An Approach to Requirements Categorization using Kano's Model and Crowds

Paula Nascimento¹, Rodrigo Aguas¹, Daniel Schneider¹,², Jano de Souza¹
¹PESC/COPPE/UFRJ, Graduate School of Engineering, Federal University of Rio de Janeiro, Brazil
²DEMAT/ICE/UFRRJ, School of Computer Science, Federal Rural University of Rio de Janeiro, Brazil
{pcn, rodrigoaguas, schneider, jano}@cos.ufrj.br

Abstract—Despite advances in the requirements elicitation techniques, researchers show that even today software engineers cannot understand exactly what a product should offer to meet user needs and expectations. This paper aims to propose a new strategy to assist the process of requirements categorization, so that it becomes possible to extract from the user what he really expects of a product. The proposed framework is based on the concepts of requirements categorization established by Noriaki Kano, and on previous studies focused on collective intelligence, to propose different approaches to extract from the crowd the necessary knowledge to classify requirements and decide what should be prioritized and what should not, according to user expectations.

Keywords—crowd; crowdsourcing; kano's model; collective intelligence; requirements validation; collaborative software design

I. INTRODUCTION

In May 2002, during a conference on agile development, Jim Johnson stated that according to a study by the Standish Group, of which he is president, 45% of the functionality of a system have never been used and only 20% are used all the time. Another study by DuPont shows that only 25% of the features implemented are really necessary [1]. Still considering the IT domain, [2] mentions the following example: "One of my students summarized the issues discussed in this book as the "rock" problem. She works as a software engineer in a research laboratory, and her customers often give her project assignments that she describes as "Bring me a rock." But when you deliver the rock, the customer looks at it for a moment and says, "Yes, but, actually, what I really wanted was a small blue rock."

Despite all facts shown above are related to software design, they illustrate what happens to most of the development processes of any kind of product: too many requirements that have no real value to the users are specified and adopted as part of the project scope. It occurs mainly because the ones who are responsible for reporting the needs that should be contemplated don’t know how to express what they really want or the software engineer responsible for collecting these informations could not quite understand the problem. In order to support this issue, in 1984, Noriaki Kano created a model to evaluate the impacts of requirements on the user satisfaction with the product. Through this model it is possible to identify which of the proposed features most stimulate users and, thus, build a unique and lean product, containing what is necessary to please and, at the same time, amaze the target users [4].

In this work, we aim to develop a mechanism capable of supporting the requirements prioritization process. Based on Kano’s model, the proposed framework aims to connect the concepts of collective intelligence and knowledge acquisition to gather, from some profiles identified in an online crowd, enough information to evaluate requirements that may compose a product and elect, among them, which are the requirements that most satisfy users that share the same profile and, based on that, support the process of designing a new product.

This is not the first time that the participation of a larger group of people in the process of creating a new product is considered. In 2008, [9] verified that using the crowd’s opinion in the relevance evaluation process of the results of an information retrieval test suite produced better results than the techniques previously used. In 2009, [10] proposed that the crowd’s wisdom should be used to design urbanism projects, believing that non-experts’ opinions would bring innovation at low costs to the projects. These researchers advocate that it is possible to delegate work to the crowd and obtain better results than the ones extracted from an expert that works individually. This strategy is known as crowdsourcing and these studies are, in fact, examples of a type of crowdsourcing called crowdvoting, where the focus lies in the audience-computer interaction [3].

The remainder of this paper is divided as follows: in section II, the concepts of crowd and crowdsourcing are explored and ways of motivating the crowd to participate are explained; in section III, Kano’s requirements categorization and prioritization model is presented; section IV describes the framework, it’s main concepts and which groups should be involved and why. And, finally, section V concludes this work with final considerations.

II. CROWDS AND CROWDSOURCING

The idea of this paper is to provide a framework that uses the crowd for classifying requirements previously raised, aiming at analyzing their importance for the project that is going to be created. As stated by Malone et al [7], it is natural, then, that two elements are considered first: who will do this work and why.
With the advent of Web 2.0, new forms of interaction over the Internet appeared, encouraging information sharing and collaboration through the Web. One, clearly, is crowdsourcing, which characterizes the act of "outsourcing tasks traditionally performed by an employee of a company, to a large and undefined group of people (the crowd)" [3]. This technique is regarded by some as a new business model where employees are distributed over the Internet and the collective contribution of different individuals generates creative solutions [12]. An example of this new type of work is the website We Do Logos. Through it, you can ask the crowd to create a visual identity for a business or company. In this model, the three parts involved get satisfied: the client has access to a greater number of options created by different designers across a single application, the designer responsible for the winning bid is rewarded financially, and the company providing the service is benefited from working with a number of professionals much greater than what would be expected in a traditional model of work.

According to Brabham [12], crowdsourcing can also be seen as "a creative way to interact with the user, not just a means between messages and people". Allied to this definition are the studies mentioned by [3] that introduce the concept of crowd-computer interaction which focus is to encourage a larger interaction among people who share the same interest, and get a positive result of this activity. According to Schneider et al [3], this interaction is accomplished through two main forms: collecting collective intelligence and crowdvoting. In the first case, the goal is to exploit the intelligence that arises from the interaction between people with different skill levels. This strategy has been proved to be highly effective and, although it is currently being presented on a new context, it is used for a long time to make people give their opinion to elect the president of their country, for example [13]. The second form includes softwares that aim to use the crowd to judge a given situation. The strategy is to get the public's views through a voting system where each individual must choose which option best suits their belief.

The framework proposed in this paper aims at exploring the concept of crowdvoting to extract from the online crowd the information needed to categorize the requirements of a product. However, to be able to extract the full potential offered by the crowd, understanding how it is organized and why it works is necessary. According to Dron & Anderson [8], we cannot treat the crowd as a single entity. According to their social organization, it is possible to identify three distinct classes:

- **Group:** groups are composed of people who have similar interests and are gathered to reach a common goal. Usually, it has well-defined roles and it is not unusual to have a hierarchy among the participants. Groups are normally formed with focus on a specific activity, such as a kick-off meeting, for example. It is impossible to predict if everyone that participates will be in touch after the meeting, but, during this task, they all know each other and feel like being part of the group that should discuss the project and compromise with everything that should be done from that moment on. This is the main difference of this class of crowd.

- **Network:** networks connect individuals who seek to obtain more information, contacts and suggestions from outside the groups they already participate. In this type of organization, the leader is elected from their contributions and participation in the community. Networks allow their users to collaborate with each other in filtering the huge amount of information that exists on the Internet and, thus, generate knowledge for each participant, without the need of knowing everybody who is connected to it.

- **Collective:** collective is characterized by being composed of people who don't see themselves as part of a group or a network, but their activities and ideas, together, create collective knowledge. In this type of organization, people end up contributing to a cause of interest of other people too, without not even knowing of each other's existence.

For each of these entities identified it is necessary to establish a way to motivate the crowd to participate in the work that is proposed. For this, Malone et al [7] proposes three main types of motivation to get people to perform the requested task:

- **Money:** in this case, people are motivated to participate for financial gain. This strategy may attract a larger audience, but possibly with lower quality. In most cases, the individual's interest is only connected to the gain promised and it may be that their contribution is not performed as well as possible. An example of this strategy can be seen at the website Camiseteria.com. Through it, designers can submit their drawings and, if they win the vote held with the public accessing the website, their art will be printed on shirts and sold online.

- **Love:** this factor is able to motivate the crowd no matter how much they will receive in return. In this strategy, people perform the task because they believe they are doing something for somebody else or because they enjoy the activity they are performing. An example of use of this strategy is Wikipedia [7], where people contribute motivated by a greater cause.

- **Glory:** in this case, people give for recognition. This applies to online forums or even a Brazilian online newspaper, G1 – its website provides a separate section where the crowd does the role of journalists by sending news. The reward here is being recognized as an expert in a particular subject, and this is achieved through the number of contributions made.

In 2010, Maher [15] proposed eight categories of motivation that should be considered when working with team collaboration. These categories can also be applied when working with crowds and collective intelligence, and were created from the analysis of previous studies in this area. They are:

- **Ideology:** very similar to Love, proposed by [7], this type of motivation considers that people will do the job “for the purpose of contributing to a larger cause” [15].
• **Challenge:** when using this type of motivation, the goal is to reach people who will do an activity just for the feeling of personal achievement. The Challenge category can be used with those who contribute for knowledge or skill acquisition.

• **Career:** this type of motivation aims at people who participate for improving their careers. This category is similar to Glory, proposed by [7], but is specialized in the professional field.

• **Social:** in this case, people get motivated with the idea of sharing an experience with other people. This is the kind of motivation that makes people in Facebook share posts or like the information someone else has posted on their profile. They don’t receive anything in return, they don’t get recognition for that action, they just want to pass on to their friends something they found to be interesting and share with them the experience they had.

• **Fun:** this is not a type of motivation considered by [7], but it is a very important one. Most people contribute best when there is excitement and enjoyment involved. An example of this case, given by [15], is Google Image Labeler, when Google used the crowd to categorize online pictures. The goal was to improve their search engine, and the means to do that was to create a game where people should add tags to images in exchange for points and gifts.

• **Reward:** when people consider this type of motivation they are after tangible benefits. This is a category very similar to Money, proposed by [7], but Reward also considers non-financial benefits, like “… points in a game, a gift or voucher” [15].

• **Recognition:** this factor is considered when people contribute looking for recognition, just like in the Glory category proposed by [7].

• **Duty:** in this case, people are motivated because they feel, somehow, obliged to complete the task they were asked to do. It is not necessary related to professional duties. It has more to do with doing something “in response to a wish or command expressed personally” [15].

The proposed framework intends to combine the different social organizations and types of motivations described above to define approaches of interaction with the target audience of the product whose requirements are being evaluated. It is believed that each of these profiles can be used differently to achieve the best result according to the purpose of the task handed over to the crowd. In section IV, these approaches will be presented and the social organization and motivation method used will be pointed out.

### III. KANO’S MODEL

Now that we have discussed who will do the work and why they will do it, we come to the following question: what methodology to use to make the crowd classify the requirements of a product?

The answer to this question is the model proposed by Noriaki Kano. It is based on the classification of the requirements and became famous for being able to identify, among the features of a product, which ones most stimulate users. In addition, the model allows collecting opinions from the crowd and verifying if features planned to be part of a product are aligned with customer expectations [5].

#### A. Requirements Categorization

According to [4], Kano defines three categories for requirements, based on how they influence customer satisfaction: must-be, linear and attractive.

Must-be requirements are those that the customer considers that the product can’t exist without. These requirements are not able to increase their satisfaction, as they represent the minimum functionality required to enable the use of the product. On the other hand, if not met, they prevent the use and eliminate one set of possible customers [4].

Linear requirements, also called one-dimensional, may have different levels of achievement on a linear scale and maintain a relationship of proportionality between the level of achievement and satisfaction of users [4]. [6] mentions, as an example of the linear requirement, the gasoline consumption of a car, that is, the more economical the car is, the more satisfied the owner of the car will be.

Finally, attractive requirements are the ones that most influences user satisfaction. Since they are not expected, they are responsible for exceeding the user’s expectations with the product. Moreover, if these requirements are not met, it does not harm the consumer satisfaction [4].

![Figure 1](image-url)

**Figure 1 - Adapted from figure 1 of [4]. Represents the model proposed by Kano.**
B. Questionnaires

According to [4], Kano proposed a process for classifying requirements into five different groups. Using a questionnaire with two questions for each requirement, the users can tell how they feel about the product when it provides a certain feature - functional form of the question - and how they feel if the product doesn’t provide – dysfunctional form. It is important that questions are always related to which user need is going to be satisfied and not how it will be resolved.

Through the responses, it is possible to identify which is the real value of a requirement for a given user. For example, if a user expects a feature is present, but can live without it, it is probably not a mandatory feature. The evaluation table below, also proposed by Kano, indicates how to combine the responses to classify the product requirements in six different groups [5]. You will notice that three other categories are used as support. They classify the requirements into categories that demonstrate that the user does not care about that feature (N: Neutral), has no opinion on it (Q: Questionable) or prefer that it would not be taken into account (I: Inverse). For these reasons, requirements with one of these three classifications can be disregarded during the development process [4].

IV. PROPOSED FRAMEWORK

After analysis of the model and the questionnaire proposed by Kano, we decided to build a framework that could make use of these tools and use the advantages offered by them to improve the process of developing a product. Based on the level of customer satisfaction, the model allows the project to be created focusing on requirements that attract potential customers and not just offer the minimum required for use of the product. In addition, a result evaluation table, also proposed by Kano, allows checking the quality of responses collected, which helps applying the model to crowds [5] and fits perfectly to the goal of this solution.

A. Approaches

Based on the concepts set out in section II, individuals who compose the crowd need incentives to collaborate on a project. We then defined three approaches of reward for the proposed framework, taking into account the different ways of selecting participants.

The first approach focuses potential consumers of a new product and users of products undergoing evolution. We consider this set of people as the network class of social organization, since they share common interests and know they are part of a group of people who consume a certain product, but not necessarily know each other or interact frequently. The incentive that fits this approach is the direct benefit the network will have by helping the product to better match their own needs, which can be seen as a form of personal gain. For this reason, the categories that best fit are Reward and Career – if the product will improve the individual’s work. The main advantage of this approach is being able to count on the commitment of those involved, since they are helping to bring improvements for themselves. And one point that can be seen as an advantage or a disadvantage is the possibility for participants to influence each other according to their interest. On one hand, because we are dealing with networks, the interaction between participants is lower, thus the chance of such situations to happen is also lower. On the other hand, just because it involves personal gain and, possibly, professional gain, it is possible that some people try to influence others so that their goals with the product improvement are achieved.

The second approach is based on a generic search for people with a pre-established profile, which configures a collective. The idea is to find people willing to contribute and reward them for their work. The advantage of this strategy is clearly in collecting completely neutral opinions that could not be driven by any personal gain with the product. This aspect, coupled with the fact that all participants have the desired characteristics for future consumers, may generate a more accurate categorization of what potential users really expect from the product. However, this method has the disadvantage of possible lack of commitment from those involved. By not having any direct interest in the product, the responses collected may not actually represent the feeling of the evaluator regarding a specific requirement. The motivation in this approach takes place through financial compensation or other forms of personal gain, which match the Money or Reward categories of motivation. As an example of another form of incentive that can be used in such cases, we highlight the
practice of Grooveshark to use its users to participate in a market research. If the user has indeed the desired profile that’s being searched and completes the survey, he earns some points that can be later exchanged for paid plans offered by the website.

The third and last approach draws upon social networks, by using communities related to the product. The form of incentive in this case is through recognition of community members in relation to those who contribute most to the categorization process, which matches the Recognition, Glory or, in some cases, Career categories. This can be done through the dissemination of a ranking of most active users. This type of initiative has a much smaller chance of influence among the participants and can also produce good results as long as there exists the commitment of those involved. Motivated to be recognized as good contributors, perhaps their answers will be of higher quality and better represent the feeling of the related community. However, it is possible that the level of participation is not good enough and that it is not possible to collect a sufficient amount of data to perform a valid categorization.

Each of the approaches shown above has its advantages and disadvantages. It is important that the environment, the scope and type of project are analyzed before deciding which strategy to use. Depending on the potential customers and the company, a method can end up doing better than another one.

B. Framework’s operation

After selecting which approach to follow based on which type of crowd one wants to involve in their project’s requirements categorization, it’s time to, in fact, use the framework proposed.

The first step is to elicit the requirements to be categorized and produce a questionnaire modeled after Kano’s proposal: for each requirement we must create the functional and dysfunctional questions and, for each question, the participant must choose one of the five possible answers. Despite the class of crowd chosen to work with, the product owner must consider the wanted number of responses and the period used to gather these opinions. The framework does not imposes these values and we believe it will vary according to the project’s scope and target.

The second step would be the evaluation of the answers collected using the evaluation table also proposed by Kano. For each requirement, we combine the two questions’ answers and extract the requirement classification for each participant. After that, we evaluate all classifications each requirement received and elect in which category it was classified for most people.

The framework suggests that this result can be exposed through graphs which shall clearly inform the distribution of ratings received by each requirement. It is important that the product owner has access to this information so that he is more confident in deciding whether a requirement should be actually implemented or not.

V. FINAL CONSIDERATIONS

Studies carried out by Standish Group and released in 2011 show that the number of failed projects has been decreasing. Even so, the routine of an IT company shows that there is always software not delivered on time, or that does not meet the requirements of the customer, or already over budget.

The framework proposed in this paper aims to assist the process of defining the scope of the project so that the problems mentioned above can be minimized during the development of a product. It is believed that combining the knowledge generated by the masses who surf the Internet and the concepts advocated by Kano, it’s possible to categorize the requirements previously raised and validate the importance of its implementation for the project and therefore to customers as well.

It is noteworthy that the results obtained from the model proposed here work only as guidance to the project owner. All the factors involved in creating the product should be evaluated and only after this analysis you can check what requirements must be really prioritized. This classification is information about the scenario that fits the system under consideration to support the decision – it does not necessarily define an order to be followed.

Future research will include the development of a prototype for performing experiments. A first investigation in this direction has been carried out by Aguas [14] in the product ScrumHalf, a web tool that supports teams who develop projects using the agile methodology Scrum. At the time the experiment was ran, the system had 1500 users subscribed and they were all asked about what new features should be prioritized in the new releases of the system. Even though this case of study did not use Kano’s model, it already had an interesting result. The users had to decide, between 6 different requirements, which ones should be released first. Asking the development team members which requirement would be chosen, they elected 3 of the 6, but only 1 of them matched the users’ choice. That result shows that, sometimes, it is not possible to predict what the users of a product want of it. Considering the result obtained with this first experiment and the noises in communication we all know exist, we believe that
the framework proposed here can significantly improve this relation between customer and producer and help to produce more value to the customer. We also believe that it is substantial to continue the study and compare results by using Kano's model and simply voting. In a second step, the intention is to propose a framework for requirements elicitation with support from the crowd. The results of these investigations will be reported in follow-up papers.

REFERENCES