New Approach of Software Development Life Cycle to improve Software Quality Management

Kunal Gupta¹, Naresh Lodha², Radha Krishna Rambola³

¹²³Department of Computer Engineering, SVKM’s NMIMS, Shirpur, India

Abstract— System development life cycle or SDLC for short is an approach for developing, constructing, and managing the information and organizational systems. Every now and then new applications and Software’s are developed which are based on System Development Life Cycle or developed with the use of SDLC. [¹] Through time, SDLC has always experienced various difficulties and disadvantages that has resulted in compelling budget swamp, irregular or suspended deliveries and disappointed customers. Nowadays, the process of developing a software has become more and more diverse and complicated. So to simplify the development process and also to improve the efficiency of an end-user product, it is utmost important for developing the software in a standardized manner such that it will be handed over within the deadline and should also have proper quality. The major problem which occurs in development is that the end-user or customer is unaware of the development process till the testing phase comes. So, this paper gives the methodology to improve the efficiency of Traditional Software Development Life Cycle and proposes a new step in addition to the eight phases which were already present in SDLC.

In this paper, we have described the traditional SDLC method along with its several disadvantages and also proposed the method to overcome those disadvantages. Here, we are only focusing on improving the SDLC phases rather focusing on any of the Software Development Life Cycle Model such as Waterfall, Iterative, Incremental, Spiral or V-shaped Models. [²]

Keywords—Development Models, SDLC, Traditional Method, UI and Function Guide, User Interaction

I. INTRODUCTION

SDLC has eight phases in its development process and each phase contributes to the development of applications. Software Development is based on the various factors such as the type of software, a module included in the software, framework required for development and more.

Basically, the large software system is difficult to develop, but with SDLC, it becomes easier to develop. [³] Also, there are some projects which are more susceptible to come across sudden, unexpected changes than others. Furthermore, some projects can better handle the changes and therefore we require SDLC for the development of Software.

A. What is SDLC?

The System Development Life Cycle is a plan that determines the step by step procedure for developing a software or application. It is a framework that is accompanied by a unit in the software development. It contains a defined procedure providing a detail about how to develop, maintain and replace particular software.

Basically, the SDLC is made of various described and unique work phases which are used by engineers and developers to plan for, design, build, test, and deliver. All these eight phases are taken one after the other and they contain a detailed plan for developing a product in accordance with customers. [⁷]

B. Advantages

- The SDLC creates a formal review at every stage that helps the management to control the development.
- Documentation is very formal and systematic and necessary process that is required to keep the track of the user or business requirement at each stage development meeting.
- SDLC is divided into eight phases which helps to check the software at the intermediate phase and make changes based on the business requirement.

C. Disadvantages

- End-user is unable to see software system between its development, which might result in the user’s dissatisfaction.
- SDLC requires documentation, which becomes expensive as it has to be created or updated after every phase and so is a time consuming process.

II. PROPOSED SOLUTION

Traditional SDLC has eight phases which are used for the development of software. The major problem which occurs in the development is that the end-user or customer is unaware of the development process till the testing phase comes. So, as to make the involvement of end-user, we have added “Design and Function Guide” phase into it.
This phase will involve the end-user or customer to verify the system and propose changes, if any.

III. FLOWCHART

![SDLC Flowchart]

Figure 1: SDLC Phases

IV. SDLC PHASES

Basically, there are eight phases in Software Development Life Cycle and we are including one more phase to enhance the efficiency and credibility of the developed application. Here, we have described all the phases in detail.

A. System Study

The very first phase of development of software system is System Study. In this phase, the developer has to investigate the system. For this phase, initial system study is required to be done and there is a need to prepare the System Proposal. This proposal has various factors involved in it such as the objective of Study, various constraints for developing the System, and also all the study of all terminologies of the system. System proposal also involves the problem definition and need to list the benefits of system. The above mentioned proposal is forwarded to Management for the Review and is updated according to their recommendation. Hence, in this phase, the system goes through the two different major steps they are as follow.

- Identify Problem and create System Proposal
- Background Analysis

B. Feasibility Study

After the acceptance of the System Proposal by the Management, the development of System is proceeded to next phase which is called the Feasibility Study. In this phase, as the name suggests the feasibility of the proposed system is examined. Feasibility study involves the study of various factors which are focused on developing the system. The factors are based on the operational, technical and resources required for development of the System. Feasibility study also includes the Economic and Social Feasibility of System. In feasibility study, the main focus is towards finding the ROI (Return on Investment). The ROI is completely based on the cost and benefits of System. The various factors on which decision made on this phase is as follow.

- Economic Feasibility of System
- Operational Feasibility of System
- Organizational Feasibility of System
- Technical Feasibility of System
- Social Feasibility of System

C. System Analysis

The formal definition of System analysis says that it is the process of understanding and identifying the problems and collecting the Factual Data which helps to understand the system better. The study of various aspect in the System analysis are listed below.

- Studying the Business Process
- Gathering the Operational Data
- Understanding the Information Flow
- Evolving the Solutions
- Identification of Data Store
- Manual Process
- Sub diving of Complex Problem

D. System Design

The most important and crucial phase of SDLC is System design. This phase is completely based on the User’s requirement and the analysis made on Proposal of the New System. The Physical system design is done on the basis of Logical system design which was done in the Analysis phase. The physical system design includes the software and hardware selection and selecting the Programming Language. It is a detailed description which shows inputs, output, forms, databases, Codification Scheme, Data structure, equipment source, workload of the System, limitation of the system, documentation, control
International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 8, Special Issue4, April 2018)

process, training, procedures of using the system, taking backups and staffing requirement [5]. The various techniques used for describing the System design are Data Flow Diagram, Decision Tables, Data dictionary and Decision Trees.

E. UI and Function Guide

After performing the four traditional phases, we have introduced this new phase, to increase the credibility of the software. At this stage, developer will prepare the guide and submit to the end-user or customer, to check or verify their system which is currently in developing phase [7].

This step will ensure that the UI should be good because developer only focuses on Implementation rather than creativity. Also, functioning can be ensured by the customer in between and hence such situations or problems can be avoided which comes after the product is developed.

F. Coding

This is also called an implementation phase. After receiving the design documents from the designing team, actual coding is started and developer aims to implement the functionality required by the customer. This part of development is the main focus for the Engineer and also it is the longest phase of whole SDLC process [8].

G. Testing

After the development of coding part, the code is tested against the conditions given by the customer. Here, a code is divided into certain modules which are checked one after the other and if any inconsistency comes, it is notified to the Developer for ensuring the correctness. In this, a table is prepared consisting of certain parameters which are taken from the customer’s perspective.

H. Deployment

After successful testing, the end product is deployed at customer’s end. At the customer’s end, it is first deployed for Beta Testing where end-users will check and report for inconsistency. This process goes for several months and when all the inconsistencies are resolved, the final product is deployed [9].

I. Maintenance

It is the support phase where the Maintenance team takes care of the final product because when users start using the final product, problems arise and they should be solved as soon as possible. There are chances where the customer wants to update the system or they want any new feature in the system so at that time maintenance team takes care to solve these many issues [6].

V. ANALYSIS

On the basis of the proposed solution, a difference has been noticed in the time management between the traditional method and the proposed method i.e. the time required to develop the software or application is comparatively less in proposed method to that of the traditional method [8].

<table>
<thead>
<tr>
<th>APPROACH</th>
<th>Development Time</th>
<th>Development Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional SDLC</td>
<td>More due to multiple</td>
<td>Based on approach</td>
</tr>
<tr>
<td></td>
<td>Checks</td>
<td></td>
</tr>
<tr>
<td>Proposed SDLC</td>
<td>Less due to Single</td>
<td>Less compared to Traditional</td>
</tr>
<tr>
<td></td>
<td>Check</td>
<td>method</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Traditional SDLC</th>
<th>Developer</th>
<th>End User Interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Study</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Feasibility Study</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>System Analysis</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>System Design</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Coding</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Deployment</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Maintenance</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Traditional SDLC</th>
<th>Developer</th>
<th>End User Interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Study</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Feasibility Study</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>System Analysis</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>System Design</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>UI and Function Guide</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Coding</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Deployment</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Maintenance</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Above tables shows the comparison of end-user interaction with the Software i.e. In Traditional Software Development Life Cycle interaction of user is less as compared to the new SDLC. So our proposed method aims to increase the End-User Interaction. This analysis has been done on the basis of the development of Proxy Cache Server Software. Thus, by using the improved approach mentioned in the paper, the development time as well as cost of the software development has been minimized.

VI. CONCLUSION

In this paper, we discussed traditional SDLC and introduced new step so to increase the efficiency and credibility of the software. We have given the new methodology of developing a software at the System level by introducing one new Step or Phase. This new phase will aim to overcome the disadvantages which were there in traditional SDLC. By using this methodology, the customer will be aware of User Interface, system functionality, and the development process. Our SDLC enables the Customer to track and mitigate the anomalies during the development phase rather solving the problems when it is fully developed. It is difficult to reform the system after the full development so it’s better the problem is solved while it is in the development phase.

References