Decision Making Process Model Framework

V. Rajiv Jetson[1], G. Satyanarayana Prasad[2], D. V. Chandra Shekar[3]

[1]Associate Professor, Dept of CSE, Kallam Haranadha reddy Institute of Technology

[2]Professor and Dean, Dept of CSE, R.V.R& J.C college of Engineering

[3]Associate Professor, Dept of CS, T.J.P.S.College

Abstract: *The paper is focused on designing a decision framework which can assist project managers selecting an appropriate software development process for the project at hand. This decision framework should be based on the main element, on which the effectiveness of a process is dependent, i.e. the characteristics of software projects. This section explains the basic design of the decision framework, in which choices are made and what this entails for the matching of software processes with the characteristics of projects. From the previous software engineering models, information is gathered based on the characteristics of the software development processes which initiated four analyses on imminent characteristics of projects. By mapping the suitability of each process (based on their main characteristics) on the characteristics of projects, a basis for the framework is made. The mappings of the processes are conducted in section 1.2 and 1.3 of this work. Section 1.1 explains the basic design of the decision support framework. This work answers the research sub question: "How this framework should be designed to be suitable for Project Manager?"*

Keywords: framework, suitability, decision support system.

1. Introduction

The research paper is focused on designing a decision framework which can assist project managers selecting an appropriate software development process for the project at hand. This decision framework should be based on the main element, on which the effectiveness of a process is dependent, i.e. the characteristics of software projects. This section explains the basic design of the decision framework, in which choices are made and what this entails for the matching of software processes with the characteristics of projects. By mapping the suitability of each process (based on their main characteristics) on the characteristics of projects, a basis for the framework is made. The mappings of the processes are conducted in section 1.2 and 1.3 of this work. Section 1.1 explains the basic design of the decision support framework.

At the end, Framework is presented to assist project managers in selecting a suitable software development process based on the characteristics of the project. The first step is to analyse whether or not the software development processes are useful when a particular characteristic reaches a certain scale. This is presented in a table in section 1.2. Section 1.3 is similar to section 1.2. However, in section 1.3 a suitability factor is given whenever a software development process fits a certain scale. Some processes are possible when a particular characteristic reaches a certain scale, however, it might be less suitable than other processes. Finally in section three the mapping of the processes is applied in the framework. Screenshots, clarification, design choices and conclusions are given.

1.1 Design of the framework

Here the significant characteristics of projects that play a role when selecting a software development process are discussed. This framework should, by indicating the scale of each characteristic, provide an answer of which process or processes should be used concerning this particular project. The design chosen for this particular framework is a questionnaire form. For each characteristic, the project manager can select the appropriate scale. Figure 1-1 presents the first rough sketch of the proposed framework. For each characteristic, the project manager can select the appropriate scale. This scale goes from 1 to 1. For example, if the budget is very low, the project manager selects, in the questionnaire i.e. the number 1. If the budget is very high, the 1st scale is selected. On each scale, certain processes are linked which are suitable for that particular scale. For each characteristic a weight can be given. By multiplying the processes resulted from each characteristic with the weights of each characteristic, the framework provides a certain score, for each process. The process with the highest score should be selected for that particular project.
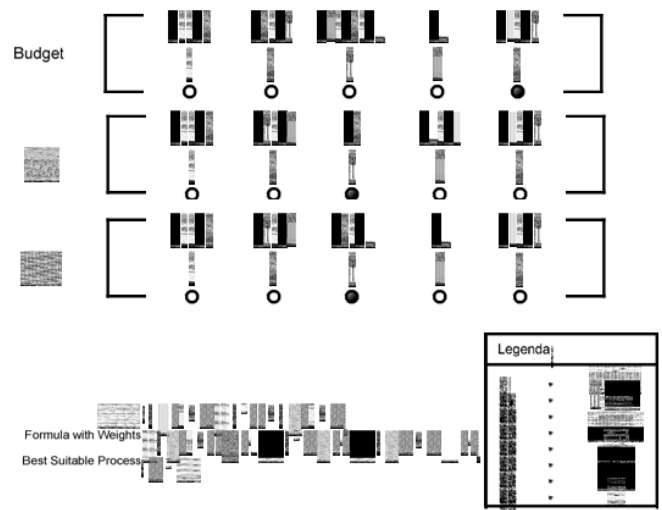

Figure 1-1: First sketch of possible decision framework

1.2 Mapping the Process to Characteristics

This section focuses on the main issue i.e. significant characteristics of each process and the characteristics of projects conducted & were found in the previous works. For the framework, these two information sources need to be combined. In this section the software development processes are matched with the characteristics of software projects. By

analysing for each characteristic whether or not the process fits, a conclusion can be made based on all the individual scores combined. The characteristics of software projects in the framework are scaled from 1 till 1, 1 being very low and 1 being very high. On each scale a decision needs to be made whether or not a process fits this. A certain process might only be useful when a characteristic, for example, the characteristic team size, is very low or low. This could entail agile processes. Agile processes have a characteristic being suitable for projects with a small project group. When the team size is very high, an agile process is less suitable. For each characteristic the processes are matched on the five scales. This overview is presented in tables 1-1 and 1-2. Table 1-1 represents the mapping of the software development processes Waterfall model, V-model, Spiral model and the RUP. Table 1-2 consists of the processes DSDM, XP, Scrum and FDD.

The decision whether or not a process fits a particular scale is based on theoretical information regarding this process and on conducted interviews. During this entire research opinions in literature and opinions in interviews were gathered. At this point in time understanding concerning the subject is very scarce. For example, the creators of FDD mention that agile processes are suitable not just for small project groups, but for larger groups as well (Nebulon Pty Ltd., 2009). However, Erickson (2001) state that agile process are limited to smaller projects with a small amount of team members. This difference in opinion does not only limit itself to this characteristic or these processes. Every author and every employee at has their own opinion regarding software development. However, in this research a framework had to be made. Therefore, certain decisions needed to be made regarding the suitability of software development processes. The following tables present an overview of when a certain process is suitable regarding the project's characteristics. These choices are discussable and are not to be seen as fact.

These are choices made based on own expertise, interviews and on literature research. Based on these tables it is possible to create a basic framework in which, when selecting for each characteristic which scale the project has, the outcome will be a suitable software development process. However, these tables only represent whether or not a process is possible when a factor has a certain scale. Multiple processes are matched on the same scale of a characteristic. How effective each process is on each particular scale, is presented next in section 1.3

The main conclusion based on these tables is that smaller projects, low budget, small time schedule, and small team size fit agile processes better than the more traditional processes. Furthermore, agile processes are more capable of handling projects surrounded by an uncertain and risky environment. The RUP process together with the Waterfall model in use at CASE STUDY PROJECT ING fits regularly the entire moderate scales. For e.g., the RUP is not suitable for very small projects and very large projects, however, it does fit everything in between.

1.3 Separation of Process by Suitability

In the previous section the software development processes were mapped on the projects characteristics. This is presented in tables 1-1 and 1-2. For example, concerning the characteristic Team Size, Scrum is only applicable when the team size is very small. However, the RUP can be used when the team size is small, medium or big. In these tables no separation was made between the processes that matched a certain scale. If two processes can both be applied when the team size is medium, this does not necessarily mean they are equally suitable. In this section the processes that fit a particular scale, are separated by suitability.

Table 1: (Part 1) Process Categorisation

	Waterfall	V-model	Spiral	RUP
Project Size	High, Very High	High, Very High	High, Very High	Low, Medium, High, Very High
Team Size	Large, Very Large	Large, Very Large	Medium, Large	Small, Medium, Large
Requirements Maturity	Very Mature	Very Mature	Mature, Very Mature	Little Mature, Medium Mature, Mature, Very Mature
Team Relationship	Very Dire, Dire, Medium, Good	Very Dire, Dire, Medium, Good	Dire, Medium, Good, Very Good	Dire, Medium, Good, Very Good
Client's Commitment	Very Little, Little, Medium, High, Very High	Very Little, Little, Medium, High, Very High	Little, Medium, High, Very High	Little, Medium, High, Very High
Scope clearness	Clear, Very Clear	Clear, Very Clear	Medium, Clear, Very Clear	Little Clear, Medium, Clear, Very Clear
Risk clearness	Clear, Very Clear	Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear	Little Clear, Medium, Clear, Very Clear
Environmental Stability	Stable, Very Stable	Stable, Very Stable	Little Stable, Medium, Stable, Very Stable	Little Stable, Medium, Stable, Very Stable
Stakeholders Flexibility	Not Flexible, Little Flexible, Medium, Flexible	Not Flexible, Little Flexible, Medium, Flexible	Medium, Flexible, Very Flexible	Little Flexible, Medium, Flexible, Very Flexible
Method of contract	Fixed	Fixed	Fixed	Fixed Pricing, Time & Material
Outsourcing	No, Yes	No, Yes	No, Yes	No, Yes

In the overview below i.e. table 1-3, this separation is presented. These separations are based on theoretical information (information regarding the characteristics, advantages and disadvantages of software development processes presented in work three) and interviews. As is depicted below, a suitability factor is given to each colour. This means that when a process is suitable on a certain scale, it gets the colour green. However, when a process fits the scale but is less suitable than other processes, they get a lower factor, for example yellow or orange. When the colour green is given, the score (which is dependent on the weight given) is multiplied by alpha. When the colour yellow is given the score is multiplied by beta and orange presents a multiplication of gamma. These three scores can be set by the user. In this research the respective scores given to these suitability factors are 1, 0.71 and 0.1.

Table 1-2: Process Categorisation

	DSDM	XP	Scrum	FDD
Project Size	Very Low, Low	Very Low	Very Low, Low	Very Low, Low, Medium
Team Size	Very Small, Small	Very Small	Very Small, Small	Very Small, Small, Medium
Requirements Maturity	Little Mature, Medium, Mature, Very Mature	Not Mature, Little Mature, Medium, Mature, Very Mature	Not Mature, Little Mature, Medium, Mature, Very Mature	Little Mature, Medium Mature, Mature, Very Mature
Team Relationship	Good, Very Good	Very Good	Very Good, Good	Good, Very Good
Client's Commitment	High, Very High	Very High	Very High	High, Very High
Scope clearness	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear
Risk clearness	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear	Not Clear, Little Clear, Medium, Clear, Very Clear
Environmental Stability	Not Stable, Little Stable, Medium, Stable, Very Stable	Not Stable, Little Stable, Medium, Stable, Very Stable	Not Stable, Little Stable, Medium, Stable, Very Stable	Not Stable, Little Stable, Medium, Stable, Very Stable
Stakeholders Flexibility	Flexible, Very Flexible	Very Flexible	Very Flexible	Flexible, Very Flexible
Method of contracting	Time & Material	Time & Material	Time & Material	Fixed Pricing, Time & Material
Outsourcing	No	No	No	No, Yes

For example, in the first table the characteristic budget is presented. When the budget is very low, Scrum or XP, which are very light processes, are both very suitable. DSDM and FDD, which are a little more complex, are suitable, but not as much as Scrum or XP. Therefore they get the colour yellow. The sequences of names do not represent any preference, only the colours do. Scrum and XP in the first scale of budget are therefore equally suitable. This manner of separation is conducted on all scales and on all characteristics mentioned in tables 1-1 and 1-2. The choice made here concerning the suitability of a process is based on the characteristics of processes found in literature and described in work 3, and on interviews and opinions. However, because there is no information regarding the suitability of processes on certain projects, these choices are discussable and are not to be seen as fact.

Table 3: Suitability of Processes on Characteristics

α	*1
β	*0.71
γ	*0.1

Suitability Factor

Scrum	FDD	RUP	RUP	Waterfall
XP	RUP	FDD	Waterfall	V-model
DSDM	DSDM	...	V-model	RUP
FDD	Scrum	...	Spiral	Spiral

Project Size	1	2	3	4	1

Scrum	FDD	RUP	RUP	Waterfall
XP	DSDM	FDD	V-model	V-model
FDD	RUP	Spiral	Waterfall	...
DSDM	Scrum	...	Spiral	...

Team Size	1	2	3	4	1

The suitability of a process often depends on the extent of the project. Whether the size of a project is expressed in function points, deliverables or man hours is left open in this characteristic. Only the scale of the size needs to be stated. When a project is very big, lightweight processes such as the agile processes, lose their suitability. For large projects often more documentation is needed because face-time is limited. Therefore, for the characteristic project size, as well as team size, the agile processes, such as Scrum and XP are only

suitable on smaller scales and the Waterfall process or the V-model only for the larger scales. This is also confirmed by Runeson & Greberg.

Scrum	FDD	RUP	RUP	RUP
XP	DSDM	FDD	FDD	V-model
…	Scrum	DSDM	Spiral	Waterfall
…	XP	XP	DSDM	Spiral
…	RUP	Scrum	Scrum	FDD
…	…	…	XP	XP
…	…	…		Scrum
…	…	…		DSDM
Requirements Maturity	1	2	3	4	1

Software development processes handle the process of defining requirements differently. Traditional processes, such as the Waterfall model, require that the requirements of a project are determined early on in the process. However, this is not always a possibility. If a client does not know exactly what his/her wishes are, a solid definition of requirements will become an issue. Agile processes accept this fact and furthermore, understand that the environment is very dynamic. Therefore, agile processes have the ability to handle uncertainties much more effectively. For the characteristic "Requirements Maturity", this means that agile processes are always possible to apply. However, the Waterfall model and even the RUP require more mature requirements to be suitable.

The characteristics "Team Relationship" and "Client's Commitment", describe the teamwork within the entire organization. These characteristics are imperative for the suitability of a process. Processes in which teamwork is of great importance, the relationship within the team must be excellent and the client must be committed greatly. This is the case for agile processes. Scrum and XP for example need the client to be onsite for the entire project. If the relationship within the team is dire, the processes lose their suitability (Palmer & Felsing, 2002; Rising & Janoff, 2000). The Waterfall model depends heavily on documentation and less on face to face time. Therefore, when team work is difficult, and the client is not at all committed, documentation is a possible solution. This heavy use of documentation however, results in much overhead when the project only concerns one department in which the relationship is excellent. The RUP also depends on a committed client and a good relationship within the team.

In dynamic markets, software projects are often covered with risks. Software development processes should be able to cope with these risks. However, not every process does this effectively. The Spiral model is a process that does cope with risks. Within this process, risk assessment is of significant importance (Boehm, 1988).

Waterfall	Waterfall	RUP	RUP	RUP
V-model	V-model	Spiral	FDD	FDD
…	RUP	Waterfall	Spiral	DSDM
…	Spiral	V-model	DSDM	Spiral
…	…	…	Scrum	XP
…	…	…	Waterfall	Scrum
…	…	…	V-model	…
Team Relationship	1	2	3	4	1

Waterfall	Waterfall	RUP	RUP	RUP
V-model	V-model	Spiral	FDD	FDD
…	RUP	Waterfall	Spiral	DSDM
…	Spiral	V-model	DSDM	Spiral
…	…	…	Waterfall	XP
…	…	…	V-model	Scrum
…	…	…	…	Waterfall
…	…	…	…	V-model
Client's Commitment	1	2	3	4	1

XP	XP	Scrum	Spiral	Spiral
Scrum	Scrum	RUP	RUP	RUP
DSDM	DSDM	DSDM	FDD	Waterfall
FDD	FDD	FDD	DSDM	V-model
…	RUP	XP	Scrum	FDD
…	…	Spiral	XP	DSDM
…	…	…	V-model	Scrum
…	…	…	Waterfall	XP
Scope Clearness	1	2	3	4	1

Spiral	Spiral	Spiral	Spiral	Spiral
Scrum	Scrum	RUP	RUP	RUP
DSDM	DSDM	DSDM	FDD	Waterfall
XP	FDD	FDD	DSDM	V-model
FDD	XP	XP	Scrum	FDD
…	RUP	Scrum	XP	DSDM
…	…	…	V-model	Scrum
…	…	…	Waterfall	XP
Risk Clearness	1	2	3	4	1

Agile processes and RUP as well, have great risk mitigation. Because of the iterative and incremental approach, risks are found early in the process. Unfortunately, the Waterfall model and the V-model do not effectively tackle these risks (the V-model does have a better approach of validation and verification comparing to the Waterfall model). When applying these processes, risks are often found at the end of the project, where costs for repair are significant. Therefore, the Waterfall model as well as the V-model can only be used when the risks are very clear (Royce, 1970; Sommerville, 2007). As was mentioned before, a project is very dependent on its stakeholders. This also counts for software development processes. Agile processes require the client to be very flexible. The client needs to accept that requirements are not defined at the very beginning.

Scrum	Scrum	Scrum	Scrum	V-model
XP	XP	XP	XP	Waterfall
DSDM	DSDM	RUP	RUP	RUP
FDD	FDD	FDD	FDD	Spiral
…	RUP	DSDM	Spiral	DSDM
…	Spiral	Spiral	DSDM	FDD
…	…	…	V-model	XP
…	…	…	Waterfall	Scrum
Environmental Stability	1	2	3	4	1

Waterfall	Waterfall	RUP	RUP	RUP
V-model	V-model	Spiral	FDD	FDD
…	RUP	Waterfall	Spiral	DSDM
…	…	V-model	DSDM	Spiral
…	…	…	Waterfall	XP
…	…	…	V-model	Scrum
Stakeholders Flexibility	1	2	3	4	1

Furthermore, a cost overview and a time schedule cannot be frozen as well, because these might change during the process. Therefore, the client must accept that he has more uncertainty concerning his deliverables. This is not the case for traditional processes. The Waterfall model states everything the client needs to know at the beginning of the project

RUP	Waterfall
FDD	Spiral
DSDM	V-model
Scrum	RUP
XP	FDD
Method of Contracting	Time & Material	Fixed Pricing

The suitability of software development processes depends on the method of contracting. If the method time/material is applied, all processes may be suitable. However, when fixed budget is applied, certain processes lose their effect. The processes that lose their effect are the agile processes. An agile process is not applicable when the price, and thus the requirements, are frozen.

RUP	Waterfall
FDD	V-model
DSDM	Spiral
Scrum	RUP
XP	FDD
Spiral	…
Waterfall	…
V-model	…
Outsourcing	No	Yes

In relationship to the previous characteristic is the characteristic "Outsourcing". In many organizations outsourcing is extensively used. Especially for software developing organizations many cost benefits as well as expertise are available when applying outsourcing, or even offshore outsourcing (outsourcing to overseas countries). This however heavily impacts the suitability of software development processes. If there is no outsourcing, all

software development processes are suitable; however, the Waterfall model and the V-model are less suitable. This is caused by its extensive use of documentation which is of less importance when no outsourcing is applied. However, when outsourcing is applied, the Waterfall model and the V-model are most suitable. The other processes, depending more on teamwork and on site clients, are less suitable (agile and RUP).

1.4 Final Decision Framework

Based on the mapping solution presented in the previous sections, a final framework is developed. This framework consists of three elements. The first element is a questionnaire form in which all characteristics found in work four are included; the second is a calculation tab and finally a conclusion tab. These three components are separately presented by screenshots and discussed in the next subsection. In subsection 1.4.2, the design choices made are explained. Finally, in subsection 1.4.3, a conclusion is given concerning the framework.

1.4.1 Screenshots
With reference to the previous, eleven characteristics of software projects were found which influence the suitability of a software development process. Each characteristic can differ in scale (as discussed in previous sections). In the first element of the framework a questionnaire form is presented. In this form it is possible for a user to state the scales of each characteristic for the particular project. However, in this final framework an average project at is presented to fully show how the framework functions. The scales and weights for this average CASE STUDY PROJECT are presented below.

Table 1-4: Scale Weights of Average Project at Case Study

Characteristic	Scale	Weight
Project Size	4	4,1
Team Size	1	2,9
Requirements Maturity	3	4,4
Team Relationship	2	3
Client's Commitment	2	3
Scope Clearness	3	3,3
Risk Clearness	1	3,8
Environmental Stability	1	4,3
Stakeholders Flexibility	2	2,8
Method of Contracting	Fixed Pricing	2,9
Outsourcing	Yes	3,1

In figure 1-2 a screenshot is presented of the questionnaire form in the final framework. The user can indicate the scale of the characteristic by selecting the circle. The goal of this component of the framework is that the project manager should indicate for each characteristic which scale it has. For example, the project size can be very big, there are a lot of team members involved and the requirements are still very immature. These eleven characteristics are all the relevant characteristics that represent a software project (relevance as in "influencing the suitability of software development processes). However, it could be a possible that a certain characteristic is not of interest for the project manager regarding a particular project. Therefore, a "Not Available" option is included in the framework. If this is selected for a

certain characteristic, it is excluded from the calculation.

For every characteristic a certain weight is given. This represents the importance of the factor (which is collected in the calculation). This weight is based on a survey distributed among the employees of and own expertise. However, it is possible to change this if the project manager and client agree on others. This is automatically changed in the calculation as well.

Screenshot of Framework – Questionnaire 1& 2

When a selection is made for every characteristic and the weight is given, the calculation tab shows each individual score. This tab is presented in figure 1-3. The first element of this tab is the "Calculation Variable". As was explained before, software development processes can differ in suitability for a certain scale. In this framework, the suitability scores are 1, 0, 71 and 0, 1. This can be changed by any user. However, it is necessary that the green colour (alpha) is higher than the other two and that the yellow colour (Beta) is higher than the orange colour (Gamma). The second and main element of this part of the framework is the calculation of scores for the software development processes on each individual characteristic. At the right of each individual calculation table a colour table is presented. This colour table represents the tables depicted in section 1.3. In this particular figure, the scale 4 is selected for "Project Size". This means that the project is rather large. In the table presented in section 1.3, it is shown that when the project size has scale four, the RUP is most suitable. The Waterfall model is suitable as well, but less. Therefore this is presented by a yellow colour. The V-model and Spiral model are also suitable nevertheless, again, less than the other two software development processes. Therefore they are given an orange colour. This is calculated in the bigger table. As the figure shows, when a process has a high suitability, it receives the score 1. If the process is presented in the yellow cells, it receives the score 0.71. Orange receives 0.1 as score. These scores are then multiplied with the weight given.

This is conducted for every characteristic presented in the questionnaire form. The score for every individual characteristic is then summed up, and presented in the conclusion tab. Figure 1-4 depicts this tab.

The graph presented in this tab gives a better overview on how the processes have scored. In figure 1-4 a second table is presented (Calculation table for best and second best processes). Because the choices made in this section regarding the suitability of the processes are not factual, other high scoring processes are presented as well. In this framework a maximal difference of 21% is chosen. However, this is possible to change. If the score of a particular process finds itself between the maximal score and the maximal score minus 21% this is presented as "other possible processes". This is also presented on the right side of the figure. The process with the highest score receives a green colour. The processes, (maximal of two) which are within the 21% score range, are presented by the colour yellow.

Figure 1-3: Screenshot of Framework – Calculation

1.4.2 Design Choices
For this framework certain design choices are made. In this subsection, these choices are discussed. Almost all aspects are based on specific arguments and are deliberated on great extent. The following design aspects are discussed:

- Colours for the framework
- The positioning of the score tables
- The Calculation Variables
- Allowing weights to change
- The "Not Available" option
- The graph in the conclusion tab

Colours of the framework
The colouration of the entire framework represents the style of ING. To provide the users with a framework which fits their comfort zone was considered very important. Furthermore, by using these colours, it clearly shows that this particular framework was designed for ING usage. The colour orange is used throughout the entire report, and it is appropriate to apply this as well to the framework.

The positioning of the score tables
The score tables are the smaller tables presented in figure 1-3, which present the suitability of the processes (green/yellow/orange). As is shown in work five, the original thought was to position these tables in the questionnaire form. However, it was decided to change this. In the original design, it was possible to see directly which consequences each scale has on the suitability of processes. Therefore, it could be possible that users would change their scales to select their favourite process to use. This overview of scores should still be present. However, it should not be obviously connected to the selection of the scales. Therefore, these score tables are moved to the calculation tab (which should not be of interest for the user).

The Calculation Variables
In the framework calculation variables are included. When multiple processes fit on a certain scale, it is possible that they differ in suitability. These calculation variables indicate their suitability. The actual variables are alpha, beta and gamma. In this particular framework these variables are given a score. These are 1, 0.71 and 0.1. It is possible to change these scores. However, it is imperative that the ranking of suitability and thus score stays the same.

Allowing weights to change
The weights presented now in the framework are based on a survey conducted at CASE STUDY PROJECT ING. However, it could be possible that a project manager, together with a business partner, decide that other weights should be given. This could for example be dependent on the current market situation. Therefore, the possibility to change these weights is included in the framework. These changes can be made in the questionnaire form. These changes are then automatically applied in the calculation tab.

Figure 1-4: Screenshot of Framework – Conclusion

The "Not Available" option

In the framework, the not available option is added. In certain projects, particular characteristics might not be of interest or unknown. This can be applied by selecting the N/A option. When this is selected, the entire characteristic is removed from the calculation.

The graph in the conclusion tab

In the conclusion tab a graph is added which presents the scores of each software development process on all the characteristics. This graph is added to provide the user with a better overview of the scores. Furthermore, the comparison between processes is clearer and the results can be communicated better to other stakeholders.

The multiple possible processes in the conclusion

In the conclusion tab it is possible that multiple processes are presented. This copes with the uncertainty in the framework. Usually one process comes out to be most suitable. However, if a second or third process is very close as well, this does not immediately mean that these are not suitable. Therefore a range is given to cope with this. When other processes fall in the range of the maximal score minus 21% they ought to be seen as suitable. This range is chosen because it deletes certain processes which are not suitable at all, but still leaves some room for discussion.

2. Conclusion

The framework designed is based on the entire research conducted. By analysing the software development processes, significant characteristics, advantages and disadvantages became clear. These were mapped on the characteristics of software projects found by analysing literature and conducting interviews. A survey was distributed among employees to review and verify the characteristics found. This mapping resulted in the framework presented in this work. The framework designed supports project managers at in selecting a certain software development process based on the characteristics of the particular project.

References

[1] Booch, G. (2007). The Economics of Architecture-First. *Software, IEEE,* Vol. 24, No. 3, pp. 10-11, September-October 2007.

[2] 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 and RD&D Expo,* held in Anaheim, California, 13-19 November, 2004.

[3] Beynon-Davies, P.; Carne, C.; Mackay, H.;Tudhope, D. (1999). Rapid Application Development (RAD): an empirical review. *European Journal of Information Systems,* Vol. 8, pp. 212-223, 1999.

[4] Boehm, B. W. (1988). A Spiral Model of Software Development and Enhancement. *IEEE Computer*, Vol. 21, No. 5, pp. 61-72.

[5] Brooks, F.P. (1987). No silver bullet: Essence and Accidents of Software Engineering. *IEEE Computer,* Vol. 20, No. 4, pp. 10-19.

[6] Dijkstra, E.W. (1972). The Humble Programmer. *Communications of the ACM,* Vol. 15, No. 10, pp. 859-866, October 1972.

[7] Dybå, T.; Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review on

[8] *Information and Software Technology,* Vol. 50, pp. 833-859, 2008.

[9] Erickson, J.; Lyytinen, K.; Siau, K. (2005). Agile Modeling, Agile software development, and extreme programming: the state of research. *Journal of Database Management,* Vol. 16, No. 3, pp. 226-237, 2005.

[10] Fayad, M.; Hamu, D.; Brugali D. (2000). Enterprise

frameworks characteristics, criteria, and challenges. *Communications of the ACM,* Vol. 43, No. 10, pp. 39-46.

[11] Georgiadou, E. (2003). Software Process and Product Improvement: A Historical Perspective.

[12] *Cybernetics and Systems Analyses,* Vol. 39, No. 1, pp. 125-142.

[13] Hesse, W. (2003). Dinosaur meets Archaeopteryx? Or: Is there an alternative for Rational Unified Process. *Softw Syst Model*, Vol 2, pp. 240–247, September.

[14] Illgen, J.D.; Gledhill, D.W. (1999). 21st Century Verification and Validation Techniques for Synthetic Training Models and Simulations. *I/ITSEC (The Interservice/Industry Training, Simulation & Education) Conference 1999: S*

[15] Kroll, P.; Kruchten, P. (2003). *The Rational Unified Process Made Easy: A Practitioners Guide to the RUP*. Addison-Wesley.

[16] Liu, L.; Horowitz, E. (1989). A Formal Model for Software Project Management. *IEEE Transactions on Software Engineering.* Vol. 15, No. 10, pp. 1280-1293.