

## **Review on Embedded Systems Software Development Life Cycle Model**

Dr. V.SureshBabu

Associate Professor

Hindusthan Institute of Technology, Coimbatore, India

### **Abstract**

The embedded device rotates around us as an integral component or subsystem in all forms of applications. The main components of the embedded computer are software and hardware. Embedded system is now assisted by software and embedded modeling of systems must be understood as appropriate. Specifications research and project management are very complex and repetitive. The Software Development LifeCycle (SDLC) assists in the proper design and execution of the project and in the handling of project specifications and communications with customers.

This paper examined SDLC models using various approaches compared to Agile Methods (AM). The Agile software development approach is a technique which evolves economically and in a timely fashion, in cooperation with self-organizing interfunctional teams, to respond to stakeholders' changing needs. In order to make the best use of consumer value from the software product, agile development is primarily inspired by providing small, self-organized teams with a painless work environment. AM encodes are not prepared, but additional documentation is needed for embedded systems due to optimized code requirements. With advanced management and technology and Lean Agile Framework the integrated SDLC process has been improved without wasting any workload.

It also tracks the collection of functions, the workforce, the time of execution, and lets the project manager determine when and where change is required. During this point, amendments to support the integrated system generation continue to be assessed, scheduled and controlled. It encourages better code refactoring which helps to optimize. Agile methods also pose possible challenges to business continuity, given the limited architecture and documentation required, and in order to overcome these risks AM must be strengthened.

**Keywords:** Embedded Systems, Agile Methods, SDLC

## 1. Introduction

The SDLC model is a system work that determines the work at any project level. It is used to describe any process. These measures are approached differently by various software development methodologies. The models are waterfall, V, Iteration, Incremental, spiral, agile[1]. These models involve. Also available are the models. Although some phases seriously overlap, some phases are concentrated or omitted. This paper offers information on software development agile technology. It explains how the SDLC model has been selected and how the various agile approaches are applicable to integrated applications. In this paper, the weather-agile methods of different SDLC models for embedded production systems are analyzed and justified.

## 2. About Agile

Agile software development methodology is a state-of-the-art technique that enables cost-effective and timely development of demands and solutions through the collaboration of self-organizing interfunctional teams in an atmosphere of sound administration[1]. An agile development event was released by a community of software practitioners and consultants in 2001. The manifesto considers that individuals and expertise are better than processes and tools, valuable for customer cooperation in contract negotiations, invaluable for software that deals with detailed documentation and valuable for responding to modifications than to plans[2]. Agile techniques were used because prior plan approximations were not applied, as seen in Figure1[3], in order for changes to be addressed in evolving environments.

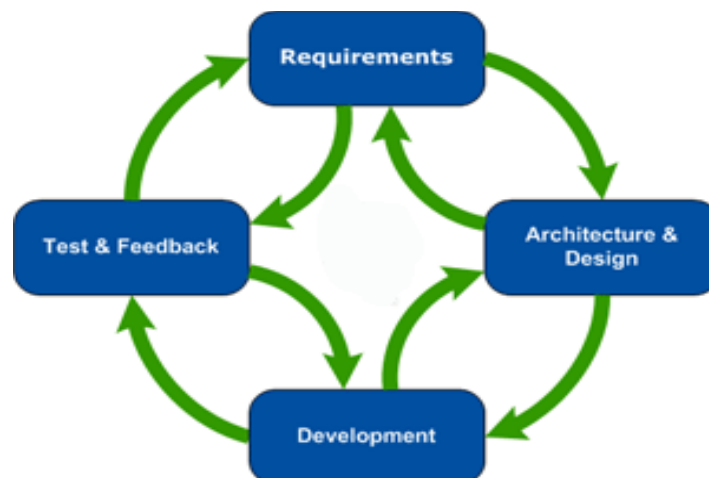


Fig. 1 Agile process

Extreme (XP), scrum, DSDM, adaptive software (ASD), crystal and functional development are the most common agile approaches in development (FDD). XP has described problems resulting from the protracted periods for producing conventional models. XP's life cycle comprises five stages: discovery, release and preparation, development and administration [2]. Scrum is an empirical approach to process creation by reintroducing concepts of diversity, adaptability and efficiency based on the principles of industrial process theory. Scrum contains environmental and technological aspects that are likely to be altered during the process[2], including specifications, timespan, resources and technologies. Have pre-game and post-game stages. The heavy technique is demonstrated in each section of the crystal (darker colour). Colors rely on the scale and critique of the project[2]. In other words, more team work and longer methodologies than smaller projects might be acceptable. Cristal clear is intended for SMEs, medium-sized Orange projects and large-scale CR projects (relatively low contact). The FDD approach does not cover the entire phase of software development, it is design-oriented[2]. The FDD underlines quality aspects and requires regular measurable deliveries and project tracking. DSDM aimed at identifying time and resources and then modifying roles. ASD offers a structure with appropriate guidance to avoid the innovation and emergency plunging of project ventures into chaos[2]. Speculation (planning), collaboration (main focus on team work) and learning are included (stresses on need to acknowledge and react to mistakes). In time-critical execution, agile approaches are primarily sought after. These strategies accelerate one or more life cycles and are used in a disciplined organization.

### **3. SDLC Model Selection**

The main factors should be the product or project features. It includes type of information system, size and complexity of program, modularity of system, integration of module levels, standards of quality, overall cost of the project, and availability of resources. The model selection criteria [4] is shown in Table 1.

Due to its durability, good project management capability, better documentation and reusability, the Waterfall model can be used on complex systems, but not when user requirements are ambiguous. Also V models, like waterfall models, have low price but also good administration, documents and reusability. They have an incredible importance. The spiral model works well in uncertain circumstances, but has disadvantages, including cost and capacity limitations on unfamiliar technologies. Gradual and iterative unknown technology models are also giving stakeholders greater visibility in the fields of complex system growth, reliability and unpredictable user needs. They are useful when handling tasks, timing, reusability and recording content. Agile methodology allows stakeholders to be exposed, good project management in

the short term, cost savings, increase in reliability, and lack of official information, documentation, reusability and unknown technology. Enhanced product quality and control by choosing a suitable model development speed. In addition to reduced project risk and overall project management, it leads to stronger customer relations.

Table 1: Factors used for SDLC Selection

Factors	Waterfall model	V-shaped model	Spiral model	Incremental/ Iterative model	Agile methodologies
Complex system	Good	Good	Excellent	Good	Poor
Reliability	Good	Good	Excellent	Good	Good
User requirements unclear	Poor	Poor	Excellent	Good	Excellent
Unfamiliar technology	Poor	Poor	Excellent	Good	Poor
Short-term schedule	Poor	Poor	Excellent	Excellent	Excellent
Strong project management	Excellent	Excellent	Excellent	Excellent	Excellent
Cost limitation	Poor	Poor	Poor	Excellent	Excellent
Stakeholders visibility	Good	Good	Excellent	Good	Excellent
Skills limitation	Good	Good	Poor	Good	Poor
Documentation	Excellent	Excellent	Good	Excellent	Poor
Component reusability	Excellent	Excellent	Poor	Excellent	Poor

#### 4. Application of Agile

The main factors should be the product or project features. This includes data form, device dimensions and complexity, system modularity, module level integration, quality requirements, overall project cost, availability of resources [5], [6].

Due to its durability, good project management capability, enhanced documentation and reusability, the Waterfall model can be used on complex systems. Even V models, like waterfall models, are well-managed, recorded and reusable for a low price. They're playing an amazing role. The spiral model works well in uncertain scenarios, but has disadvantages, including cost and power limitations for unfamiliar technologies. In the areas of complex system development, dependability and unpredictable users' requirements, stakeholders have more visibility in progressive and iterative unknown technology models. They are useful in projects, timestamps, reusability and monitoring of content. The flexible approach exposes the absence of

official records, documentation, reusability and the unfamiliar technologies to stakeholders, good short-term project management, cost efficiency and reliability. Improved quality of the product and control by selecting the model development speed required. In addition to a reduced project risk and overall project management, it leads to stronger customer relations [7],[8].

## **5. Lean Agile method**

The Lean Method is the philosophy of Toyota production framework. Lean's theory is to reduce waste in the production of embedded systems. The lean and agile combination results in more integrated software development with much better re-calibration and refurbishment where the requirements change without affecting quality or time. The main components of the Lean Agile approach are:

Waste disposal: Code, workers, signals and PCBs will be developed. Build a trustworthy and sustainable team.

Better learning: Build projects and teams for better learning

Production of integrity: Integration and strengthened project management team

Deliver as early as possible: with the honesty of the team, the project will take place as soon as possible to meet the project deadline. Lean Agile Methodology deals with the development of user history in the integrated system development process. User stories demonstrate who, what and why in a realistic and reliable way. These user stories help us to define the functions needed for each aspect of the creation of the device user application. It also helps us to understand the features needed in every section of the embedded device framework to satisfy user requirements. The Lean Agile methodology makes emphasis on the articular properties of the integrated system in order to prioritize enhanced business values in system growth. Lean Agile provides the following advantages:

- Fast turn-around: the production of embedded systems would be far quicker to ensure effective customer connectivity and satisfaction than conventional SDLC methods
- Direct input: Design and development team reviews and customer feedback offer a great way of developing an integrated system.
- Priority projects: this will lead to increasing the performance of the most critical components of the device and other testing problems and to satisfying and contributing users.

The emphasis will be on the provision of added value to company and the elimination of low value. It also focuses on improving the project's cost value.

## **6.Conclusion**

Agile methodologies can be adapted to evolving needs, and enable customer engagement in fast cycles to concentrate on reducing workloads and software. These features are however useful for system growth because they add value to the system and the business. It also improves customer engagement and feedback. It also helps to reassess and refine the code. The drawbacks of agile methods create possible risks to the stability of organizations if the required design and documentation are not insisted on.

## **References**

- [1] Moniruzzaman A.B.M., Hossain S.A., (2013) Comparative Study on Agile software development methodologies, Retrieved on October 15, 2015.
- [2] Abrahamsson P., Salo O., Juhani Warsta J.R., (2002) Agile software development methods Review and analysis. Issue 478 VIT Electronics, [Online]. Retrieved on October 17, 2015.

- [3] Highsmith J.A., (2002) Agile software development ecosystems, 1 Edn: Addison-Wesley Professional.
- [4] Mohamed Sami, (2012) Software Engineering Practices, Retrieved on November 1, 2015.
- [5] Ajay Deep , (2012) An empirical study of agile development, International Journal of Latest trends in Engineering and Technology , 1(1), p p . 35-40.
- [6] Doug Dahlby , (2004) Applying Agile Methods to Embedded Systems Development, Retrieved on November 1, 2015.
- [7] Chiranjeevi Aradhya. "A detailed survey and evaluation of various aspects of embedded software robustness testing." International Journal of Advance Research, Ideas and Innovations in Technology 7.1 (2021).
- [8] Chiranjeevi Aradhya. "Utilizing lifecycle management system approach, boost airworthiness certification of software-centric avionics systems" International Journal Of Advance Research And Innovative Ideas In Education Volume 7 Issue 1 2021 Page 1583-1589