Using a Liquid Democracy Tool for End-user Involvement in Continuous RE

Jonathan Seesink[1], Stijn Hoppenbrouwers[2,1]

[1] Radboud University, Department of Software Science,
Toernooiveld 212, 6525 EC Nijmegen, the Netherlands
j.seesink@yahoo.nl
[2] HAN University of Applied Sciences, Model-Based Information Systems Group,
Ruitenberglaan 26, 6802 CE Arnhem, the Netherlands
stijn.hoppenbrouwers@han.nl

Abstract. This paper reports on a case study exploring the idea that e-democracy approaches, more in particular the 'liquid democracy' variety, can support the ongoing communication about requirements involving an application's user community and requirements practitioners. A small scale explorative study was carried out in a real life context, focusing on the user community of an application used internally in a consultancy company and deploying an existing Liquid Democracy Tool. Interviews and a questionnaire were used to inquire about effects of the system in view of requirements understanding, and the motivation of users to participate in requirements-related activities. While the recorded use of the system was disappointing, we believe our results provide some worthwhile insights into factors at play in involving users in continuous RE through using a (Liquid) Democracy Tool.

Keywords: liquid democracy, requirements, user communities, stakeholder involvement

1 Introduction & background

Traditional Requirements Engineering approaches such as interviews and workshops only gather requirements based on a small group of users (Johann & Maalej, 2015). When elicitation of requirements is based on a small number of users, potential requirements may not be captured and a systematic user-oriented prioritization of requirements may be hampered. The idea underlying this paper is that user involvement could be improved by letting an (in principle) unlimited number of people participate in the requirements decision process. This concept is not new. The field of e-democracy aims to communicate to politicians individuals' opinions relating to public issues (Thomas & Streib, 2005). User involvement can be a big success factor in building software applications (Bano & Zowghi, 2015). Application reviews by users on online platforms indicate that users are intrinsically willing to contribute to the product's success (Maalej & Pagano, 2011).

Direct democracy means that every person can vote for any proposal. Indirect democracy means that people vote for a representative who makes direct decisions. Liquid democracy is characterized by the possibility that people vote directly for proposals and delegate their voting power to others (Paulin, 2014); people can adjust their votes or delegations anytime if they want to. Liquid democracy can be considered a new democratic voting system that mixes properties of two existing democratic voting systems: direct democracy and representative democracy.

A representative democracy entails a voting system in which every individual has the right to vote on a group of representatives that makes decisions for the group for all or selected options. A direct democracy entails a voting system in which every individual has the right to vote for every option. Options are designed by groups of individuals (political party). A liquid democracy is a voting system in which an individual has the right to vote for every option but can also copy votes of other individuals (which are called proxies). Proxies can copy votes from other individuals resulting in a ‘delegate cascade’. The liquid democracy concept is sometimes also called ‘proxy voting system’, ‘delegated voting’ or ‘direct/proxy voting system’ (Green-Armytage, 2015). The Pirate Party in Germany, Austria, Italy, Switzerland and Brazil choose to call it liquid democracy (Green-Armytage, 2015), a term which was adopted by other researchers (Johann & Maalej, 2015). In the implementation used in the case study, votes are copied wholesale from a delegate, not for selected options.

Software engineers created Liquid Democracy Tools (LDTs) but these were never used for the purpose of requirements engineering or ‘Online Requirements Gathering’ (ORG). Liquid democracy techniques may have the potential to let RE practices involve the total crowd of end users of software applications. The field of Large Scale Social Requirements Engineering aims for communities to formulate requirements collaboratively. A tool that is made based on this philosophy is Requirements Bazaar (Renzel and Klamma, 2014). It is a requirements elicitation tool that focuses on open source projects. The tool implements a voting system for proposals. However, it does not support delegation of voting power and therefore misses an important aspect of the liquid democracy concept. Besides, it focuses on a negotiation process between users and developers which is out of the scope for the current study, which focuses more on the idea that LD techniques may be valuable as source of inspiration from end users for other stakeholders such as developers. Requirements Bazaar or other liquid democracy tools create online communities. Typically, only a small fraction of users really use such tools. Designers may use social psychology insights to leverage participation in online communities and thus increasing the online community's usefulness (Ling et al., 2005). Recognizing the public through liquid democracy techniques could potentially motivate users to participate, increasing the potential value of a LDT for RE (Johann & Maalej, 2015).

In a small case study we used a tool that implemented Liquid Democracy techniques for receiving feedback by end users about an existing software application in a real context (consultancy company, CC). We set out to investigate the effects of using the tool for RE, and the motivation of users to participate in the requirements decision process. The outcomes of the research, while admittedly disappointing in terms of actual use of the LDT, do provide an indication of the possible added value of using liquid democracy techniques for the continuous ORG processes related to an existing software application, and in particular point to challenges in using (liquid) democracy tools for requirements engineering. Findings may also be used for informing the (improved) design and deployment of web applications making requirements elicitation possible based on the crowd of users, and for developing the concept of ORG in general.

2 Research Setup

Our main research question was: *how can liquid democracy techniques contribute to the RE processes of existing software applications?*

The following sub questions were formulated:

1. What are the effects of using LD techniques on the requirements understanding of software developers/designers and users?
2. What are potentials and areas of concern when applying LD techniques for gathering end user requirements of an existing web application?
3. Do end users feel more motivated to participate in the RE processes by LD techniques?
4. How can findings be used to contribute to the RE processes of a specific software supplier?

Fig. 1. shows the consecutive, interrelated data gathering steps of the research setup. General RE practices of the CC (Consultancy Company) were determined by conducting 10 interviews with *requirements practitioners* of three out of the fifteen different business units of the CC. We define a requirements practitioner as “person working in his profession with the elicitation, analysis, documentation and change management of requirements for software projects”. An LDT was set up in order to gather end user requirements related to a Software Application (SA) that the CC uses for internal work processes. Using a questionnaire (25 respondents) and interviews with SA requirements practitioners, it was investigated whether the results added value to the requirements understanding of the requirements practitioners of the SA and the end users who used the LDT. Also it was investigated whether LD techniques can be seen as motivational incentives for end users to participate in requirements activities. Two interviews were conducted with the requirements practitioners of the SA, one interview before and one interview after the application of the LDT. Based on the findings of the LDT application case study a second round of interviews with requirements practitioners of the CC was conducted to investigate potentials and areas of concern of using a LDT for RE purposes of the CC. The interviews were recorded, transcribed and coded for processing.

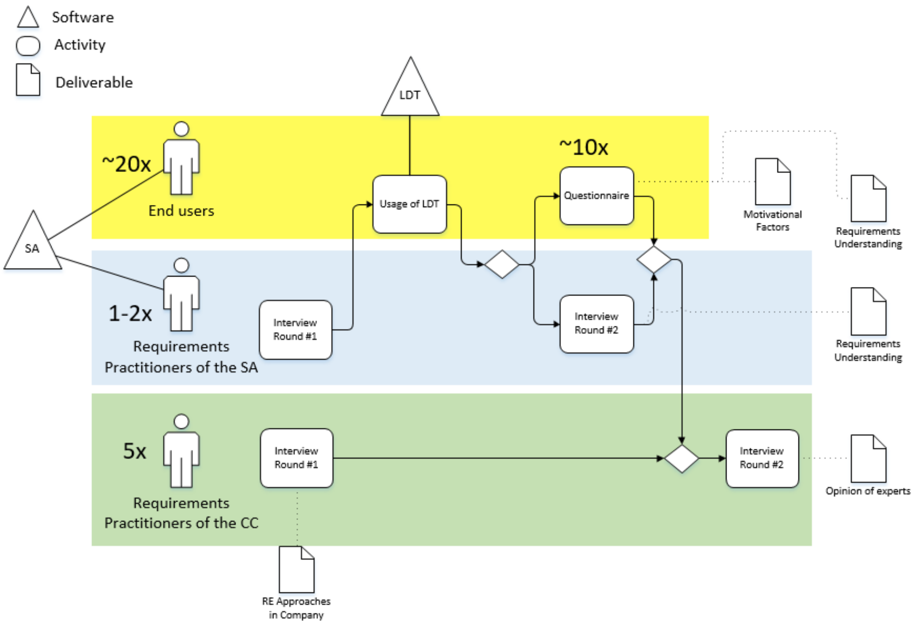

Fig. 1: Research setup for data gathering

The following criteria were determined for the selection of a LDT:

- The tool implemented the main aspect of LD (delegate voting power to other users and vote on proposals directly)
- The source code of the LDT was publicly available, changes to it were allowed, and the source code was documented
- The LDT could be easily installed and run on a standard server environment
- The interface was perceived as simple to use

The only tool (of four tools considered) that matched these criteria was the web application Liquidizer (2012 version) by Stefan Dirnstorfer (Dirnstorfer, 2010). Therefore, this application was chosen as the basis for the LDT applied in the investigation.

Beside the main aspect of LD (delegate voting power to other users and vote on proposals directly) the LDT provided the functionality to create new proposals, adjust the votes anytime, see the popularity of other users (based on delegated voting power and their votes) and the possibility for a user to post one comment for each proposal.

The term 'requirement' was avoided in the slightly modified interface of the LDT because it might suggest that end users could expect their (votes on) proposals to be actually implemented. Instead, all the pages included the clause “wishes for the SA” (while the LDT displayed the real name of the software application instead of SA).

3 Findings & Discussion

What are the effects of using LD techniques on the requirements understanding of software developers/designers and users?

The usage of an LDT slightly changed the requirements understanding of software developers and users. However, the number of respondents participating in the LDT was too low to consider the LDT results reliable. Disappointingly but also tellingly, the main aspect of LD, namely the possibility to vote representatively, was not used through the LDT. Therefore, the noted changes in requirements understanding have not been achieved through LD techniques, but by the more general means of using an online requirements gathering tool. Activities for obtaining vote delegations suggested in the LDT were perceived as very time consuming.

There ought to be significant evidence that the information provided in the LDT is reliable. Further research could focus on criteria that indicate when input by end users would be reliable and how to reach such a reliable input. The number of end users participating in the LDT is of course crucial for this.

The interviews provide indications that end users did not fully understand the concept of LD or did not accept the mechanisms of the LDT that were supposed to implement the LD concept. Further research and designs should pay more attention to good explanation of the LD concept to users and to concrete mechanisms/policies implementing LD, fitting the needs of users both functionally and communicationally.

Finally, an area of concern is the audience that is allowed to use a LDT for RE. This group of people should be more clearly defined. Should all stakeholders participate in such a tool? Should key stakeholders or minorities of end users obtain a higher voting weight (arguably damaging the true democratic nature of the approach)? Should developers be allowed to comment on the input of end users for acceptance or expectation management? How can a distinction be made between the different stakeholders (e.g. a key stakeholder can be also an end user in some cases?).

Do end users feel more motivated to participate in the RE processes by LD techniques?

All LDT end users responded mainly because they were asked by the researcher or unit manager to do so. No LD technique used in the LDT during the case study can be considered a motivational incentive for end users to participate; further research should more tenaciously investigate the motivation for stakeholders to participate. Attention should not only be paid to end users but also to the requirements practitioners who need to facilitate the requirements process.

How can findings be used to contribute to the RE processes of the CC?

All requirements practitioners of the CC stated that they would not immediately use an LDT for ‘Online Requirements Gathering’ (ORG) activities. Main reasons given were the fact that an LDT (or an ORG tool in general) cannot be used in every case and that the quality of results of the LDT application was lacking. ORG added value to a developer of a web application in which end user satisfaction was considered most important. In the case study company (CC), end user satisfaction does not always have the highest priority. The need for direct communication with end users is therefore low for two of the three business units studied. Within the other business unit, LD techniques for RE could add value for the CC.

The CC could experiment with an LDT with more resources than were available for this case study. We used a LDT without heavily customising it for the context of RE. A further step could be that the CC develops a (L)DT that better fits RE activities. Such an implementation could address potentials and areas of concern reported in this research.

The LDT should ideally be in common use by many people, with everyone sufficiently understanding the main LD principle and minimal hindrance to the RE activities. This might be better possible through offering a common portal for gathering requirements across all software applications used in an organization, or even across a number of organisations, making it part of a general 'improvement culture'. A company such as the case CC could, for example, promote inclusion of links to such a portal in applications they implement or advise on.

Conclusion to the main question

This research confirms, albeit not decisively, that ORG can contribute positively to RE processes. However, LD techniques likely did not significantly influence the results in the case study as such. The applied LDT as one further example of ORG added value in comparison to other techniques the developers of the SA frequently use, but any specific contribution to this by use of the LD variety has not been confirmed.

In the mean time, a number of potentials and areas of concern have been identified. Two of them were not mentioned in existing literature (Johann & Maalej, 2015): i. the need for common use and ii. the possibility that stakeholders do not understand the concept of LD. Other issues, like security/privacy, were hardly touched upon by the subjects interviewed, but should nevertheless be addressed in further research.

Despite the results, LD techniques might still contribute to ORG and therefore also to the RE processes in general. It should, however, be ensured that at least the areas of concern mentioned in this study are covered in a LDT that is used in a production environment. For this, more research is needed as suggested in the answers to the sub questions.

The areas of concern of *common use* and *understanding of specific democratic systems* may be specifically addressed through some application implementing main LD techniques and its use and understanding by a majority of people involved, but can also be added to the generic set of concerns for ORG.

From the perspective of LD as such, the study reported on is of course only one example of applying the LD concept, in a somewhat eccentric context. Many areas of concern are not limited to the RE field and are also relevant for other fields of decision making. The concept of LD as such is, after all, not yet a mature one.

This paper is a condensed report derived from the Master's thesis of Jonathan Seesink, part of the Information Science curriculum of Radboud University, Nijmegen, the Netherlands. Many details of background, method, and results were necessarily left out. For the original thesis (Seesink, 2017), please contact the authors.

References

Bano, M. & Zowghi, D. (2015). 'A systematic review on the relationship between user involvement and system success', Information and Software Technology 58, 148-169.

Dirnstorfer, Stefan (2010). A Voting System for Internet Based Democracy. Available at SSRN: https://ssrn.com/abstract=1679002 or http://dx.doi.org/10.2139/ssrn.1679002

Green-Armytage, J. (2015). Direct Voting and Proxy Voting'. In: Constitutional Political Economy 26(2), 190-220.

Johann, T. & Maalej, W. (2015). Democratic Mass Participation of Users in Requirements Engineering?, in proceedings of the 23rd International Requirements Engineering Conference (RE 2015), , pp. 256-261. IEEE.

Renzel, D. & Klamma, R. (2014). Requirements bazaar: Open-source large scale social requirements engineering in the long tail. In: IEEE Computer Society Special Technical Community on Social Networking E-Letter 2.

Ling, K., Beenen, G. Gerard, Ludford, P., Wang, X., Chang, K., Li, X., Cosley, D., Frankowski, D., Terveen, L., Al Mamunur, R., Resnick, P., Kraut, R. (2005). Using Social Psychology to Motivate Contributions to Online Communities. In: Journal of Computer-Mediated Communication 10(4). Blackwell Publishing Ltd.

Maalej, W. & Pagano, D. (2011). On the Socialness of Software. In: Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE 9th International Conference on Dependable, Autonomic and Secure Computing, pp. 864-871. IEEE.

Paulin, A. (2014). Through liquid democracy to sustainable non-bureaucratic government, in 'Proc. Int. Conf. for E-Democracy and Open Government', pp. 205_217.

Seesink, J. (2017). Liquid Requirements Engineering. Master's thesis, Institute for Computing and Information Sciences, Radboud University, Nijmegen, the Netherlands.

Thomas, J. C. & Streib, G. (2005). E-democracy, e-commerce, and e-research: Examining the electronic ties between citizens and governments, Administration & Society 37(3), 259-280.