RESEARCH PAPER
PROTOTYPE TOOL FOR REUSE SUPPORT SYSTEM

Amir Ansari
Software Engineer
CTB Solutions
Puchong, Selangor, Malaysia

Abstract: ***The software industry has developed many applications which are using in various platforms for the different work and also huge numbers of application available and it has a great impact of human's life but now time is changing so the users' demand are also changing they want some advanced features in their existing application like WhatsApp every day add some additional new features in the software and attract to the users so company also cannot put any limitation on their work because they know in future some extra component will be needed. The software component is methodology which is playing an essential role and here developer can add any feature in existing software at any time without repeating the same code again and this methodology is also called reusability which is the first choice of the IT expert and now most of the companies are developing product using this methodology because It saves much time and money and can deliver products to the users in less time but it is not easy sometimes it fails because of lack of resources or knowledge and software industry lost much money. Component Based Software Engineering (CBSE) is also another methodology which is frequently used to reused the component with an application and it provide methods, models and guidelines etc. The CBSE has reduced the risk to reuse the component and provides the best platform for the developers who are working reusability methods.***

1. Introduction

The reused methodology has given many ways to implement one component with any application easily. The effectiveness of a component can be analysis by its reusability and portability. A component based software development saves much money and human efforts but if a component is too complex to implement in further development, than it is not be called a Worthy Component (Muthu Ramachandran, 2014). These components can be implemented with many another server-side application only some changes will be required. CBD(Component Based Development) architecture allows the components to intercommunicate with each other and this make them interoperable (M. Rizwan Jameel Qureshi, 2007). The interface is the most important because of this user will interact with a component and the component express themselves through interface. If an interface changed the user needs to know that how to use new version of it. If the API change the user should recompile the code (Ivica Crnkovic). Online System has changed the users 'view and now they want everything online so the online sim recharge will help those users who are addict to online recharge. This system will help to the users and it is different from other online system because in this system there is no need to register themselves it is only for guest user after giving some information recharge will be successful. This component have different sub component which can be reused individually with the project as whole component without developing same component. The Development of "**Case tool**" component which is the combination of many items and these items can be used individual with another application depending on the project requirements.

2. Literature Review

2.1. System Description

The development of **"Case tool prototype** "which include many menu-items, sub menu-items with a Component Framework Tool and these items or whole component can be reused in future with another applications using CBSE methodology. Another part of the project is development of "**online sim recharge system**" which is a component and can be reused with any server side application where it will be needed. The best thing of the system is that there is no need to provide more information during recharge process only give few information and recharge will be successful.

2.2 Development for reuse and with reuse

Software Engineering is a set of disciplined activities that are based on well defines standards and procedures. In Software Design we use guidelines that help us to identify a suitable design criterion when faced with de- sign decisions (Ramachandran, 2012). There are two aspects of reuse components

(i) Development for reuse: This component is developer side the main objective of this aspect of reusability is that when the developers are going to develop some application they should develop application such a manner which can be reused in future with the different software as a component of the project.

(ii) Development with reuse: This is not an easy task because developers have to analyze component from the given applications and try to implement with the different projects and try to maximize the use of existing components. In this aspect developer use code, design etc. This component can be developed by same developer who are going to implement with the project or may be different developer.

2.3 Component for reuse

This is the development of two application first one "Case tool" which have menu, sub menu-item and it can be reused with another application menu-items, sub menu-items or as a whole component and another one is development of "online recharge system" which will give an interface and can be reused with any server side applications. It also provide sub items and can be reused individually depends on the project requirements.

3. Research Design and Methodology

1. UML Modeling for Component

The UML diagram will show the exact accessibility of components and there are two sides first one is user side and another one is developer side which will reused the existing components with a new component or in same manner develop a component and try to implement in a new project. This diagram will show that when and how component can be reused.

1.1. Use Case Diagram

There are two use case diagrams first one is to access the project component and reused with the student form and another one is the online sim recharge form and how the users can access these two components frequently and it can be clearly seen by the below diagram.

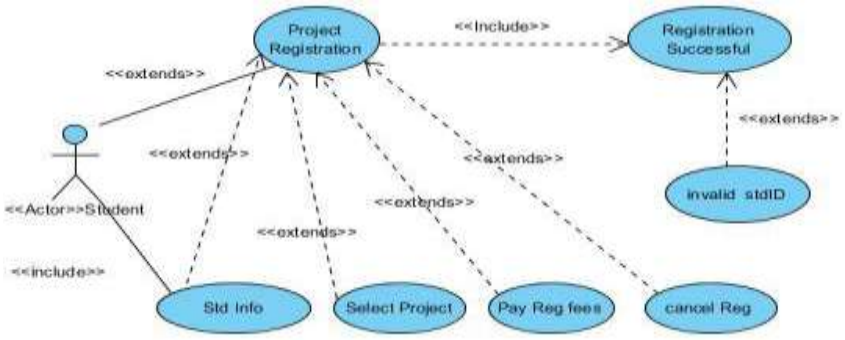

Fig 1.1 Use case diagram project component reusability

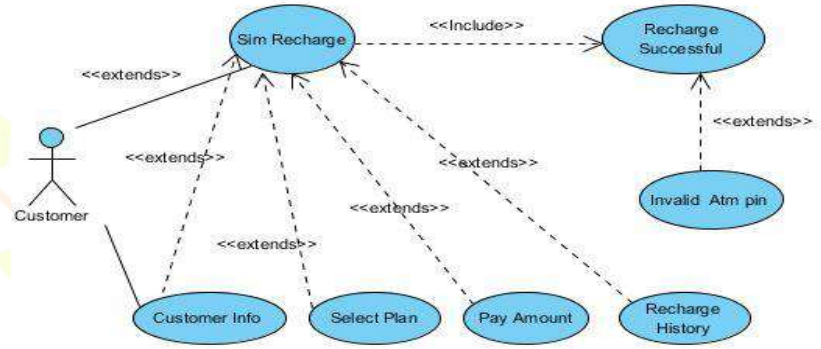

Fig 1.2 Use Case diagram online sim recharge component

1. 2. Activity diagram

The activity diagrams of the two components are below and can be seen that it is representing the way of accessibility and also showing that when the process can be fail and which way is the correct way. It is combination of success and failure an application. This diagram saves much money before implementing a project because it shows all the way if customer satisfy then it can be implementing otherwise it will be stopped.

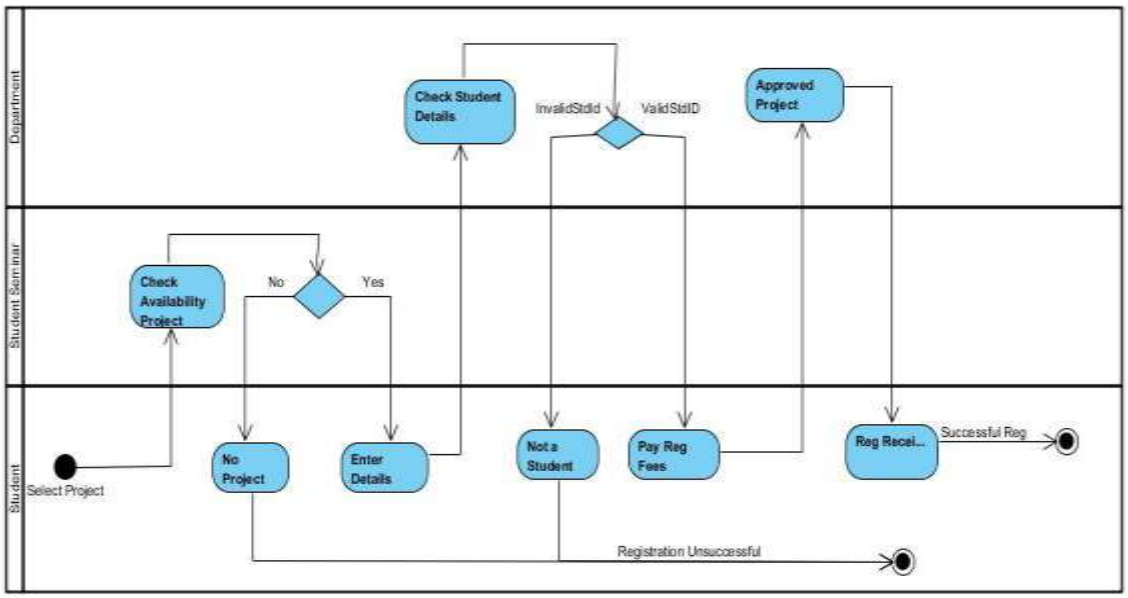

Fig 1.2 Activity diagram reusable component of project

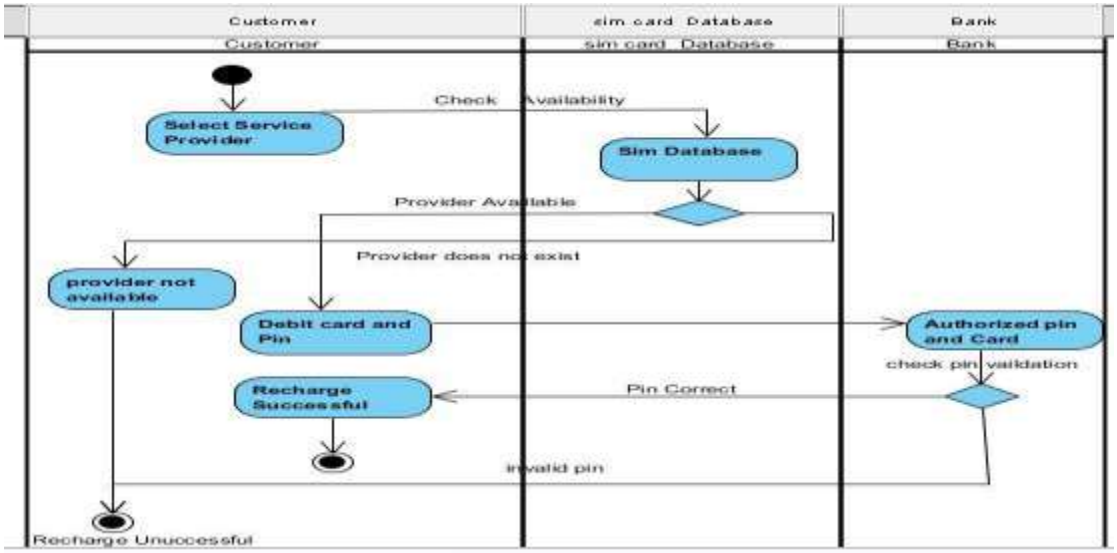

Fig 1.3 Activity diagram of online sim recharge component

1.3. User Sequence diagram

This diagram is user side that how the user can do online recharge and what are methods to pay money all steps are very easy so users will not feel any problem when they will do recharge using this component. The below diagram is also showing whole process of online recharge.

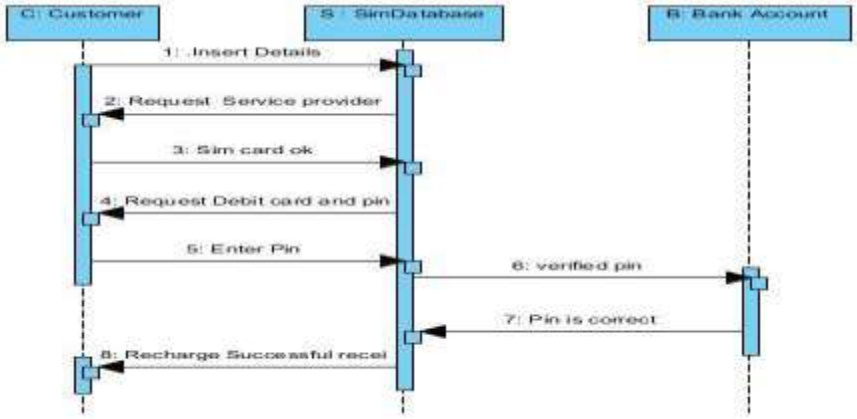

Fig 1.3. User Sequence Diagram for online sim recharge

1.4. Developer Sequence Diagram

The sequence diagram of developer side only means reusability of component. It gives an idea that first complete those part of application which you know when component will be needed it can easily include. There are two sequence diagram first one is that how developer is including project component with student form and another one is how they are including state component with the project because both are different components are the demand of the project.

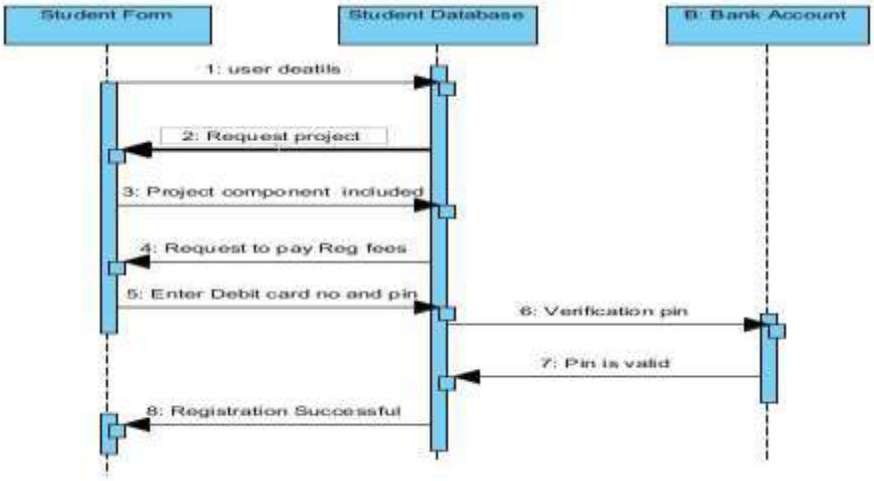

Fig 1.4. Reusability project component to student form

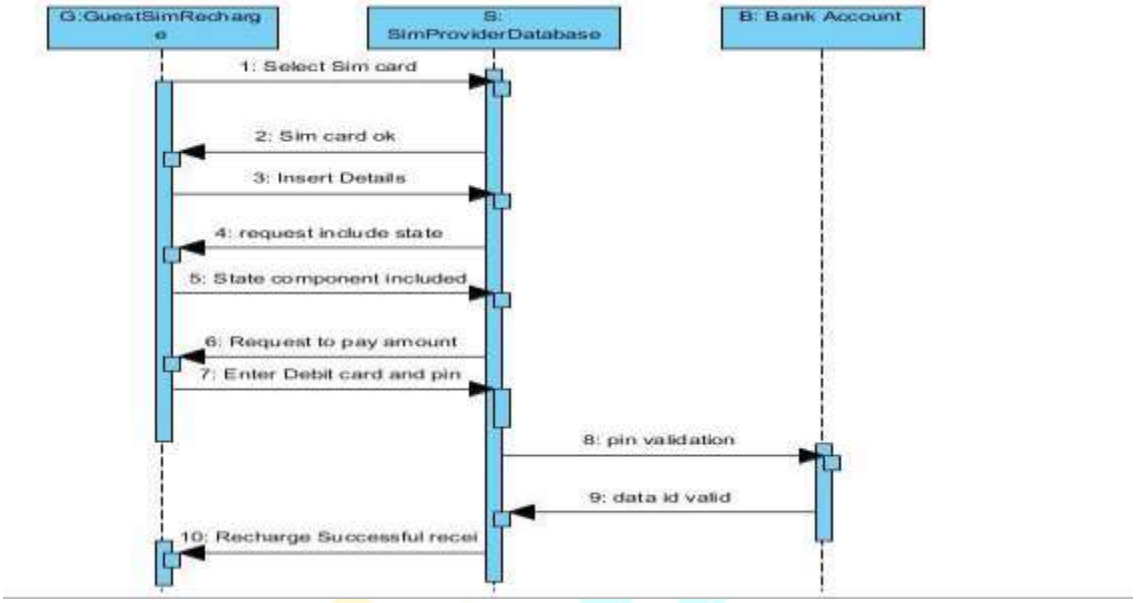

Fig 1.4.1 Sequence diagram reusability state component with user form

1.5. Class diagram

The class diagram will show all the entities of the project which is included to making online recharge system and also will tell the possibilities of dependency with each other.

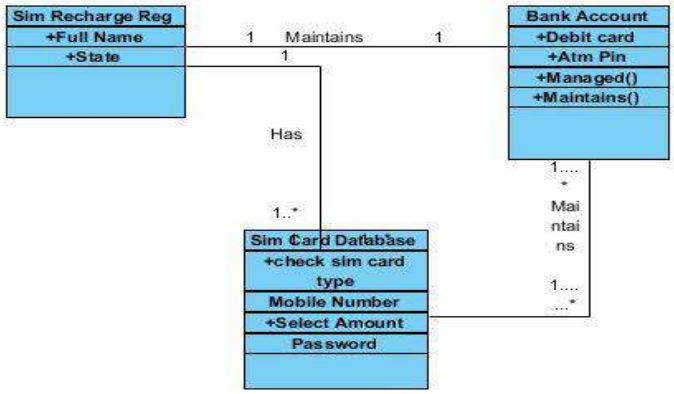

Fig 1.5. Class diagram online sim recharge system

1.6 Component diagram

The component diagram describes two things first one is provided which means what information project will give to the environment (users) and another one is that what the project expect from the environment it is called required. This is the interaction between users and entity and it is also useful security purpose because developers do not want to show all the things to the users so it can be hide. The below are two diagram first one is project component reusability and another one is online recharge system.

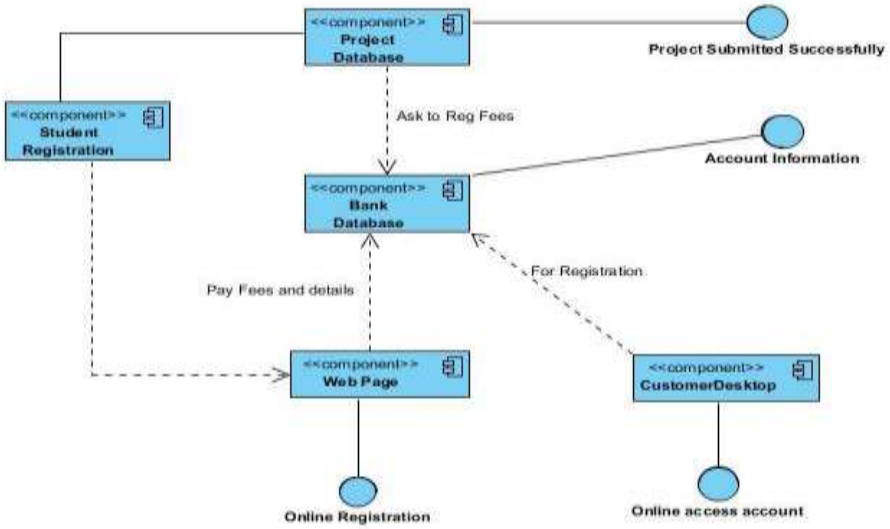

Fig 1.6. Reusable project component with student form

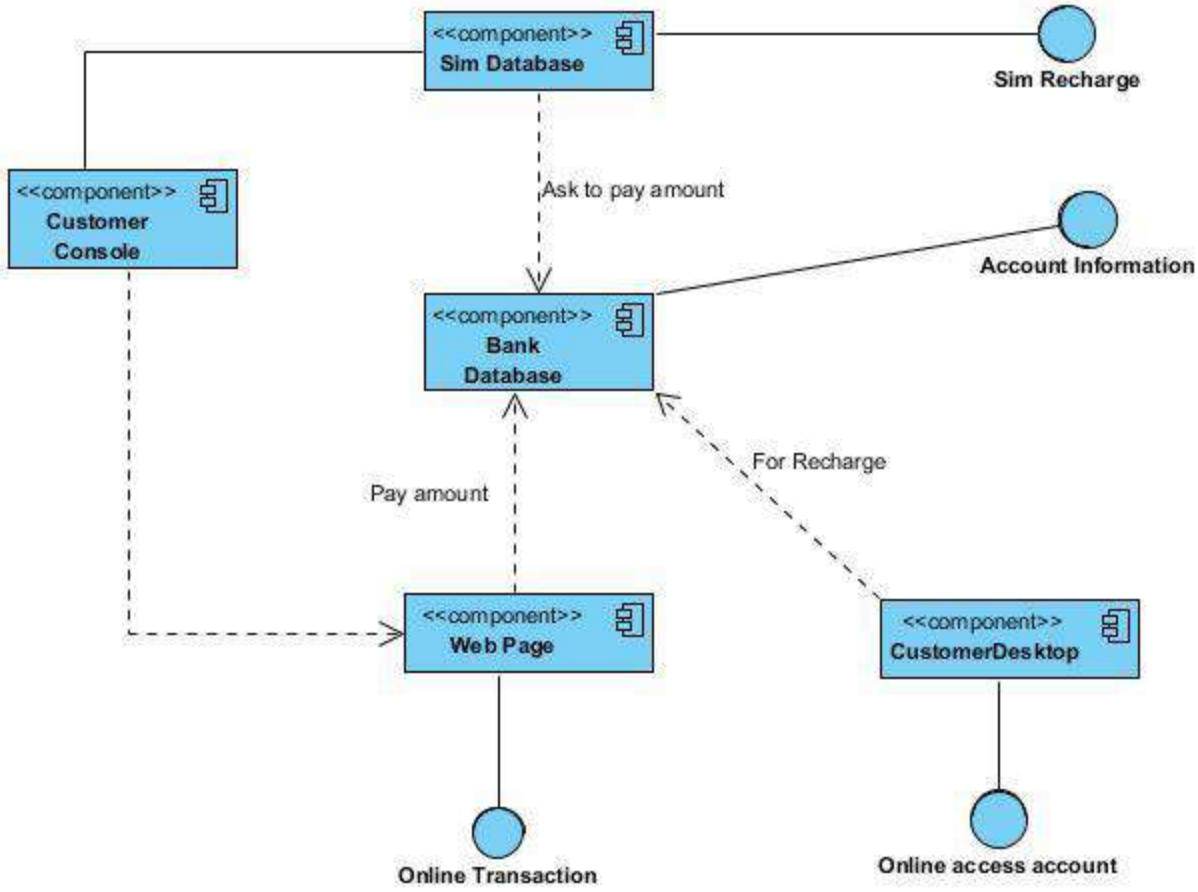

Fig 6.2 Component diagram of developing online recharge system

4. Results and Discussion

1. End User Component with Reuse

The development of the project is good for all those users who want to recharge soon without giving much information on the website but also they can register themselves if they want but there is no compulsion on it. The best thing of this component is that it can be implemented as a sub component or whole component with a new project using CBSE methodology. As the above code, it can be seen that that state is used with user form so this small sub component can be reused with many applications. The overall, this component is user friendly and everything is clearly mention before submit the form.

2. Development Effort

It was not an easy task to develop online recharge system which should be different with another and how this component can be included with another component on server side but this code is user friendly and without any problem can be implemented the big challenge was to make a connection between "AWT" with server side language because both work on different platform but it reduce the problem and project successful.

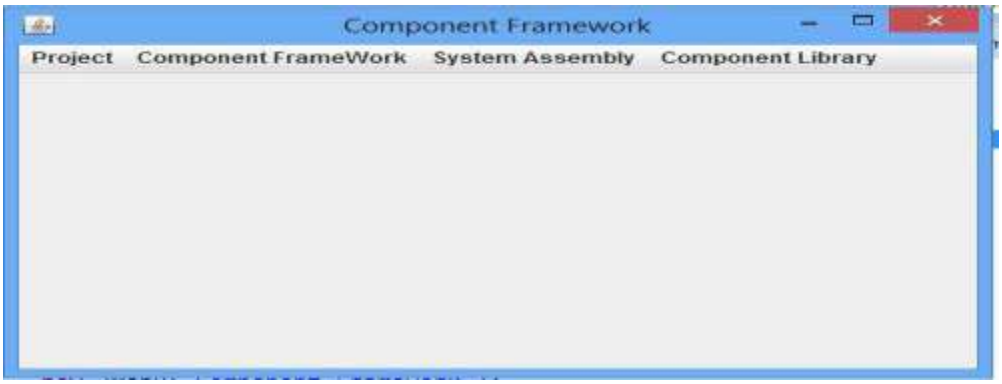

Fig 4.1 Component Framework Tool

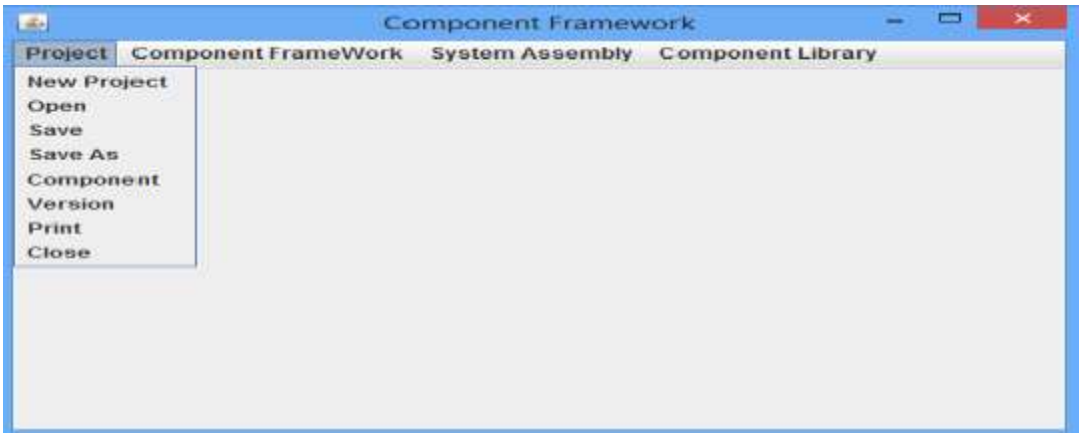

Fig 4.2 Menu-items of Project

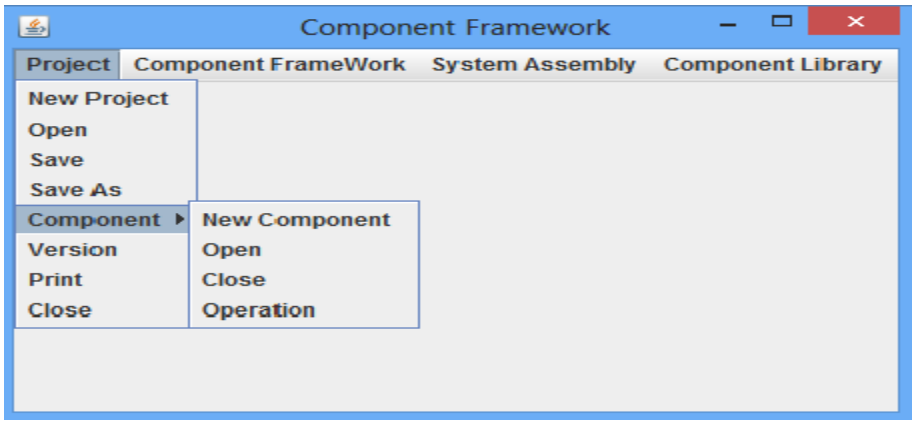

Fig 4.3 Sub menu-items Component

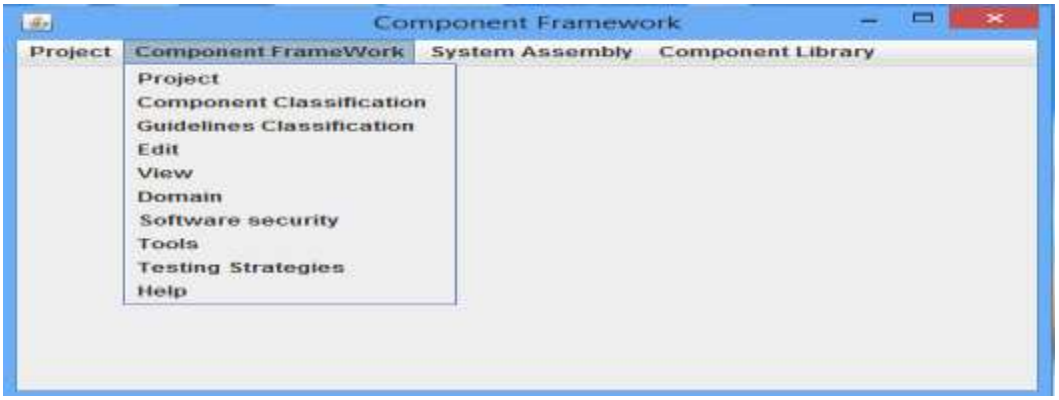

Fig 4.4 menu-items Component Framework

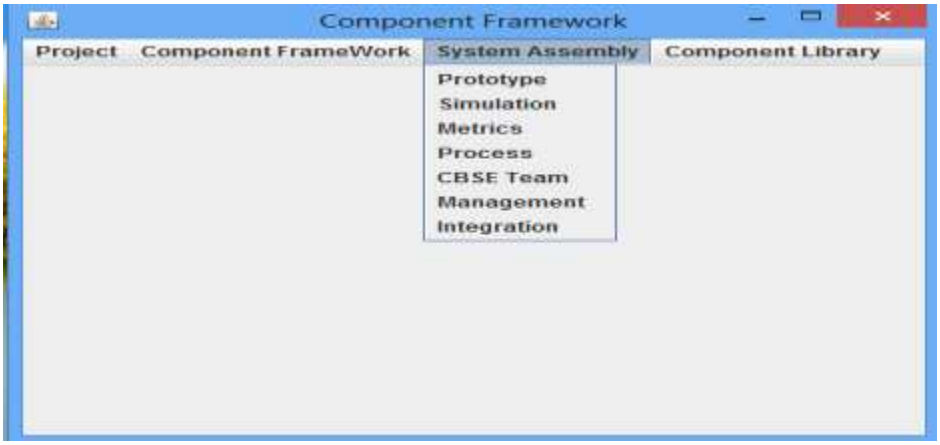

Fig 4.5 menu-items of System Assembly

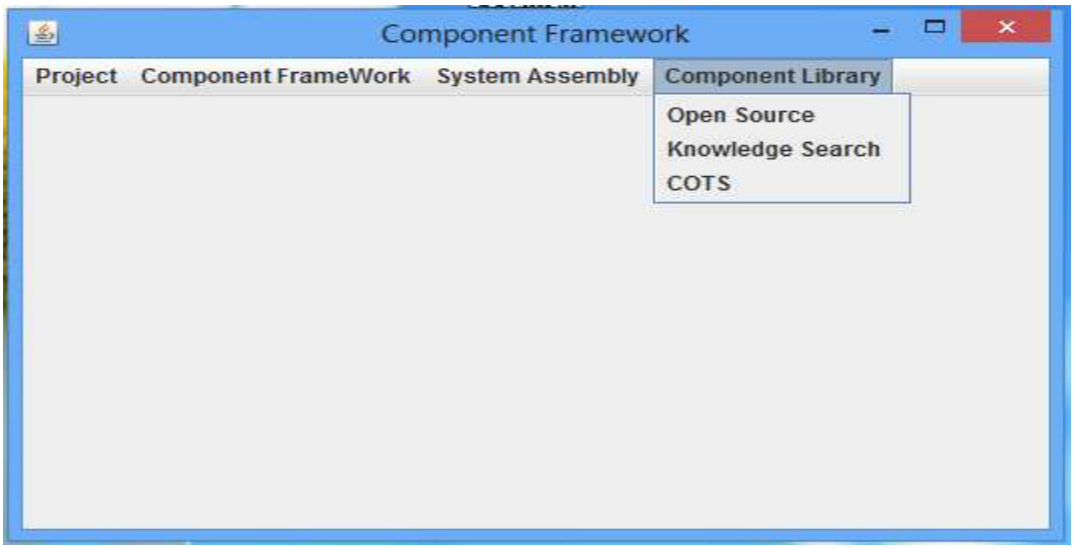

Fig 4.6 menu-items Component Library

5. Conclusion

To sum up, the reusability of the component saves much time to develop the same component which already available and only needs to apply these components in a project as it is. CBSE (Component Based Software Engineering) has already defined everything methods, architecture, design and it reduces the problem of the developers. Now the reusability is the first choice of the IT sector and now it is growing but there is also a negative point is that failure of the project but if we ignore some negative the overall reusability is good. Users can get same product very soon with an additional feature and can enjoy with the project.

REFERENCES

[1] Ivica Crnkovic, M. L. (n.d.). *Component-Based Software Engineering – New Paradigm of Software Development* . Mälardalen University .

[2] M. Rizwan Jameel Qureshi, S. A. (2007). THE ARTIFACTS OF COMPONENT-BASED DEVELOPMENT . *SCIENCE INTERNATIONAL*, 2-3.

[3] Muthu Ramachandran, G. S. (2014). Developing Reusable .NET Software Components. *Science and Information Conference* , (pp. 4-5). London, UK .

[4] Ramachandran, M. (2012). Guidelines Based Software Engineering for Developing Software Components. *Journal of Software Engineering and Applications*, 1-2.