Trend of Component Based Software Development and Traditional Software Development in Peshawar

Ayesha Tariq, Dr. Samad Baseer Khan

Abstract— Component based software development (CBSD) is attaining a lot of popularity day by day due to fast software development as compare to traditional software development. In this competing world, customer and developer wantsfast product development and rapid delivery. For that, CBD is one of the best option. Traditional approach is not any more fascinating to software developers. Using pre-existing components is a better option for developing software instead of coding each component from scratch.These are already components, so there is less chance of error or failure. It saves time, cost, reduces budget and delivers efficient software. Through distributing questionnaires, data is gathered from software houses. This paper shows the current trend of CBD in Peshawar region. In addition, it indicates reasons of adopting CBD by the developers.Moreover, it proves that software developers of Peshawar also agree on using pre-existing components reduces development time, cost, and improve efficiency.

Index Terms— Component based software development, Cost, Efficiency, Traditional software development, Time consuming.

1 Introduction

Developing any software from scratch is a hectic task. Because traditional software development includes many phases from requirement gathering to the maintenance phase. And each phase requires effort. Reference [1] illustrates that developing software from scratch takes large development time. As compare to traditional software development the approach is becoming very famous which is called component-based software development (CBSD) or component-based software engineering (CBSE). In component based development (CBD), software development becomes easy because there is no need to start development from scratch. In CBD, software are developed through already existing components.

It is mentioned that the idea of componentizing software became visible with Douglas McIlroy's address at the NATO conference in 1968. The repaid growth of CBD made possible to develop software in less time and cost. According to Fred Brooks CBSE embodies the “buy don’t build” philosophy. The main aim of CBD is to develop high-qualitysoftware with low cost and less time. Similar to traditional software development, in CBD first requirement elicitation is done then architecture design is formed. After architectural design before moving towards detailed design, requirements are again examined to make sure that whether architecture meets the requirements or not. If requirements are difficult to implement with the available components then requirements are revised. If it is not possible to change the requirements, new components are developed using conventional engineering methods. Further activities initiate for those requirements which can be implemented using available components. And the activities are Component qualification which ensures quality factors like performance, functionality, reliability. Component adaptation to change unwanted features of components. Component composition to assemble components with suitable infrastructure. And component update to change existing software as different versions of components become accessible. low cost with high-quality software development in less time are factors due to which CBD is gaining importance. These components are already tested and developers have to choose the appropriate components for software development. Reusable components save the time of development, save the cost of development, less chance of errors and less effort is requires [2].

Reference [3] stated that the CBD approach can be used for developing software within the country and across the globe. And the term used is global software development. Different components of the software are developed at different locations and then integrated together in the end. According to [4] due to less labor cost and less development time this technique is better. The development team has less work burden. But there are many challenges which should be investigated. Some most important challenges are communication, coordination control and trust with the key stakeholders. Causes of these issues are culture, language, and time zone diversity.As said by authors [5] software can be delivering within reasonable time using CBD.

According to authors [6], in today’s techonolgical world every organization needs a software for running business. Most of the customers want less sofatware price and conventional approach is not achieving such low cost software with high quality. One of the way for improving quality can be the use of components. Initially, when CBSE was developed, its focus was end-user computing (EUC) means developing applications on personal computers. Use of COTS are increasing. Also, Java and web technologies are using CBD. Active X, CORBA and JavaBeans are some examples. CBD contains module based architecture and software is developed in incremental way.

2 Literature Review

Nowadays all developers want to develop their software products in minimum time and in less cost. The approach through which these goals can be achieved is "Reuse". CBD allow developers to use preexisting component to develop software in less time and less cost. In CBD, there is a difference between developing reuse components and developing a system using reusable components. For example, Toyota Company build different components like brakes and engine. These are single components where the focus of component-based system is to integrate these single components to build a car. Different models of CBD are their which can be used during development. Software engineers are working and introducing new life cycle models specifically for CBD, to improve overall development process. One model is called Y model proposed by Capretz for CBD. This model consists of different phases. Some new phases are introduced in this model. These phases are: domain engineering, frame working, assembly and archiving. The focus of this model is reusability also new components are developed for future use. Another model which is used for developing components or system using CBD is known as W model. Main phases of this model are component design and the other is component deployment. Another model proposed by Lata Nautiyal etal., is Elite Life Cycle Model (ELCM) and during development the main focus of the model is also reusability. The difference between these models and traditional life cycle model is that model developed for CBD have reuse feature explicitly [7].

Aim of CBD is to integrate pre-existing components for developing new software. CBD produce software in less time and cost also it reduces overall effort and burden. CBD approach follows selecting components then using technique whereas non-CBD approach follows designing and testing method. Fuzzy logic based retrieval technique was introduced for reusing components and the purpose was to retrieve specification based components using Fuzzy interface. Other component reuse technique includes semantic-based retrieving model, automated component retrieval and adaption model, characterization and architecture of component based model and component classification and retrieval using data mining. CBD also maintains component repository, which is used to stock those components, which are very common in use. Components can be find in repository through search feature [8].

Authors [9] said that, selecting suitable commercial off the shelf (COTS) software components is a main goal of CBD. Fast software production is possible through CBD technique. Whereas traditional development take large time. Traditional approach is different from CBD. In CBD, software components are globally developed. Development process of CBD is also different. It includes selecting correct COTS components during requirement phase. For CBD cost model was developed known as COCOMO II. Various versions of this model also exist. Using this model developers prepare plans for projects related to effort, schedule, and cost. System integration, infrastructure software projects and application generator are estimated through three stages provided by COCMO model. Extensive form of COCOMO II is constructive COTS (COCOTS) model. This model is not able to estimate effort; it is developed for cost estimation. So using CBD overall effort and time can be reduce and CBD is one of the best approach for software development.

3 Methodology

3.1 Data Collecting Method

For collecting data, questionnaires were prepared. First pilot testing was performed on this questionnaire after that these questionnaires are distributed in software houses of deans center Peshawar.

3.2 Population and Sampling Procedure

Research region for this research paper is KP but due time limitation, questionnaires are distributed in 30 registered software houses of deans center. Total 150 questionnaires were distributed. 137 questionnaires were returned out of 150. Out of 137, 7 questionnaires were discarded due to bad writing and incomplete questionnaires.The collected data is analyzed by using statistical package for the social sciences (SPSS) tool. All data collected through questionnaires are entered in SPSS to construct tables and graphs.

4 Results And Discussions

Firstly, we have discovered that software developers of Peshawar are using CBD approach or traditional approach. Fig. 1 shows that in Peshawar more than 80 percent developers are using CBD.

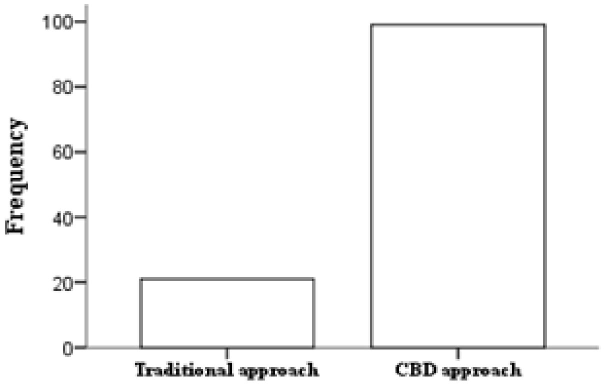

Fig.1.Which approach is mostly adopted in your organization for development?

The reason for selecting CBD approach is to develop efficient software. Fig. 2 confirms the positive responses of developers in favor of CBD.

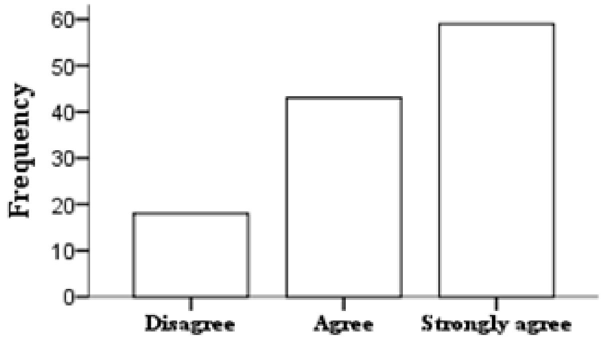

Fig. 2. Efficient software can be deliver using component based development

Moreover, traditional approach develops software from scratch whereas CBD involves pre-existing components and these are an already tested component that is why they are more efficient. Similarly, pre-existing components reduces overall burden because developers have to select appropriate component instead of coding and developers of Peshawar agree as shown in Fig. 3.

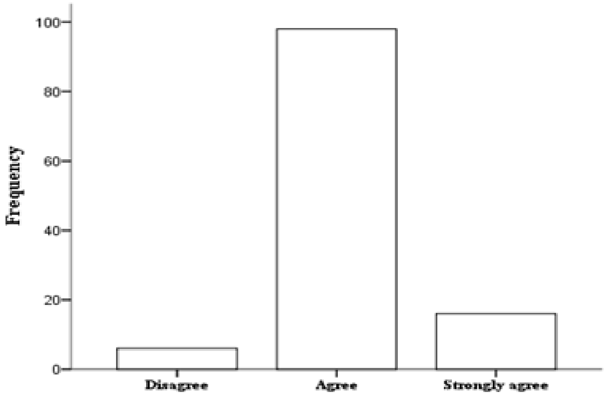

Fig. 3.Component based development reduces team burden

As developing software using per-existing components involves less or sometimes even no coding which eliminates huge coding cost. Fig. 4 shows that 85 percent of total developers agree with the fact that reusable components reduces overall budget.

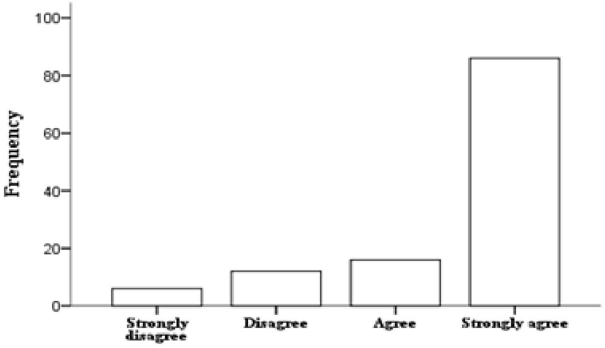

Fig. 4.Reusable components reduces overall budget

According to data collected, 95 percent developers agree with the statement that starting development from scratch is a time consuming task as shown in Fig. 5.

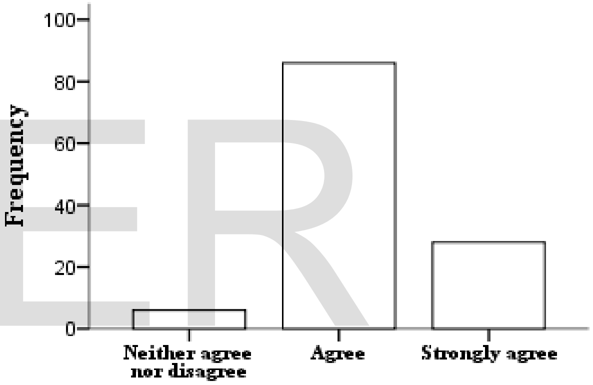

Fig. 5. Development of and software from scratch is a Time-consuming task

In Fig. 6, developers also agree on integrating CBD into software engineering curriculum so that graduates know about the importance of CBD in software development because still there is a need to know about different perspectives and processes of CBD.

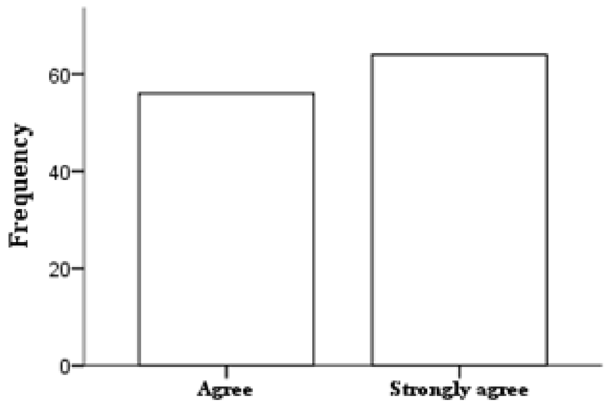

Fig. 6. CBSD should be integrated into software Engineering curriculum

5 Conculsion

In CBD pre-existing components are used which may involve importing components across the globe. Data collected from Peshawar region indicated that 82.5 percent developers prefer CBD as compare to non-CBD because non-CBD is a time-consuming approach.Using CBD; efficient software can be developed in less time and cost.Whereas traditional software development is a long time-consuming task. Furthermore, CBDshould be taught in universities to software engineering students so that fresh graduates got the idea of the importance of CBD and its process.

References

[1] D. G. T. Anshula Verma, "A Comparative study of Traditional and Component based software engineering approach using models," *International Journal on Recent and Innovation Trends in Computing and Communication,* vol. 2, no. 9, pp. 2676-2681, 2014.

[2] D. BOSE, "Component Based Development: Application in Software Engineering," *INDIAN STATISTICAL INSTITUTE,* 2010.

[3] A. K. M. A. a. M. K. Khalid Khan, "Quality Assurance Assessment in Global Software Development," *World Applied Sciences Journa,* vol. 24, no. 11, pp. 1449-1454, 2013.

[4] M. Tariq, "The Context of Global Software Development: Challenges, Best Practices and Benefits.," *Information Management & Business Review,* vol. 3, no. 4, pp. 193-197, 2011.

[5] P. J. Clemente, "Aspect component based software engineering.," in *The Second AOSD Workshop on Aspects Components and Patterns for Infrastructure Software ACP4IS* , 2003 .

[6] M. Aoyama, "New age of software development: How component-based software engineering changes the way of sofware development?," in *International workshop on CBSE*, 1998.

[7] N. -u.-Q. ,. U. A. K. Asif Irshad Khan, "An Improved Model for Component Based Software Development," *Scientific & Academic Publishing,* vol. 2, no. 4, pp. 138-146, 2012.

[8] N. K. Mohan, " Component Reuability for Component Based Software Engineering.," *International Journal of Computer Science and Telecommunications,* vol. 4, no. 1, 2013.

[9] H. H. Khan, " Factors generating risks during requirement engineering process in global software development environment.," *International Journal of Digital Information and Wireless Communications (IJDIWC),* vol. 4, no. 1, pp. 63-78, 2014.