4-14-2014

OF THE USER, BY THE USER, FOR THE USER: ENGAGING USERS IN INFORMATION SYSTEMS PRODUCT EVOLUTION

Adarsh Kumar Kakar
Alabama State University, akakar@alasu.edu

OF THE USER, BY THE USER, FOR THE USER: ENGAGING USERS IN INFORMATION SYSTEMS PRODUCT EVOLUTION

Adarsh Kumar Kakar
Alabama State University
akakar@alasu.edu

ABSTRACT

Collectively users constitute a source of massive amounts of product innovation (Vonn Hippel, Ogawa and de Jong, 2012). When users are viewed merely as recipients of innovation, the firm does not have access to user knowledge and experience developed through product use (Sawhney, Verona and Prandelli, 2005). Additionally, it has been suggested that the product evolution should be innovative in the users' frame of mind not the developers' (Fellows and Hooks, 1998). New product features that do not resonate with the users create wasted development effort, delay in time-to-market, increased complexity and operational costs of the product. Keeping this context in view, this empirical study assesses existing promising methods for selecting new product features through involvement of users. The results of this study show that the Kano survey method demonstrated potential in not only identifying those product features that add value to the user but also those which do not.

Keywords

Feature selection, requirements prioritization, kano survey method

INTRODUCTION

Involving users as active contributors in the product development process rather than as passive participants has been expressed in literature for a long time (Gardiner and Rothwell, 1985; Leonard-Barton, 1995; Rothwell, 1976; von Hippel, 1988; Witell, Lofgren and Gustaffsson, 2011). Users are co-creator of value (Vargo and Lusch, 2006) and useful partners in the innovation process (Kristensson et al; 2004). Magnusson et al. (2003) found that ordinary users were the best source for new ideas in terms or originality. Through their unique knowledge derived from use of the product ordinary users were better equipped to generate ideas that are relevant to customers than R and D (Research and Development) employees and engineers (Kristensson et al, 2004; Magnusson, 2003).

Therefore to actively engage users organizations have evolved various mechanisms. Of these the use of websites for capturing and prioritizing user requirements is becoming increasingly prevalent..The websites include both forums and collaborative tools, and are designed to allow large numbers of users to participate in the requirements gathering and analysis process. The success of websites and collaborative tools that have been used to gather inputs from users, demonstrates that, given the opportunity, users too are willing to take the time to contribute feedback and ideas (Laurent and Cleland-Huang, 2009).

However, by actively engaging the users, more new feature requests are often elicited than are needed to build into the system. While on the one hand excluding a high value feature may mean losing users to a competing product, on the other hand including a requirement that is unneeded creates wasted development effort, delays in time-to-market, and increased complexity, maintenance and operational costs of the product. Keeping this context in view, this study first identifies 5 promising methods for feature selection from requirement engineering literature, product development and quality literatures. It then assesses which of these methods demonstrate greater efficacy in engaging users for identifying features which add value to the users of information systems (IS) product as well as features which do not.

SELECTION OF FEATURE SELECTION METHOD

Binary Search Tree. Racheva, Daneva and Buglione (2008) reviewed a number requirements prioritization techniques and classified them into two main categories: techniques used to prioritize small number of requirements (small-scale) and techniques that scale up very well (medium-scale or large-scale). Bebensee, van de Weird and Brinkkemper (2010) observed that as software products are developed for the market rather than a single customer, one can expect a larger number of feature requests from users. Hence techniques that scale up well are most appropriate for software products. They found that the Binary Search tree method scales up well for software products with medium-scale requirements. Another study by Ahl

(2005) investigating the five ranking techniques of requirements prioritization - AHP, Binary Search Tree, Planning Game, 100 Points Method and PGcAHP (Planning Game combined with AHP) - found that Binary search tree was superior to all other methods on many counts including accuracy of results and scalability. Binary search tree was therefore chosen from among the non-grouping (ranking) techniques as the first technique to be assessed in the study.

Priority Groups Method. Medium-scale or large-scale prioritization techniques might be based on relatively complex algorithms or at least due to the large amount of requirements need tool support (Rachdeva, Daneva and Buglione, 2008). However, sophisticated prioritization techniques are found to have limited ability to support requirements prioritization in market-driven product development with professionals in industry preferring simple tools instead (Lehtola and Kauppinen, 2006; Berander and Andrews, 2006). The Priority groups method is one such simple classification technique which ranks requirements into three priority categories, High, Medium and Low (Wiegers, 1999). It is among the most traditional and best known (Lehtola and Kauppinen, 2006). Priority Groups technique was therefore chosen as the second technique for comparison.

Kano Survey Method. The three factor theory is widely used for product feature selection and classifies user requirements into three categories that add value to the user in distinct ways – Basic, Performance and Excitement. A review of the advantages and disadvantages of techniques for feature selection based on the three factor theory such as the Direct Classification method, Importance Grid method, Penalty-Reward contrast analysis method and Kano survey method by Mikulic and Prebez (2011) suggests that the Kano method was the most suitable. It was found to be both a valid and a reliable method for categorizing feature requests according to the three factor theory. Another study by Witell and Lofgren (2007) comparing Direct Classification method, Importance grid method, Kano survey method and a variant of the Kano survey method which used a 3 level questionnaire rather than the 5 level questionnaire of the Kano survey method came to the same conclusion and recommended that practitioners continue to use the Kano survey method. For this study the Kano survey method was therefore chosen from among the various techniques based on the three factor theory as the third technique for evaluation.

Dual Questioning Method. One of the limitations of the techniques listed above is that they do not take in consideration market factors such as the availability of the features being assessed in competitive products. As this study is exploring a suitable technique for market-driven software products, it will also investigate the potential of the determinant attribute approach (Myers and Alpert, 1968) using the dual questioning technique as the fourth technique for evaluation.

Hybrid Method. In addition a fifth technique which is a combination of Dual questioning method and the Kano survey method is suggested for comparing its efficacy in feature selection. Although the three factor theory allows producers to make a strategic choice through classifying product feature requests into the three categories, it does not rank features within a category. In addition, it does not take in consideration market factors such as the availability of these features in competitive products. In the hybrid method, detailed in the experimental treatments section, the Dual questioning approach is expected to complement the Kano technique by providing a method for ranking the features within each category, keeping competition in view, after they have been categorized using the Kano method. This we expect will be relevant for producers of market-driven software products. It will provide them with additional information to select a lean set of features that give maximum user impact for the resources invested while simultaneously keeping the strategic options open for the management.

METHOD

139 subjects who were actual users of Gmail participated in two rounds of an experiment of which 122 valid responses were obtained. The valid responses from 69 females outnumbered the valid responses from 53 male subjects. The average age of the subjects was 21.3 years with the female subjects averaging 21.3 years and the male subjects averaging 21.2 years. 10 Feature of Gmail were used as the test instrument (see Table 1 for a sample).

No	Feature description
1	Allow sending emails/ replies to emails at a later time or date. Presently if the user has to send an email or a reply to email at a later date she can only save the draft and remember to send it when the date arrives.
2	Allow user to have another view of their inbox below the message they are composing. This will allow users to reference information from one or more emails, if required, while composing
3	Provide preview of media stored on other sites within an incoming Gmail message when the sender

	includes only a link. Users get tired of clicking on links to get to the videos and photos of friends

Table 1. Sample of Feature Description in Test Instrument

EXPERIMENTAL PROCEDURE

A brief description of each of the methods used by the subjects in feature selection is described below:

Binary search tree method treatment. The Binary Search Tree Method has been used previously for software product feature prioritization. It provides a ranked list of requirements according to user preference. Prioritizing software requirements using this technique involves subjects constructing a binary search tree consisting of nodes equal to the number of candidate requirements. First a single node holding one requirement is created. Then the next requirement is compared to this node. If it is of lower priority than this node then it is assigned to the left of this node else it is assigned to the right of this node. This process continues until all requirements have been inserted into the binary search tree. The node at the extreme left of the binary search tree is of the lowest priority while the node at the extreme right is of the highest priority. If the nodes in a binary search tree are traversed in *in order*, then the requirements are listed in a ranked order of priority. Thus using the binary search tree approach involved subjects selecting the requirements one at a time and creating a binary search tree and then traversing the binary search tree in order to generate a ranked list.

Priority groups method treatment. The Priority Groups Method has been used previously for software product feature prioritization. It is based on grouping requirements into different (highest to lowest) priority groups, with clear and consistent definitions of each group. Although the number of priority groups may vary the use of three groups (High, Medium and Low) is the most common (Leffingwell and Widrig, 2003). The description for these groups is as follows (Wiegers, 1999):

> **Definition: High priority requirements** are mission critical requirements; required for next release
>
> **Definition: Medium priority requirements** support necessary system operations; required eventually but could wait until a later release
>
> **Definition: Low priority requirements** are a function or quality enhancement; would be nice to have someday if resources permit

Subjects used this description to categorize each Gmail feature request into one of the three groups.

Kano survey method treatment. The Kano Survey Method involved subjects responding to two questions for the every product feature request: the functional question "How do you feel if this feature is present?" and dysfunctional question "How do you feel if this feature is NOT present?" The first question concerns the reaction of the user if the product includes that feature, the second concerns his reaction if the product does not include that feature. The user has to choose one of the five possible options for the answers for both the functional and dysfunctional question: 1. I like it this way, 2. I expect it this way, 3. I am neutral, 4. I can live with it this way, 5. I dislike it this way. Asking both functional and dysfunctional questions helps product managers assess user priorities. If the user expects some feature to be present, but can live without the feature, it is not a mandatory feature. The user response for each feature request are then mapped on a 5 x 5 grid to determine the requirement category – Basic, Performance or Excitement - to which it belongs based on plurality.

Dual questioning technique treatment. In the Dual Questioning Technique consumers are:

1. asked which features they consider important and then
2. asked how they perceive this feature as differing among the competitor products

Features ranked high in rated importance (5- Extremely Important 1 – Not Important) but not thought to differ much (4 – Very Different, 1- Very Similar) among the various products may not be the most determinant factor. The product of attribute importance and difference among products determines the ranking of feature requests. Attributes that are ranked high in importance and difference ratings among products in the same product category are considered more determinant than attributes that are ranked low in importance and difference ratings among products.

Complexities of Feature Selection. User Requirements expressed in the form of user feature requests are not stand alone artifacts (Dahlstedt and Persson, 2003). They may exhibit complex interdependencies among each other and therefore cannot be treated independently (Regnell, Paech, Aurum, Wohlin, Dutoit and Natt och Dag, 2001; Carlshamre, Sandahl, Lindvall, Regnell and Natt och Dag, 2001) . Feature selection must therefore include approaches for managing requirements interdependencies to fully support producers of software products (Karlsson, Olsson and Ryan, 1997). If user requirements had no interdependencies then feature selection would only involve selecting the top 'n' features from a given set of feature requests which add maximum value to the user. However, the interdependencies between requirements make feature selection extremely complex. For example, for only ten feature requests it will ideally require seeking satisfaction feedback from users on 1023 scenarios (2^10 – 1 = 1023feature subsets excluding null subset). This will require a very large sample size of users if one considers the norm of having the number of observations at least 10 to 20 times the number of treatments (scenarios). In addition the cognitive load on the users will make data collection infeasible.

We therefore decided to collect the user feedback for only two critical feature subsets, one subset that is likely to add value to the user and the other that is not expected to add value to the user. The underlying assumption in this approach is that while the interaction (multiplicative) effect may be expected for features that add value there is not likely to be significant interaction (zero or marginal multiplicative effect) among features that do not add value. The efficacy of the method can then be determined by whether the value added feature subset determined by the method increases the user satisfaction significantly as well as whether the non-value added feature set does not impact user satisfaction significantly. Comparing the efficacy of 4 feature selection methods will require user response on only 4 x 2 = 8 scenarios.

Determining the feature subsets for comparison of efficacy. The Priority Groups method provides both a subset of features that are expected to add significant value to the product (High and Medium categories) and those that are not expected to add significant value to the product (Low category). The Priority groups method was therefore used as the baseline. For instance, if the Priority groups method identifies ‘n’ features (High + Medium) that are likely to add value to the product, then in the complementary subset there are ’10-n’ features (LOW) that are not likely to add value to the product. The top ranked ‘n’ features selected by the Kano survey method, Binary Search tree method and Dual method were then chosen for comparison of efficacy in identifying features that provide value and bottom ’10-n’ features were chosen for comparison of efficacy in identifying waste. For the ranking methods such as Binary Search tree and Dual questioning method the set of ‘n’ value-add and ’10-n’ non-value added feature sets could be easily derived from the rank order. For Kano survey method the ranking order was determined by Basic > Performance > Excitement and for each feature within each category by the descending order of the number of users who selected the feature in that category. Past research has shown that the feature importance from user perspective is Basic > Performance > Excitement (Matzler and Sauerwein, 2002, Zhao and Dhokalia, 2009). The number of features that add value for the Hybrid method was determined by considering only the common value-added features identified by Kano survey and Dual questioning methods.

Round 1. In round 1 of the experiment each subject provided their requirement prioritization of the 10 feature requests by users of Gmail through a paper-based instrument that included questions related to the Binary Search Tree Method, Priority Grouping Method, Kano Survey Method and Dual Questioning Technique using the methods detailed in the previous section. The data obtained from the subjects in Round 1 was used to select a subset of features that added value to the software product and a subset of features that do not using the process described in the previous section. Additionally subjects in Round 1 also provide their user satisfaction response to the current version of Gmail.

Round 2. The data obtained from each subject in Round 1 was used to tailor questions specifically for that subject in Round 2 of data collection conducted a week later. As the same subjects were involved in providing data on both independent and dependent variables the temporal separation between Round 1 and Round 2 was expected to mitigate potential effects due to Common Method Variance (Sharma, Yetton and Crawford, 2009). Subjects in Round 2 rated their perceived satisfaction with implementing the two feature subsets obtained from each method in round 1 Perceived user satisfaction was used as a dependent variable because the producer would want to know the impact of the feature subsets before rather than after implementing the features. Subjects rated their satisfaction for each of these experimental conditions using a single item 7 point scale (Andrews and Withey, 1976) with a neutral midpoint of 4, terrible at one end of the scale (1) and delighted at the other end of the scale (7): 1 - Terrible 2 – Unhappy 3 – Mostly Dissatisfied 4 – Neither Satisfied nor Dissatisfied 5 – Mostly Satisfied 6 – Pleased 7 – Delighted.

A number of control procedures were used to mitigate effects due to extraneous variables. The extraneous variable i.e. “user segment” of Gmail users was controlled through the use of a homogeneous sample of student subjects. The “sequence effect” of manipulating different treatments a counterbalancing design using Latin squares (Sheehe and Bross, 1961) was used to get subject responses for different methods of feature selection. Every fifth subject got the same sequence (see Table 3).

Round 1: Feature Selection Method				
Subject 1	Priority groups	Kano	Binary Tree	Dual
Subject 2	Kano	Dual	Priority groups	Binary tree
Subject 3	Dual	Binary Tree	Kano	Priority Groups
Subject 4	Binary Tree	Priority Groups	Dual	Kano

Table 3. Sequencing of Methods

RESULTS AND ANALYSES

The descriptive statistics of the mean user satisfaction (V=value added feature subset, NV=non-value adding feature subset) under different experimental treatments is shown in Table 4.

EXPERIMENTAL CONDITIONS	Mean User Satisfaction (NV)	Mean User Satisfaction (V)
Current version (1)	4.541	4.541
Kano Survey Method (2)	4.655	4.984
Priority Groups Method (3)	4.721	4.679
Dual method (4)	4.789	4.936
Binary Search tree method (5)	4.749	4.656
Hybrid method (6)	4.771	4.682

Table 4. Descriptive Statistics

To determine if there is a significant difference between satisfaction with the various feature subsets that add value a repeated measure ANOVA was performed as each subject took part in all experimental conditions. The Bonferroni post-hoc test results summarized in Table 5 were examined (row - column) to discover which specific means differed significantly. Looking at the column values representing Current version and titled ‘1’ in Table 5 we see that the Kano survey method (row 2) demonstrated superior efficacy (significantly higher perceived user satisfaction) in identifying a feature subset that add value to the users of the software product compared to all other methods expect the Dual questioning method. To determine if there is a significant difference between satisfaction with the various feature subsets that did not add value to the software product a repeated measure ANOVA was performed again. The Bonferroni post-hoc test results summarized in Table 6 were therefore examined (row - column) to discover which specific means differed significantly. Among the five methods only the feature subsets identified by Kano survey method and Priority groups method demonstrated efficacy in identifying non-value added feature subsets that did not impact user satisfaction significantly (see column 1, Table 6)..

	1	2	3	4	5	6
1	0					
2	0.443**	0				
3	0.138	-0.306**	0			
4	0.395**	-0.048	0.257*	0		
5	0.115	-0.328**	-0.022	-0.280*	0	
6	0.141	-0.302*	0.003	-0.254*	0.026	0

p < .05 ** P < .01 ***p<.001

Table5. Difference in User Satisfaction (V)

	1	2	3	4	5	6
1	0					
2	0.114	0				
3	0.180	0.066	0			
4	0.248*	0.134	0.068	0		
5	0.208*	0.094	0.028	-0.040	0	
6	0.230*	0.116	0.050	-0.018	0.022	0

*** p < .05 ** P < .01 ***p<.001**

Table 6. Difference in User Satisfaction (NV)

CONCLUSION

Thus the results show that overall the Kano survey method demonstrates promise as a method of selecting features (*for the user*) from a given set of user feature requests (*of the user*) through user engagement (*by the user*). While the Dual method performed better statistically than the Binary Search tree method, Priority groups method and the Hybrid method in identifying the features that add value to Gmail, it did not demonstrate efficacy in identifying features that do not add value to Gmail. Also, while the Priority groups method performed better statistically than the Dual method, Binary Search tree method and the Hybrid method in identifying the features that do not add value to Gmail, it did not demonstrate efficacy in identifying features that add value to Gmail. Only the Kano method demonstrated efficacy both in identifying features that add value to the users of IS product and in identifying features that did not add value to the users of IS product. The reason for its superior performance may lie in its unique approach of taking user responses for both including the feature as well as for not including the feature in an IS product. This could be an interesting area for in-depth investigation in future research.

REFERENCES

1. Ahl, V. (2005) An Experimental Comparison of Five Prioritization Methods. Master's Thesis, School of Engineering, Blekinge Institute of Technology, Suecia.
2. Andrews, F. M. and Withey, S. B. (1976) Social indicators of well-being, New York: Plenum Press.
3. Bebensee, T., van de Weerd, I. and Brinkkemper, S. (2010) Binary priority list for prioritizing software requirements, *Requirements Engineering: Foundation for Software Quality*, 67-78.
4. Berander, P. and Andrews, A. (2006) Requirements Prioritization, in Engineering and Managing Software Requirements, A. Aurum and C. Wohlin, eds., Springer, 69–94.
5. Carlshamre, P., Sandahl, K., Lindvall, M., Regnell, B. and Natt och Dag, J. (2001) An industrial survey of requirements interdependencies in software product release planning, In Requirements *Engineering Proceedings, Fifth IEEE International Symposium*, 84-91.
6. Dahlstedt, A. G. and Persson, A. (2003) Requirements interdependencies-moulding the state of research into a research agenda, *In 9th International Workshop on Requirements Engineering–Foundation for Software Quality*, 55-64.
7. Fellows, L. and Hooks, I. (1998) A Case for Priority Classifying Requirements, *in Third International Conference on Requirements Engineering (ICRE'98)*, 62-65.
8. Gardiner, P. and Rothwell, R. (1985) Tough customers: good designs, *Design Studies*, 6, 1, 7-17.
9. Karlsson, J., Olsson, S. and Ryan, K. (1997) Improved Practical Support for Large-scale Requirements Prioritisation, *Requirements Engineering Journal*, 2 (1), 51-60.
10. Kristensson, P., Gustafsson, A. and Archer, T. (2004) Harnessing the creative potential among users,. *Journal of Product Innovation Management*, 2, 1, 4-14.
11. Laurent, P. and Cleland-Huang, J. (2009) Lessons learned from open source projects for facilitating online requirements processes, In *Requirements Engineering: Foundation for Software Quality* 240-255, Springer Berlin Heidelberg.
12. Leffingwell, D. and Widrig, D. (2003) Managing Software Requirements: A Use Case Approach, 2nd edition, Addison-Wesley, USA, 124-125.
13. Lehtola, L. and Kauppinen, M. (2006) Suitability of Requirements Prioritization Methods for Market Driven Software Product Development, *Software Process Improvement and Practice*, 11, 1, 7-19.
14. Leonard-Barton, D. (1995) Wellsprings of knowledge: Building and sustaining the sources of innovation, Harvard Business Press.
15. Magnusson, P. R. (2003) Benefits of involving users in service innovation, *European Journal of Innovation Management*, 6, 4, 228-238.
16. Matzler, K. and Sauerwein, E. (2002) The factor structure of customer satisfaction: an empirical test of the importance grid and the penalty-reward-contrast analysis, *International Journal of Service Industry Management*, 13 (4), 314-332.
17. Mikulic, J, and Prebez, D. (2011) A critical review of techniques for classifying quality attributes in the Kano model, *Managing Service Quality*, 21, 1.
18. Racheva, Z., Daneva, M. and Buglione, L. (2008) Supporting the Dynamic Reprioritization of Requirements in Agile Development of Software Products, *In: Proceedings of the Second International Workshop on Software Product Management*, 49-58.
19. Regnell, B., Paech, B., Aurum, A., Wohlin, C., Dutoit, A. and och Dag, J. N. (2001) Requirements Mean Decisions!–Research issues for understanding and supporting decision-making in Requirements Engineering, *In First Swedish Conference on Software Engineering Research and Practice: Proceedings*.
20. Rothwell, R. (1976) Marketing: a success factor in industrial innovation", *Management Decision*, 14, 1, 43-53.
21. Sawhney, M., Verona, G. and Prandelli, E. (2005) Collaborating to create: The Internet as a platform for customer engagement in product innovation, *Journal of interactive marketing*, 19, 4, 4-17.
22. Sharma, R., Yetton, P. and Crawford, J. (2009) Estimating the Effect of Common Method Variance: The Method–Method Pair Technique with an Illustration from TAM Research, *MIS Quarterly*, 33, 3, 473-490.
23. Sheehe, P. R. and Bross, I. D. J. (1961) Latin squares to balance immediate residual and other effects, *Biometrics*, 17, 405–414.
24. Vargo, S. L. and Lusch, R. F. (2006) Service-dominant logic, *The service-dominant logic of marketing: Dialog, debate, and directions*.
25. Von Hippel, E. (1988) The Sources of Innovation, Oxford University Press, New York, NY.
26. Von Hippel, E., Ogawa, S. and de Jong, P. J. (2011) The age of the consumer-innovator. Susumu Ogawa, 53, 1, 27–35.

27. Wiegers, K.E. (1999) Automating requirements management, *Software Development*, 7, 7, S1–S6.
28. Witell, L. and Löfgren, M. (2007) Classification of quality attributes, *Managing Service Quality*, 17, 1, 54-73.
29. Witell, L., Löfgren, M. and Gustafsson, A. (2011) Identifying ideas of attractive quality in the innovation process, *The TQM Journal*, 23, 1, 87-99.
30. Zhao, M. and Dholakia, R. R. (2009). A multi-attribute model of web site interactivity and customer satisfaction: An application of the Kano model, *Managing Service Quality*, 19, 3, 286-307.