Group Decision Support for Requirements Negotiation

Alexander Felfernig[1], Christoph Zehentner[1], Gerald Ninaus[1], Harald Grabner[1], Walid Maalej[2], Dennis Pagano[2], Leopold Weninger[3], and Florian Reinfrank[1]

[1]Applied Software Engineering, Graz University of Technology, Inffeldgasse 16b, A-8010 Graz, Austria
[2]Applied Software Engineering, Technische Universitaet Muenchen, Boltzmannstrasse 3, D-85748 Garching, Germany
[3]wsop, Muellnergasse 4, A-1090 Vienna, Austria
{alexander.felfernig,christoph.zehentner,gerald.ninaus,harald.grabner, florian.reinfrank}@ist.tugraz.at
{walid.maalej,dennis.pagano}@in.tum.de
LWeninger@wsop.at
www.ist.tugraz.at,www.tum.de,www.wsop.at

Abstract. Requirements engineering is one of the most critical phases in software development. Requirements verbalize decision alternatives that are negotiated by stakeholders. In this paper we present the results of an empirical analysis of the effects of applying group recommendation technologies to requirements negotiation. This analysis has been conducted within the scope of software development projects at our university where development teams were supported with group recommendation technologies when deciding which requirements should be implemented. A major result of the study is that group recommendation technologies can improve the perceived usability (in certain cases) and the perceived quality of decision support. Furthermore, it is not recommended to disclose preferences of individual group members at the beginning of a decision process – this could lead to an insufficient exchange of decision-relevant information.

Keywords: Group Recommender Systems, Requirements Engineering

1 Introduction

Requirements engineering is considered as one of the most critical phases in software projects [1] and poorly implemented requirements engineering is a major risk for the failure of a project [2]. Requirements themselves are a verbalization of decision alternatives regarding the functionality and quality of the software [3]. Related individual as well as group decisions are extremely difficult due to the increasing size of requirement models as well as contradicting preferences of stakeholders [4, 5].

In this paper we analyze the impact of applying group recommendation technologies [6, 7] to improve the quality of decision processes in the context of *requirements negotiation* which is the process of resolving existing conflicts between requirements and deciding which requirements should be implemented. Functionalities often provided by group recommenders are the visualization of the preferences of other group members, recommendations for individual and group decisions, and recommendations for conflict resolutions in the case of inconsistent stakeholder preferences [6, 7]. Our motivation for applying group recommendation technologies is to improve the *usability* and the *quality of decision support* in requirements engineering environments (especially in the context of requirements negotiation).

Note that decision models based on rational thinking [8] are not applicable in most requirements negotiation scenarios since stakeholders do not exactly know their preferences beforehand [4, 9]. Furthermore, preferences are not stable but rather change over time which is an important aspect to be taken into account by requirements negotiation environments [4, 9]. The group recommendation technologies discussed in this paper are based on incremental preference elicitation [10] and thus are key technologies for preference construction [11].

For the purpose of supporting preference construction in requirements negotiation we have developed the IntelliReq decision support environment. In our scenario, student teams are allowed to configure the set of requirements that should be implemented in their software project. Note that our goal was to develop recommendation technologies which can be flexibly exploited in requirements negotiation; it is not our intention to replace existing requirements negotiation approaches (see, e.g., [12]) but to provide useful extensions.

The contribution of this paper is the demonstration of the applicability of group recommendation technologies in requirements negotiation. We show that group recommendation technologies can improve the perceived usability (in certain cases) and quality of decision support.

The remainder of this paper is organized as follows. In Section 2 we introduce the IntelliReq environment which supports group decision processes for requirements negotiation. In Section 3 we present our hypotheses defined for the empirical evaluation of IntelliReq and discuss the corresponding study results. In Section 4 we discuss related work. The paper is concluded with Section 5.

2 IntelliReq Environment

2.1 Application Scenario

IntelliReq is a group decision environment that supports computer science students at the Graz University of Technology in deciding on which requirements should be implemented within the scope of their software projects. Typically, a project team consists of 6–8 students who implement a software system with an average effort of about 8 man months. At the beginning of a project, students have to evaluate a set of requirements which have been defined by the course instructors and to figure out which requirements they will implement within the scope of their project (requirements negotiation phase). For example, the task could be the implementation of a tourist recommender application – the corresponding decision alternatives are depicted in Table 1. We will use this simple set of decision alternatives as a working example throughout the paper.

Table 1. Example decisions to be taken by the project teams – taken decisions are interpreted as agreement between the project team and the course instructors. The fulfillment of the selected requirements is an evaluation criteria.

ID	Question	Decision Alternatives
1	which application domain?	20 destinations in Austria; world-wide
2	persistence management?	relational databases; XML; Java objects
3	which type of user interface?	text-based; Java Swing; Web application
4	recommendation algorithms?	knowledge-based; collaborative & content-based
5	evaluation by whom?	students of own university; other univ.; instructors
6	type of user manual?	HTML-based; .pdf based
7	type of acceptance procedure?	live-demo; slide presentation with screenshots

2.2 User Interface & Functionalities

Example screenshots of the IntelliReq user interface are depicted in Figures 1–3. With the goal of supporting the achievement of a common group decision, the IntelliReq user interface supports the following functionalities (the IntelliReq entry page is shown in Figure 1):

- Each stakeholder is enabled to define, adapt, and store his/her preferences (*add/change personal preferences*).
- Each stakeholder can comment on and discuss already defined preferences of other users (*show and comment on preferences of group members*).
- Each group can view and discuss recommendations for group decisions determined on the basis of already defined user preferences (*show group recommendation*).
- Define and store a group decision; this can only be done by the project manager (*edit current group decision*).
- Each IntelliReq user can evaluate the application (*evaluate* IntelliReq); this user feedback has been analyzed within the scope of an empirical study.

3 Empirical Study

In order to evaluate the provided IntelliReq functionalities, we conducted an empirical study within the scope of the course *Object-oriented Analysis & Design* organized at the Graz University of Technology. The major focus of this study was to analyze the impact of group decision technologies on the dimensions *usability* of the system and *quality of decision support.*

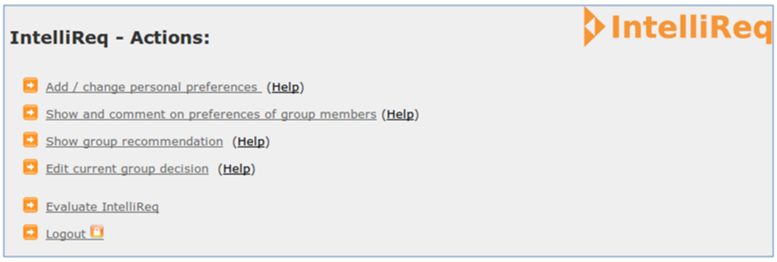

Fig. 1. Activities supported by the IntelliReq user interface. Each group member can define and adapt his/her own preferences. These preferences can be seen and discussed by other group members. On the basis of articulated user preferences and a system-determined group recommendation, the team (represented by the project manager) can define and store the team (group) decision. Team decisions can be reviewed and adapted later on (until the submission deadline for team decisions has passed).

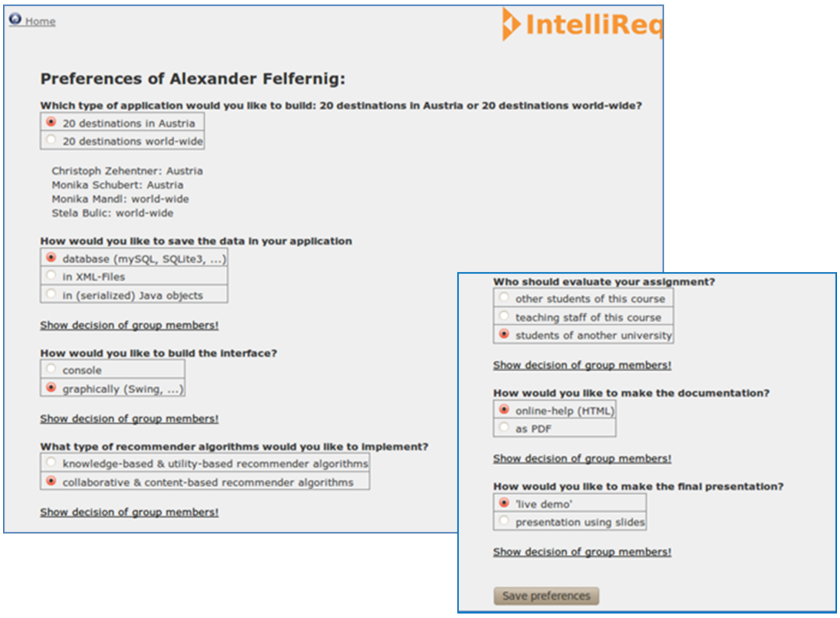

Fig. 2. IntelliReq preference specification: each group member articulates his/her own preferences and – during this process – has insights into the preferences of other group members.

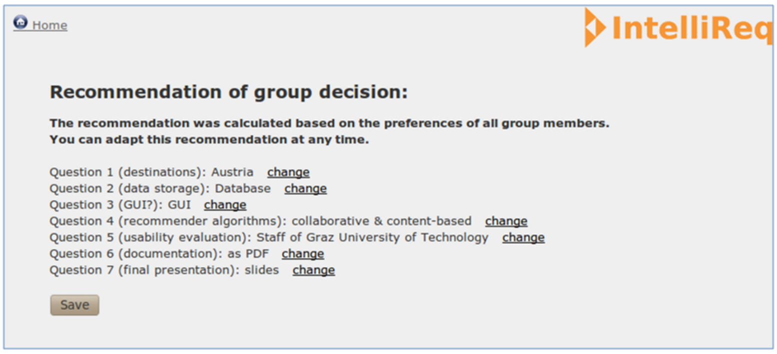

Fig. 3. IntelliReq group recommendation.

3.1 Study Design

For the purpose of the empirical study we provided the IntelliReq environment in *four* versions. In order to analyze our hypotheses, we decided to implement a 2x2 study with the variation points *group recommendations available* (*yes*/*no*) and *preferences of other users visible* (*yes*/*no*) – these versions are shown in Table 2. Both, group recommendations and preference visibility, are key functionalities provided by state of the art group recommendation environments [6, 7]. On the basis of this empirical study we wanted to investigate to which extent these functionalities are applicable within the scope of requirements negotiation.

Table 2. The four used *IntelliReq* versions. Variation points: *group recommendation supported* (*yes*/*no*) and *preferences of other team members are visible* (*yes*/*no*).

	with recommendation	without recommendation
preference view	version 1	version 3
no preference view	version 2	version 4

N=293 participants (computer science students at the Graz University of Technology, 23.1% female and 76.9% male) selected their preferred requirements using the IntelliReq environment. The participants were randomly assigned to one of 56 different groups (the development teams) and defined (stored) 3733 individual preferences and 101 group decisions. For each development team the last stored group decision was interpreted as the final decision; after the published deadline no further adaptations of the taken decisions were possible. After a user had successfully articulated his/her requirements, he/she had the possibility to give feedback on the *usability* and the *decision support quality* of IntelliReq (*evaluate* IntelliReq link in Figure 1) on a 10-point Likert scale.

3.2 Study Hypotheses

The empirical study is based on hypotheses derived from existing research in the areas of requirements engineering [4, 12, 13], group recommender systems

[6, 7, 14], and decision & social psychology [9, 15, 16, 17]. The corresponding list of hypotheses is shown in Table 3.

Table 3. Hypotheses (H) for evaluating the IntelliReq decision support environment.

H	description
H1	*group recommendations* improve the perceived system usability
H2	*group recommendations* improve the perceived quality of decision support
H3	*group recommendations* trigger more discussions
H4	*preference visibility* for all deteriorates the perceived usability
H5	*preference visibility* for all deteriorates perceived decision support quality
H6	*preference visibility* for all triggers less preference adaptations
H7	*preference visibility* triggers a decision bias
H8	winning strategy: use *group recommendation* but not support *preference visibility*
H9	*unconsidered preferences* deteriorate perceived usability & decision support quality

Group Recommendation (Hypotheses 1–3) Existing research in the field of recommender systems [6, 7, 14] points out the potential of group recommendation technologies to significantly improve the quality of group decision processes. *First* we wanted to investigate the potential of group recommendation technologies to improve the quality of the dimensions *usability* and *decision support* in a requirements negotiation scenario. With *Hypothesis 1* we express the assumption that recommendation technologies can improve the overall system quality in terms of *usability. Hypothesis 2* expresses the assumption that recommendation technologies can help to improve the perceived *quality of decision support. Second* we wanted to know whether the availability of group recommendations has an influence on the frequency of applying discussion functionalities (*Hypothesis 3*) – the underlying assumption is that the availability of group recommendations intensifies discussions between group members. This phenomenon is well known and exploited by critiquing-based recommenders where the system proposes recommendations and the user can give feedback in terms of critiques [11]. Studies in social psychology show that frequent information interchange can improve the decision quality [16, 17].

Visible User Preferences (Hypotheses 4–7) Existing research in the field of group-based recommendation points out the advantages of preference transparency in group decision making [6, 7]. In contrast, literature in social psychology points out the fact that suboptimal outcomes of group decision processes are correlated with the visibility of individual preferences of other group members [17, 16]. The reason for groups not being able to take optimal decisions (hidden-profile identification problem) is explained by an insufficient exchange of decision-relevant information triggered by the initial disclosure of individual preferences (focus shift from information interchange to preference comparison). *First* we wanted to investigate whether the group-wide visibility of individual preferences has an influence on the perceived usability and decision support quality (*Hypotheses*

4 and *5*). *Second* we wanted to figure out whether the group-wide visibility of individual preferences has an influence on the frequency of preference adaptation (*Hypothesis 6*). One underlying assumption here is that persons follow the phenomenon of *social proof* [15], i.e., are doing or accepting things that others already did (accepted). The other underlying assumption is that persons tend to stick with their current decision due to the phenomenon of *consistency* [15], i.e., the effect that published personal opinions are changed less often. *Third*, a lower frequency of information exchange can lead to a different decision outcome [16]. With *Hypothesis 7* we wanted to investigate whether the group-wide visibility of preferences can lead to a decision bias (the phenomenon of *social proof* [15]).

Winning Strategy (Hypothesis 8) We wanted to provide an answer to the question which of the four different IntelliReq versions will be evaluated best regarding usability and quality of decision support. With *Hypothesis 8* we want to express the assumption that group recommendations improve the system usability as well as the decision support quality. In contrast, making preferences of other group members visible in the group decision process deteriorates the system evaluation. Consequently, *version* 2 (see Table 2) should be evaluated best.

Distance Matters (Hypothesis 9) Finally, we wanted to provide an answer to the question whether the distance of a users's preference to the final group decision has an impact on the overall system evaluation. With *Hypothesis 9* we express the assumption that users with a low number of considered requirements will not be satisfied with the system usability and the decision support quality.

Group recommendation heuristics The *majority* rule (applied in our empirical study) is a simple but very effective heuristic in group decision making [18]: each decision is taken conform to the majority of the votes of the team members. In addition to the majority rule, there exist a couple of further heuristics [6] which can be applied when generating recommendations for groups, for example, the *fairness* heuristic which guarantees that none of the group members will be disadvantaged.[1]

3.3 Study Results

In order to identify statistically significant differences in the user quality feedback depending on the used IntelliReq version we conducted a series of two-sample t-tests. We will now discuss the results of our analysis.

Hypothesis H1 has to be rejected since the *usability* of IntelliReq versions with recommendation support is only better on the descriptive level (mean of 7.0 with vs. a mean of 6.42 without recommendation support) compared to versions without a recommendation support (see Table 4).

[1] Note that due to limited number of subjects (N=293) we were not able to compare the different recommendation heuristics with regard to the dimensions usability and quality of decision support. Such comparisons will be in the focus of future work.

Hypothesis H2 can be confirmed since we could detect a significant better evaluation of the IntelliReq *decision support* for recommendation-enhanced versions (p<0.001) compared to versions without a recommendation support. Table 4 summarizes the results of this evaluation.

Table 4. User feedback on recommendation support (mean, SD=std.dev.).

recommendation	usability	SD	decision support	SD
yes	7.0	1.67	7.07	2.03
no	6.42	2.47	5.21	2.96

Hypothesis H3 can be confirmed as well since the number of comments on individual preferences is significantly higher in versions with provided group recommendations (p<0.0015) – see Table 5. Thus we can interpret group recommendations as a stimulating element for information interchange among group members which is a key factor for high-quality group decisions [17, 16].

Table 5. Impact on information exchange frequency (SD=std.dev.).

recommendation	#comments (mean)	SD
yes	7.96	5.974
no	3.53	2.71

Hypotheses H4 and *H5* can not be confirmed since users with no access to the preferences of other group members did not provide a significantly better rating for usability and quality of decision support. However, on the descriptive level the evaluation of versions *without* preference visibility for all group members is better compared to versions *with* preference visibility (see Table 6).

Table 6. User feedback on IntelliReq preference accessibility (mean, SD=std.dev.).

preference access	usability	SD	decision support	SD
yes	6.46	2.09	6.16	2.72
no	7.0	2.08	6.25	2.64

Hypothesis H6 can be confirmed since the number of adapted individual preferences is significantly *lower* in versions with access to the personal preferences of other group members (p<0.001). This can be explained by the fact that – due to preferences visible for other users – the current user inclines to be *consistent* [15] with his/her original requirements, i.e., the willingness to change articulated preferences decreases if preferences are accessible for other users [15].

Hypothesis H7 can be confirmed since users having access to the preferences of other group members articulate preferences which are more similar to the final group decision (see Table 7). Being confronted with the preferences of other group members, persons base their decisions on the already known preferences and do not focus on the exchange of decision-relevant information which is extremely important for finding optimal decisions [16]. There is a significant biasing effect due to the visibility of preferences ($p<0.001$). This effect can be explained by the phenomenon of social proof [15] which triggers group members to do things or accept things that other group members are doing (accepting).

Table 7. Impact of preference accessibility on user preferences (mean, SD=std.dev.).

preference access	distance of indiv. preferences	SD
yes	0.28	0.09
no	0.43	0.13

Hypothesis H8 can not be confirmed. However, users with recommendation support and without insight into the preferences of other users (IntelliReq version 2 – see Table 2) provided the highest ranking for both, *usability* and *quality of decision support* (see Table 8). Versions with recommendation support outperform versions without recommendation support in terms of *decision support quality* (see Tables 4 and 8) ($p<0.001$) and versions with recommendation support and without a view on the preferences of other users clearly outperform all other versions in terms of *usability* ($p<0.001$) – see Table 8.

Table 8. Impact of preference visibility and group recommendation on usability and decision support quality (mean, SD=std.dev.).

version	recommendation	preference view	usability	SD	decision support quality	SD
1	yes	yes	6.37	1.84	7.03	2.04
2	yes	no	7.62	1.21	7.11	2.06
3	no	yes	6.56	2.38	5.13	3.07
4	no	no	6.29	2.59	5.29	2.91

Hypothesis H9 can be confirmed since users with preferences having a higher distance from the final group decision rated the IntelliReq environment significantly worse in terms of *usability* ($p<0.05$). This result conforms to the *win-lose* situations discussed in [12] which typically turn into *lose-lose* situations. We could not detect a difference in the *quality of decision support* (see Table 9).

Table 9. Relationship between the distance of individual preferences to the final decision and perceived usability and decision support quality (mean, SD=std.dev.).

#answers different from group decision	usability	SD	decision support quality	SD
≤ 2 answers	7.05	2.04	6.18	2.72
> 2 answers	6.15	2.08	6.15	2.75

4 Related Work

Group recommender systems support human decision making by taking into account factors such as beliefs (knowledge) about the opinion of other group members, knowledge about individual motivations, and personal preferences. The major goal of group recommenders [6, 7] is to achieve consensus among the members of the group – such a consensus is achieved by different heuristics such *majority voting* (for each decision preferences with an underlying majority are selected) or *fairness* (a fair consideration of the preferences of each stakeholder).

In contrast to the results reported, for example in [7], showing individual preferences to other group members is not always a good choice since this can lead to a lower perceived usability and decision support quality. This result is consistent with results of empirical studies conducted in the area of social psychology [17] where the outcome of group decisions significantly deteriorated when group members knew about the preferences of other group members. *Psychological studies* on the role of individual preferences in group decision making clearly show biasing effects in terms of significantly different outcomes of the decision process depending on whether preferences of other group members are known or not (see, e.g., [17]). This phenomenon can be explained by the fact that group members predominantly base their decisions on preferences known beforehand and not on the information generated in the decision process. As a consequence, optimal decisions (solutions) can only be identified if group members are not(!) confronted with individual preferences before starting a decision process. The failure of groups to identify acceptable or optimal decisions (so-called hidden-profile identification problem) can be explained by the insufficient discussion of unshared information (triggered, for example, by the articulation of initial preferences) and the resulting premature consensus of a group on an alternative which is not optimal [16].

Typical functionalities of group support systems in the requirements engineering context are brainstorming, idea organization, voting mechanisms, discussion forums, and shared drawing [12]. Group support systems can help to significantly reduce requirements engineering efforts and achieve higher-quality results [12]. Compared to integrated group support environments (see, e.g., [12]), IntelliReq focuses on the specific aspect of group recommendation. For existing requirements engineering environments (see, e.g., [12]), the concepts presented in this paper can contribute to achieve more effective decision processes. Note that IntelliReq technologies can be applied in the context of different negotiation constellations [19] such as sales meetings, application requirements definition, reactive product line scoping, and release planning. Finally, we want to emphasize that models of human decision making on the basis of rational thinking [8] are not applicable in requirements engineering scenarios since preferences of stakeholders are not stable, i.e., change over time. Existing requirements engineering environments neglect this important aspect [4]. The group decision technologies presented in this paper are based on incremental preference elicitation [10] which provides a solid basis for handling unstable preferences [11].

The *application of recommendation technologies* in the context of requirements engineering is a constantly evolving research field [4, 5]. The current research focus is on the application of machine learning approaches to the generation of coherent sets of requirements [5]. An example for the application of these technologies are users of open source CRM environments who perform badly when having to identify the appropriate discussion forum for a certain feature request. Another application of clustering techniques is introduced in [20] where an intelligent requirement grouping mechanism is applied to support the construction of feature models. As far as we know there do not exist any applications of group recommendation technologies in the context of requirements negotiation. We see the results presented in this paper as a first step to improve the overall decision quality in different phases of the requirements engineering process (e.g., evaluation, negotiation, and planning). For a comprehensive overview of potential application areas of recommendation technologies in requirements engineering we refer the reader to [21].

5 Conclusions

In this paper we introduced the IntelliReq decision support environment which is used at the Graz University of Technology for supporting group decision processes in small-sized software projects (6–8 team members). Each group in the empirical study interacted with exactly one version of IntelliReq – the four versions provided differed in terms of the *availability of recommendation support* (yes/no) and the possibility to take a look at the *preferences of other users* (possible/not possible). The major results of this experiment were that group recommendation can improve the perceived usability (in specific cases) and quality of decision support. It is not recommended to disclose the preferences of individual group members at the beginning of a decision process since the knowledge of the preferences of other group members can lead to an insufficient exchange of decision-relevant information. The results of our study clearly indicate that deep knowledge about human decision making can help to improve the overall quality of decision support environments. The investigation of further psychological issues is within the scope of our future research.

Acknowledgments

The work presented in this paper has been conducted within the scope of the IntelliReq project (829626) funded by the Austrian Research Agency (Intelligent Recommendation Technologies for Software Requirements Engineering).

References

[1] K. Pohl. *Process-Centered Requirements Engineering*. John Wiley and Sons, 1996.
[2] H. Hofmann and F. Lehner. Requirements engineering as a success factor in software projects. *IEEE Software*, 18(4):58–66, 2001.

[3] A. Aurum and C. Wohlin. The fundamental nature of requirements engineering activities as a decision-making process. *Information and Software Technology*, 45(14):945–954, 2003.
[4] B. Alenljung and A. Persson. Decision-making activities in the requirements engineering decision processes: A case study. In *ISD 2005*, pages 707–718, Karlstad, Sweden, 2005.
[5] C. Castro-Herrera, C. Duan, J. Cleland-Huang, and B. Mobasher. Using data mining and recommender systems to facilitate large-scale, open, and inclusive requirements elicitation processes. In *16th IEEE Intl. Conf. on Req. Engineering (RE'08)*, pages 165–168, Barcelona, Spain, 2008.
[6] J. Masthoff. Group recommender systems: Combining individual models. In *Recommender Systems Handbook*, pages 677–702. Springer, 20101.
[7] A. Jameson, S. Baldes, and T. Kleinbauer. Two methods for enhancing mutual awareness in a group recommender system. In *ACM Intl. Working Conference on Advanced Visual Interfaces*, pages 48–54, Gallipoli, Italy, 2004.
[8] D. McFadden. Rationality for economists? *Journal of Risk and Uncertainty*, 19(1):73–105, 1999.
[9] J. Bettman, M. Luce, and J. Payne. Constructive consumer choice. *Journal of Consumer Research*, 25(3):187–217, 1998.
[10] J. Blythe. Visual exploration and incremental utility elicitation. In *14th National Conference on Artificial Intelligence (AAAI02)*, pages 526–532, Edmonton, Alberta, 2002.
[11] P. Pu and L. Chen. User-involved preference elicitation for product search and recommender systems. *AI Magazine*, 29(4):93–103, 2008.
[12] B. Boehm, P. Gruenbacher, and R. Briggs. Developing groupware for requirements negotiation: Lessons learned. *IEEE Software*, 18(3):46–55, 2001.
[13] B. Nuseibeh and S. Easterbrook. Requirements engineering: A roadmap. In *Conference on the Future of Software Engineering*, pages 35–46, 2000.
[14] A. Felfernig, M. Mandl, M. Schubert, W. Maalej, and F. Ricci. Recommendation and decision technologies for requirements engineering. In *2nd International Workshop on Recommendation Systems for Software Engineering (RSSE10)*, pages 11–15, 2010.
[15] R. Cialdini. The science of persuasion. *Scientific American*, (284):76–81, 2001.
[16] T. Greitemeyer and S. Schulz-Hardt. Preference-consistent evaluation of information in the hidden profile paradigm: Beyond group-level explanations for the dominance of shared information in group decisions. *Jrnl. of Personality & Soc. Psych.*, 84(2):332–339, 2003.
[17] A. Mojzisch and S. Schulz-Hardt. Knowing other's preferences degrades the quality of group decisions. *Jrnl. of Personality & Soc. Psych.*, 98(5):794–808, 2010.
[18] R. Hastie and R. Kameda. The robust beauty of majority rules in group decisions. *Psychological Review*, 112(2):80–86, 2005.
[19] W. Heider, P. Gruenbacher, and R. Rabiser. Negotiation constellations in reactive product line evolution. In *4th International Workshop on Software Product Management (IWSPM'10)*, pages 63–66, 2010.
[20] K. Chen, W. Zhang, H. Zhao, , and H. Mei. An approach to constructing feature models based on requirements clustering. In *13th IEEE Intl. Conf. on Req. Engineering (RE'05)*, pages 31–40, Paris, France, 2005.
[21] W. Maalej and A. Thurimella. Towards a research agenda for recommendation systems in requirements engineering. In *2nd Int. Workshop on Managing Requirements Knowledge (MARK09)*, Atlanta, USA, 2009.