

# The Novel Life Cycle Model for Component Based Software System Based on Architecture Quality Using KCW Framework

**S. Kaliraj**

Assistant Professor, Department of Computer Science and Engineering, Kongunadu College of Engineering and Technology, Trichy, Tamilnadu, India  
Email: kaliraj.se@gmail.com

**N. Premkumar**

Assistant Professor, Department of Computer Science and Engineering, Kongunadu College of Engineering and Technology, Trichy, Tamilnadu, India  
Email : send2premkumar@gmail.com

**A. Bharathi**

Professor, Department of Information Technology, Bannari Amman Institute of Technology, Sathyamangalam, Tamilnadu, India  
Email: abkanika07@gmail.com

**Abstract**— With the growing size of software's and increasing needs of people we need a complex software system to provide a solution to the many problems among the humans. For that, we need to develop the software system with high quality and limited amount of time. Due to this reason, the software development process is moved to the component based software development (CBSD). The preliminary life cycle models of software development is not compatible with component based software development. For that we need a novel life cycle model to develop the software system using components with high quality. Some of the life cycle models are developed by some researchers, but when we are considering the quality of the software system those models are not assured the highest quality. Because of this we have developed one novel life cycle model based on v-model using KCW framework which assures the quality of the architecture of a software system itself.

**Index Terms**— KCW Framework, Life Cycle Model of the CBSS, Process of Component Based Software Development, Quality Based Life Cycle Model.

## I. INTRODUCTION

Component-based software engineering (CBSE) (also known as component-based development (CBD)) is a branch of software engineering. It is a reuse based approach to defining, implementing and composing loosely coupled independent components into systems[5]. Even though the software system is developed from the components, the development process is very important to maintain stability, control, quality and cost of the software system. The development process of component based software system is distinct from the development process of an object oriented software system [13]. More activities are varied from these two development approaches. Specifically, the implementation process is

varied in component based development and object oriented development. In CBD approach, the software system is developed by using the available existing components. Because of that, the traditional development models are not compatible with the component based software system development. So that we need a specific life cycle model for developing the component based software system. From the last decades some life cycle models are developed for component based software system. But all the life cycle models are not given the assurance to the quality of the software system. According to the ISO 9126 (the reliability is one of the important factors to ensure the software quality)[3]. The architecture of the software system is very important to achieve the quality software system. For that, we concentrate on the architectural quality and also planned to incorporate this in the life cycle process. The V model [4] is a software development process of the component based software system. This is somewhat better than the all other models. Even though this model is better, it's also lacking in component evolution and quality factors. So that, we have developed one novel life cycle model based on the V model with incorporating the component evolution and architecture evaluation using KCW framework [5][13]. The KCW framework is a recently published simple framework to analyze the reliability of component based software system. This will be useful to develop the quality based life cycle model of component based software system. We hope that our model will be used to develop the quality component based software system to fulfill the requirements of industries and peoples.

This paper is organized neatly for the easy understanding of the reader. The motivation of this research is given in section 2 and literature reviews of the

preliminary models are given in section 3. The importance of KCW framework in the reliability analysis of design and condensed form this framework process is given in section 4. The V model and its limitation for the component based software development is explained in section 5 and the proposed model using KCW framework and its importance, a comparison with traditional life cycle is given in section 6. Section 7 contains the merits of our model and the future idea of applying this in real time software development.

## II. MOTIVATION

The motivation of this process model comes from the depth research in reliability models of component based software system. This study enables us to design the robust reliability analysis framework (KCW) of component based software system [5]. After that, we were concentrating on the development process of component based software system because of the absence of a standard life cycle model to develop higher quality software system. After the analysis of life cycle models of component based software system, we found V model [4] is somewhat better than the all other models. Due to that, we modified the V model with a KCW framework to get the complete and standard quality based life cycle model of component based software system. Finally our model has developed by incorporating the KCW framework to know the reliability of architectural design of CBSS and component evolution.

## III. LITERATURE SURVEY

Software development life cycle process is a sequence of steps that are performed in a respective order in order to develop the quality software system. In a component based software system development, the many life cycle models have been proposed for the last decades. Among those models we have chosen some better models to compare with each other for the understanding of limitation of each one. The models for CBSD are Kotanya's Model (2003) [7], Sommerville's Model (2005), Capretz Model (Y Model 2005) [3], Crnkovic Model (V Model 2006) [4] and Qureshi and Hussain's Model (2008) [12]. All preliminary models are having some limitations due to the lacking of some features. The Kotanya's model is lacking in domain analysis. The domain analysis and component evolution are needed in Sommerville model. The V and Y models[4][12] are lacking component evolution. The domain analysis and component evolution is absence in Qureshi and Hussain model. From the above statement, we can conclude that, none of the models having the entire feature to develop the quality software system.

## IV. KCW FRAMEWORK IN SOFTWARE LIFE CYCLE

KCW (Kaliraj Chandru Wahi) [5] is the recently published simple framework which will be used to

estimate the reliability of component based software system from the modular structure of the system[10]. This framework was designed by using the Kal-Chan path selection algorithm [13].

Because of the simplicity of this framework, we can incorporate this into the component based software life cycle to improve the quality of the software system. Because the component based development is based on the design of software components. So that, the new life cycle is designed with the aim of achieving the highest quality software systems using KCW framework based on V model. We hope that, the KCW framework will play the major role in the life cycle process to develop the highest quality software system.

The condensed form of KCW framework [5] is shown in fig. 1. The use of the KCW framework of the modified V model development process of component based software system is described in below section.

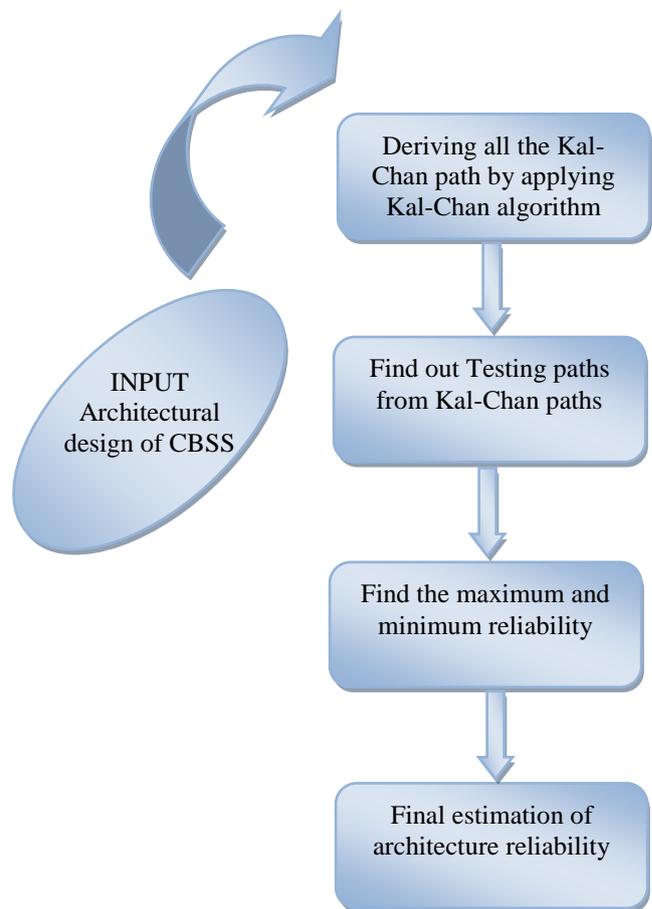


Fig. 1. Condensed form of KCW framework

## V. MODEL FOR COMPONENT BASED SOFTWARE DEVELOPMENT

The V model [4] for the component based software development is designed by the crnkovic in 2006. This V model having all the phases to develop the component based software system except the component evolution.

But when we are following this type of life cycles we cannot assure the quality of the system. Because of this

some modification in the process of V model is required to achieve the highest quality software system. The existing V model is shown in fig. 2.

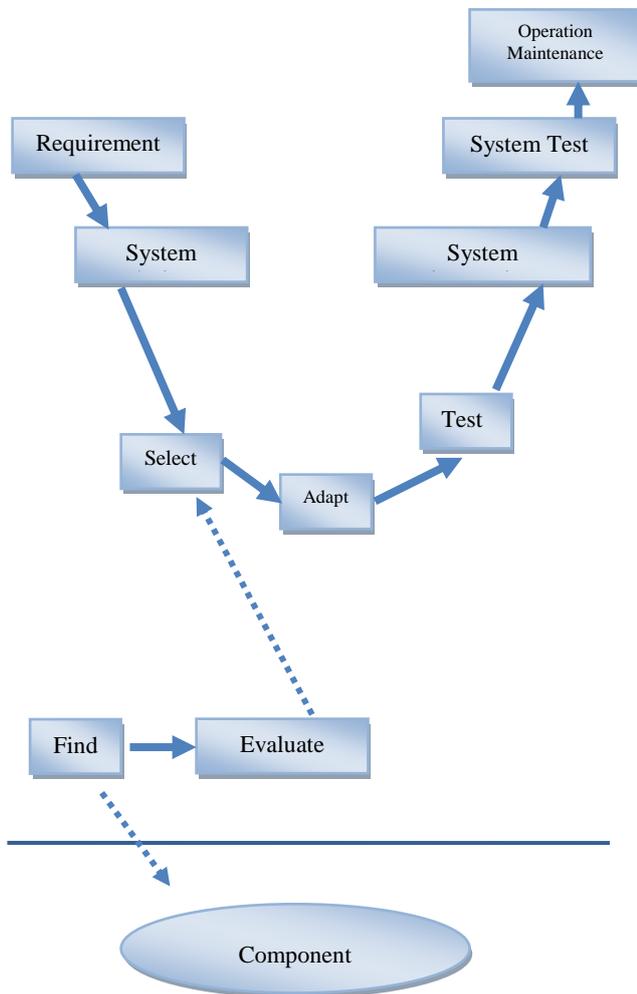


Fig. 2. V model development process of CBSS

In the above V model [4], the component evolution does not exist mean, the up-gradation of the software system using updated components is not considered. This will affect the software system maintenance and also resist the system modification based on the availability of updated components. The modification V model with a KCW framework [5] and component evolution is designed and described in below section.

## VI. PROPOSED LIFE CYCLE MODEL OF CBSD WITH KCW FRAMEWORK

Our proposed life cycle model is focused on the architectural (design) quality of the software system with the intent of achieving reliable and quality software system using KCW framework [5]. The traditional software development process need to know before going for our quality based life cycle model.

### A. Traditional software development life cycle

The general steps involved in traditional software development are,

- Feasibility study
- Requirement analysis
- System design
- Implementation and unit testing
- Integration and system testing
- Maintenance

Here, all the phases contain some subtask. We have given the abstraction level of traditional SDLC.

#### 1) Feasibility study

The feasibility study is nothing but, it is a self-analysis of organization and human resource capability to do the particular project within a particular period of time. This task should be completed before entering into the project.

#### 2) Requirement analysis

The actual requirements of the particular project are defined. The goals and overall specification of the software system is declared.

#### 3) System design

An overall architecture of the software system is established, followed by the abstraction of the software system and the relationships between them are discovered.

#### 4) Implementation and unit testing

The architecture of the software system is converted into the real time software system in the implementation phase. The unit testing takes place after the completion of each unit by the developer.

#### 5) Integration and system testing

After the successful implementation of each unit, all the units are integrated to make an information system, followed by system verification and validation takes place to assess the quality of the system. Then acceptance testing is conducted by the end user before delivery of the system.

#### 6) Maintenance

The maintenance phase involves some activity that is used to keep the system up to date with good performance.

### B. Traditional SDLC Vs Component Based SDLC

The component based software development life cycle model is differentiated from the traditional life cycle model in some phases. The requirement analysis phase of the CBSDLDC takes place by considering the availability of the software components in the market [10]. The requirements may be changed or revised based on the availability of the software components.

So, this phase fully depends on the component availability, not on the customer exact requirements. The implementation and unit testing in the traditional SDLC not happened in the CBSDLDC. Instead of that, the components are selected, evaluated, tested and adopted in the system design. Then system testing followed by operation and maintenance takes place.

### C. Proposed SDLC with KCW framework

Our proposed framework is based on the KCW framework [5] and V model of the component based

software development. Here the KCW framework is incorporated after the system design. This will be used to predict the probability of system design reliability. If the design reliability is less than 0.5[11], then we can redesign the system to improve the quality. This parameter may be fixed by the system tester, it may vary based on the project size and complexity. So, by using the KCW framework we can get a good quality design. It

will automatically leads to the highest quality system. The second thing in proposed life cycle is including the component evolution. This will be used to maintain the system up to date. The component evolution is coming under the maintenance phase.

The proposed life cycle of the component based software development is shown in fig. 3. The explanation of all the phases of our proposed SDLC is given below.

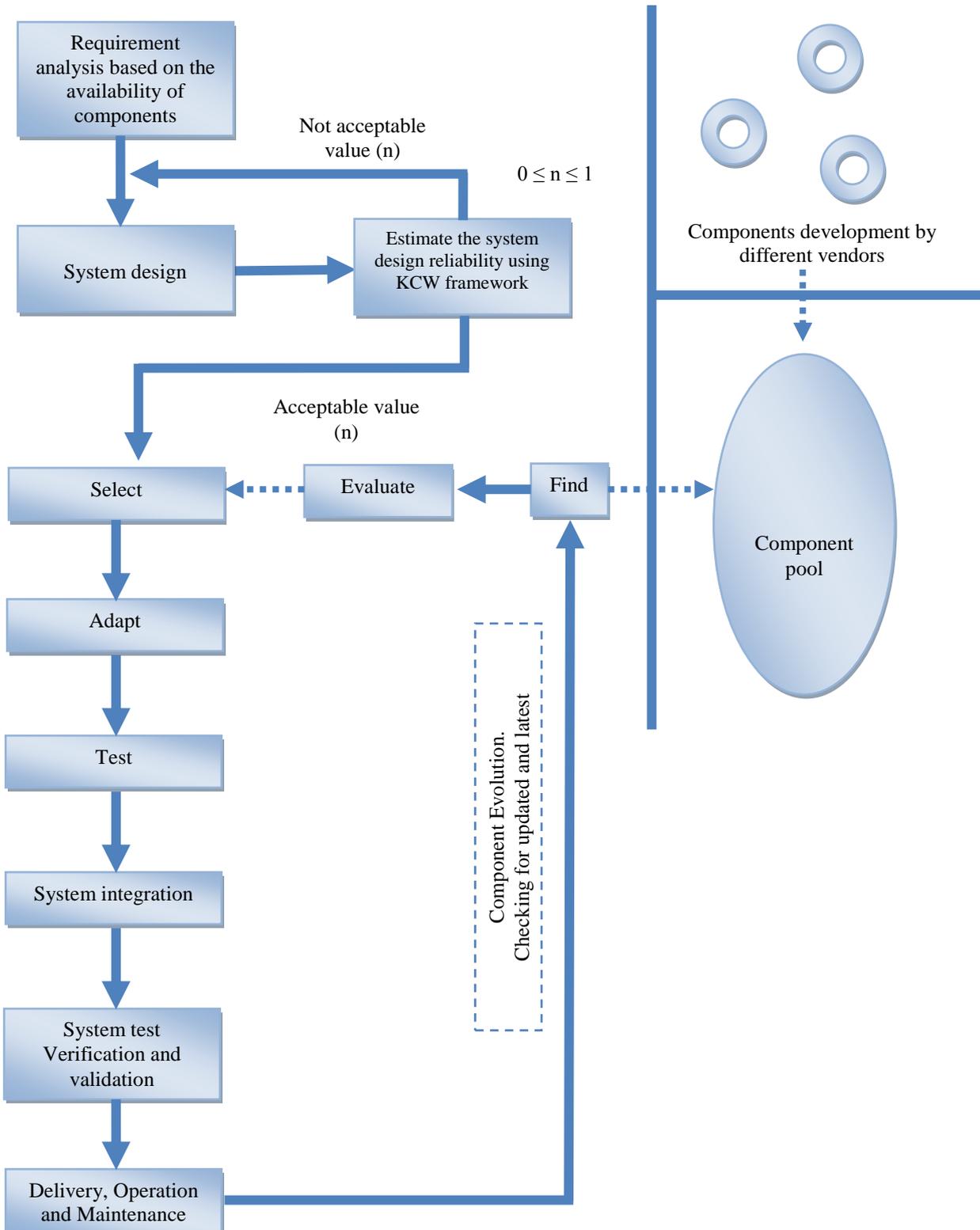


Fig. 3. Proposed life cycle model with KCW framework

### 1) Requirement gathering based on the availability of components:

Similar to the traditional software development life cycle, the requirements gathering comes after the feasibility study. But the difference here is, the requirements are defined based on the component availability in the market. Because, mostly the components are from the third party vendors. The source code of the components are not available to the developer of the software system. They cannot modify anything in the component, they have to use the components with the specification what given by the vendors.

### 2) System design:

The system design of the software system in CBS DLC is similar to the traditional software development life cycle. The overall architecture of the software system is established based on the requirements [12]. The component will be adapted to this architecture in next phase instead of implementation and unit testing in traditional SDLC.

### 3) Reliability estimation using KCW framework:

The reliability estimation of system design is calculated by the KCW framework [5]. If the probability value of the system design is acceptable, then next phase is executed. In next phase components will be selected and adapted to the system design. If the value of the system design is not acceptable, then go for the redesign. Generally, this probability value varies based on the number of components and size of the software system.

### 4) Select, Adapt, Test:

After the validation of the system design, the components are selected from the component pool, which will be evaluated, tested and adapted to the system design.

### 5) System integration:

The system integration is taking place after the successful adaptation of software components. This phase is similar to the system integration in traditional SDLC.

### 6) Verification and Validation:

The verification and validation process in component based software system is varied from the traditional SDLC. The components are tested and corrected individually. But when we are integrating all into the system, it will reflect some errors based on interfaces [10]. The error in one component may be caused by some other components. So, separate testing activity is needed for component based software system [6].

### 7) Delivery, Maintenance and Component evolution:

After the successful completion of a software system, the software product will be delivered to the end user. Then maintenance phase takes place after the delivery to the customer. The overall system performance is monitored. The component evolution is very important to keep the updated system. During this phase, the availability of the updated software components is checked in component pool.

## VII. CONCLUSION AND FUTURE WORK

We have designed this quality based life cycle, using KCW framework [5] with incorporating component evolution. Our model is very simple and may be applicable to the most of the component based software system development. The KCW framework of our model is used to maintain the system quality and component evolution is used to maintain the system up to date. The preliminary models [4][12] do not assure the quality of the system and also not considered the component evolution. So, we hope that our model will be useful to develop the quality component based software system and also maintain the system up to date. In the future, we have planned to apply this life cycle model to two real time software systems to know the efficiency of our model compared to other model.

### ACRONYM

CBSD	Component Based Software Development
CBSS	Component Based Software System
CBSE	Component Based Software Engineering
SDLC	Software Development Life Cycle
CBS DLC	Component Based Software Development Life Cycle
KCW	Kaliraj Chandru Wahi
CBD	Component Based Development

### REFERENCES

- [1] Ivica Crnkovic and Magnus Larsson (editors), "Building Reliable Component- Based Software Systems", *Artech House Publishers*, ISBN 1-58053-327-2, 2003 (1)
- [2] Morisio M., Seaman C. B., Parra A. T., Basil V. R., Kraft S. E., and Condon S. E., "Investigating and Improving a COTS-Based Software Development Process", *In Proceedings, 22nd ICSE*, ACM Press, 2000. (2)
- [3] Capretz, L.F., "Y: A new component based software life cycle model", *Journal of computer science*, 1.1: 76-82. 2005. (3)
- [4] Crnkovic, M. Chaudron and S. Larsson, "Component based development process and component life cycle," *In Proc. of International conference on Software Engineering Advances*, 44, Oct. 2006 (4)
- [5] S. Kaliraj, N. Chandru and Dr.Amitabh Wahi. "A Reliability Analysis Framework of Component Based Software System Using Kal-Chan Path Selection Algorithm". *International Review on Computers and Software*, vol. 8, n. 2, pp. 605-612. Feb 2013. (5)
- [6] S. Kaliraj and N. Chandru, "Introducing path selection algorithm in reliability analysis of component based software system", Presented at 6th Int. Conf. Quality, Reliability, Infocom technology and Industrial technology management (ICQRITIM 2012), University of Delhi, India, Nov. 23-25, 2012. (6)
- [7] G.Kotonya, I.Sommerville and S.Hall, "Towards a classification model for component based software engineering research," *In Prco. of the 29<sup>th</sup> IEEE Euro micro conference*, 01-06, Euromicro, 43-52. Sept, 2003. (7)
- [8] M.R.J. Qureshi and S.A. Hussain, "A reusable software component based development process model", *Advances*

- in *Engineering Software*, Elsevier Ltd, Amsterdam, The Netherlands, 39.2. 88-94. 2008. (8)
- [9] K. Goseva-Popstojanova and M. Hamill, "Architecture based software reliability: Why only a few parameters matter?," in *Proceedings of the 31st International Computer Software and Applications Conference (COMPSAC 2007)*, Beijing, China, pp. 423-430. Jul 2007. (9)
- [10] Bass L, Clements P, and Kazman R, "Software Architecture in Practice", Addison- Wesley, 1998. (10)
- [11] A.Bessam, M.Tahar Kimour and A. Melit, "Behaviour metamodelling for software architecture description", *International review on Computers and Software*, vol. 2, n. 6, pp. 602-608. Nov 2007. (11)
- [12] Capretz, L.F. and M.A.M. Capretz, "Object Oriented Software: Design and Maintenance", *World Scientific Press*, 1996. (12)
- [13] Royce, W.W., "Managing the development of large software systems," *Proceedings of 9th IEEE International Conference on Software Engineering*, pp: 328-338. 1987. (13)
- [14] K. Goseva-Popstojanova and K .S. Trivedi, "Architecture based approach to reliability assessment of software systems", *PerformanceEvaluation*, vol. 45, no. 2/3, pp. 179-204. Jul 2001.
- [15] Garland D, Allen R and Ockerbloom J, "Architectural Mismatch: Why Reuse is so hard", *IEEE Software*, Vol.12, n. 6. 1995.
- [16] Chao-Jung Hsu and Chin-Yu Huang, "An adaptive reliability analysis using path testing for complex component based software system", *IEEE Trans. Reliability*, vol. 60, pp. 158-170. Mar 2011.
- [17] W. L. Wang, D. Pan and M. H. Chen, "Architecture-based software reliability modeling", *Journal of Systems and Software*, vol. 79, no. 1, pp. 132-146. Jan 2006.
- [18] V. E. Johnson, A. Moosman and P. Cotter, "A hierarchical model for estimating the early reliability of complex system", *IEEE Trans. Reliability*, vol.54, no. 2, pp. 224-231. Jun 2005.
- [19] K. Kaur and H.Singh, "Candidate Process model for component based software development" *Journal of Software Engineering*. 4.1. 16-29.
- [20] Khulood Salem Albeladi, M. Rizwan Jameel Qureshi, "Improvement of Component Integration Testing Technique" *International Journal of Information Technology and Computer Science*. vol. 8, N.5, pp.109-122, July 2013.
- [21] Mohd. Anjum, Md. Asraful Haque, Nesar Ahmad "Analysis and Ranking of Software Reliability Models Based on Weighted Criteria Value" *International Journal of Information Technology and Computer Science*. vol. 5, N.2, pp.1-14, Jan 2013.

#### Authors' Profiles



**S. Kaliraj**, got an M.E degree in Software Engineering from Bannari Amman Institute of Technology, Sathyamangalam, Tamilnadu, India. He is currently working as an Assistant Professor in the department of Computer Science and Engineering at Kongunadu College of Engineering and Technology, Trichy, Tamilnadu, India.

His research interests are Software Engineering, Component based software development, Reliability models of Component based software system and Cloud computing. He has published

his first research paper in 6<sup>th</sup> International Conference on ICQRITTM-2012 at University of Delhi, India followed by he published a novel framework for reliability analysis of CBSS in International Review on Computers and Software which has the highest impact factor. (IRECOS, Italy).

Mr.S.Kaliraj is a Reviewer of International Journal of Engineering Research and Technology and also the International Journal of Engineering Science and management.



**N. Premkumar**, got an M.E degree in Computer Science and Engineering from J.J College of Engineering and Technology, Trichy, Tamilnadu, India. He is currently working as an Assistant Professor in the department of Computer Science and Engineering at Kongunadu College of Engineering and Technology, Tamilnadu, India.

His research interests are Software Engineering, Software development process model, Cloud Computing, Software Engineering in Green Computing.



**A. Bharathi**, got a Ph.d in Computer Science and Engineering from Anna University, Tamilnadu, India. She is currently working as a Professor in the department of Information Technology at Bannari Amman Institute of Technology, Sathyamangalam, Tamilnadu, India.

Her research interests are Software Engineering, Data Mining and Software Project Management. She has more than twenty five publication in the reputed journals and conference proceedings.

Dr.A.Bharathi is a member of ISTE, IAENG and also she is Reviewer of International Journal of Computer Science and Electrical Engineering and IEEE International Conference on Electronics and Optoelectronics.

**How to cite this paper:** S. Kaliraj, N. Premkumar, A. Bharathi, "The Novel Life Cycle Model for Component Based Software System Based on Architecture Quality Using KCW Framework", *International Journal of Information Technology and Computer Science(IJITCS)*, vol.6, no.9, pp.74-79, 2014. DOI: 10.5815/ijitcs.2014.09.10