A Case Study for Prioritizing Features in Environments with Multiple Stakeholders

Eduardo Cristiano Negrão
Aeronautical Institute of Technology
São José dos Campos, São Paulo – Brazil
eduardocnegrao@gmail.com

Eduardo Martins Guerra
Aeronautical Institute of Technology
São José dos Campos, São Paulo – Brazil
guerraem@gmail.com

Abstract

In corporations where the focus is a very dynamic business inserted in web environments, agile methods can fully meet almost all needs. However, in some particular companies, there are multiple stakeholders, who represent different interests in prioritizing activities. There is, consequently, a heavy challenge to implement agile methodologies which deal with such conflicts in order to prioritize the features of the system. It is important to focus on higher earned value as possible and consider the technical risks exposed by the development team. These barriers often lead these companies to abandon such agile methods, incorporating an approach of a chaotic work environment. This paper proposes an agile technique for prioritizing features in environments with multiple stakeholders and reports a successful experience in its usage.

Categories and Subject Descriptors D.2.9 [**Software Engineering**]: Management – cost estimation, productivity, programming teams, software process models, time estimation.

General Terms Management, Measurement, Experimentation, Human Factors, Theory, Verification.

Keywords Prioritization, Estimating, Planning, Agile, Stakeholders, Conflict of Interests.

1. Context

The software product, which is contextualized in this paper, involves the work of a development team aligned to another team of business analysts. It consists in an e-commerce system focused on health care. It involves the marketing of high cost products of hospitals, such as orthosis, prostheses and other special surgical materials. The application has features to meet the needs of five different actors: hospitals buyers, product suppliers, health plan operators, hospital service providers and system administrators.

The agile methodology used to manage the work is the Scrum [4]. Three developers, a professional in quality assurance and a Scrum Master compose the team that meets the demands of maintenance and evolution of this software.

In this case, the application does not have a single product owner. Actually, it has one person responsible for each area, giving a total of five main stakeholders. All of them participate in the planning meeting in order to prioritize product backlogs and sprint backlogs, bringing demands from customers that they represent. When possible, they have tried to prioritize it within general consensus. However, this reality is often very different: there is a wide disparity of interests, leading to conflicted priorities and increasing the planning efforts.

Some proposals [1] attempt to minimize the planning efforts. However, they do not address all solutions to solve the problems in contexts which involve more than one stakeholder from different business areas and a small development team to attend the needs. This paper documents the adopted solution to define a technique to prioritize software requirements impartially, organizing the product backlog and the sprint backlog, aiming solely at increasing the return on the corporation's investment. This work also describes the search for consolidated existing techniques and how they can be adapted and combined into the given context.

Nowadays, with the high dynamism and diversity of internet businesses, this problem is common and recurrent in such corporations. As a consequence, it is extremely important that studies aimed at mitigating these problems are considered and evolved.

This study documents the solutions adopted to define a technique for prioritizing software requirements in environments with multiple stakeholders by organizing the artifacts in an impartial way for product maintenance planning, in order to maximize the return on the corporation's investment. It also describes the researches performed to go through already established techniques in the market; and how and why they were combined and adapted to suit this context. The proposed technique has been implemented in a real corporate context and its results were collected and then evaluated by means of some established metrics.

SPLASH'11 Companion, October 22–27, 2011, Portland, Oregon, USA.
Copyright © 2011 ACM 978-1-4503-0942-4/11/10…$10.00.

2. Major Occurred Problems

As mentioned previously, the company has faced the stakeholders' conflict of interests to prioritize their customers' demands. As a result, this context increases the development team's effort to expose the barriers and technical risks, which are important to maintain a higher quality of the software.

Stakeholders have thus considered abandoning the agile methods, as these had been understood by them as being very inflexible, since they did not allow new business demands to interrupt an iteration in course. But, actually, such demands had often been considered urgent, however, after being developed, these features were never or rarely used. This situation avoids the development of other demands that could be developed primarily to deliver greater earned value to the application, meeting the real needs of a higher number of customers.

Therefore, it has been extremely important to the project's success to define better ways to prioritize features mitigating the risks of building low-value software to the customers and avoiding the abandonment of agile techniques.

3. Initial Approach: Researching Solutions

Given these problems, the major objectives were to define a planning technique to prioritize development features in this complex environment. The approach was to use agile values and techniques in order to decrease those planning efforts and balance the existing technical risks and business interests.

As a research base for studies of existing solutions and for the realization of the tailoring of the existing processes, the book "Agile Estimating and Planning" [1] was used. It proposes the usage of already established techniques in environments that are using agile methodologies. Then, the resulting studies combined those theories with traditional methods of risk management from the Professional Management Body of Knowledge (PMBOK) [3].

Some solutions have been chosen for implementation in the current work, in the use of Scrum. The proposed technique has then been implemented in a real corporate context and the results were collected and evaluated by means of some metrics.

The researched solutions are based on already established techniques in the agile world such as the Relative Weighting, Kano, Theme Screening and Theme Scoring [1].

The Relative Weighting method was chosen as the approach which best fits the solution of the encountered problem.

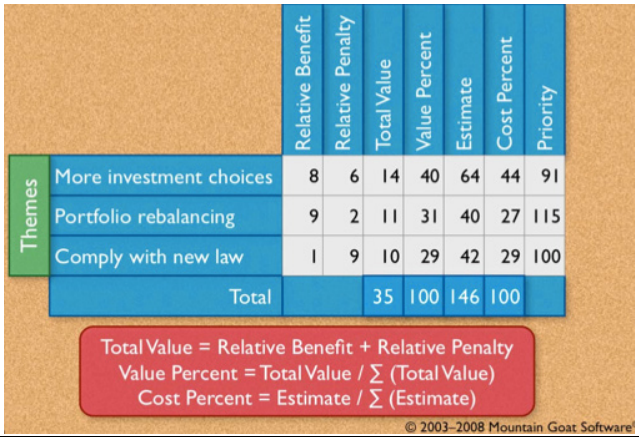
This method has been adopted because it provides a more efficient way to classify the priorities for each requirement. In this technique, not only the relative benefit of adding that feature is considered, but also how much the product would be hurt if it were not included. To get the complexity of each story [2], story points are estimated by methods such as the planning poker [1].

The Relative Weighting contributed positively to the planning activities, since it reaps the business value and technical costs score in a more democratic approach.

To get the value of each story, it is necessary that the stakeholders score, from 1 to 9, the benefit that it brings, being 1 the story with the lowest benefit and 9 the one with the higher benefit. Then, the story needs to be scored, also from 1 to 9, to punctuate the penalty a story can bring if it is abandoned. It is given 1 point to the story that brings no penalty and 9 the story that has a higher penalty to be abandoned.

The cost of the software development, in this method, is achieved by reaching a relative measure on how big is each story. The development team uses the estimated size in points, known as Story Points, obtained by methods like the Planning Poker, in which the development team scores the size or the cost of each story according to its technical complexity and importance, a story over another, following the Fibonnacci scale [1].

Given all three scores, then sum up the benefit added to the penalty, resulting in a number known as "Total Value". Having then the total value and the cost of each feature, it is still necessary to obtain another derived value: the percentage that each of these two scores represents on the whole. Finally, to give the value of the priority of each feature, the percentage of each total is divided by the percentage of each corresponding cost [1]. These ideas are better explained in figure 1.



		Relative Benefit	Relative Penalty	Total Value	Value Percent	Estimate	Cost Percent	Priority
Themes	More investment choices	8	6	14	40	64	44	91
	Portfolio rebalancing	9	2	11	31	40	27	115
	Comply with new law	1	9	10	29	42	29	100
	Total			35	100	146	100	

Total Value = Relative Benefit + Relative Penalty
Value Percent = Total Value / ∑ (Total Value)
Cost Percent = Estimate / ∑ (Estimate)

© 2003–2008 Mountain Goat Software®

Figure 1. Relative Weighting Method [7]

One of the other major approaches researched (though this one comes from projects based on more traditional methods) is the risk management suggested by the Project Management Institute (PMI). With the risks identified, it is possible to classify how they could impact on the organization, being also possible to measure if they would be low, medium or high.

A tool named Probability and Impact Matrix can be used as a concept to represent the analytical structure of risks defined, in which it can view the level of the numbered risks and classify them into four quadrants. As follows, it is viable to get the degree of the possibility that the risk can occur; what impact it could cause if it occurred; which actions should be taken; who would be those responsible for addressing the identified risks and how should it be proceeded to monitor the risk. As a result, it is possible to evaluate whether the project that is being planned is a project of high, medium or low risk and if it is feasible to continue.

The Probability and Impact Matrix and its four quadrants are represented in Figure 2.

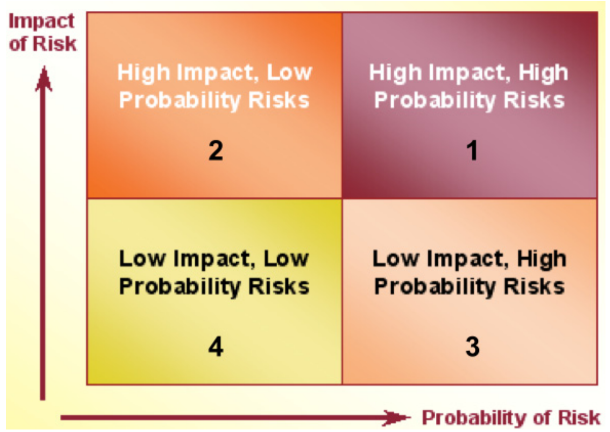

Impact of Risk
High Impact, Low Probability Risks
2
High Impact, High Probability Risks
1
Low Impact, Low Probability Risks
4
Low Impact, High Probability Risks
3
Probability of Risk

Figure 2. Probability & Impact Matrix *[8]*

The quadrants are classified as follows [3]: (1) High probability of risk that can very aggressively impact the project; (2) Average probability of risk, yet can be very costly to the company and should be monitored routinely; (3) High probability of risks and cause little impact to the company; and (4) Risk of low probability and low impact on the project. As a conclusion of these researches, achieving a mature idea about the use of these techniques to face the existing problems in the corporation, it was necessary to adapt and combine them in order to solve mainly the conflicts of interests. As a result of this, a new process model for agile planning has been created.

4. Proposed Agile Planning Technique: Brew Model

During the implementation of Relative Weighting, some deficiencies have been encountered. In short, the conclusion is that the existing methods aim to prioritize the activities under the business optics at the expense of priorities associated with the technical risks. Consequently, we face situations in which some features should be taken as a technique premise to the other one, but its business assigned value becomes a low priority. Then, it has conflicted desired approaches by the development team with business interests from the stakeholders.

The main original Relative Weighting shortcoming in this context for the desired prioritization based on technical risks is that it proposes the division of the value by the cost (technical complexity), consequently, the higher the complexity, the lower priority has the feature. This factor is contrary to what is stated in the Scrum guide [4]: *“Products are built iteratively using Scrum, wherein each Sprint creates an increment of the product, starting with the most valuable and riskiest”*.

Another factor that has been changed in the proposed technique is the scale 1-9 used to measure the values of the requirement. Since sequence numbers are a bit comparative, the use of a Fibonacci-based scale is more suitable in this situation. This is because with the Fibonacci scale, it is possible to have an increasing range of numbers, which make each one more comparative with the increase of range of the scale. If a stakeholder, for example, has a User Story scoring 13 and immediately looks to score a second User Story, he compares it to the previous one.

The stakeholder can then have a doubt between punctuating 8 or 13, but seeing that 13 is relatively much larger than 8 and that this feature is not enough to have the same importance as the first, they decide more safely to keep the score 8.

Therefore, in the proposed technique, the scale was defined by the following numbers: 1, 2, 3, 5, 8, 13 and 20. The last number was rounded off in order to fix the end of the scale.

To obtain a form to qualify and justify the technical risks in a principled way, this technique uses a traditional approach, often not considered in agile environments: the Probability & Impact Matrix proposed by PMBOK [3]. This approach offers ways to address the risks much more thoroughly, predicting and assessing impacts at different levels of organization and providing ways to analyze the actions to be taken. A fundamental premise of this approach that fits exactly in the objectives is the fact that Probability & Impact Matrix proposes that the highest risk of technical requirements must be considered first, and precisely driven to the context of these real interests. This approach does not hurt agile principles, since it demonstrates ease of understanding, facility of communication between team members and ease of maintenance.

Then, as a gain from the combination of these practices, the proposed planning technique suggests a visual way to represent and communicate the priorities to all stakeholders named as Attractiveness versus Risks Matrix (Figure 3). It represents the features organized in a table composed of quadrants of risks combined with attractiveness.

Each feature has a location defined by coordinates provided by the business value versus technical risk or cost. Using this matrix, it is possible to achieve a greater transparency in the priority choices.

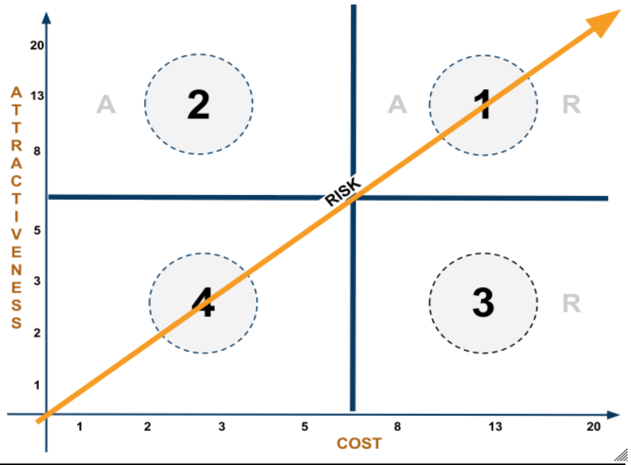


Figure 3. Attractiveness versus Risks Matrix.

This matrix suggests that the more is the earned value (attractiveness) and cost, the higher is the priority. This is because the cost for the development team is also considered based on the risks that each feature can represent to the probability of the non-delivering by its complexity.

The earned value, following the definitions of the Relative Weighting method, consists of two values: benefit and penalty, both at the same scale defined by just an axis, represented by the Attractiveness. It has been proposed accordingly that the result of each value belonging to this axis derived from the arithmetic average of the two scores. Therefore, it is defined:

$$\text{Attractiveness} = (\text{benefit} + \text{penalty}) / 2.$$

As an adaptation based on the ideas of the Probability & Impact Matrix [3], which considers the risks for all aspects of the project, this new matrix has been adopted for only the software requirements in a little different view. For the software development team, the features within the quadrants provide the following vision: (a) Quadrant 1 – A feature in this quadrant is exceedingly complex, it may be such as a definition of an architectural feature or a creation of a new component, for example, being often a basic premise to build the other features. It characterizes a key feature to be implemented, because besides its complexity, it adds higher earned value; (b) Quadrant 2 – A feature in this quadrant is considered as a lower risk, however, it provides a relevant earned value; (c) Quadrant 3 – A feature in this quadrant is often a minor feature that adds a little or irrelevant earned value. It is also complex to implement or has a high risk to the project that should be monitored, prevented or delayed if necessary; (d) Quadrant 4 – A feature in this quadrant is something simple and has low priority, since it causes little impact to the project and also provides the fewest earned value. It should be implemented only when the features of higher value have already been made.

In relation to existing works, this approach provides a more effective way of valuing a requirement according to their degree of risk and the impact that it may cause to the project, also offering subsidies to the design decisions based on its risks. In addition, it becomes a visual tool, which helps the development team to evaluate the risks of each feature and expose these facts to all those involved in a clearer and more honest communication approach. This visual presentation approach to the entire team gives an explanation based on the PMBOK risk management, which also results in more efficient ways to monitor or even mitigate those risks.

So, after the main tools for the processes of prioritization have been defined, the ideas of the model's dynamic were created. Therefore, the proposed approach was called "BREW Model", in which "BREW" stands for "Benefit-Risk Effective Weighting". It also settles the idea from its principle, of "brewing" the processes of prioritization in the context of agile methodologies.

To demonstrate these proposed processes of prioritization by the BREW Model it is necessary to explain them in terms of successive steps. The dynamic is then described in Figure 4.

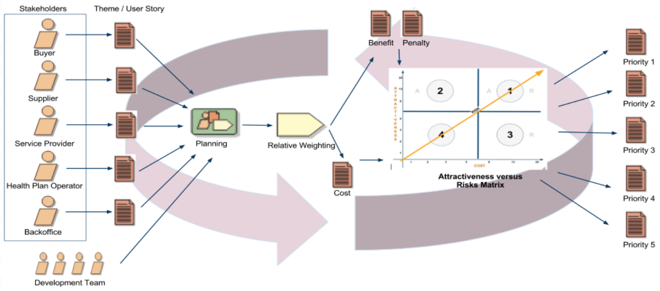


Figure 4. Steps for execution of the BREW Model.

First, the stakeholders bring their features to the planning meetings. This discussion should involve all the stakeholders and the development team. Here, within a consensus, the demands have their benefits and penalties scored, one against the other. In the second part of the meeting, only the development team discusses the complexity of each feature and its corresponding technical risks. Then they place each feature over the Attractiveness versus Risks Matrix, following the values of the scores as coordinates. They obtain the most valuable features, which are farther from the zero axis. So, the development team presents the order of priority to the stakeholders and gives their views on the priorities according to all the concepts of risk analysis.

Finally, all involved in the prioritization process can achieve the understanding about business and technical issues. The communication among them becomes clearer.

5. An Example Of Case Study

This section aims to describe a fictional story about planning to develop features related to an e-commerce platform of electronic sporting goods. The aim here is to illustrate the use of the BREW model in detail to unfold the dynamics of its implementation in a software project.

It is a system developed by a startup, which was released to the market with some features and even a few limitations. It was given only basic functionality which allowed customers to make their purchases with payment by boleto, without the possibility of monitoring the delivery of the products and without an online help chat, features that are considered as essential in a system of that category. The planning to be held, therefore, has the objective of increasing the product with all these features, in the shortest time as possible and ordering activities efficiently and productively to make it more attractive to the market and thus able to keep up with competitors.

The project has three stakeholders, who work directly in researches of new capabilities that may evolve the system to gain market competitiveness and improve relationships with their customers, thus increasing the volume of sales through the site. They will be called here as John, representing the purchase module, Mary, representative of the customer care module and Anthony, representative of the management module in orders and shipping. The development team has two developers, one test analyst and a Scrum Master.

The stakeholders travel throughout the country and take turns in tracking the activities of the development team, serving as the Product Owners of the project. Consequently, it requires that meetings of time-boxes have short duration, so that everyone can be present and thus participate in the defense of their prioritizations.

The company that develops software uses Scrum to manage the development processes. With the imminent need to release a new version of the system with all required increments, a Sprint Planning meeting was scheduled, in which all stakeholders participate in bringing their needs.

As the scope of new valuable features to be released, some requirements have been collected according to the customer's opinion as well as perceived needs through the use of the system. Then the Sprint Planning meeting started. Each stakeholder, therefore, presented their demands through their description of User Stories.

John brought the following stories:

• US01 - "As a buying customer, I want to pay by credit card, so that I can get more speed in my shop";

• US02 – "As a buying customer, I want the appropriated discounts calculated in cash payment by boleto, so that I can get advantages in this kind of payment".

Mary brought a few other needs:

• US03 - "As a buying customer, I want to take my doubts online with an attendant, so that I can solve them quickly and gain more agility in my shop";

• US04 – "As a buying customer, I want to be able to assess each help received by an attendant, so that I can continuously improve the way I met on it".

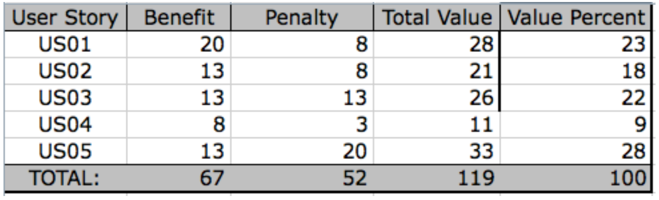
Anthony also brought his need:

• US05 – "Just like the carrier's delivery, I want to be able to signal the status of shipping for tracking, so that customers can have a schedule for delivery, increasing the reliability for the customer in making requests and the credibility in the transport service".

All these stories here were labeled with the initials "US" (User Story) followed by a sequential number for easy identification of each one along the explanations.

Each stakeholder presented their benefit and penalty and scores according to the degree of importance given to each of their stories and scores based on historical data of past meetings of prioritization. Each score is shown according to Table 1.

Table 1. Scores of benefit and penalty for the case study.



User Story	Benefit	Penalty	Total Value	Value Percent
US01	20	8	28	23
US02	13	8	21	18
US03	13	13	26	22
US04	8	3	11	9
US05	13	20	33	28
TOTAL:	67	52	119	100

The benefit is mainly ruled by the last clause of the sentence of each User Story, because it is where the importance is explained precisely, the reason for the existence of the story. In planning meetings, often in order to enhance the justification, the argument becomes necessary for each stakeholder in order to defend their priorities.

So, John thought about the reasons for the creation US01: "[...] so that I can get more speed and ease in my shop" and he appreciated with the maximum benefit. He thought, therefore, that as he represented the main focus of the system (the purchases), shopping would bring greater sales with more speed and ease.

However, the penalty to abandon this feature for the next iteration would not be severe since it would not prevent the customers from continuing doing their shopping, even though in somehow obsolete ways those were already implemented, such as payment by boleto.

The second story, US02, John relativized in comparison to the first, as being lesser attractive than the payment by credit card. In addition, paying by boleto, customers no longer receive this discount and live well with it. The system could then "survive" for one more iteration without this functionality. So it has the same penalty as the previous one.

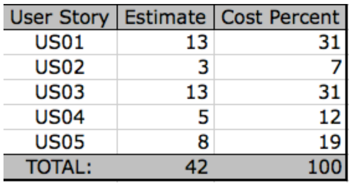
Then Mary, considering good service as primordial for an electronic procurement system as it would provide an increase in the purchase volume, judged the benefit of US03 as thirteen, the same as US02, as it would be so attractive to the customer as having payment discounts. The penalty to abandon this feature in the current Sprint, in her view, could further alienate customers and cause some loss of fidelity of regular customers. It would be severe to delay this feature.

The functionality described for US04 would be dependent on the previous one. So, its benefit and penalty were scored based on this premise. It would therefore be a story with less priority.

As previously reported, there are no current features that show the status of deliveries. Therefore, Anthony would like to prioritize his needs considering this. Although it had less priority than the functionality of US01, (payment by credit card) even though, that would be fundamental in this Sprint. The functionality has therefore a maximum penalty if it is abandoned at this moment.

After all the discussion of the scores in the view of business, the meeting assumed a technical focus. The development team then scored the development cost of each feature as follows in Table 2.

Table 2. Scores of cost for the case study.



User Story	Estimate	Cost Percent
US01	13	31
US02	3	7
US03	13	31
US04	5	12
US05	8	19
TOTAL:	42	100

The team judged the functionality of US01 as being of high complexity. The need to implement the communication with the credit card system operators was considered and it would be necessary to study the solution. So this feature presents a high risk of not being completed on time.

The US02 story would be something regarded as simple to develop since it does not incur significant risks. But the US03 functionality has more complexity, so the development team scored with similar cost to the US01 because it would be necessary to implement an online chat with communication via socket technology, which would also be necessary to study.

The feature expressed in the US04 story would be relatively simple to implement, then the cost of taking it was five, with only a little more complexity than US02. And finally, the feature US05 would be a little more complex, since the team took into account the implementation of a service for external access by the carrier. There is therefore a more technological risk from this new form of communication.

After this stage, the development team held all the stories on the Attractiveness versus Risks Matrix and carried out the risk and impact assessment. They analyzed each story to fit in the quadrant that corresponds exactly to the design they had on their corresponding risk to the project. The matrix was then presented to the stakeholders and the necessary technical arguments were exposed.

In order to achieve a better visual differentiation between stories belonging to each one involved, they were represented by different colors as follows in Figure 5.

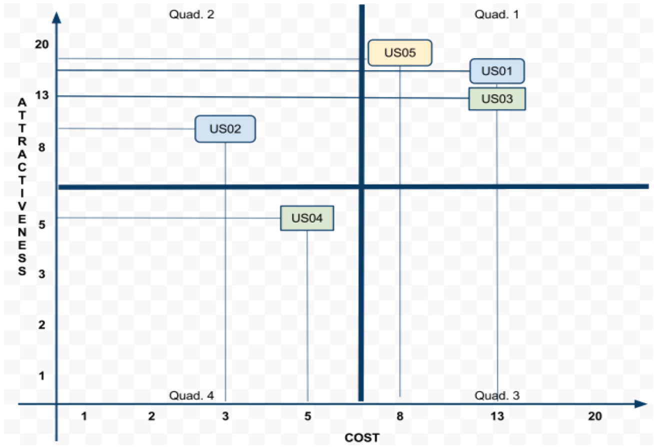


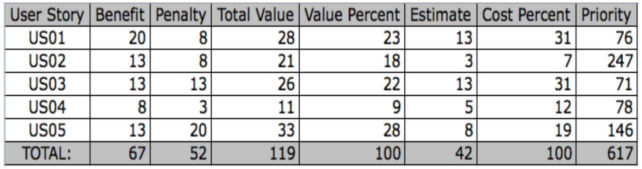
Figure 5. Attractiveness versus Risks Matrix for the case study.

The order prioritized for implementation was as follows: US01, US03, US05, US02 and US04. It may be noticed that the risks related to the development of technologies that make use of new communication protocols have been prioritized. Therefore, it can be developed to the components that can be reused by subsequent other features.

Thus, the system is componentized, achieving increased productivity and quality in software development with this emerging architecture, giving conditions to meet other demands with greater agility.

The original Relative Weighting approach, compared to this method, if applied in this project, would have a totally different order as shown in Table 3.

Table 3. Application of the case study in the relative weighting approach.



User Story	Benefit	Penalty	Total Value	Value Percent	Estimate	Cost Percent	Priority
US01	20	8	28	23	13	31	76
US02	13	8	21	18	3	7	247
US03	13	13	26	22	13	31	71
US04	8	3	11	9	5	12	78
US05	13	20	33	28	8	19	146
TOTAL:	67	52	119	100	42	100	617

The resulting order, considering the highest score to the lowest, would therefore be: US02, US05, US04, US01 and US03. It becomes clear that priorities with lower risk and technological complexity would be of higher priority such as the story US02, as it would not add anything innovative on the technical point of view. In addition, some demands that were the premise for the creation of others, for example, the US03 and US04 stories, had their priorities reversed.

6. Evaluation To Adopting The New Model

In order to measure the effort spent in the prioritizing processes, it has been necessary to apply some methods that would provide an efficient way to compare and know if a proposed solution would be effective. A plan has been made to define the required period to apply the validation. The main objective of this period was to collect a body of consistent data for evaluation. Then, a necessary interval of time involving a Planning Meeting and a number of Sprints that would be sufficient for its completion were defined.

The evaluation of the existing scenario occurred in two perspectives: qualitative and quantitative. It is because analyzing qualitatively makes it possible to analyze the degree of satisfaction of the stakeholders in maintaining the existing prioritization processes. Analyzing quantitatively is a way of getting the performance of this approach in the time-boxes meetings as an essential indicator to get concrete data to demonstrate the existing communication problems. So there were two tests to reach the necessary conclusions about the low quality of the implemented model: under different views; and collecting the key points, the focus to apply the improvements.

In order to implement all the proposed assessment in a qualitative view, a questionnaire containing some objective questions to the stakeholders was created.

The questions were initially asked to analyze the entire period that had preceded the implementation of an improvement.

It had been planned to analyze them again a second time after a new implementation, seeking to analyze the results obtained by the stakeholders in the prioritization processes intended to evaluate the evolution of this view with the implementation of the new model. All data was analyzed using the method of the Grounded Theory (GT).

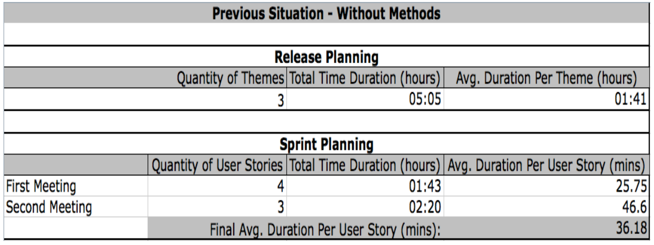
GT is a both qualitatively and quantitatively technique suitable for studying human behavior and organizational culture [5], so the choice to use it as a basis for analyzing the results of this study. GT is aimed at generating explanations for the actions of individuals, focusing on why and how certain groups interact with other groups in specific situations, according to a context delimited from the reality experienced by these groups. So, the questions answered by the stakeholders were analyzed by this theory. This technique in software engineering areas is even scarcer, however, there are successful experiments in software engineering, which also use it, such as investigations of the practice of software process improvement in the Irish industry [6].

The defined period for observation and analysis of the actual situation without a consistent method of prioritizing in fact lasted for a Release Planning Meeting and two Sprint Planning Meetings, a sufficient number of Sprints to deliver this release. The duration of each meeting and each demand prioritized were registered, and after that, the questionnaire was applied and answered by all of the five stakeholders.

The qualitative conclusion of the evaluation this time was that the main problem appointed was caused by the great effort found to prioritize features, often caused by difficulty in executing prioritizing, combined to the fact of not having a concrete method on how to do it. The lack of transparency demonstrated by the software development team to represent their difficulties was also one of the problems.

The quantitative evaluation showed the following results as demonstrated by Table 4.

Table 4. Metrics before implementation of the new method.



Previous Situation - Without Methods			
Release Planning			
	Quantity of Themes	Total Time Duration (hours)	Avg. Duration Per Theme (hours)
	3	05:05	01:41
Sprint Planning			
	Quantity of User Stories	Total Time Duration (hours)	Avg. Duration Per User Story (mins)
First Meeting	4	01:43	25.75
Second Meeting	3	02:20	46.6
	Final Avg. Duration Per User Story (mins):		36.18

7. Use Of The Brew Model And Results Assessment

In the first meeting, the Release Planning, it was necessary to explain how the new method works in terms of steps, to start the process of implementing it. This explanation took about twenty minutes and then the whole dynamic was put in place to prioritize the themes brought up to this discussion. There were many questions over the use of this process still they were promptly answered. The theme that had a higher earned value in the view of all stakeholders and that also had a higher technological risk to the system in the view of the development team was prioritized. The data for the duration of the meeting was properly collected.

With the theme properly prioritized, it was necessary to start planning the execution of the order of their coding features, at the level of their User Stories, from the first Sprint Planning meeting. Then, the proper benefit and penalty punctuations of each started being scored and put on the Attractiveness versus Risks Matrix. Nevertheless, the conclusion that some priorities did not match the real need for this first delivery was reached, which resulted in the prioritization of too complex items. The team would stand to deliver a unique User Story for this Sprint and that this would not bring any earned value in a business view. The stakeholders then disagreed with this approach for this kind of meeting.

They came to the conclusion that in this particular case, the approach of the Relative Weighting would be better. So, that was used and the team finally managed to reach a proper prioritization, with three User Stories prioritized, two less complex than the others that would add good earned value of business and a third more complex that would bring good benefits to the system, mitigating some architectural risks.

The second Sprint Planning Meeting has already occurred with the process model adapted to the reality of the company, with the deficiencies corrected. Then all the scores were submitted to the Relative Weighting approach and the User Stories were properly prioritized. An important fact to notice this time was that the stakeholders were concerned about bringing the new demands now properly punctuated. This fact showed that the use of a concrete technique to enable them to have ways to prioritize and justify their priorities had good motivational effects. There were five User Stories involved in this discussion for prioritizing.

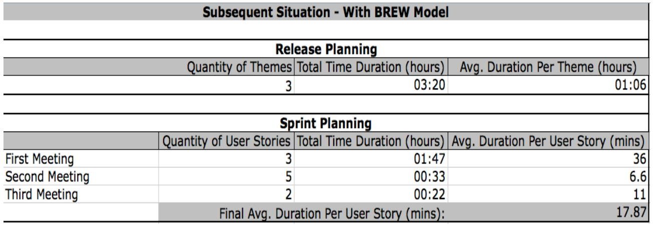
There was a third Sprint Planning Meeting for this release. The results of prioritizing were better than the previous discussions and the meeting lasted for less time. The understanding of the method sounded clearer and its adoption was consolidated. The total of this release was then delivered and the time was once more registered.

Then, it was time to make a new qualitative evaluation and compare both results, before and after this approach, in order to get the performance at the quantitative metrics realized and the level of adherence to the method, at the qualitative evaluation. The same questionnaire used at the prior time to the use of the technique was applied, this time, to review the new processes in place. Then it was employed to qualitative analysis, using the method of Grounded Theory again, on the responses collected.

The analysis of the answers post-of the questionnaire after the implementation of BREW Model revealed a quite positive result. As responses to the questionnaire, there were citations indicating that the model provided simple demands and necessary effort, which brought good results to the team. It was concluded as a consensus that there was a significant decrease in the process of prioritizing efforts to the implementation of the new technique.

For the quantitative analysis, a decrease in the time for each story or theme means that the team communication has been improved and the number of conflicts was decreased. The results for this new metric, after the use of the technique, are in the table 5.

Table 5. Metrics after the implementation of the new method.



Subsequent Situation - With BREW Model			
Release Planning			
	Quantity of Themes	Total Time Duration (hours)	Avg. Duration Per Theme (hours)
	3	03:20	01:06
Sprint Planning			
	Quantity of User Stories	Total Time Duration (hours)	Avg. Duration Per User Story (mins)
First Meeting	3	01:47	36
Second Meeting	5	00:33	6.6
Third Meeting	2	00:22	11
	Final Avg. Duration Per User Story (mins):		17.87

So, for the Release Planning meeting, the total time spent compared to the previous meeting without the technique was approximately 47% lower. It can be concluded that the method, combined to the motivation provided to stakeholders in bringing their demands has already punctuated to this meeting, has brought more efficiency to this process. For the Sprint Planning meeting there was an average reduction of about 51% at the time of prioritization for each User Story.

This fact is evidence that the use of the suggested model reached the desired improvement and efficiency for the prioritization. Therefore, it shows that the conflicts of interests have been mitigated.

In conclusion, the evaluation evidenced some positive results in the use of the proposed technique. As a result, the evaluation pointed out that the conflict of interests and the time spent with prioritization discussions was drastically reduced, giving more time to refine the solution and to keep the team motivated within the agile approach.

8. Important Improvements

As previously reported, some improvements resulting from real use of the BREW Model could be applied to refine the technique according to the perceptions of those involved.

This gain is presented as something significant to the continuous improvement of its processes. So with this reformulation of the process model, figure 6 shows the optimized version of its details, presenting the model with different execution steps under each of the meetings that it applies to.

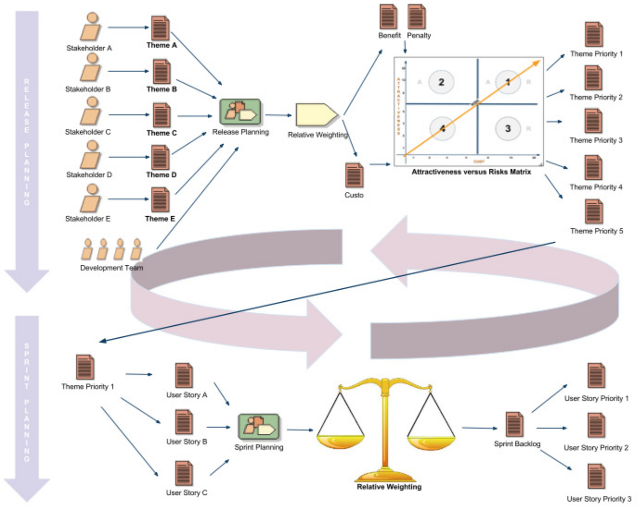


Figure 6. Steps for execution of the improved BREW model

Now, this methodology basically proposes an execution of two different ways into two time-boxes: the Release Planning Meeting and the Sprint Planning Meeting. For the Release Planning, it is proposed to use the Attractiveness versus Risks Matrix to prioritize the themes. After that, for the Sprint Planning corresponding to such theme, it is used the complete and original Relative Weighting method to prioritize the User Stories from that theme.

Therefore, in the first meeting, it is possible and necessary to prioritize the themes (group of related User Stories) according with their degrees of risk, and after that, in order to maximize the productivity for the main User Stories, in the second meeting, the Relative Weighting model may be used, prioritizing the most important one to the business and the simplest to develop.

So, in the first part of the prioritization processes, for the Release Planning, the analysis is less detailed in order to consider a greater quantity of demands. The second part, for the Sprint Planning processes, presents a deeper detailed approach in order to analyze the importance of each feature in its particularities.

9. Conclusion

This paper discusses planning activities in the context of agile software development, especially in scenarios where there are multiple stakeholders involved. Activities intrinsic to this approach are related to estimation and prioritization of work, which require inspection and adaptation depending on the context or environment where the agile methodology is implemented. Therefore, it proposes some adaptation and combination on consolidated theories, to tailor the processes used to achieve full compliance within a dynamic environment with multiple stakeholders. As a result, the proposed model solves conflicts of interest between them, evidenced by the reduction of the time duration of the planning meetings.

The original creation of this work is a compilation of studies that have already been consolidated, combined with ideas and principles suggested as ideas resulting of researches. As a result, the elaboration of new concepts involved solutions that have not been covered in its fullness yet, such as the effective resolution of interests in environments whose product focuses on various areas of business and a more transparent risk management to everyone involved. It already incorporates the best-known approaches both in agile methodologies and traditional approaches.

The main contribution of this work is the proposed BREW Model practices, that aims to decrease the conflict of interests in a common environment found nowadays at the corporations: teams of business analysts with different interests in the same application. This approach was applied in a real project and the assessment techniques used indicates that it was able to reduce the team conflicts and increased their communication, resulting in a decrease of the prioritization efforts.

Another contribution is the proposed way for the presentation of risks to the whole team, since the classification of risks in the Attractiveness versus Risks Matrix provides a more visible and understandable view of them to everyone involved. The compilation of all these studies resulted in a model known as BREW Model, which achieves improvements in the prioritization techniques to a specific environment that is commonly found nowadays.

Aiming to know if the proposed approach achieves its goals, assessment techniques were used intending to measure the stakeholders degree of satisfaction, its ease of use in the planning meetings and whether there was a performance optimization in the planning activities.

The first evaluation was based on questionnaires that were distributed to the stakeholders before and after its implementation in order to analyze impacts of the new approach. These reports were submitted to a qualitative assessment technique known as Grounded Theory [5].

A parallel evaluation approach used metrics based on the time measurement for the prioritization activities. These metrics assess the time spent in planning meetings and the time used for priority discussion in between them. The goal was to verify if the proposed technique directed the discussions and increased the consensus among the stakeholders.

In conclusion, the evaluation evidenced some positive results in the use of the proposed technique. As a result, the evaluation pointed out that the conflicts of interest and the time spent with prioritization discussions were drastically reduced, giving more time to refine the solution and to keep the team motivated within the agile approach.

Many current efforts are aimed to improve planning in agile projects, but there is still too much to be explored in this wide area. Therefore, this study contributes to software engineering, intending to advance the existing studies and expand the horizon of other forms of improvement about agile methodologies, without ruling out major contributions that have been offered by more traditional approaches.

References

[1] COHN, Mike. “Agile Estimating and Planning”, Addison Wesley, TBD.

[2] COHN, Mike. “User Stories Applied”, Addison Wesley, 2004.

[3] PMBOK. “A Guide to the Project Management Body of Knowledge”, 2004.

[4] SCRUM GUIDE. “Scrum Guide”, Ken Schwaber and Jeff Sutherland. Scrum.org. 2009. http://www.scrum.org/scrumguides. Retrieved 2010-02-03.

[5] GROUNDED THEORY. “What is Grounded Theory?”, Jillian Rhine. 2009. http://www.groundedtheory.com/what-is-gt.aspx. Visited in 2010-10-15.

[6] COLEMAN, G.; O'Cconnor, R. 2007. “Using grounded theory to understand software process improvement: A study of Irish software product companies, Information and Software Technology”. Journal of Information and Software Technology, v. 49, n. 6, p. 654-667, jun.

[7] COHN, Mike. “Prioritizing Your Product Backlog Presentation”. 2009. http://www.mountaingoatsoftware.com/system/presentation/file/101/ Cohn_NDC09_Prioritizing.pdf?1267636319. Retrieved 2011-02-05.

[8] JISC infoNet, “A JISC Advance service”. 2009. http://www.jiscinfonet.ac.uk/InfoKits/infokit-related-files/riskmatrixpic/view. Visited in 2011-06-22.