

A Review on Software Engineering Practices in SaaS Application Development

Rohan Kulkarni¹, Abhishek Tripathi², Santosh Sharma³, Swathi.J.N⁴

*School of Computing Science and Engineering
Vellore Institute of Technology, Vellore, Tamil Nadu, India*

Abstract— A web application is a utility that is developed to serve a specific purpose in the industry. It is an application that follows a Client-Server architectural model for communication and functioning. This paper is a review of the methodologies proposed for web application development. A general software engineering approach for the development is exemplified first. Secondly, an AGILE based approach for development of web applications is explained in this paper. AGILE is a new methodology for efficient software development being utilized in the industry today. The third review contains a methodology for project estimation for AGILE based web applications. Project estimation has become a cumbersome task today considering the great magnitude of today's web applications. Last but not the least, the automated testing of web applications is presented using a search based software engineering method.

Keywords- Software engineering, Web applications, AGILE methodology, Search based software testing, Project Estimation

I. INTRODUCTION

A web app runs on a browser and is created using browser supported programming languages like HTML, CSS, JavaScript and PHP. Web applications have been very successful because of their ubiquitous nature which makes them easily usable and easy to maintain all without having to install it on multiple workstations. The cross-platform compatibility of web apps also make them popular choices of software development. With the advent of the cloud computing era, many businesses prefer the concept of SaaS (Software as a Service) over the traditional locally installed softwares. This is because These programs allow the user to pay a monthly or yearly fee for use of a software application without having to install it on a local hard drive. Thus owning an application becomes economically affordable and convenient for the users. This has in turn provided an impetus for the application providers to make their application available over the internet as a SaaS application. Hence web based application development has gained a lot of momentum over the last decade or so.

Application development though is a complex job involving a large group of people working over a fixed timeframe and deadlines. Thus this process needs to be streamlined by using Software engineering

methodologies. Incremental delivery, Evolutionary approach, Rapid Application Development (RAD) and AGILE methods are in common use today. This paper contains a review of using some of these methods while developing web applications with improvements suggested to make the process even more efficient.

II. MOTIVATION

There has been a steep rise in SaaS services due to the advantages provided by them such as their easy usage, maintenance, cost effectiveness, low software requirement from the client and better scalability. Similarly a large number of students also prefer web based designs for developing their applications. Thus it becomes necessary to review the current methodology used for developing the same in the industry today and compile the observations into a single compendium so that web application developers benefit from the study of the review paper.

III. PROBLEM DEFINITION.

Today's industry requires development of web applications on a large scale. This development activity becomes cumbersome if not guided by a definitive software development process. There are many generalized software development processes but there are none which are specific to the development of Web applications. Providing a framework for the software engineering process specific to web application development has thus become very important. A comprehensive study about the current trends in software development over the web has thus become necessary in today's scenario.

IV. LITERATURE SURVEY

Software Engineering for Web Application Development by: Samuel Hsieh [1] contains a through methodology for web application development especially suggested for student application development. The paper suggests three main steps in any web application development process - structural design, detailed design and implementation. While structural design addresses the issue of intra and inter page difficulties, the implementation is divided into four stages: client side scripting, server side scripting, implementation of a visual interface and implementation of server side processes without visual interfaces. The paper illustrates these steps with a lucid

example and also suggests ways to make the same system better

A. Problem Statement:

The schema contains the partial database of a student course registration system. The student table contains the details of the student. The enrolment schema records the student roll number and his pass/fail status in the registered subject. The courses table maintains information about all available courses and information related to the same. The problem is to design a web based application that enables a student to register for a course, add a course and drop a course. The system is expected to provide immediate confirmation about every activity to the student as well as maintain his information securely. The relationship between the tables in the schema is found to be one-many as is the case most of the times in real life applications.

B. Structural Design:

In web application development, a module can be a page, a group of pages or a server side script which is not visible to the client or the server as a page. The structural design phase aims at identifying the structure of pages. It also aims at unearthing the functionalities and relationships between the pages in the system. The following are defined as the goals of the structural design phase:

- Pages in the system.
- Functionalities of the pages
- Structure of the pages.
- Relationships among pages
- Decisions on the visibility of the page and need for server side scripting.
- Decision about client side processing.

To decide whether a page requires server side scripting, it is assessed whether the page requires data stored in the server i.e. dynamic data or static data stored in the client side. The pages requiring dynamic data require server side scripting while those using static data are coded on the client side. To reduce the burden on the server side, all pages that can be implemented on the client side are implemented in client side scripting languages. The structure of the user interface is also finalized in this stage. The pages that are linked are represented in a tree like structure for better visualization. The number of links to a page should be as limited as possible. The output of the structure phase is a page layout document which contains all the aforementioned functionalities.

C. Detailed Design:

Generally, a detailed design encompasses writing the pseudo-code and the functions for the system as well as the intra module interaction of the procedures. But web pages also need to keep a record

of some client information. Thus detailed designing for web apps includes the following:

- Detailed layout of pages
- Client side information storage.
- Tasks to be implemented as functions.
- Pseudo code for procedures.

The procedures which are used very often by many pages are created as functions which reduce code redundancy. For example in the above case study, the authenticity of the client request has to be checked many times by many pages, thus it is implemented as a separate function itself. To preserve the client information in the web pages, JavaScript cookies are used in the design.

D. Implementation:

While structural design and detailed design are platform independent, implementation phase is very platform specific. Minute details about feasibility and scalability have to be taken into consideration before starting the implementation. Implementation is followed by White and Black box testing using generated test cases. In Web applications, test cases should incorporate testing of dynamic web pages thoroughly. Four logical steps in implementation are:

• **Creation of GUI:** Web pages like the login page shown in figure 2 should be created by using any development language like HTML. Pages can be aesthetically improved as long as the functionality is maintained.

• **Client side scripting:** This step involves adding functionalities so that the client can interact with the web pages.

• **Server side scripting:** This step involves writing codes to dynamically use data in the web pages if required.

• **Write backend scripts:** Write scripts which are not visible to the client, but are instrumental in enhancing the functionality and the security of the system. For example, for the above design problem, validation scripts are written in this stage.

An Efficient Approach to Agile Web based project estimation by- Ratnesh Lithoriya and Abhay Kothari [2]: This paper addresses the issue of project cost estimation for various web based projects developed using Agile methodology. Project cost estimation is a very difficult task due to changing customer requirements and different individual capabilities. Hence a separate model called AgileMOW has been suggested in this paper which aims to model the process of cost estimation through changing customer requirements as well as environmental constraints. Earlier, cost estimation processes took into account only one parameter ie program code to estimate the completion time for a particular project. But recently, newer models have

proliferated that also account for other factors like financial and strategic planning. There are two types of cost estimation models:

- **Algorithmic models:** Algorithmic models are based upon past project data. The interesting patterns in the historical data are mined and project cost is estimated. Examples include CO-COMO II and Function point algorithm.

- **Non Algorithmic models:** Non algorithmic models are based on expert judgments ,price to win and machine learning. Examples of these techniques in artificial neural networks ,fuzzy logic and genetic algorithms.

E. Agile based web application development:

Developers now a days use high level editors like joomla,dreamweaver, wordpress etc to develop web applications quickly so that their product reaches the market first. But by doing this they compromise on cost estimation and size metrics. The reason for big firms to shift to Agile project development is the urgency to provide a better product at a faster rate using component based software engineering model. Agile methodology promises good quality software with the use of minimum amount of resources and rapid development time. This reduces the cost of development by a large margin. The need for experienced software engineers in Agile development methodology is also very less ,thus further reducing the project cost. Agile methods promote collaboration, teamwork and adaptability in order to accomplish project development. Short term planning is done to incorporate changes in the requirements using iterative development and planning. The current trend in the software industry thus demands a systematic cost estimation methodology for agile development especially for Web based applications as SaaS is the order of the day.

Typical Agile development methods include:

- Extreme Programming
- Scrum
- Lean development
- Adaptive Software development.

F. Challenges in Agile Web development:

Web application development using Agile methods poses new challenges which are different than those that occur in conventional web development paradigms. They are:

- Accurately measuring cost and duration of project.
- Effort estimation is difficult as traditional regression formulae are inapplicable here.
- Schedule estimation cannot be estimated by the conventional cube root relationship with effort.

- Calibration of models is difficult as it cannot be done using past project measures as new COTS are used for every project.

G. Proposed Model for Agile Cost Estimation:

The AgileMOW approach proposed in this paper is a adaptive method which focuses on determining the right attributes for cost estimation of a project every time for a new web based project. In order to understand the attributes that determine the cost of project development, certain inbuilt environmental factors are considered in accordance with the COCOMO II model. This model considers the human/people factor to be more influential than the process factor. But efficient process cost estimation requires that the attributes that the prediction is based upon to be predictable which is not the case here. Other factors like risks, emerging requirements and complicated issues also compound this matter. To address this issue a table is maintained containing the key quality attributes affecting Agile development along with their ratings.

Size estimation: Web projects can be small and moderate sized. This entices the developers to make highly risky decisions due to the lack of efficient size estimation techniques for Agile development methodologies. AgileMOW takes care of size estimation for Agile web development using Halstead equation of volume. The equation is given below:

$$S=(W1+W2).\log_2(w1+w2)$$

where:

W1 and W2: total occurrences of web estimators.

w1 and w2:number of unique web operands estimators.

Effort estimation: The Effort estimation technique suggested in the AgileMOW approach makes use of the traditional regression analysis techniques as well as the more recent expert judgment based analysis for calculating Effort required for the development. The equation for calculating the effort estimation is as follows:

$$\text{Web Effort [E]} = A.\text{prod} [cfwa_i (\text{size})^{P1}]$$

Where:

A: constant

P1: power factor

cfwa: Agile manifesto attributes for Agile web development.

size: number of web objects.

Agile Development Methods for Developing Web application in small software firms. by-Foudziah Ahmed, Fouziah Baharom & Moath Husni[3]:This paper addresses the issue of unorganized application development using Agile software development in small software firms. Small software firms are mostly given contracts to develop web applications since the resources required to developed SaaS applications are

negligible. This paper aims to propose a specific model for web application development in small firms. A comparative study on suitable Agile development methods has been published here that ultimately aims to choose the most effective model for small firms. The study has been conducted in two phases. The first phase studies different Agile development methods with respect to small scale firms while the second study suggests certain enhancements for the same.

H. Suitable Agile development methods for small development firms:

The above paper consists of the data about the number of people required for different types of Agile software development. This criteria has been chosen because, for small firms the most restricted resource is human capital apart from financial constraints. Accordingly the best methods for such firms are found to be scrum and extreme programming as the personnel needed are only 2-10.

I. Enhancements Needed:

The enhancements needed were found out by comparing selected Agile development methods based on the following five criteria:

- **Development process:** The development process should be systematic and disciplined.
- **Project management:** Project management activities like planning, designing, measuring, coordinating have to be performed during the development process.
- **Requirements elicitation:** Requirements must be collected directly from the use/customer.
- **Testing process:** Testing should be performed on all components and the code coverage should be maximum.
- **Design:** Web page design must be simple and quick to implement.

Based on the criteria of development process, it was found that while the methodology of extreme programming was found to be 80% effective, that of scrum was effective only till 20%. On comparing Scrum and XP over project management, it was found that Scrum focused more on this phase than XP. Considering requirements elicitation, both were found to be equally good, but neither focused on requirements reuse and traceability. In the testing criteria, XP uses Test Driven Development (TDD) that ensures all implemented features are tested and hence proves to be better than Scrum. As far as the design considerations are looked at, XP has a clean and simple coding style while Scrum does not have any standard coding style. Thus both XP and Scrum do not have a measurement program to ensure the quality of product and process. Both of them do not cater to requirements reusability and reuse. Scrum does not have a specific testing procedure. Some researchers have therefore suggested the combination of both

these methods to ensure inclusion of management and development practices in one development method.

Enhancements suggested in these methodologies are:

- Combine important XP practices to scrum method.
- Establish a simple requirements repository that will help in requirements reuse and tracing.

Automated Web Application Testing Using Search Based Software Engineering by-Nadia Alshahwan & Mark Harman[4] introduces three related algorithms for efficient software testing. It also introduces a tool SWAT for the same. These algorithms significantly increase branch coverage while testing by 54% and decrease test effort by 30%. Each improvement has been separately evaluated and explained using real world examples in this paper. With internet users having grown by about 400% in the last five years, online retailing has also witnessed a significant growth. For activities like online banking, online shopping, online recharging and bill payment, it becomes paramount to have the customers' faith and confidence in your SaaS application. And for this to happen, the application should be extensively tested for not only the functional requirements, but also for the non-functional requirements. This is the reason why testing is such an important process in the web application industry.

J. Search Based Software Engineering(SBSE):

SBSE is an approach that formulates software engineering problems into optimization problems. It works as follows:

- Possible solutions for the problem are encoded in such a way that similar solutions are clustered together in the search space.
- A fitness function is defined which compares the different clusters of solutions.
- Finally, operators that alter the unsuccessful solutions are selected in such a way that directs the search to a better solution.

Hill climbing and Alternating Variable Method (AVM) are examples of traditional and widely used SBST methods.

K. Test data generation algorithms:

All algorithms proposed here are based on the hill climbing approach using Korels AVM. When a target branch is selected, AVM is used to mutate each input in turn while all other inputs remain fixed. When the selected mutation is found to improve the fitness value, it is selected as the next traversal. This method is also called the exploratory approach.

NMS: Overall test data generation algorithm:

Top level units are extracted from the file tree analyser results. Each unit is called with no parameters

to get an initial work list of reached branches .For each work list branch, the input vector is mutated iteratively until the branch is covered or the stopping criterion is satisfied. Near misses and co-lateral coverage are recorded for a later use.

V.ENHANCEMENTS WE PROPOSE:

While developing SaaS applications ,there may arise many different issues and risks related to the network that the application uses. The risks can be as follows:

- **Injection:** Injection flaws occur when untrusted data is sent to an interpreter as a query. The attacker can take advantage of this to trick the interpreter into executing malicious code.
- **Broken Authentication and Sessions management:** The attacker can use this to compromise passwords and other identification details.
- **Cross Site Request Forgery:** The CSRF forges a fake HTTP request over the victims logged in browser and sends client side information to vulnerable applications.
- **Unvalidated redirects and forwards:** Web applications direct users to untrusted sources without validating the access rights of the URL. This can make the client vulnerable to malicious attacks.
- **Unannounced network failures:** This can cause a serious problem especially in monetary transactions done over the web

Though the development process of SaaS applications is very well explained in the papers reviewed above, none of them propose a specific model for RISK Identification and Management which is of vital importance in developing web applications also. Thus we propose that along with the above mentioned processes, the RISK management feature of the spiral model should be incorporated at every stage of development to identify risks and find possible precautionary and corrective measures for the same.

VI . CONCLUSIONS.

SaaS application development is a major challenge in itself considering the magnitude of the projects in today's world and the risk factors involved. Thus Web application development should be a process governed by software engineering principles to ensure better quality software development for the customer and that too at a quick pace.

REFERENCES

- [1]Software Engineering For Web Application Development by: Samuel Hsieh
- [2]An Efficient Approach to Agile Web based project estimation by-Ratnesh Lithoriya and Abhay Kothari

- [3]Agile Development Methods for Developing Web application in small software firms. by-Foudziah Ahmed,Fouziah Baharom & Moath Husni
- [4]Automated Web Application Testing Using Search Based Software Engineering by-Nadia Alshahwan & Mark Harman.
- [5] Morrison, M. and Morrison, J., *Database-Driven Web Sites, 2nd ed.* Course Technology, 2003.
- [6] Riccardi, G., *Database Management with Web Site Development Applications.* Addison Wesley, 2003.
- [7] Sebesta, R.W., *Programming the World Wide Web*, Addison Wesley, 2002. page 9
- [8] Martin Bravenboer, Karl Trygve Kalleberg, Rob Vermaas, and Eelco Visser. Stratego/xt 0.17. a language and toolset for program transformation. *Science of Computer Programming*, 72(1-2):52–70, 2008.
- [9] Naveen Donthu and Garcia Adriana. The internet shopper. *Journal of Advertising Research*, 39(3):52–58, 1999.