Association for Information Systems
AIS Electronic Library (AISeL)

BLED 2011 Proceedings

BLED Proceedings

2011

Participatory Design of Web 2.0 Applications in SME Networks

Martina Peris
Hamburg University, Germany

Alexander Sperling
Hamburg University, Germany

Nadine Blinn
Hamburg University, Germany

Markus Nüttgens
Hamburg University, Germany

Nick Gehrke
NORDAKADEMIE Elmshorn, Germany

Follow this and additional works at: http://aisel.aisnet.org/bled2011

Participatory Design of Web 2.0 Applications in SME Networks

Martina Peris, Alexander Sperling, Nadine Blinn, Markus Nüttgens

Hamburg University, Germany

Nick Gehrke

NORDAKADEMIE Elmshorn, Germany

Abstract

In increasingly complex and dynamic markets, small and medium sized enterprises (SME) face new challenges. Amongst others, these are innovativeness and technological expertise. In order to counteract the challenges, SMEs cooperate in corporate networks. Here, information and communication technologies are main drivers. At this point, Web 2.0 technologies are uttermost important. Until now, the development and imple- mentation of Web 2.0 applications in SMEs was proceeded independently from the future users. We aim at bridging this gap by developing a participatory procedural model. The presented model includes the futures users from the beginning of the development process. The model respects SME specific characteristics.

Keywords: Web 2.0, SME, Social Network, Procedural Model, Participatory Design

1. Introduction

Due to growing competition, small and medium sized enterprises (SME) are confronted with increasingly complex challenges. In order to counteract the challenges, SME have to develop innovative information technology (IT)-based solutions. With respect to their restricted resources and limited capacity for innovation, SME cooperate in networks (Street, C. T. & Cameron A. F. 2007). Web 2.0 applications are seen as adequate tools for SMEs in order to increase productivity as well as proximity to the market (De Saulles, M. 2008) (Wyllie, D. 2008). We present a procedural model in order to support the development and integration of a Web 2.0 application in SME networks. In this context, participatory design is an adequate approach to involve the prospective users in the software development process. The paper focuses on the development and presentation of a procedural model, which allows the software developers and future users to contribute equally and in conjunction to the software development process. The

benefits of the presented model are the practice-oriented collection and incremental implementation of multidimensional requirements in order to achieve an improved integration into the application context.

The remainder is structured as follows: In section two, we present basic definitions and concepts. Subsequently, existing procedural models are analysed with a focus on the given research problem. Based on the results, the procedural model and a corresponding role model with respect to SMEs' characteristics are developed.

2. Basic definitions and concepts

2.1 SME and corporate networks

The European Commission defines small and medium sized enterprises (SME) according to three criteria. Hence, the following enterprises belong to the class of SME: enterprises with less than 250 employees and an annual turnover of less than 50 million € or an annual balance with less than 43 million € (Europäische Gemeinschaften 2006).

Since SMEs represent 99 % of all European Enterprises and provide 75 million jobs, they are of high social and economic importance within Europe (Europäische Gemeinschaften 2006). Due to their size and structure, SMEs are relevant sources for innovations. The company management of SMEs is usually „flat", that means a low percentage of executive or leading managers. The slight distance between the company management and employees supports a high identification with the enterprise and an increased working motivation. Mostly, SMEs are family businesses, i.e. the property rights as well as the managements rights, are united within the person of the entrepreneur (Institut für Mittelstandsforschung 2010). This fact influences the quality of decisions. As the enterprise is led in autocratic way, decisions are often taken without employee involvement. Delegation scope is rarely used and many decisions are misinterpreted as leading decisions (Sattes, I. & Conrad, H. 1998). Strategic or corporate-policy aspects are rarely or not at all considered, as decisions are often intuitively taken. In sum, the success of an enterprise is strongly related with the personality of the owner and business manager. Nevertheless, for the enterprise's success, further factors play a role. Besides the corporate structure, most notably cooperation is a mentioned factor. In the face of increasing competition due to big enterprises and globalised markets in the environment of a SME, corporate networks enable SMEs to bear up in the markets. Corporate networks are organisational forms with economic activities, that act formal independent but economically more or less dependent enterprises, who are connected through complex-reciprocal, rather cooperative than competing and relatively robust relations (Sydow, J. 1992). Through cooperation, a collective increase of efficiency and the strengthening of the individual competitive situation are expected.

2.2 Web 2.0 and Enterprise 2.0

The term „Web 2.0" represents the second generation of World Wide Web (WWW), that differs from the passive consuming attitude of the Internet. The term was coined by Tim O'Reilly in 2004. Web 2.0 is hence a development stage of the Internet that represents a „business revolution in the computer industry caused by the move to the

internet as platform" (O'Reilly, T. 2006). The contents are not read passively consumed, in fact they support an active communication and participation in the Internet in order to exchange and commonly develop ideas (McAfee, A.P. 2006) (O'Reilly, T. 2005). Web 2.0 applications support collective intelligence, whereas network effects by user participation represent the key to success (O'Reilly, T. 2005).

The implementation and application of Web 2.0 technologies into a corporate context is called "Enterprise 2.0". Enterprise 2.0 refers to the intra-corporate or extra-corporate usage of Web 2.0 technologies in order to support the exchange with business partners and customers (McAfee, A.P. 2006). The core represents new forms of web 2.0 supported collaboration that enables employees to participate in intra-corporate and extra-corporate processes. Besides the technical aspects, Enterprise 2.0 relies on the principle of self-organisation, that calls for a reduction of hierarchies in order to establish creativity and innovation (Stamer, S. & Baier, T. 2008). Within SMEs, a trend towards an intensive Internet usage is in evidence. The Internet is primarily used in order to support communication by e-mail or information research. In the field of sales and customer services, web-based applications are increasingly used, but the application of Web 2.0 technologies remains an exception. Only few entrepreneurs believe in the influence of Web 2.0 towards their business. Also at this point, the potential of Web 2.0 technologies are not recognised. In fact a majority of SMEs refers Web 2.0 to risks like data misuse or damage of reputation through manipulated information. Not least for this reason, the intra-corporate implementation of Web 2.0 technologies is considered sceptical (Kautz, K. 2010). In this context it becomes evident, that the challenges for a successful implementation and usage are not only technically caused, but rather influenced by organisation and corporate culture (Raabe, A. 2007).

2.3 Participatory Design

Participatory Design is an approach for the assessment, design and development of technical and social systems, that focuses on the active inclusion of the working practice in the context of design and decision processes (CPSR 2005) (Kautz, K. 2010). This refers to the principle of giving future users the right to a say in a matter in the whole software development process (CPSR 2005).

The implementation of the core idea of Participatory Design – positioning the user to the centre of development – can be interpreted in manifold manners or instanced in different ways. Participatory Design is not a new method or a new model, but a new point of view or dimension in order to extend software development (Kautz, K. 2010).

The main goal is to include future users in the whole development process in order to avoid undesirable developments at early stages. Failures occurring at later stages can be expensive in time and money or not even revisable. Moreover, a holistic integration supports an increased acceptance for the application in use. In order to realise a holistic integration, proper procedural models are demanded.

3. Suitability of existing software engineering models

Due to its complexity and the people involved software engineering is a process that has to be planned and must be structured according to systematic criteria. This process is embedded in a software life cycle and describes the process for building this software.

In general a procedural model is used to describe the pattern of the development process based on a development scheme that summarises the ideas and models for action strategies instructions and descriptions (Wolf, M. & Thränert, M. 2007).

Existing software engineering models have specific strengths and weaknesses in terms of user participation. To get an integrated evaluation they should be assessed on various aspects related to the suitability within the present context. This concerns in particular the respect of the use in small and medium-sized businesses and their associations to networks. Therefore, criteria must be found that allow designing a meaningful comparableness of existing models. Four criteria are relevant in this application context: communication, extensibility, flexibility and specific SME features.

During the whole development process a clear and effective *communication* between users and developers has to be warranted. This does not only reduce errors in coding based on communication errors bonded with higher costs of correction, but also increases the acceptance of the application to be introduced in the work context. The aspect of *extensibility* is used for rapid development of directly executable systems that can be further developed. This allows the contemporary introduction into the application context to early identify problems or new requirements together with the user. *Flexibility* concerns in particular the responses to changing requirements in the application context that are not understood as a problem but as part of the development process. The attention of *specific SME features* is essential, as these are crucial to the success of the implementation. This applies to limited budgets, limited staff resources, patriarchal forms of governance and heterogeneous IT skills. In the implementation of IT projects especially SMEs dispose limited to no experiences on which they can access
.

The phase models, with the *waterfall model* as a representative (Himmelreich, J. 2006), don't offer support for communication between users and developers during the development process. Given that, due to the linear hierarchy each phase has to end with a clear result before the next phase starts sequentially, all requirements must be included in the phase of analysis. Because of potential end users cannot be assumed that expert knowledge of the methods is present, a clean requirements analysis with regard to different levels of abstraction seems to be difficult. The aspect of extensibility is not observed, because an incremental development and early use are not provided. Likewise, the waterfall model is not flexible enough to respond to the resulting problems.

STEPS represents evolutionary procedural models characterised by a strong focus on participatory development application. The cyclical development that is characterised by stages of production and use doesn't need a complete requirements analysis done with regard to functions to be implemented (Floyd, C. 1984). Building on experiences from the use new requirements can be formulated and implemented in a subsequent cycle. On this way communication problems between user and developer can be reduced. In each cycle, it is possible to react to changing conditions or requirements. Those changes cause the parallel adjustment of the supplied documentation. It should be noted that this process takes time and financial resources. Additional, this model binds human resources from the application context. In particular, this aspect can be described

as negative, since this resource is shorthanded in SMEs. Overall, STEPS useful clues are predestined to be kept in mind for the participatory model. Nevertheless in its whole, this model is overweight. Especially the binding of human resources is in SME projects an exclusion criterion.

Feature Driven Development as a procedural model, which implements the principles and methods of Extreme Programming Modelling (Yakut, Y. 2008). This lightweight and flexible approach is particularly suitable for the development of application software in smaller projects. The constant communication between users and developers as an elementary principle of agile software development is an essential aspect of the quality of the application. The early delivery of software components and incremental development characterise the extensibility and flexibility of the model. The avoidance of documentation tasks quickly leads to a ready application, saving time and financial resources.

4. Role and procedural models in Participatory Design of Web 2.0 applications in SME networks

4.1 Requirements and Characteristics

As the assessment of existing procedural models already pointed out, flexible organised software development processes are proper for manageable software developments as the development of Web 2.0 applications in SMEs. Procedural models of agile software development are hence predestined as they offer sufficient space for individual developments. In order to point out, why the decision for a certain tool was taken, the goals for the procedural model to be developed are concentrated as follows. Against the backdrop of a future application within SME corporate networks, these are:

- The procedural model should be designed for all development stages *transparent* for all stakeholders. This includes not only tracking of the project advance, but the continuous inclusion of requirement analyses and decision processes. This leads to trust in the application to be developed and reduces misunderstandings.
- The procedural model should be designed *flexible,* in order to be able to react on changing requirements and to implement them in the current development. In this context, it should be possible to formulate the requirements in an abstract manner. This allows focusing on a common exchange rather than focusing on the technical implementation. This advances a professional understanding of the application on both the developer and the customer side and supports the identification of important functionalities.
- The procedural model should lead *early* to executable software artifacts, that can be further developed in an *incremental* manner. This possibility of early practical tests leads to a fast user feedback. On the one hand, this eases the work of the developer as he can focus on core functionalities. On the other hand, the user gets impressions of the application skill, so he can develop realistic expectations towards the application. A fast availability of an application that inherent specified core functionalities, lead to flexible decision processes concerning the ending of the development process through the user. Due to limited money and times resources, he decides whether to extend the application

with less relevant requirements or to end the development process. The results of the development process are always executable applications that only differ in their functionalities depending on the development stage.

- The permanent *communication* between developer and user belongs to the core principles of agile software development and is essential for the procedural model to be developed. In order to support this, different communication channels must be made available. Generally, the personal communication should always be preferred. In practice, this is not always realisable. Hence, communication methods must be available that are always present and are transparent for all stakeholders.

- A major point of critique towards agile software development is the rudimental implementation of the *documentation* into the procedural model. Particularly in relation to the requirements analysis with focus on positive or negative decision towards the implementation of functionalities must be fixed in a written way. Also at this point, an integrated communication can provide support.

4.2 Role model

In the context of software development, a role model assigns tasks and responsibilities to all participants. The usage of a role model increases the degrees of freedom and supports the collaboration, as the boarders are fluently defined. Besides a good communication, the flexibility of the team members concerning their tasks and responsibilities are key success factors (Hanser, E. 2005).

The role model to be developed should be oriented at agile model's procedural structure. As a component based development is aspired, the core programming effort is reduced to a minimum. Hence, the role model to be developed must respect the modified conditions. It is reasonable to keep the group of people responsible for the development of the application to a minimum. In this manner, a differentiation of the roles independently from the technical implementation is feasible. The role model should define responsibilities in a higher level and hence respect the heterogeneous composition of the different participants with their differing expertise. In this context, the paradigm of agile software development is to be mentioned, according to which all participants are equal and their opinion is equal to the opinion of other participants. Hence, it is possible to consider all forming, whether they focus on technical or application-specific aspects, in a common discussion.

4.2.1 Stakeholder

The stakeholders are the representatives of the SME corporate network. These experts for the application context are mostly the business managers. At this point, it is to be kept in mind, that the business managers are experts for the application context without a doubt, but caused by their personality structure and the differing business areas, they are influenced by different motivations. Because of this fact, it is reasonable to identify so called value-stakeholders that – as deputy for the other stakeholders – are stronger involved in the process. One selection criterion, besides the interest in participating, is the temporal availability. According to their "pioneer role" important tasks are the requirements definition, the validation of the different prototype development stages and the final acceptance, in order to implement the platform in the application context.

4.2.2 Developer

The developer is responsible for the technical concept and documentation of the application. He should inherent knowledge in the area of web-based platform development, especially in component-based prototyping. Moreover, it is reasonable that the developer possesses a certain degree of basic knowledge of the application context in order to be available as a critical discussion partner during the requirements analysis. In this context, he analysis cyclical the requirements together with the user and discusses them in the context of feasibility. As a result, the developer designs incremental, executable prototypes that provide a base for further analysis and finally result in the Web 2.0 application. Besides the examination of faults and change requests, he should support the users with the handling of the application.

4.2.3 Project manager

The project manager is responsible for the compliance with the project accompanying constraints. He supervises the application development concerning the compliance of time and budget. Furthermore, he organises meetings for further requirements analysis and protocols the decisions and requirements in order to keep the platform specification up to date. Moreover, the project manager organises and executes workshops. Hence, the project manager should possess basic knowledge in this field.

4.2.4 Key User

The key user is a representative of the future users. He is the conjunction between implementation context and application development. Besides the participation on meetings for requirements analysis, his main task is the support of the project manager. The key user as the direct actor in the application context has a closer link to the other users than the project manager. The main goal of the user involvement is primarily the intervention of media competencies. Hence, he becomes the contact person concerning questions on the application. This is important, as he should be able to administrate the application after the end of the development project (Zorn, I., Büschenfeldt, M. & Schelhowe, H.).

4.3 Procedural model

In order to face potential resistance during the development and implementation of Web 2.0 applications in SME networks, a lightweight procedural model is developed, which is close to organizational features without extensively binding human resources. For this purpose, the focus is on communication and participation. Figure 1 shows the model.
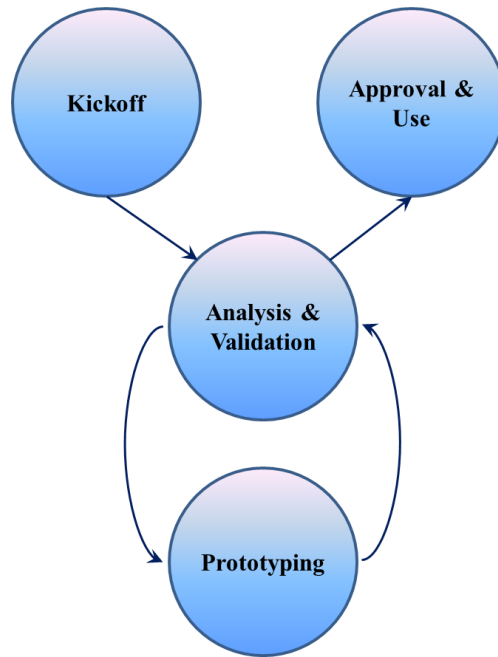
Figure 1: Procedural model for participatory designing Web 2.0 applications in SME networks

4.3.1 Kickoff

The aim of the kickoff is the introduction of all participants in the project and methods used to develop the Web 2.0 applications. It should be stressed that participation and communication play a key role in the procedural model and thus stand out from traditional methods. The kickoff should take place promptly after starting the project as a meeting of all involved persons. The Project Manager organizes and implements the kickoff. The kickoff includes in particular the placement of the objectives pursued by the applications to be developed and the resulting benefits for all SME representatives. In particular, the differences of Web 2.0 applications to conventional applications (creation of infrastructure rather than content) and the need to dispense with external business-related hierarchical structures have to be shown. The promise of an iterative development should increase the interests of stakeholders in order to reduce barriers to active participation. This particularly concerns the motivation for an intensive communication with the developers, since an efficient application development is guaranteed. Result of this meeting is a project plan that captures the various stages of development. Furthermore, the key user should be defined after his role was explained. All parties have an idea of their role in the project and the resulting responsibilities.

4.3.2 Analysis and Validation

The aim of this step organised by the project manager is the discussion of the requirements involving all stakeholders in a workshop. The analysis and validation of requirements is the first step in an iterative cycle that leads by repeated passages through an evolutionary application development, prototypes will be expanded incrementally on the basis of new requirements. All stakeholders are enabled to directly formulate new requirements. The form of requirements analysis can be flexible, in which in addition to interviewing the method of the management game seems to be applicable (Rittenbruch, M., McEwan, G. et al 2002). It focuses on so-called user stories

that describe a requirement or a work situation using with some sentences and from which functionalities can be derived. These user stories are informally held on story cards, without regard to technical implementation. Then stakeholders and developers prioritise them according to various criteria. Stakeholders and key users differentiate the requirements of their value to the application context into:

- critical to the running of the application,

- less important but of good value and

- nice to have.

The developer arranges them according to the underlying risks into:

- requirements, whose timetable for its implementation can be determined and provided,

- requirements, whose timetable for its implementation can be estimated reliably, and

- requirements which cannot be estimated over time.

Based on this prioritization, requirements are collected that are important and implementable in time. In collaboration stakeholders and developers discuss in particular controversial requirements, to either reject them or to work out an alternative solution. Due to the iterative requirements analysis and its implementation into executable prototypes the stakeholders get a better idea about the limits and possibilities of development.

Result of this analysis may be the selection of new or changing requirements and functionalities to be implemented in the next iteration. They are hold in an iteration plan and build the basis of the developers work in the subsequent phase. Furthermore, this must be documented in a list of requirements. If the participants come to the conclusion that the prototype fully meets all the requirements and can be productively used in the application context, the final approval takes place.

4.3.3 Prototyping

Based on the iteration plan the collected requirements are implemented in the responsibility of the developer supported by the key user. The separation of stakeholders and key users reduces the staffing resources without constraining the benefit of participatory development limit. Using an established communication channel, the possibility of continuous feedback is given so that the key user still can take direct influence on development. This is possible because of the client-server architecture of the Web 2.0 application allows the key user having access to a recent release of the prototype independently of location and time.

The developer initially models roughly the requirements to limit the solution space. In a next step, the functionality is integrated into the prototype. As soon as an executable version exists, the developer releases it as preliminary release. This version is the basis for the feedback of the key user. Once the Web 2.0 application has a certain level of maturity and has similarities to a productive environment, it is treated to an extended

circle of users that switches over to work with real information and thereby to test more realistic. This increases not only the commitment but also the quality of feedback regarding identified problems and the formulation of new requirements (Zorn, I., Büschenfeldt, M. & Schelhowe, H. 2008).

The feedback can be divided in bugs, handling problems and additional requirements. The key user supports the developer in processing, sorting and aggregating the given feedback. The processed bugs and handling problems are cleaned up in the same iteration. Additional requirements are collected and will be integrated into the analysis and validation phase of the next iteration.

The result of this phase is a stable release prototype that meets the requirements in the iteration plan completely through implementing the appropriate functionality. This forms the basis for the analysis and validation of a further iteration of the development cycle.

4.3.4. Approval & Use

After assessing that the prototype was fully implemented with regard to the requirements and objectives the approval follows. This takes place in a separate meeting organised by the project manager. The purpose of this meeting is to official handover the Web 2.0 application and motivating to use it. Therefore, it is advantageous if the participants are not the same as those who were involved into the development process. There should also be other future users from the application context. Using a role-playing, the concepts and methods of the developed Web 2.0 applications can be demonstrated. It is crucial to convey that the use of the platform enables an advanced form of collaborative cooperation that allows achieving specified objectives. A special role is given to the key user, who deals with the administration of the application and who has to be presented separately. Furthermore, the project-related documentation with information on technical and functional specifications can be handed over. This will facilitate the incorporation, if the application is part of a further developed in a new project.

5. Summary

The objective of this paper was to present a participatory procedural model which points out the user-centered development of Web 2.0 applications in the context of SME networks. The integration of Web 2.0 applications into the context of SME networks has the potential to increase competitiveness. The benefit comes out in the collaborative cooperation, using the Web 2.0 platform as the infrastructure. Currently, however, a more sceptical attitude towards this new form of cooperation can be observed especially in SMEs, as this means major changes at the organizational, human and cultural level. Therefore, a procedural model was developed that can contribute to facilitating integration. The participatory approach can enhance the acceptance of the application and thus contribute to the success significantly. The characteristics of SMEs play a decisive role. Generic procedural models with classical properties show problems in the implementation when observing in the context of SME networks.

In summary, this paper shows that for successful development and launch of a Web 2.0 application in SME networks, a participatory approach generally can be more promising

than the use of classical models, because the consistent focus to the needs of the application context performs more practical solutions. The model has been approved in a single case study (www.kmu20.net).

References
- **Books**

De Saulles, M. (2008). SMEs and the Web - Executive Summary. University of Brighton.

European Community. (2006). Die neue KMU-Definition, Benutzerhandbuch und Mustererklärung.

Floyd, C. (1984). A systematic look at prototyping. In Budde, R., Kuhlenkamp, K., Matthiassen, L. & Züllighoven, H. (Eds.), Approaches to prototyping, (1-18), New York: Springer.

Hanser, E. (2005). Rollenveränderung in der Agilen Software-Entwicklung. In Georg Herzwurm, Martin Mikusz (Eds.), Industrialisierung des Software-Managements. GI Lecture Notes in Informatics.

Raabe, A. (2007). Social Software im Unternehmen: Wikis und Weblogs für Wissensmanagement und Kommunikation. Saarbrücken: VDM Verlag Dr. Müller.

Sattes, I. & Conrad, H. (1998). Erfolg in kleinen und mittleren Unternehmen: Ein Leitfaden für die Führung und Organisation in KMU. Zürich: vdf Hochschulverlag AG.

Sydow, J. (1992). Strategische Netzwerke. Wiesbaden: Gabler.

Wolf, M. & Thränert, M. (2007). Begriffsbestimmung und Metamodell für Vorgehensmodelle. in Fähnrich, K.-P., Thränert, M. & Wetzel, P. (Hrsg.), Leipziger Beiträge zur Informatik. Bd. VI: Integration Engineering : Motivation – Begriffe – Methoden – Anwendungsfälle. Leipzig: Eigenverlag Leipziger Informatik-Verbund (LIV).

Yakut, Y. (2008). Agile Softwareentwicklung.

- **Proceedings from conferences**

Himmelreich, J. (2006). Agile Software Entwicklung nach Winston Royce. In Agiles Beiträge zur Konferenz InterPM, April 2006 (123-134), Glashütten: dpunkt.verlag.

Kautz, K. (2010). Participatory Design Activities and Agile Software Development. In J. Pries-Heje et al. (Eds.). Human Benefit through the Diffusion Information Systems Design Science Research, IFIP AICT 318, (303–316), Berlin Heidelberg New York: Springer.

Rittenbruch, M., McEwan, G., Ward, N., Mansfield, T. & Bartenstein, D. (2002). Extreme Participation - Moving Extreme Programming Towards Participatory Design. In Participatory Design Conference, June 23-25 2002, Malmo.

- **Reports**

Lindermann, N. et. al. (2009). Netzwerken 2.0 in KMUs - Kleine und mittlere Unternehmen im Zentrum Web 2.0 basierter Kooperation. In von Kortzfleisch, H.F.O, Jung, R.H, Nüttgens, M. & Scherrer, B.U. (Eds.). Working paper from the project KMU 2.0, 1/2009.

- **Journal Articles**

McAfee, A.P. (2006) Enterprise 2.0: The Drawn of Emergent Collaboration. Sloan Management Review. 47 (3). 21-28.

Stamer, S. & Baier, T. (2008). Enterprise 2.0: Mit Selbstorganisation zu mehr Effizienz und Kreativität. Computer und Arbeit. 8-9, 19-23.

Street, C. T. & Cameron A. F. (2007). External Relationships and the Small Business: A Review of Small Business Alliance and Network Research. Journal of Small Business Management. 45 (2), 239-266.

Zorn, I., Büschenfeldt, M. & Schelhowe, H. (2008). Kooperative Softwareentwicklung einer Sekretariatsplattform als Bildungsprozess. In Michael Herczeg & Martin Christof Kindsmüller (Eds.), Mensch & Computer 2008: Viel Mehr Interaktion, Interdisziplinäre Fachtagung, 7.10. September 2008 Lübeck, (337-345). München: Oldenbourg Verlag.

- **Web pages**

CPSR. (2005). Participatory Design. March 2010, from http://cpsr.org/issues/pd/.

Institut für Mittelstandsforschung. (2010). Definition Familienunternehmen im engeren Sinn. Mai 2010, from http://www.ifm-bonn.org/index.php?id=68

Wyllie, D. (2008). Blogs, Wikis, Social Networks – Warum der Mittelstand Web 2.0 braucht. January 2009, from http://www.computer-woche.de/knowledge_center/mittelstands_it/1866746/.

O'Reilly, T. (2005). What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software. May 2010, from http://www.oreilly.de/artikel/web20.html.

O'Reilly, T. (2006). Web 2.0 Compact Definition: Trying Again. May 2010, from http://radar.oreilly.com/archives/2006/12/web-20-compact.html.