing results in bringing more positive feedback stating about the additional features required rather than faults found in the service. After completion of testing of beta version, the final service is released for general usage of customers.

Since we design service oriented software a constant update must be kept in recent advancement in the services. This can either be obtained through customer feedback or through in-house domain experts. The service is modified based upon the inputs obtained from analyzing the customer feedback and expert opinion.

h) Quality through Intelligence

This is a phase unique to SaaS model. This is phase where we design a intelligence system which learns the operations of the system from the users end. This is similar to that of business intelligence. Based upon the customer feedbacks and overall functionality of the system a special module is designed which runs in parallel to service learning the operational mode of various functions. Based on the reports generated by the module the quality assurance team generates solution structure for the desired requirement specified by the learner module.

Various types of specifications provided by the learner module are the time reduction specification, entry reduction specification, frequently used specification and user environment specification.

1) Time Reduction Specification: this functionality learns the time taken to complete each functionality available in the service for various users and identifies the minimum time taken to complete an operation. Based upon the identification it identifies the operational mode of that particular user and reports it to the quality assurance team.

2) Entry Reduction Specification: this functionality learns the entry structure and identifies the correlation between each entry and when a similar entry is encountered it automatically completes the form with minimal entry time.

3) Frequently used: this functionality identifies frequently used functions and creates a short in the main screen to enable quick access to the user.

4) User Environment: this functionality modifies the working environment based on certain parameters like time, gender etc.. this enables the user to work with ease.

V. CONCLUSION

SaaS is an emerging concept which is being adopted by numerous software firms to hold their ground in this highly customer oriented market. SaaS model will definitely bring out a software system with high degree of quality without compromising the cost factor, since its lower levels of production and the specialized learning module in the Quality through Learning concentrates on delivering a customer oriented quality service.

REFERENCES RÉFÉRENCES REFERENCIAS

1. Bin Xu, "A Service Oriented Model for Role Based Global Cooperative Software Development", Zhejiang Gongshang Univ., Hangzhou, Nov. 2007
2. Rodriguez-Martinez, L.C. Mora, M. Alvarez, F.J," A Descriptive/Comparative Study of the Evolution of Process Models of Software Development Life Cycles (PM-SDLCs)", Comput. Sci., Univ. Autonoma de Aguascalientes, Aguascalientes, Mexico, Sept. 2009
3. Ming Huo Verner, J. Liming Zhu Babar, M.A., "Software quality and agile methods", Nat. ICT Australia Ltd., Australia, Sept. 2004
4. McConnell, S.," 10 most powerful ideas in software engineering, Construx, WA, May 2009
5. <http://www.wikipedia.org/>
6. Eric A. Marks and Michael Bell, "*Service-Oriented Architecture (SOA)*",
7. Melvin B. Greer Jr, "Software as a Service Inflection Point: Using Cloud Computing to Achieve Business Agility"



This page is intentionally left blank