ISSN: 2321-7782 (Online)
Volume 2, Issue 8, August 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

A Case Study on Decision Process Model

V.Rajiv Jetson
Associate Professor
Dept of CSE
Kallam Haranadhareddy Institute of Tech
India

***Abstract:** Software development processes should support organizations in developing software projects in a structural and organized manner. Unfortunately, many large organizations still struggle with applying these processes appropriately. A large number of processes are available for organizations. However, just a few processes are often actually in use. Organizations apply processes for projects without considering the characteristics of these projects and the possible other processes available. The success of using a software development process for a software project heavily depends on the characteristics of the project itself and the characteristics of the used process. There is no ideal process according to Sommerville, 1996. A solution is to use different processes for different projects. By providing project managers a decision framework for choosing the right process, valuable time is saved and the quality of the final product is increased. The main research question in this research was:*

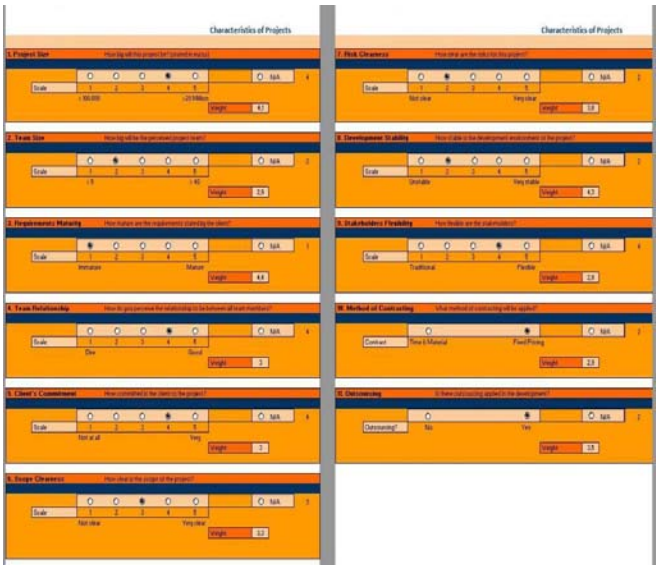
Which decision framework will assist project managers in large organizations in choosing a suitable software development process?

The first step was not only to analyze software development processes but software projects as well. This case study at SOFTWARE INDUSTRY provided enough information to get an overview of all the relevant characteristics of software projects. Literature was a great source of information for the software development processes. For this particular decision framework the main problem was to map the characteristics of software development processes and the characteristics of the software projects. This mapping problem was solved by analyzing the suitability of each process on each characteristic individually. By dividing the software project's characteristics in five different scales, an overview was created of the suitability of each process on each possible scale.

The developed framework is further elaborated on in section 1.1. Explanation and screenshots are presented. Section 1.2 reflects on this entire research. Questions such as "Which issues were overcome and which were not?" are answered. Furthermore, generalization possibilities for the developed framework are presented. Finally, in section 1.3, recommendations are formulate with regard to this research. These consist of all the implications this research has on SOFTWARE INDUSTRY. A separate subsection is devoted to recommended future research

I. The Framework

Based on the mapping solution explained before, a decision framework has been developed. The current framework consists of an MS Excel model. The decision framework is presented as follows by presenting screenshots of each element. A fictitious project is used to present the mechanics of the framework. In subsection 1.1.2, the usability of the framework is discussed.



Characteristics of Projects

1. Project Size

2. Team Size

3. Requirements Maturity

4. Team Relationship

5. Client's Commitment

6. Scope Clearness

7. Risk Clearness

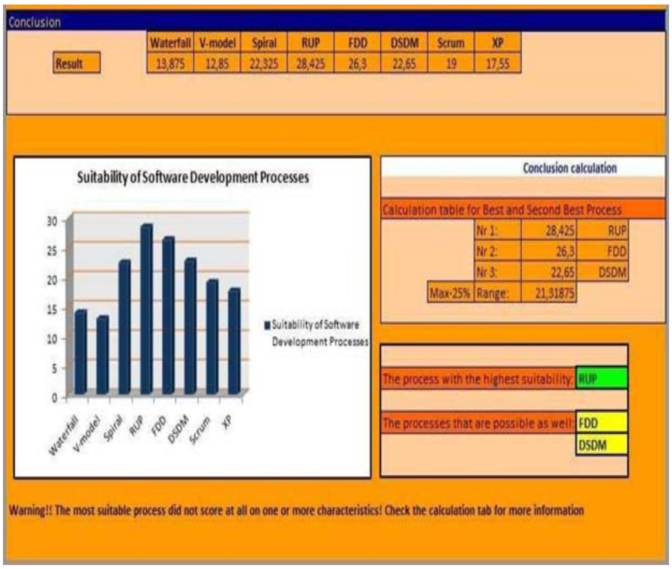
8. Development Stability

9. Stakeholders Flexibility

10. Method of Contracting

11. Outsourcing

FIGURE 1-1 Framework Questionnaire



Conclusion

	Waterfall	V-model	Spiral	RUP	FDD	DSDM	Scrum	XP
Result	13,875	12,85	22,325	28,425	26,3	22,65	19	17,55

Suitability of Software Development Processes
30
25
20
15
10
5
0
Waterfall
V-model
Spiral
RUP
FDD
DSDM
Scrum
XP
■ Suitability of Software Development Processes

Conclusion calculation

Calculation table for Best and Second Best Process

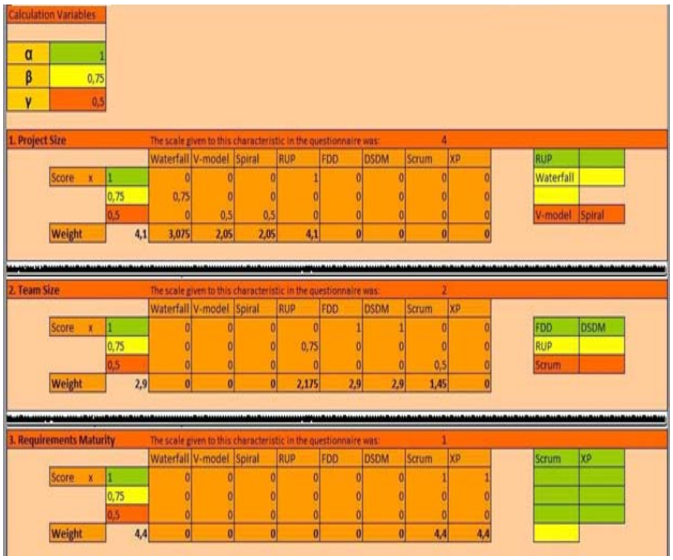
	Nr 1:	28,425	RUP
	Nr 2:	26,3	FDD
	Nr 3:	22,65	DSDM
Max-25%	Range:	21,31875	

The process with the highest suitability: RUP

The processes that are possible as well: FDD
DSDM

Warning!! The most suitable process did not score at all on one or more characteristics! Check the calculation tab for more information

FIGURE 1-2 Framework Conclusion



Calculation Variables

α	1
β	0,75
γ	0,5

1. Project Size — The scale given to this characteristic in the questionnaire was: 4

		Waterfall	V-model	Spiral	RUP	FDD	DSDM	Scrum	XP
Score x	1	0	0	0	1	0	0	0	0
	0,75	0,75	0	0	0	0	0	0	0
	0,5	0	0,5	0,5	0	0	0	0	0
Weight	4,1	3,075	2,05	2,05	4,1	0	0	0	0

RUP	
Waterfall	
V-model	Spiral

2. Team Size — The scale given to this characteristic in the questionnaire was: 2

		Waterfall	V-model	Spiral	RUP	FDD	DSDM	Scrum	XP
Score x	1	0	0	0	0	1	1	0	0
	0,75	0	0	0	0,75	0	0	0	0
	0,5	0	0	0	0	0	0	0,5	0
Weight	2,9	0	0	0	2,175	2,9	2,9	1,45	0

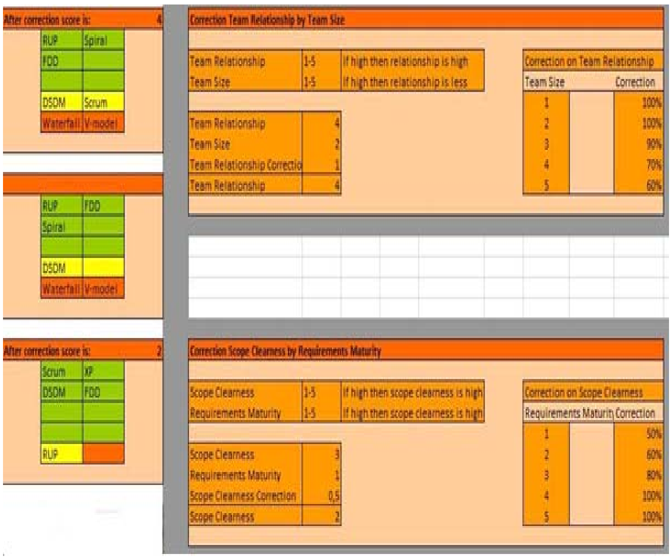
FDD	DSDM
RUP	
Scrum	

3. Requirements Maturity — The scale given to this characteristic in the questionnaire was: 1

		Waterfall	V-model	Spiral	RUP	FDD	DSDM	Scrum	XP
Score x	1	0	0	0	0	0	0	1	1
	0,75	0	0	0	0	0	0	0	0
	0,5	0	0	0	0	0	0	0	0
Weight	4,4	0	0	0	0	0	0	4,4	4,4

Scrum	XP

FIGURE 1-3 Framework Calculation Part 1



After correction score is: 4

RUP	Spiral
FDD	
DSDM	Scrum
Waterfall	V-model

Correction Team Relationship by Team Size

Team Relationship	1-5	If high then relationship is high
Team Size	1-5	If high then relationship is less

Team Relationship	4
Team Size	2
Team Relationship Correctio	1
Team Relationship	4

Correction on Team Relationship

Team Size		Correction
1		100%
2		100%
3		90%
4		70%
5		60%

RUP	FDD
Spiral	
DSDM	
Waterfall	V-model

After correction score is: 2

Scrum	XP
DSDM	FDD
RUP	

Correction Scope Clearness by Requirements Maturity

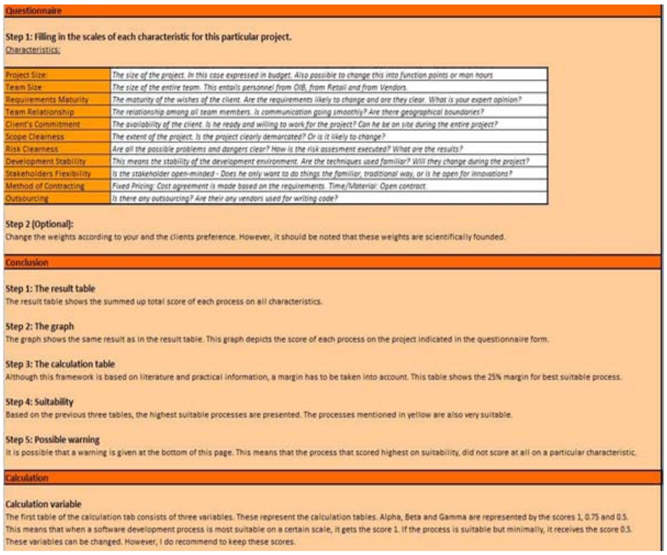
Scope Clearness	1-5	If high then scope clearness is high
Requirements Maturity	1-5	If high then scope clearness is high

Scope Clearness	3
Requirements Maturity	1
Scope Clearness Correction	0,5
Scope Clearness	2

Correction on Scope Clearness

Requirements Maturity		Correction
1		50%
2		60%
3		80%
4		100%
5		100%

FIGURE 1-4 Framework Calculation Part 2



Questionnaire

Step 1: Filling in the scales of each characteristic for this particular project.

Characteristics:

Project Size:	The size of the project. In this case expressed in budget. Also possible to change this into function points or man hours
Team Size	The size of the entire team. This entails personnel from OIB, from Retail and from Vendors.
Requirements Maturity	The maturity of the wishes of the client. Are the requirements likely to change and are they clear. What is your expert opinion?
Team Relationship	The relationship among all team members. Is communication going smoothly? Are there geographical boundaries?
Client's Commitment	The availability of the client. Is he ready and willing to work for the project? Can he be on site during the entire project?
Scope Clearness	The extent of the project. Is the project clearly demarcated? Or is it likely to change?
Risk Clearness	Are all the possible problems and dangers clear? How is the risk assesment executed? What are the results?
Development Stability	This means the stability of the development environment. Are the techniques used familiar? Will they change during the project?
Stakeholders Flexibility	Is the stakeholder open-minded - Does he only want to do things the familiar, traditional way, or is he open for innovations?
Method of Contracting	Fixed Pricing: Cost agreement is made based on the requirements. Time/Material: Open contract.
Outsourcing	Is there any outsourcing? Are their any vendors used for writing code?

Step 2 (Optional):
Change the weights according to your and the clients preference. However, it should be noted that these weights are scientifically founded.

Conclusion

Step 1: The result table
The result table shows the summed up total score of each process on all characteristics.

Step 2: The graph
The graph shows the same result as in the result table. This graph depicts the score of each process on the project indicated in the questionnaire form.

Step 3: The calculation table
Although this framework is based on literature and practical information, a margin has to be taken into account. This table shows the 25% margin for best suitable process.

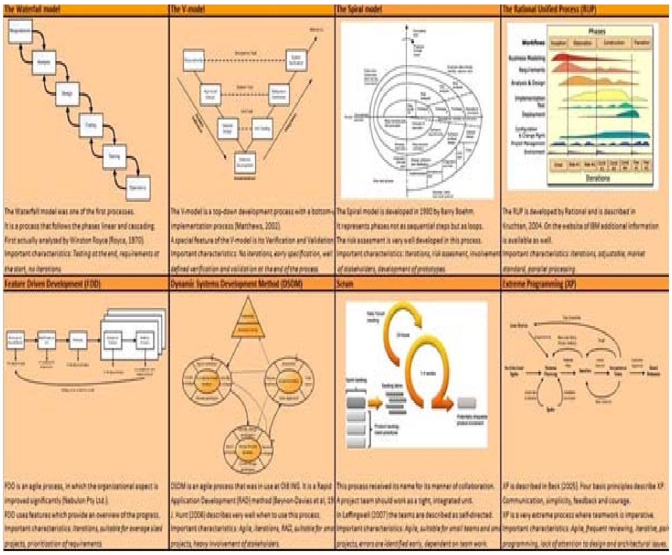
Step 4: Suitability
Based on the previous three tables, the highest suitable processes are presented. The processes mentioned in yellow are also very suitable.

Step 5: Possible warning
It is possible that a warning is given at the bottom of this page. This means that the process that scored highest on suitability, did not score at all on a particular characteristic.

Calculation

Calculation variable
The first table of the calculation tab consists of three variables. These represent the calculation tables. Alpha, Beta and Gamma are represented by the scores 1, 0.75 and 0.5. This means that when a software development process is most suitable on a certain scale, it gets the score 1. If the process is suitable but minimally, it receives the score 0.5. These variables can be changed. However, I do recommend to keep these scores.

FIGURE 1-5 Framework Explanation



The Waterfall model

The Waterfall model was one of the first processes. It is a process that follows the phases linear and cascading. First actually analyzed by Winston Royce (Royce, 1970). Important characteristics: Testing at the end, requirements at the start, no iterations.

The V-model

The V-model is a top-down development process with a bottom-up implementation process (Matthews, 2002). A special feature of the V-model is its Verification and Validation. Important characteristics: No iterations, early specification, well defined verification and validation at the end of the process.

The Spiral model

The Spiral model is developed in 1980 by Barry Boehm. It represents phases not as sequential steps but as loops. The risk assesment is very well developed in this process. Important characteristics: iterations, risk assesment, involvement of stakeholders, development of prototypes.

The Rational Unified Process (RUP)

The RUP is developed by Rational and is described in Kruchten, 2004. On the website of IBM additional information is available as well. Important characteristics: iterations, adjustable, market standard, parallel processing

Feature Driven Development (FDD)

FDD is an agile process, in which the organizational aspect is improved significantly (Nebulon Pty Ltd.). FDD uses features which provide an overview of the progress. Important characteristics: iterations, suitable for average sized projects, prioritization of requirements.

Dynamic Systems Development Method (DSDM)

DSDM is an agile process that was in use at OIB ING. It is a Rapid Application Development (RAD) method (Beynon-Davies et al, 19 J. Hunt (2006) describes very well when to use this process. Important characteristics: Agile, iterations, RAD, suitable for sma projects, heavy involvement of stakeholders.

Scrum

This process received its name for its manner of collaboration. A project team should work as a tight, integrated unit. In Leffingwell (2007) the teams are described as self-directed. Important characteristics: Agile, suitable for small teams and sma projects, errors are identified early, dependent on team work.

Extreme Programming (XP)

XP is described in Beck (2005). Four basic principles describe XP: Communication, simplicity, feedback and courage. XP is a very extreme process where teamwork is imperative. Important characteristics: Agile, frequent reviewing, iterative, pai programming, lack of attention to design and architectural issues.

FIGURE 1-6 Framework Process Page

II. FRAMEWORK USABILITY

A significant element of this research was the validation of the framework. Is the framework easy to use, and does it actually do what it intends to do? Three important validation questions were formulated to research the validity of the framework:

1. Is the design of the decision framework appropriate for users at SOFTWARE INDUSTRY?
2. Are the conclusions in accordance with the expectations?
3. Is the decision framework an appropriate tool to assist project managers in selecting a suitable software development process based on the software project at hand?

A validation process was developed to find an answer to these questions. After testing the system with actual users and experts a positive conclusion resulted. The users and experts all stated that the framework would indeed be a useful tool for conducting software projects. The users especially considered the framework to be a real eye-opener. By using the framework, projects would be conducted after thoroughly specifying imperative characteristics. An interesting result followed from one particular test conducted in the validation process. In this test almost all processes scored equally in suitability. This suggests that the project needs to be specified further. Therefore, the framework does not only indicate which processes are suitable, but also whether or not the project is specified enough. It helps in discussing unsuitable projects.

Although the final result was positive, some useful and interesting remarks were made during the **validation** process. The most interesting suggested improvements are the following:

- Framework explanation
- A page with a link to all processes for more information
- Questionnaire should be more clear, own interpretation should be minimal
- A warning when the most suitable process does not score on one (or more) characteristic(s)
- Dependences between characteristics

All these suggestions are included in the framework. It is therefore possible to state that the framework is a useful tool for large organizations and that it is in accordance with the wishes of users and experts. However due to the lack of time, a comprehensive validation is still lacking. It is imperative to analyse the usability of the framework even further, as well as a data-comparison process between projects executed with the framework and projects without.

III. Reflection

In this section the research conducted is reflected upon. In subsection 1.2.1 issues are discussed that result from this research. Finally the generalization possibilities are presented in subsection 1.2.2.

1.2.1 Issues

Disagreement in literature and practice

A main issue that came across during this research was the lack of agreement between experts and practitioners regarding the characteristics of software development processes. In this research a decision framework had to be created in which the suitability of software development processes was presented. However, there is no uniformity in literature. Some sources stated for example that agile processes are useless when applying outsourcing. This was subsequently contradicted in other literature. Furthermore, in practice, at SOFTWARE INDUSTRY, opinions differentiated as well. Therefore, the most difficult task in this research was to decide when a software development process was suitable or not. The lack of confirmation from the scientific society adds uncertainty to the framework. This should encourage the users to further validate the framework to eliminate risks. The framework at this point in time reflects the way employees at SOFTWARE INDUSTRY work with a combination of some scientific background together with my own expertise.

The lack of literature regarding characteristics of software projects

Although there was no agreement in literature regarding software development processes, there was an abundance of information available. Unfortunately, this was not the case for the characteristics of software projects which influence the suitability of processes. The characteristics found are partly based on scientific sources. However most of these sources refer more to the characteristics of a process. Other great sources are interviews, surveys and lessons learned documents. This lack of scientific foundation regarding the characteristics of software projects could be an issue for the developed decision support framework. Although an extensive validation process was executed in this research, it is imperative to continuously test and validate the framework even further, especially because this field of research is quite new and uncertain.

Experience

The final issue for this research and the framework developed is the factor *experience*. In this research the experience an organization has with software development processes is not included. This means that the framework might suggest a software development process for a particular project which, however, is not known within the organization. This framework partly functions as an eye-opener for the user. If experience was included in the framework, probably a small selection of processes (which are already in use at the organization) will often be the outcome. By removing experience from the framework, users might be confronted with the fact that other processes, which are not in use at the moment, are far more appropriate to apply. The aim of this current framework is to show the users which other software development processes might be of interest, which possibly lead to the evolution of an organization such as SOFTWARE INDUSTRY. It will be imperative that they include these interesting processes into their portfolio and mature their organization into a highly experienced software developing team with a broad array of experience. If this is all conducted in the coming years, it is possible to include experience into the framework.

1.2.2 Generalization possibilities

The framework developed in this research is based on a case study executed at SOFTWARE INDUSTRY. In this subsection the possibilities to generalize this framework to other market segments or other organizations is discussed.

One of the aims of this research was to interview developers outside of SOFTWARE INDUSTRY to find whether or not any differences exist in the manner of developing software. Unfortunately this was not feasible due to the lack of time. Therefore it is difficult to state the exact generalization possibilities for this framework. However, after the validation process conducted in this research, it seems that the framework is quite general. In this research, an expert validation was conducted. Jos Vrancken tested the framework on a project executed by *Rijkswaterstaat*. According to this expert, the framework was indeed a very helpful tool. Although further testing is a necessity, I think this framework can be generalized to other organizations in the same market segment. Even for organizations outside this market segment, this framework offers some benefits. This framework is developed for ICT related projects. Software projects in the food industry will not differ significantly from projects in the financial industry. Furthermore, the same software development processes are often used. Although this framework is useful for any software developing organization, some changes are needed. First of all the style of the entire framework should be adjusted. The current framework is designed to fit the style. Other changes, which are not necessarily needed, might be the weights and the processes. The weights currently used in the framework are partly based on the opinions and experience from personnel at SOFTWARE INDUSTRY. This might differ in other organizations or market segments. The software development processes included in this framework could also change when generalizing it to other organizations. Some processes, such as DSDM, are included because of the experience the organization has with this particular process.

IV. RECOMMENDATION

In subsection 1.3.1 the implications this research has on SOFTWARE INDUSTRY, large organizations in general & the scientific society is discussed. Sections 1.3.2 discusses recommended future research.

1.3.1 Implications

The result of this research brings about many implications for SOFTWARE INDUSTRY and for organizations in general. The first very important element is how projects should be initiated. By using this framework, team members, clients and stakeholders are forced to extensively reflect and analyze the project at hand. Each characteristic of the project is of great importance and should therefore be defined thoroughly. Furthermore making adjustments to the underlying calculations of this framework should be very limited. Although the framework might differ from the users own expert opinion and reflection, no changes should be made. These calculations are based on scientific research and practical validation. The result should be interpreted as an eye opener, not as an actual fact. However, if experience in using the framework increases, changes in factors and scales are possible.

A different and very important aspect is the usage of software development processes. At this point in time only the Waterfall model and the RUP are in use at SOFTWARE INDUSTRY. The RUP is not even used throughout the entire organization. It is of the utmost importance to train employees in using RUP much more intensively. The Waterfall model is, according to this framework, a very unsuitable process for the majority of projects executed at SOFTWARE INDUSTRY. Therefore, it is important to start expending the knowledge within the organization on RUP and possibly other processes as well.

Besides the RUP and the Waterfall model, six other software development processes are included in this framework. It is not necessary to be experienced in all these processes. However, it is highly recommended that, when a certain process is consistently considered most suitable, research concerning this process should be conducted and included in the organization. For some of these processes this means even changing organizational wide rules. To use agile processes, changes in an organization, such as changes in method of contracting and outsourcing, are imminent (Leffingwell, 2007).

1.3.2 Future research

In this research the focus was on developing a framework to help project managers at selecting an appropriate software development process for a certain project. During this research, other possibilities came forth. These possibilities were outside of the scope of this research. However, these are discussed here as they are recommended to be researched.

Single standard process

One possibility that was mentioned multiple times (especially in interviews and surveys) was to standardize one particular process. Especially the RUP was mentioned as a great process to standardize. This could be a possible solution. However, if one process is used as a standard process, it will be imperative to create three or multiple versions of this process. This means that for a smaller project, a tailored RUP process is used which fits smaller projects. If there are three different types of RUP processes developed (for small projects, average projects and large projects), this could definitely be a solid solution. It is recommended to research this possibility.

Multiple processes within one project

A different solution that was found during this research was to use multiple processes for one single project. Not only projects differ from each other, but also phases within a particular project could be very different as well. Therefore it might be a possibility to use two or three different processes for one project. For example, if the requirements are not clear, an agile process can be used for requirement management. However, because outsourcing is applied, the Waterfall model should be used in the coding phase. For this solution, the framework developed in this research could be very useful, as long as the project is correctly separated in different phases.

1.3.3 Validation

The conclusion of the validation process was generally positive. However, at this point in time it is not feasible to state that the framework is absolutely valid. Continuous testing is necessary. This framework focuses on an area in which literature does not suffice. Because of this, insecurity might become an issue. The most significant method to find whether or not the framework does what it intends to do is to compare projects conducted while using the framework and projects conducted without using the framework.

Although the framework is a valid, it is still believed to be a helpful and interesting tool to use at this point. The fact that it forces the user to evaluate and discuss the project at hand is a significant benefit.

References

1. Georgiadou, E. (2003). Software Process and Product Improvement: A Historical Perspective. Cybernetics and Systems Analyses, Vol. 39, No. 1, pp. 125-142.
2. Hesse, W. (2003). Is there an alternative for Rational Unified Process.SoftwSyst Model, Vol 2, pp. 240–247, September.
3. Rising, L.; Janoff, N.S. (2000). The Scrum software development process for small teams. IEEESoftware, Vol. 17, pp. 26-32, 2000.
4. Boehm, B. W. (1988). A Spiral Model of Software Development and Enhancement. IEEE Computer, Vol. 21, No. 5, pp. 61-72.
5. Booch, G. (2007). The Economics of Architecture-First. Software, IEEE, Vol. 24, No. 3, pp. 10-11, September-October 2007.
6. Dijkstra, E.W. (1972). The Humble Programmer. Communications of the ACM, Vol. 15, No. 10, pp. 859-866, October 1972.
7. Kleijnen, J.P.C. (1995). Verification and Validation of Simulation Models. European Journal ofOperational Research, Vol. 82, pp. 145-162.
8. Dybå, T.; Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. Information and Software Technology, Vol. 50, pp. 833-859, 2008.
9. Aughenbauch, J.M.; Paredis, C.J.J. (2004). The Role and Limitations of Modeling and Simulation in Systems Design. Paper presented at the ASME International Mechanical Engineering Congress andRD&D Expo, held in Anaheim, California, 13-19 November, 2004.
10. Georgiadou, E. (2003). Software Process and Product Improvement: A Historical Perspective. Cybernetics and Systems Analyses, Vol. 39, No. 1, pp. 125-142.
11. O'Keefe, R.M.; O'Leary, D.E. (1993). Expert system verification and validation: a survey and tutorial in Artificial Intelligence Review, Vol. 7, pp. 3-42, 1993.
12. Fayad, M.; Hamu, D.; Brugali D. (2000). Enterprise frameworks characteristics, criteria, and challenges. Communications of the ACM, Vol. 43, No. 10, pp. 39-46.

AUTHOR(S) PROFILE

Under the Guidance of Dr. G. Satyanarayana Prasad received Ph.D. degree in Computer Science in the Faculty of Engineering in 2006 from Andhra University, Andhra Pradesh. He Completed is MS in Computer Science & Engineering from A & M University, USA, 1984; He is currently working as Professor in the Department of Computer Science and Engineering, R.V.R & JC Engineering College, Guntur. His current research is focused on Image Processing. He has published several papers in IEEE journals and various National and International Journals