Abstract—“Customer is king”. This is the principle which drives most of the business across the globe. If a business has to succeed, the customers must be happy. Only then, the profit can be made. Software Industry is no exception. Can the customers be made happy always? If yes, at what cost? “It is difficult to satisfy a customer always”. This is one of the most common problems stated by many industry experts. Why does this happen? In this competitive world, customers expect a product not only to just meet their requirements but also give the best performance. The market keeps changing as people have lots of options to choose from. In this kind of a situation, it becomes very essential to implement the changes frequently as required by the customers, in minimum time with high quality. This is one side of the problem. The other face is more challenging. If the requirements are not properly understood and implemented, then the right product can’t be built. This means a loss to the company both in terms of resources & in terms of the client. Hence, effective Requirement Change Management is one of the key factors for the success of any project. Often, requirement changes are not addressed in a systematic manner and the documents are not kept updated and consistent. This necessitates a systematic approach to handle the Requirement changes. Requirement Change Management is a process to address the effective management of the changes in Software Requirement. Typically the objective is to minimize disruptions to the normal flow of SDLC activities. Many software failures and defects are the results of poor Software Requirements Management and definition. In some cases, when a client comes up with an additional issue, the resources required to resolve the same is very high. This remains subject of negotiation between the company and the client. Change control is the centerpiece of Requirements Management. So, an effective change control process is an essential part of software maintenance. While organizations often implement such a process to handle bug reports, it also helps to manage other elements like enhancement requests, functionality modifications, and changes to requirements for new systems under development.

**Keyword:** CR:Change Request POC:Proof of concept CCB: Change control board

I. INTRODUCTION

In order to sustain the business, it becomes important for a company to implement the changes required by the client at right time[1]. When the changes to the requirements affect the project plan, it is very important to have a process to understand the impact of the same. This necessitates the need for a process to understand, estimate and implement the changes in an effective & efficient manner with minimum resources. This process is the Requirement Change Management process[2]. Change management can be conducted on a continuous basis or on a regular schedule. Effective change management programs are frequently sequential and gradually evolving programs that support compliance, acceptance, and internalization of Software Requirement changes. Hence there is a need for a standard process to maintain the integrity of the change management [3]. If the client requires a change to be implemented in the proposed system (software), the change management model makes sure that the required set of activities are done at right intervals with proper approvals by effective communication to the stakeholders.

II. REQUIREMENT CHANGE MANAGEMENT PROCESS

A simple Requirement Change Management consists of:

1. **Client** : The Change Initiator.
2. **Project Manager**: Understands and analyses the Change and tracks its progress during implementation.
3. **Team Members** : The team which implements the Change request
4. **Change Control Board** : The body of experts that decides whether to implement the change or not.

The client raises a request for the implementation of the new requirement by providing the required details to the system. This new request is routed to the CCB via the Project Manager. It is his responsibility to make the team members and CCB aware of the new requirement change suggested by the client. The typical activities done by the Project manager is as shown in Fig2. The Team Member(s) do the impact analysis and submit the same to the Project Manager Fig:3 The Change Control Board is the final committee which decides to implement the new change or not and their decision is final.
Fig 2

Fig 3

Fig 4
In general when the client comes with the new requirements (may be for existing project or ongoing project or completed project) implementing the new request for the customer is not always cost effective because efforts & time also come in to the picture and lot of changes take place in the software development activities. The result of this is partial implementation of the requirement. This leads to an unsatisfied client. In this way, there are chances of losing the client. If the contract period is over, then client has to bear additional amount which may be high and not affordable. So a model has to be developed in such a way that new requirements can be implemented by the company in a cost effective and flexible manner without many efforts. Therefore a prototyping model is required to describe some new changes to the software requirements in a systematic manner.

III. CONCLUSION

The model has to describe change process to all people involved and the information should be complete, unbiased, reliable, transparent, and timely. Implementing new change for the requirement should be aligned with organizational objectives, macro environmental trends. It helps in identifying gaps and opportunities for improvement in requirement engineering activities, defining and implementing best practice and establishing process improvement roadmaps. Any change shall be assessed before accepting it and changes shall be authorized and documented. Authorized users who can request changes and also approval mechanism for change request shall be defined. At any point of time change initiator should know the status of his new change request.

IV. ACKNOWLEDGEMENT

We wish to thank all the authors for the providing the informational support

V. REFERENCES

1) G.Kotonya and I Sommerville Requirement engineering Process
2) Requirement Process Improvement Model : JSystem and software
3) I Sommerville Requirement Engineering :Good Practice Guide www.cmmi.edu