ISSN:0975-9646

Confident Software Development Process Model

S.Thulasi Krishna M.E., (Ph .D),
Research scholar
Rayalaseema University
Kurnool (dist.,)
Andhra Pradesh

Dr. S .Sreekanth, Ph.D
Professor,
Dept of CSE
SITAMS,Chittoor(dt),
Andhra Pradesh

Abstract: Software Development Life Cycle Model gives theoretical guide line regarding development of the software. Software development models are very important for developing software in a systematic manner such that it will be delivered within the time deadline and also should satisfy quality parameters. Generally we have many different techniques and methods used in software development life cycle. An overview of the more common system development process models used to guide the system Analysis, Design, Development and Maintenance of developed software. I propose new Software development process model that is "Confident Software Development Process Model". In this model we have not frozen requirement phase, one can move easily from design phase to requirement phase if a new requirement surfaces. We can easily incorporate small to large changes in any phase. This is very useful to developers to develop new software.

I. INTRODUCTION

A software process is a sequence of tasks intended to produce a high quality software product on time and with in budget [3]. A software development process is structure imposed on the development of a software product. Similar terms include software life cycle and software process. The System Development Life Cycle framework provides a sequence of activities for system designers and developers to follow for developing software. It is often considered as a subset of system development life cycle. Any software development process is divided into several logical stages that allow a software development Company to organize its work efficiently in order to build a software product of the required functionality within a specific time frame and budget. All software projects go through the phases of Requirements gathering, Business analysis, System design, and Implementation and Quality assurance testing [1]. A Software Development Life Cycle Model is a set of activities together with an ordering relationship between activities performed in a manner that satisfies the ordering relationship that will produce desired product. A software development life cycle model is broken down into distinct activities and specifies how these activities are organized in the entire software development effort. In response to traditional approaches to software development, new light weight methodologies have appeared [2].

II. STEPS TO BE FOLLOWED IN DEVELOPING SOFTWARE:

- Software life cycle
- Software development language (C, C++, Java Etc.,)
- Document with well-defined requirements.
- Designing model for requirements.
- Transforming a model into actual software (End user product).
- Verifying and validating the developed software.
- Delivery of developed product to end user.
- The software life cycle is the sequence of phases that a software Development project goes through.

Output = Test Code + Design +Analysis + Requirements
Requirements=> Analysis => Design => Software development => Verification = **End product**.

III. CONFIDENT SOFTWARE DEVELOPMENT PROCESS MODEL

The Confident process Development Process is also called as "C Process Model" This model which we have proposed has seven phases, namely; Feasibility study/Requirement, Requirement Based Analysis, Logical Design, Confident Code, Logical Testing, Implementation & Deployment and Maintenance.

It is a flexible model not restricting the developers and enabling them to move both Front and back from any given stage to any other stage during its development lifecycle. Each phase is further divided into sub phases, each specifying a criterion which has to be met to move to the next phase. This criterion also points out which phase to back track in case of failure.

The model starts with the Feasibility Study/ Requirement and Analysis phase in which requirements are gathered as well as validated by the user. Next phase inline is the Logical Design, We will design our software .Next phase Confident Code, we will write code in confident manner. Next phase Logical testing phase in which we will test software after coding. Next phase is Deployment, its last phase, our software and later on it is tested and is validated by the user.

But meanwhile if some of the requirements changes or new requirements surface during this phase, then we can loop back to requirement phase to accommodate new requirements with the existing one. Following this phase is the Implementation and Deployment phase which ultimately leads to Maintenance Phase.

Architecture of Confident Software Development Process Model

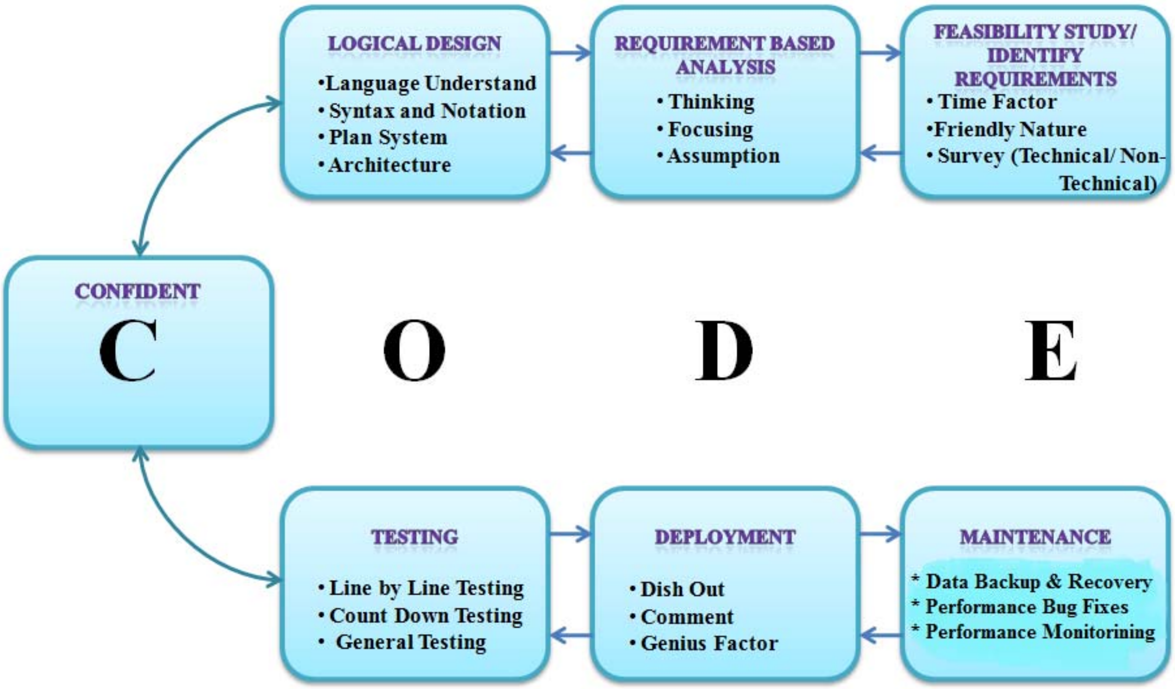


Fig 1. : Architecture of Confident Software Development Process Model

A. Identify *Requirements & Requirement Based Analysis (IR&RBA):*

First, Feasibility study, System Based Requirement and Model Based Requirement, Identify requirements is divided into "Must have" and "Should have" requirements as shown in below fig. 1.1. The "Must have" requirements are those which are most important to the system and must be the part of the system. These mandatory requirements are expected to be implemented in the first version itself. While "Should have" requirements are those which have lesser priority, [owing to trivial implementation their they are] delayed till later versions. "Should have" requirements are equally important and must not be left out in later versions. We have divided requirements to have more and more focus on the "Must have" requirements, and also to make our work simple and easier to carry out. The requirements are then initially Verification is validated by the user and we are left with the refined requirements only. Initial Verification and Validation by the user will ensure whether or not the developer is going in the right direction and that all the concerned parties are committed with the requirements being gathered. The refined requirements are also revalidated again by the user in the Final Validation step to get the clear and obvious picture of the user requirements. Then all the requirements i.e. "Must have" and "Should have" are integrated to have the overall scenario of requirements for the software.

B. Logical *Design with Validation:*

Beginning of Design phase is marked by the existence of an Initial Architectural Model. Architecture Model consists of all basic modules that our system will have. The next step is the Risk Analysis and Risk Resolution of the Architectural Model in the Designing phase. A risk is a potential for loss or damage to development of the software from materialized threats. Risk Analysis attempts to identify all the risks and then quantify the severity of the risks. If it occurs, it exploits vulnerability in the security of a computer based system. Here, different risks involved are identified and resolved ensuring secure development and advancement of the design towards its deliverable version. This step leads to achievement of a Design. Following this step is the Model Validation step. Here the identified Architectural Model is validated ensuring all basic modules have been covered. The validation's major objective is to ensure that all concerned parties are equally satisfied by the output. Once the design is validated and is found to be up to the mark, next step is to give a more detailed complete Design shown in Fig 1.2.

Each Validation sub phase in Design phase commands fulfillment of a certain criteria. Criteria fulfillment allows moving downward/ forward in the flow towards the next phase whereas criteria non-fulfillment restricts moving forward and loops back to the phase decided by the criteria. The flow works its way all the way back towards the phase it looped back from. The criteria fulfillment is tested again and next decision is taken accordingly.

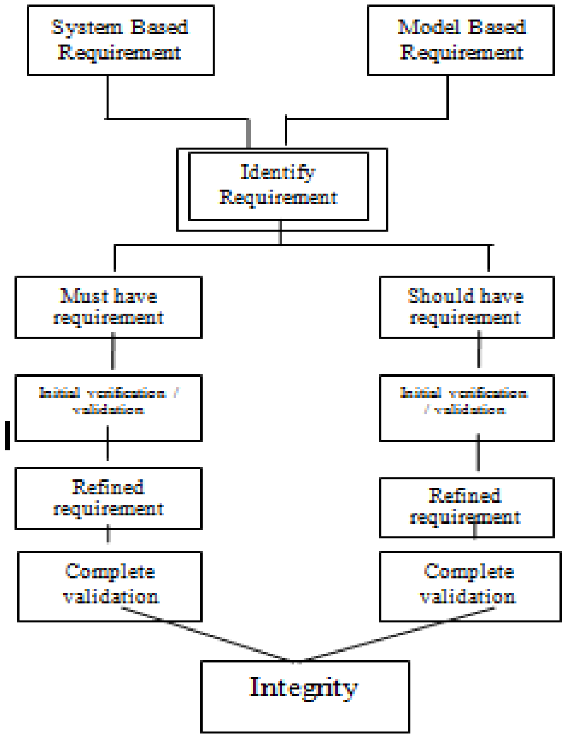


Figure 1.1: Identify Requirement &Requirement Based Analysis

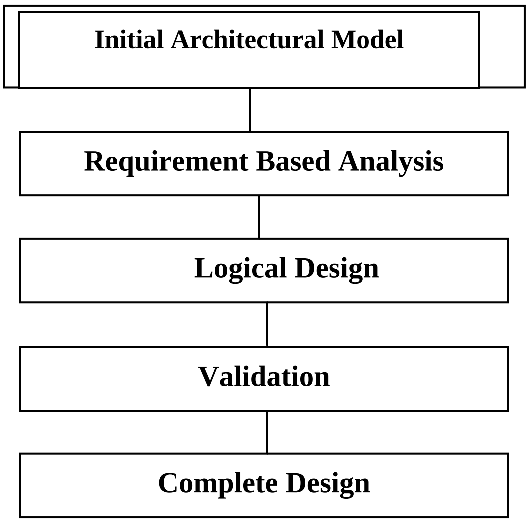


Figure 1.2: Logical Design & Testing

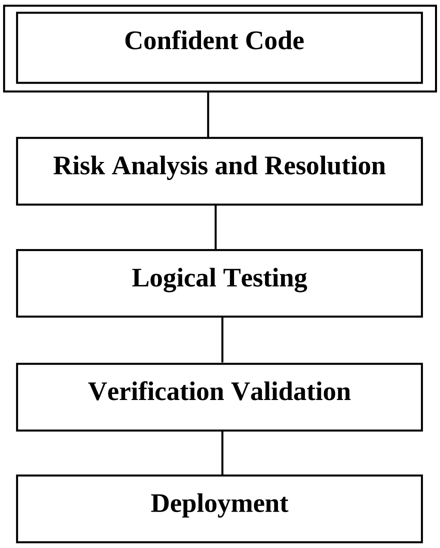


Figure 1.3: Confident Code & Deployment

C. Code *&Deployment:*

The first sub phase of Code & Deployment is Coding. Next this coding is analyzed for Risks (if any) and their Resolution. Risk Analysis and Resolution (if any) like technology risk, platform risk, programming language risk, and last but not least the cost risk. Then, whole system is tested; here the System Testing is involved. System testing of software is testing conducted on a complete project, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic. Now, the software is validated and verified by the concerned authority. After which the system is deployed. Deployment should be interpreted as a general process that has to be customized according to specific requirements or characteristics.

D. ***Maintenance:***

Maintenance refers to modification in the product after delivery, to correct errors or to improve the performance. Maintenance phase in our model may include many activities like creation of management plan, analysis of the problem process. It includes other activities like coding of modification, checking whether the software conforms to particular environment or not whatever is needed.

III.I. There Are Seven Phases, And Sub-Phases Namely:

1. Feasibility Study/Identify Requirements
    - Time Factor
    - Friendly Nature
    - Survey (technical and non-technical requirement)
2. Requirement Based Analysis
    - Thinking
    - Focusing
    - Assumption
    - Chart preparation
3. Logical Design
    - Language Understand
    - Syntax and Notation
    - Plan System Architecture
4. Confident Code
    - Logical Design => Code (software)
5. Testing
    - General Testing
    - Line by Line Testing
    - Countdown Testing
6. Deployment
    - Dish Out
    - Genius Factor
    - Comment
7. Maintenance
    - Data Backup & Recovery
    - Performance Bug Fixes
    - Performance Monitoring.

1. Feasibility Study/Identify Requirements:

The first phase includes the study of feasibility/identifying requirements. Based on Development activity it is further divided into “must have” and “should have” phases. Feasibility study defines Time Factor, Technical Requirement and Non technical Requirement for developing project. Friendly environment need to be provided between customer and developers to collect requirements in detail.

2. Requirement based Analysis:

By thinking Requirements we get Time Management schedule, and it is evaluated by Assuming the development activity in Future .This phase mainly focused on Preparing SRS Document.

3. Logical design :

In order to define System Architecture a developer needs to understand Language for Syntax and Notation used in Documentation.

4. Confident Code:

Transformation of Logical Design into Software Code that is set of instruction given to system to satisfy user Requirements.

5. Testing:

General Testing: Verifies Behavior of a system that is Black box testing which Validates output for various inputs.

Line by Line Testing: This Testing phase checks every line of software for Identifying bugs .

Count down Testing :This is module based Integration testing for module interfacing.

6. Deployment :

Dish Out is also known as Result, which is Final outcome of testing phase with Standard Genius Factor. After deployment an usual feedback is taken from customer for further development and satisfaction.

7. Maintenance:

Once an application developed it is need to be monitored for performance, bug fixes and data backup/Recovery.

III. II. Principles Of Confident Software Development Process Model

The Basic Principles are:

1. Project is divided into set of sequential phases, without overlap between phases.
2. Back Tracking is possible at any phase. If Requirements are Changed.
3. Mange Requirements at any phase.
4. In Every phase, user involvement is possible.
5. Standard system Analysis, Design, Confident Code can be fitted into this Frame work.
6. Continuously verify Quality Factors in each phase.
7. Key Objective is for fact development and Delivery of High Quality System a Relatively low Investment Cost.
8. Automatically increase the Milestones in every phase.

III . III. Advantages:

- Simple and easy understand
- Sequential step by step growth process
- Well understand milestones
- Easy to mange each phase as specific deliverable to end-user
- All upcoming requirements to be frozen and to be accommodated in the later version of the product.
- We haven frozen requirements phase one can move easily from design phase to requirement phase if a new requirement (S) surface.

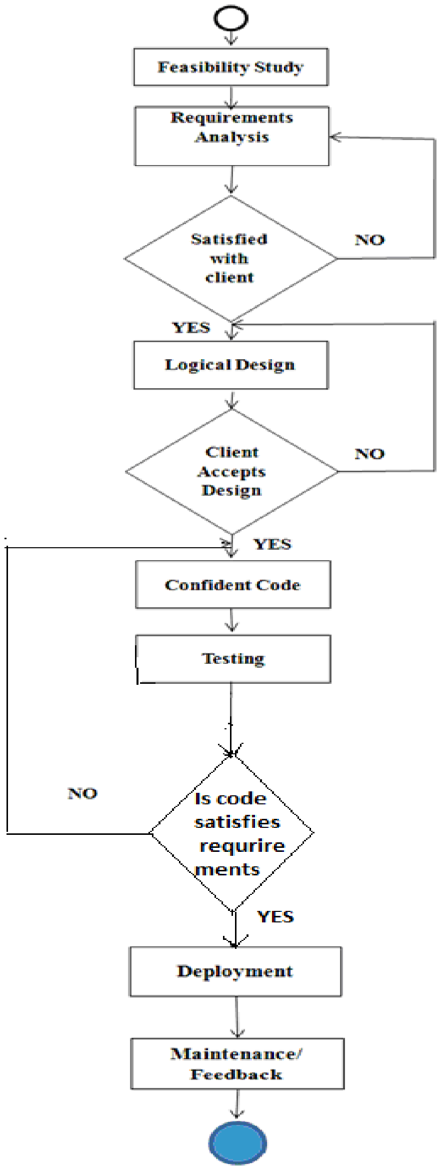
III. IV. Disadvantages:

- If any changes happen in midway, then test document along with Requirement document as to be update.
- Risk factor is high.
- Increase downward flow of defects.

IV. V. Application Of Confident Software Development Process Model

- Small to large projects
- commercial application development

V. VI : Data Flow Diagram For Confident Development Process Model



IV. Conclusion:

In this paper we have proposed Confident Software Development Life Cycle Model on the basis of certain features like- Requirement specifications, Risk involvement, User involvement, Cost etc. On the basis of these features for a particular software project one can decide which of these software development life cycle model should be chosen for that particular project. Selecting the correct life cycle model is extremely important in a software industry as the software has to be delivered within the time deadline and should also have the desired quality. This Model allows changes at any phase, so that it can be used from small to large scale projects.

References:

[1]. Klopper, R., Gruner, S., & Kourie, D. (2007), “Assessment of a framework to compare software development methodologies” Proceedings of the 2007 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries, 56-65. doi: 10.1145/1292491.1292498.

[2]. Fowler, M. (2000), "Put Your Process on a Diet", Software Development".

[3]. Software project management Real Word Guide success “person Education, pp.25

Authors Description

S.THULASI KRISHNA received the B.Tech Degree in Computer Science and Engineering from Jawaharlal Nehru University, Hyderabad, India in 2005, M.E from Sathyabama University Chennai, and currently pursing Ph.D in Rayalaseema University, Kurnool. Presently working as an Associate Professor in Kuppam Engineering College. He is also a member of International Association of Engineering.

Dr. S. SREEKANTH has obtained Ph.D. Degree from S.V.University, Tirupathi. Presently he is working as a Professor in the Department of Computer Science and Engineering in Sreenivasa Institute of Technology and Management Studies, Chittoor, Andhra Pradesh, with the experience of 20 years. He has published 15 research papers both in national and international journals.