

Design and Management of Web-Based Innovation Communities: A Lifecycle Approach

Stefan Hallerstede

Information Systems I - Innovation & Value Creation, University of Erlangen-Nuremberg, Nuremberg, Germany,
stefan.hallerstede@wiso.uni-erlangen.de

Angelika Bullinger

Information Systems I - Innovation & Value Creation, University of Erlangen-Nuremberg, Nuremberg, Germany,
angelika.bullinger@wiso.uni-erlangen.de

Kathrin Möslein

IS I: Innovation & Value Creation, University Erlangen-Nuremberg, Nuremberg, Germany, kathrin.moeslein@wiso.uni-erlangen.de

Follow this and additional works at: <http://aisel.aisnet.org/amcis2012>

Recommended Citation

Hallerstede, Stefan; Bullinger, Angelika; and Möslein, Kathrin, "Design and Management of Web-Based Innovation Communities: A Lifecycle Approach" (2012). *AMCIS 2012 Proceedings*. 5.
<http://aisel.aisnet.org/amcis2012/proceedings/VirtualCommunities/5>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISEL). It has been accepted for inclusion in AMCIS 2012 Proceedings by an authorized administrator of AIS Electronic Library (AISEL). For more information, please contact elibrary@aisnet.org.

Design and Management of Web-Based Innovation Communities: A Lifecycle Approach

Stefan H. Hallerstede

University of Erlangen-Nuremberg
stefan.hallerstede@wiso.uni-erlangen.de

Angelika C. Bullinger

Chemnitz University of Technology &
University of Pennsylvania
awi@tu-chemnitz.de

Kathrin M. Moeslein

University of Erlangen-Nuremberg
kathrin.moeslein@wiso.uni-erlangen.de

ABSTRACT

Nowadays, web-based communities are a popular means to integrate external innovators into the innovation process of organizations. Combining extant research in innovation management and IS management, we integrate open innovation and application lifecycle management (ALM) to present an integrated understanding of the design and management of innovation communities. Therefore, the paper draws on an in-depth explorative case study. We describe the process of community design and management along the phases of ALM. From a socio-technical systems perspective, the manager of an innovation community has to specialize in designing and managing the social subsystem rather than the technical subsystem of an innovation community. Accordingly, we reveal that the community manager's core asset is a specialized backend that supports these management tasks.

Keywords

Open Innovation, Innovation Community, Application Lifecycle Management, Socio-Technical Systems, Case Study.

INTRODUCTION

The capability to generate innovative products and services is a critical success factor for organizations in nowadays dynamic markets (Christensen, 2006). Prior work has shown that organizations rely more and more on integrating the innovative potential of external actors like consumers, customers, partners and other stakeholders (Chesbrough, 2006; Leimeister, Huber, Bretschneider and Krcmar, 2009; Piller and Walcher, 2006). Web-based communities for innovation are a popular means to do so (Bateman, Gray and Butler, 2010; Füller, Bartl, Ernst and Mühlbacher, 2004).

Whilst communities differ in structure, topical focus and the extent of social ties (Armstrong and Hagel III, 1996; Füller, Bartl, Ernst and Mühlbacher, 2006; Lynn, Aram and Reddy, 1997; Muniz Jr. and O'Guinn, 2001; Rheingold, 2000), they are mainly based on shared enthusiasm and knowledge concerning a particular domain (Kozinets, 1999). In a web-based context, also the technical basis allowing for interaction among the community members, i.e. the community platform, has to be considered. On an innovation community platform, community members for instance discuss new ideas for products or how to improve them (Füller, Matzler and Hoppe, 2008; von Hippel, 2005). The willingness to contribute is thereby dependent on the particular characteristics of the community, like language, netiquette, norms and the individual motivation of the community members (Armstrong and Hagel III, 1996; Bateman et al., 2010; Butler, Sproull, Kiesler and Kraut, 2007; von Hippel, 2005; Williams, 2000). Consequently both the design, which influences for instance the community characteristics, and the management of a community, which influences for instance the netiquette and norms, are critical for its success (cf. Armstrong & Hagel III, 1996; Butler et al., 2007; Füller et al., 2004; Williams, 2000).

Current research offers insights into both aspects: community design (Bullinger, Neyer, Rass and Moeslein, 2010; Ebner, Leimeister and Krcmar, 2010) as well as community management (Adamczyk, Boehler, Bullinger and Moeslein, 2011; Bateman et al., 2010). As researchers and practitioners though still lack guidance on which activities to focus on and how they interrelate, this study bridges these two streams of research by applying application lifecycle management (ALM), a traditional software lifecycle management approach (OGC, 2002), to the field. Thereby, we aim at an integrated

understanding of the design and management of innovation communities. This study thus asks: *How to structure the design and management of a web-based innovation community using application lifecycle management?*

In order to answer the research question, we present an in-depth case study of a company called *innosabi* that designs and manages its innovation community *userAller*. The paper proceeds as follows: Chapter 2 will lay the theoretical foundations of communities and ALM. Chapter 3 will give an overview of the methodology of the case study. Chapter 4 introduces the case setting followed by the findings in Chapter 5. The paper closes with a discussion and conclusion in Chapter 6.

THEORY

Web-based communities for innovation can be characterized by a set of design elements to describe their design and management (Bullinger et al., 2010). These design elements contain both social as well as technical dimensions. Thus, when designing and managing an innovation community, the social *and* the technical subsystems have to be considered as already proposed by socio-technical systems thinkers in a multitude of settings (Neyer, Bullinger and Möslein, 2009; Pasmore, Francis, Haldeman and Shani, 1982; Trist and Bamforth, 1951). Innovation literature mainly deals with the social subsystem (e.g. Harhoff, 2003; Leimeister et al., 2009), whereas IS literature tends to focus on the technical subsystem (e.g. Hoffmann & Beaumont, 1997; OGC, 2002; Sommerville, 2007). In order to develop an integrated approach to design and manage the social as well as the technical subsystem of innovation communities, we combine the strands of innovation and IS research in accordance with Pasmore et al. (1982) and Neyer, Bullinger and Möslein (2009). To analyze the design and management of web-based innovation communities, we draw on application lifecycle management (OGC, 2002). From an ALM perspective and in the conception of this paper, a web-based innovation community is an application that has to be designed and managed. ALM is particularly well suited for this case, as it represents – opposing to other software lifecycle models – a model which balances the design and management activities in the lifecycle of an application, i.e. an innovation community (OGC, 2002; Sommerville, 2007).

Within ALM, application creation and service management are distinguished (OGC, 2002). Whereas application creation refers to the ALM phases *requirements*, *design* and *build* in the lifecycle, service management refers to the phases *deploy*, *operate* and *optimize* (Figure 1). The phases do not necessarily run consecutively, but can overlap due to parallel circles or iterations (OGC, 2002). Thus, optimization for instance can be a continuous activity.

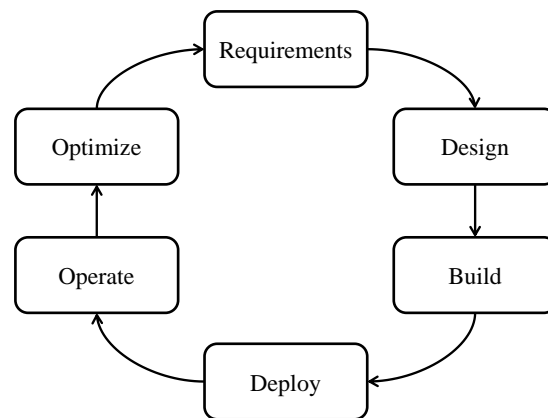


Figure 1: Phases of the application lifecycle management framework (OGC, 2002)

The lifecycle of an application, i.e. a community platform, begins with the gathering of functional and non-functional *requirements*. In the *design* phase these requirements are translated into feature specifications with the aim to design a socio-technical system. In the *build* phase the application and architecture are realized, namely the implementation of the technical subsystem starts. New components are purchased or built and subsequently integrated and tested. Once the system is built, *deployment* starts. Therefore the (changed) architecture has to be implemented to existing systems and the application has to be made available. During *operation*, support has to be given to the users and changes in the requirements have to be noticed. The final phase in the ALM circle is *optimization*. During this phase the results from operations are analyzed. Therefore, feedback of the users has to be collected and other means of evaluation have to be taken. Based on the analysis required changes are assessed and initiated (OGC, 2002). Drawing on ALM, the paper continues to analyze the lifecycle of a sample innovation community.

METHOD

In order to better understand the lifecycle of web-based communities for innovation and potential structuring thereof, an explorative research approach is chosen (Schnell, Hill and Esser, 2008). We perform an in-depth case study on an innovation community in the field which is designed and managed by a specialized organizer (Runeson and Höst, 2008; Yin, 2003). The community is active since 2010. Five semi-structured, explorative interviews with executives of the organizer are conducted in 2011 to assess their approach to design and manage the community (Table 1).

Position of interviewee	Duration [sum in minutes]	Major insights
CEO (3 interviews)	177	Strategic positioning and outlook; development of the company and its community; community management
CIO (1 interview)	81	IT organization; design of the community; origin and prioritization of requirements
CTO (1 interview)	98	Architecture and versions of the community; division of tasks

Table 1: Interview partners

Topics of the interviews included the company in general, its competencies and processes, the concept of their community and its lifecycle management. The interviews were conducted in the company's offices using the critical incidents technique (Flanagan, 1954). Transcripts from the interviews (342,241 characters) were independently coded by two of the authors in MAXQDA 10 (Lindsay, 2004). Coding started with a codebook *deducted* from application lifecycle management. To *inductively* explore activities of innovation community design and management, we also used grounded theory methodology. We chose this *abductive* approach, i.e. the combination of deductive and inductive proceedings because of the ability of grounded theory to analyze qualitative data systematically and to uncover underlying relationships. New aspects of innovation community design and management were initially coded as concepts and by constant comparison (Strauss and Corbin, 1998) iteratively condensed to a higher level of abstraction.

In addition to the interview data, further material was used for the case study, including the company's online presence¹, internal documents (strategy presentations, visual material, documentation etc.) as well as its community platform. The different data sources are triangulated (Yin, 2003).

THE INNOVATION COMMUNITY AND ITS MANAGER

innosabi sees itself as a service provider that offers consulting on the integration of consumers into a company's innovation process. innosabi's core asset is the innovation community unserAller, where companies can jointly develop products with consumers (Figure 2). Their clients include dm, Görtz, LEXA and other large, medium and small-sized producers of consumer goods. innosabi has been honored as the most successful business start-up in the field of information and communication technology by Germany's federal minister Dr. Philipp Rösler in June 2011 (BMW, 2011).

innosabi offers its innovation community of approximately 13.000 users (as of February 2012) to their clients. Thereby innosabi is responsible to design the community platform and carry out the community management. Consumers can participate in the development of new products based on a standardized innovation process developed by innosabi. The steps in each phase of the innovation process are the same: it starts with a period of suggestions and discussions by the community, continues with a short review period for the client and ends with an evaluation by the community. This setup ensures the strategic as well as technical fit of community-innovations with the client whilst it maintains a sense of ownership by the community. A comprehensive description of innosabi, the innovation process of unserAller and evaluation mechanisms can be found in Hallerstede, Bullinger and Möslin (2012).

¹ <http://www.innosabi.com/>; <http://www.unserAller.de/>

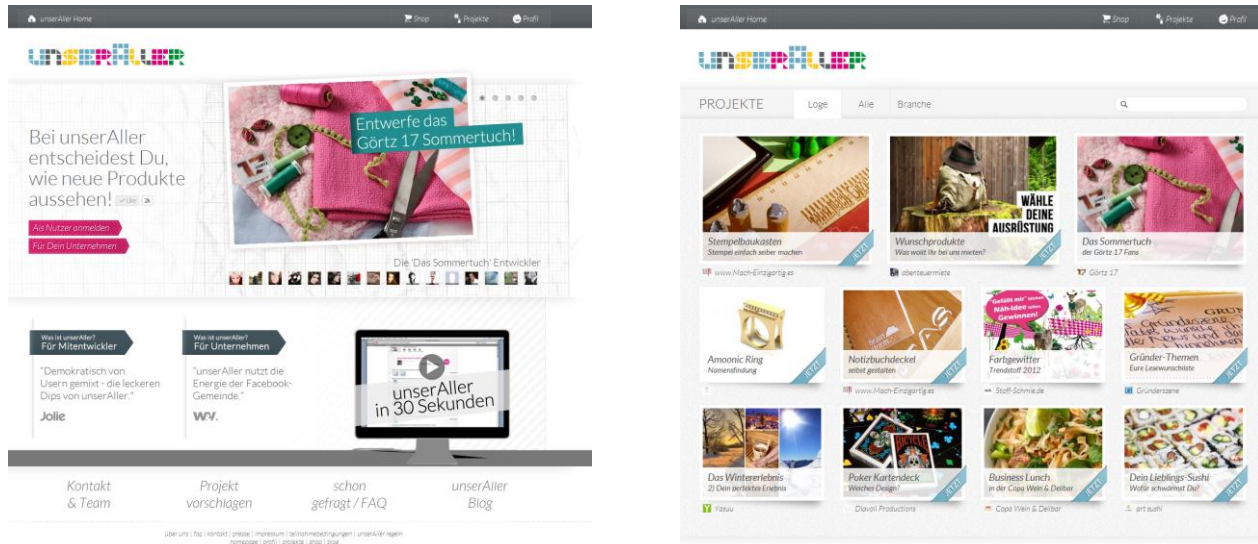


Figure 2: Homepage and project overview of unserAller (version Pegasus)

ALM AT INNOSABI

The following details describe the process how innosabi designs and manages their community unserAller.

Requirements

The initial idea of unserAller originates from prior experience of the founders who have been consultants in the field of open innovation for an average of more than five years. From their projects, they see a demand for user integration in the new product development of consumer goods. They strive to build a platform to collaboratively develop innovative products that meet the requirements of consumers. Next to brainstorming within the team, innosabi includes their Facebook community to develop ideas for such a platform.

Having a first concept in mind, they include both potential customers (i.e. producers) and consumers (i.e. innovators) to further gather requirements. At this point, it is an unstructured approach. From the first group of potential customers, they elicit requirements by short discussions with multiple parties and detailed discussions with single companies. They present their initial idea and ask how they would prefer the process and what they are concerned about. The mayor requirements cover target group considerations and producibility of the results. From a target group perspective, the idea is to integrate all potential consumers, rather than a special group of them, e.g. designers or scientists, as it can be found in other communities. The major aim is to get closer to consumers than with traditional approaches.

„I you look at it from the perspective to get to know the user or that you want creative ideas, you will face a lot of frustration with traditional methods. If I do market research and statistics, I will have nice figures, but after all it will not represent reality.“ (CEO)

Requirements elicited from the consumer perspective are a playful experience and the reassurance that their contributions are acknowledged and realized. At the end of the day, they want to profit from the developed products. This leads to the necessity that products which are developed by the unserAller community are available in stores within a reasonable amount of time. This limits the selection of entitled products whereupon convenience goods are particularly suited.

“The basic idea of unserAller arose from the observation that people identified themselves with co-created products. We supposed that it would work even better for convenience goods. Meaning things, I build a relation to anyway and with which I identify myself and my lifestyle somehow. Things that are quickly producible. Not like a car that needs 10-15 years of preparation but rather things we [in cooperation with the client] can bring to market within half a year.“ (CEO)

With these basic requirements and ideas, innosabi starts to design the initial version of unserAller.

Design

In the design phase innosabi uses a trial-and-error approach. They draw rough wireframes that show different options to realize the requirements. They thereby focus on designing the social subsystem.

„You can see it at this [screenshot of the] prototype. We have always been rather design driven. A good look was always important. So you will have fun to participate and it will work somehow. [...] That's how unseerAller was developed. A lot of prototypes and trial-and-error. Having a spark and simply trying it. If it does not work, we leave it be. And that is it.” (CEO)

Discussions lead to a first design draft of unseerAller in which innosabi strives for a well-managed community. Thus from the beginning, they consider a backend to actively manage the community, e.g. by reviewing suggestions and comments. They know from their experience that this is crucial for an innovation community's success.

Though the basic structure of the social subsystem is stable, details of it vary over time (Figure 3): The first version of unseerAller (Donkey) can be seen as a proof-of-concept for the innovation process, which was developed in a trial-and-error approach, whereas the second version (Zebra) is a trial-and-error sandbox for details of the social system around the innovation process. innosabi for instance introduced opportunities to attach pictures and documents to suggestions or asked additional profile information to test acceptance among their users. Also frequent changes to the user interface are performed to highlight certain aspects of the social system (e.g. prominent commenting). During the runtime of the first innovation project, innosabi figured to send out prototype packages to connect the online and the offline world. Now, this is one of the key value propositions of unseerAller.

„What motivates them is fun doing it. If you get such a package and can try out things, it becomes very playful. There is this term of gamification and that is a valid point. If you enjoy doing it, you will do a lot.” (CEO)

Also, with the experience from the first projects, the community management backend is refined to meet the requirements of the community managers in order to facilitate their work.

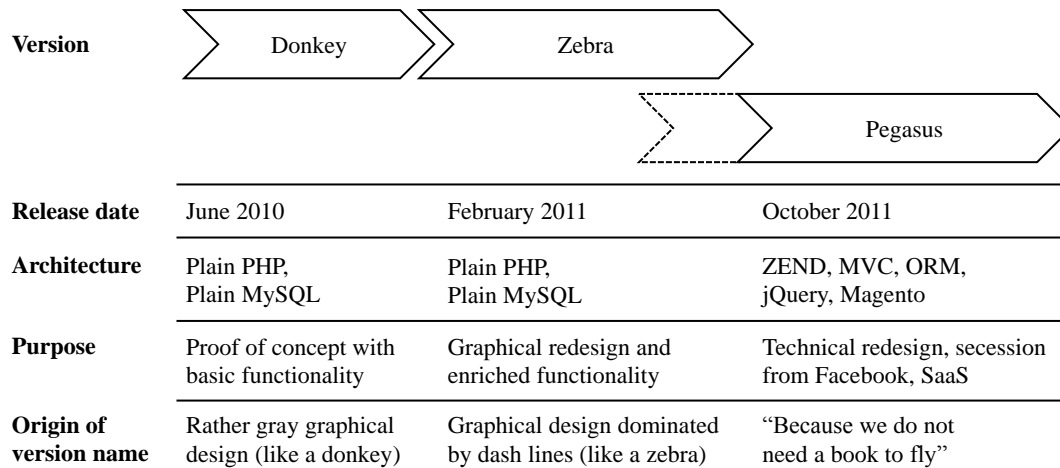


Figure 3: Major versions of unseerAller

The technical subsystem is designed to complement the purpose of the different versions. This means that Donkey and Zebra are designed in a very lean way with plain PHP and MySQL. This enables flexibility and quick responses to the still volatile social subsystem. The integration with Facebook allows a quick and easy access to user authentication and start of the community. Once the social subsystem gets permanent in Zebra, the architecture is adapted to the more stable requirements: Pegasus is redesigned from scratch to complement the current social and technical requirements while building upon state-of-the-art technical concepts. It therefore includes ZEND as a framework with a model view controller (MVC) architecture in PHP, Doctrine2 as an object relational mapper (ORM) to design the data structure and jQuery for the user interface. Additionally, Magento is selected as a shop system as it is based on the ZEND framework as well. Tracking is realized by Piwik. Also the close link with Facebook is dissolved and replaced by an own user management. All technical components are thus harmonized.

„We now know what the product [unserAller] will look like and have the vision how it shall be and will be. We did not have that for the first version. [...] At the beginning of the year, when large clients approached us, scalability became a subject [...] and that is why a complete redesign from scratch was justifiable anyway.” (CTO)

Build

Three developers of innosabi work full time on the development of unserAller. Outsourcing is out of scope, as innosabi sees itself as a technology provider. They consequently want to keep their core competency in-house.

„We actually have a high technological competency in our team. We can quickly realize many things on ourselves and iterate and try out things and see what works best. That is because we do not always have to ask an agency to program it for us. [...] That was actually key to bring unserAller to this technical realization which it is today.” (CEO)

In the early phases, they coordinate their work using a SCRUM-like technique with Post-Its (Figure 4).

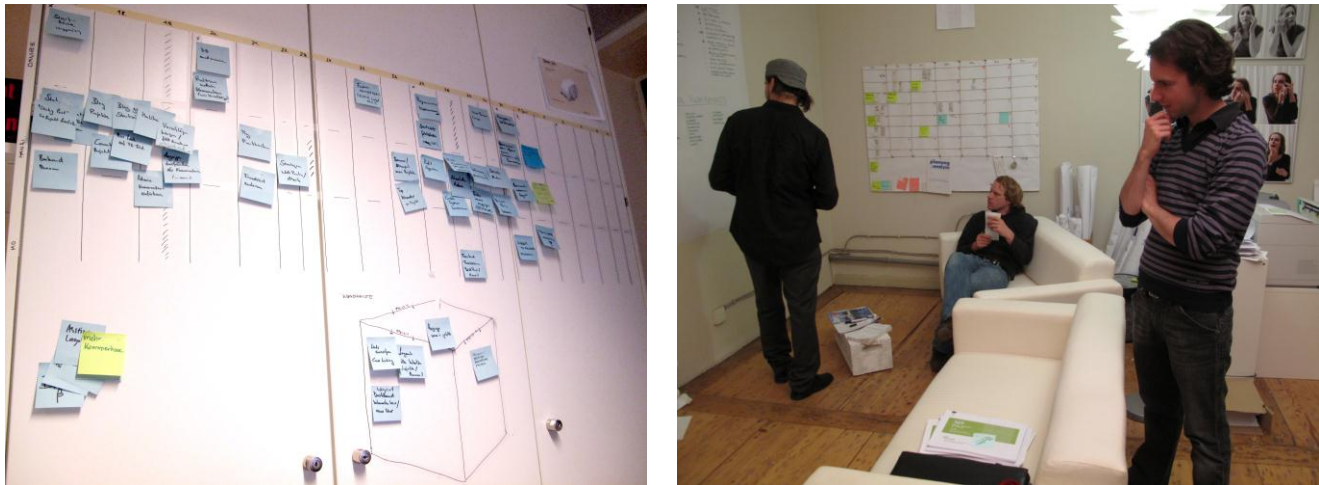


Figure 4: SCRUM-like coordination of tasks at innosabi

Tasks are mainly assigned by competence: one developer takes care of the interface while the other one takes care of the database and program logic. The codebase is stored in a repository (i.e. Beanstalk) that is linked to a development and live system. Central functionality is developed using a 4-eyes principle. Ideas are implemented as rough prototypes prior to the final implementation. All functionality of unserAller is implemented as modules. Thereby, usage of existing open source components and reusability is a key concept. Quality is assured by intensive testing especially with respect to cross browser compatibility. Therefore, innosabi does not merely rely on internals, but also on external beta testers from the community.

„We have a list of 15 beta testers. Those are mainly companies which are completely different. We have software developers that want to test the interfaces, caterers, small producers, large producers and also community members.” (CEO)

Reported bugs are often solved within minutes. This also applies for small feature requests. Bigger ones are prioritized with the team.

Deploy

Due to the managed code repository, the technical deployment of unserAller can be done from Beanstalk, as no deployment to foreign systems is required. This is also required, as smaller changes are published immediately in a perpetual beta paradigm. Next to these small updates, bigger updates (like for instance the introduction of the online shop) are deployed according to a milestone plan. Only these bigger updates are communicated to the community. Smaller changes occur in a silent rollout in order to reduce spam to the community.

The initial deployment of unserAller took place at a launch party with stakeholders. Prior to the launch, the first project is set up. It is occupied during the launch party in order to have content when the online community starts to work on the project.

Marketing for userAller is carried out via Facebook. innosabi thereby uses its client’s Facebook communities to address potential innovators.

„We were able to build the community really fast without spending a single Euro for advertising. This was actually supported by the viral effect on Facebook.” (CEO)

Besides the rollout to the community, innosabi takes care of a proper integration of the community’s results into its client’s organization.

Operate

A proper management of the userAller community is a core value proposition of innosabi. In order to run innovation projects within the community, innosabi employs dedicated community managers that ensure ten hours per day, seven days a week community management. Each contribution by the community is reviewed and modified or deleted if necessary. Thereby the innosabi team ensures compliance with the community rules and copyright restrictions. The review is supported by innosabi’s backend that for instance lists open review tasks and identifies duplicate ideas and proposes a merge of them (Figure 5).

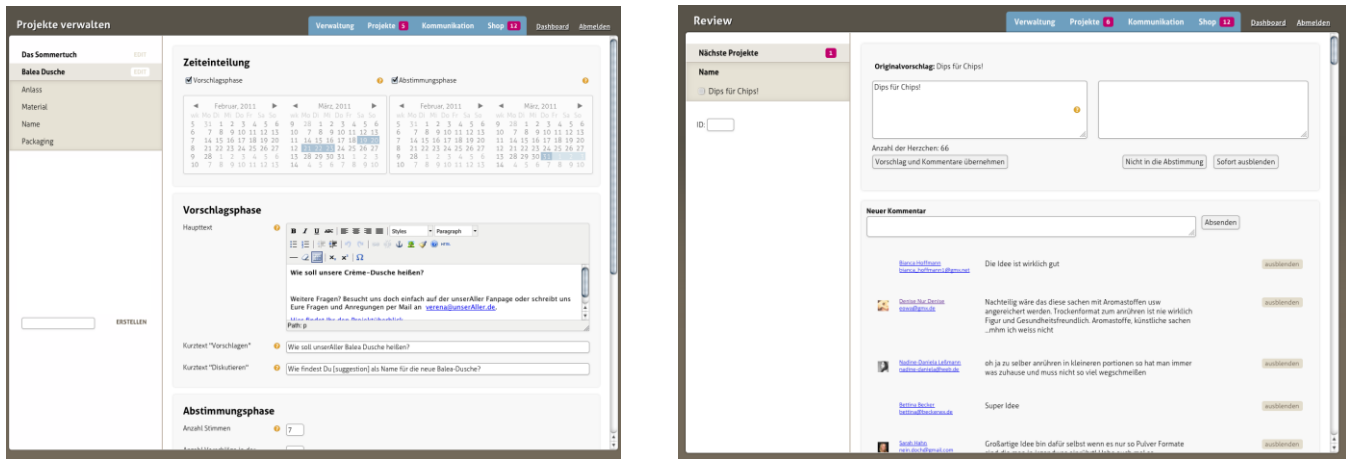


Figure 5: Community management backend of userAller (version Zebra)

This supportive backend is required in order to deal with up to 2,300 suggestions and 6,600 comments per project.

„We read every contribution of the community, review it and answer every single question. If we have 450 design proposals and 2,500 comments like in the [name of project] we have to approve all of them. This is quite a big effort, but that guarantees us that we can quickly react on questions, that we can avoid disputes and that the community feels taken seriously.” (CEO)

Next to the community management, innosabi informs its client with reports of each phase’s result and the overall outcome of an innovation project.

Optimize

As already outlined in the design phase, innosabi continuously optimizes userAller. This does not only apply to bugs but also to new features. Requests thereby origin from the community as well as from existing or potential clients of innosabi. Though having standardized feedback functionality, most feedback is given personally. This is a result of the close link to the community. To develop new functionality, innosabi starts with a single client and transfers the outcome to other clients.

„[The client] tells us which functionalities he wants and how he imagines the project. Then we build in the way that it is perfect for him and hope that it is transferrable to our other clients.” (CEO)

innosabi runs a trial-and-error approach with multiple iterations to develop new features. They are especially striving to integrate features that can enhance their business model or foster community activity. Anyhow, they try to not change basic functionality in order to not confuse the community:

„We try to not break functionality, which is of course not always possible. [...] We try to silently introduce new functionalities for our project partners. Now and then there is a new button and you can try it.” (CIO)

Besides feedback from externals, innosabi also uses tracking analyses to identify problems and evaluate features.

„We do not always activate the heat map [of Piwik]. We only activate it if we integrate something new in order to evaluate it.” (CTO)

To sum it up, userAller thereby evolves in small steps and some big leaps. The next big step of userAller is the introduction of a mobile version that integrates the online and offline world.

„That is the model behind the next version of userAller. From a development perspective, it evolved step by step with the projects we ran. We had a vision of a platform in mind right from the outset. Nevertheless we said that the vision is too big for a first step and that we need to gain experience.” (CEO)

DISCUSSION & CONCLUSION

In order to answer our research question on *how to structure the design and management of a web-based innovation community using application lifecycle management*, we applied the application lifecycle management framework to the field of innovation communities. We presented a case study of innosabi and its approach to design and management their innovation community userAller. Application lifecycle management with its six stages has been shown to be suitable to structure the activities during the design and management of this community, as all aspects elicited in the case study are represented in the framework. However, going beyond a traditional software project, which is at the heart of ALM, design and management of an innovation community is more challenging and requires additional activities.

The most important finding is that the community manager has to hold particular competencies in the field of open innovation and innovation communities. As suggested by Pasmore et al. (1982), our data shows that technical knowledge to design the community platform has to be accompanied with knowledge on the social subsystem of innovation communities. Our case company builds up this expertise with prior knowledge and trial-and-error approaches in pilot projects. While striving for a solid technical subsystem as an enabler that reduces barriers of participation and enables scalability, innosabi focuses on the management of the community which they see as the crucial activity in an innovation community's lifecycle. This result corresponds with the findings of Adamczyk, Boehler, Bullinger and Moeslein (2011). In order to accomplish community management, innosabi's core asset is a specialized backend that supports the management tasks. This is an interesting finding, as future research should investigate systems that facilitate the management of communities, which is a so far neglected aspect.

According to the described social focus in the community lifecycle management activities, the focus lies on the following phases of the application lifecycle management framework: Much effort is spent to assess, design, evaluate and manage the social subsystem of an innovation community in order to foster both the voluntary contributions of the community and a viral effect. Accordingly the phases *design*, *operate* and *optimize* are crucial for an innovation community's success and should thus be emphasized. Additionally the activities in the deployment phase shift from technical deployment towards marketing activities to recruit potential innovators.

The results of our case study have to be evaluated against its shortcomings. We present findings from a single case that is limited to a community that builds upon a playful experience of the community. Results might differ among organizations and cultures. Maybe other organizers realize different approaches to design and manage innovation communities. Also the type of community (Hallerstede and Bullinger, 2010) might influence an appropriate lifecycle management.

Future development of userAller should be an interesting trajectory: while the case illustrates that AML can be used to structure the design and management of innovation communities, it stresses in particular the social aspect. The introduction of a Software-as-a-Service-version of the userAller platform is of special research interest. innosabi's clients will be able to configure and manage their own projects. This will offer the unique opportunity to investigate the influence of the organizer's experience on the community management.

ACKNOWLEDGEMENTS

We thank all open innovators who are part of our ongoing innovation research journey and gratefully acknowledge support by the German Federal Ministry of Education and Research (project: BALANCE, FKZ 01FH09153).

REFERENCES

1. Adamczyk, S., Boehler, D., Bullinger, A. C. and Moeslein, K. M. (2011) Facilitating interaction in web-based communities: the case of a community for innovation in healthcare, in H.-U. Heiss, P. Pepper, H. Schlingloff, and J. Schneider (Eds.) *Informatik 2011. Informatik schafft Communities. Lecture Notes in Informatics (LNI) - Proceedings, Series of the Gesellschaft fuer Informatik*, , 228.
2. Armstrong, A. and Hagel III, J. (1996) The Real Value of on-line Communities, *Harvard Business Review*, 3, 134-141.
3. BMWi. (2011) BMWi-Pressemitteilungen: Rösler prämiert erfolgreichste IKT-Unternehmensgründungen, Retrieved January 14, 2012, from <http://www.bmwi.de/BMWi/Navigation/Presse/pressemitteilungen,did=406252.html>.
4. Bateman, P. J., Gray, P. H. and Butler, B. S. (2010) The Impact of Community Commitment on Participation in Online Communities, *Information Systems Research*, 1-16.
5. Bullinger, A. C., Neyer, A.-K., Rass, M. and Moeslein, K. M. (2010) Community-based innovation contests: Where cooperation meets competition, *Creativity & Innovation Management*, 19, 3, 290-303.
6. Butler, B., Sproull, L., Kiesler, S. and Kraut, R. (2007) Community effort in online groups: Who does the work and why?, *Human-Computer Interaction Institute*, 90.
7. Chesbrough, H. W. (2006) *Open business models: how to thrive in the new innovation landscape*, McGraw-Hill Professional.
8. Christensen, C. M. (2006) *The innovator's dilemma*, Harper Business, New York, NY.
9. Ebner, W., Leimeister, J. M. and Krcmar, H. (2010) Community Engineering for Innovations: The Ideas Competition as a method to nurture a Virtual Community for Innovations, *R&D Management*, 40, 4, 342-356.
10. Flanagan, J. C. (1954) The critical incident technique, *Psychological bulletin*, 51, 4, 327-358.
11. Füller, J., Bartl, M., Ernst, H. and Mühlbacher, H. (2004) Community based innovation: a method to utilize the innovative potential of online communities, in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, IEEE, Hawaii.
12. Füller, J., Bartl, M., Ernst, H. and Mühlbacher, H. (2006) Community based innovation: How to integrate members of virtual communities into new product development, *Electronic Commerce Research*, 6, 1, 57-73.
13. Füller, J., Matzler, K. and Hoppe, M. (2008) Brand community members as a source of innovation, *Journal of Product Innovation Management*, 25, 6, 608-619.
14. Hallerstede, S. H. and Bullinger, A. C. (2010) Do you know where you go? A taxonomy of online innovation contests, in *Proceedings of the XXIth ISPIM Conference*, Bilbao.
15. Hallerstede, S. H., Bullinger, A. C. and Möslein, K. M. (2012) Community-basierte Open Innovation von der Suche bis zur Implementierung – der Fall des Innovationsintermediärs innosabi, in *Multikonferenz Wirtschaftsinformatik 2012 (MKWI)*, Braunschweig.
16. Harhoff, D. (2003) Profiting from voluntary information spillovers: how users benefit by freely revealing their innovations, *Research Policy*, 32, 10, 1753-1769.
17. von Hippel, E. (2005) *Democratizing Innovation*, MIT Press, Cambridge MA.
18. Hoffmann, M. and Beaumont, T. (1997) *Application Development: Managing the Project Life Cycle*, Mc Press.
19. Kozinets, R. V. (1999) E-Tribalized Marketing? The Strategic Implications of Virtual Communities of Consumption, *European Management Journal*, 17, 3, 252-264.
20. Leimeister, J. M., Huber, M., Bretschneider, U. and Krcmar, H. (2009) Leveraging Crowdsourcing: Activation-Supporting Components for IT-Based Ideas Competition, *Journal of Management Information Systems*, 26, 1, 197-224.
21. Lindsay, V. J. (2004) Computer-assisted qualitative data analysis: application in an export study, in *Handbook of qualitative research methods for international business*, Edward Elgar, Cheltham, 486-506.

22. Lynn, L., Aram, J. D. and Reddy, N. M. (1997) Technology communities and innovation communities, *Journal of Engineering & Technology Management*, 14, 129-145.
23. Muniz Jr., A. M. and O'Guinn, T. C. (2001) Brand Communities, *Journal of Consumer Research*, 27, 412-432.
24. Neyer, A.-K., Bullinger, A. C. and Möslin, K. M. (2009) Integrating inside and outside innovators: a sociotechnical systems perspective, *R&D Management*, 39, 4, 410-419.
25. OGC. (2002) *Application management*, The Stationery Office, Norwich.
26. Pasmore, W., Francis, C., Haldeman, J. and Shani, A. (1982) Sociotechnical Systems: A North American Reflection on Empirical Studies of the Seventies, *Human Relations*, 35, 12, 1179-1204.
27. Piller, F. T. and Walcher, D. (2006) Toolkits for idea competitions: a novel method to integrate users in new product development, *R&D Management*, 36, 3, 307-318.
28. Rheingold, H. (2000) *The virtual community: Homesteading on the electronic frontier*, The MIT Press.
29. Runeson, P. and Höst, M. (2008) Guidelines for conducting and reporting case study research in software engineering, *Empirical Software Engineering*, 14, 2, 131-164.
30. Schnell, R., Hill, P. B. and Esser, E. (2008) *Methoden der empirischen Sozialforschung*, Oldenbourg Wissenschaftsverlag, München.
31. Sommerville, I. (2007) *Software engineering*, Addison-Wesley, Harlow.
32. Strauss, A. L. and Corbin, J. M. (1998) *Basics of qualitative research: techniques and procedures for developing grounded theory*, Sage Publications, Thousand Oaks.
33. Trist, E. L. and Bamforth, K. W. (1951) Some Social and Psychological Consequences of the Longxwall Method of Coal-Getting: An Examination of the Psychological Situation and Defences of a Work Group in Relation to the Social Structure and Technological Content of the Work System, *Human Relations*, 4, 1, 3-38.
34. Williams, R. L. (2000) Four Smart Ways To Run Online Communities, *Sloan Management Review*, 4, 81-91.
35. Yin, R. K. (2003) *Case Study Research: Design and Methods*, Sage Publications, Thousand Oaks.