

# Determinants of Success in Crowdsourcing Software Development

Hamed Tajedin  
Schulich School of Business  
4700 Keele St.,  
Toronto, Ontario, Canada, M3J 1P3  
(647) 774-4657  
Hamed.tajedin@gmail.com

Dorit Nevo  
Schulich School of Business  
4700 Keele St.,  
Toronto, Ontario, Canada, M3J 1P3  
(416) 736-2100  
dnevo@schulich.yorku.ca

## ABSTRACT

With the advent of digitization, recent years have witnessed a surge toward collective undertaking of production process different from traditional ways of organizing. In this vein, crowdsourcing has lent itself into a successful emerging mode of organizing and firms are increasingly using it in their value creation activities. However, despite popularity in practice, crowdsourcing has received little attention from IS scholars. Specifically, what the determinants of success in this model are remains an unexplored area of research that we strive to address in this paper. We focus on software development via crowdsourcing and drawing on studies from IS success, OSS and software development, we build a model of success that has three determinants: the characteristics of the project, the composition of the crowd and the relationship among key players. Finally, we describe our research methodology and conclude with potential contributions of our work.

## Categories and Subject Descriptors

H.0 [General]

## Keywords

Crowdsourcing, outsourcing, opensource software, software development

## 1. INTRODUCTION

With the advent of digitization, recent years have witnessed a surge toward collective undertaking of production process different from traditional ways of organizing in which required collocation of individuals. Opensource software movement (OSS) has shown that production of software can happen outside firm boundaries through contribution of talents who are globally spread. The success of this model has attracted managers' attention to crowdsourcing model through which firms can harness the power of a crowd of interested individuals who can virtually gather around a production goal. As crowdsourcing has lent itself into an efficient mode of organizing, a growing number of firms have been using it. An estimate of the amount of money paid to the crowd of over 2 million people for a sample of

crowdsourcing service providers is about \$750 million in 2009. More recent estimates show that the number of crowd workers is growing in excess of 100% a year while Nearly 77% of all workers have a primary job. In terms of revenue for service providers, crowdsourcing demonstrated a 50% growth in 2010 and 75% in 2011. Despite this popularity growth in practice, crowdsourcing has received little attention from IS scholars. Specifically, what the determinants of success in this model are remains an unexplored area of research that we strive to address in this paper.

Firms have been using crowdsourcing for a diverse set of purposes, from problem solving [1] to accomplishing part of their operation [2], to harnessing the external knowledge of individuals beyond their boundaries in order to come up with new ideas for business development [3]. While there are numerous applications for crowdsourcing, software development is one with special importance to IS scholars since it bears the IT artifact that shapes the core of our research domain [4]. However as an emerging phenomenon, successful accomplishment of such method of software development is a challenge for practitioners [5]. Software development via crowdsourcing can happen directly, i.e. a firm reaches out to the crowd of individuals through an open call, or it can happen indirectly via an intermediary in which the intermediary bridges the crowd to the firm. The latter is more popular in practice since the intermediary has the advantage of possessing a crowd of coders over time that participates in various projects. This is the instance of crowdsourcing that we have focused on and we would like to study the determinants of success in such setting. More specifically, we try to answer the question of what are the key success factors of crowdsourcing software development with the existence of an intermediary.

We find three areas of research in IS to be relevant to our work and can help us in framing our research direction. First, given that crowdsourcing counts as a form of sourcing arrangement [6] the findings from strong body of knowledge on information technology outsourcing (ITO) can shed light on different aspects of our setting that has been explored in a more general context. In spite of the fact that ITO is a well explored area in IS, there is a dearth of research on emerging sourcing arrangements; with a few exceptions, e.g. [7], the literature is mainly focused on a dyad level, i.e. client and vendor. In our setting, which an intermediary or 'middleman' acts as a service provider to both client and vendor, the whole structure of the relationship is changed and further research is required to determine the extent to which the findings of traditional sourcing arrangements apply to these settings.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*SIGMIS-CPR '13*, May 30–June 1, 2013, Cincinnati, Ohio, USA.  
Copyright © 2013 ACM 978-1-4503-1975-1/13/05...\$15.00

Second, the similarities that opensource software (OSS) and crowdsourcing share make the former an apt area from which inferences about the latter can be drawn. However, we believe the existence of the firm at the focal point of crowdsourcing has several consequences that translate into the differences between this model and OSS. First, unlike the opensource spirit of non-proprietary use of software and General Public License (GPL), the firm appropriates the outcome of crowdsourcing through developing legal frame-works that address issues of transferring intellectual property [8]. Second, projects in opensource are defined arbitrarily by interested developers and receive contribution through self-selection mechanisms. This, in turn, results in uneven distribution of developers' attention and their contribution among projects. SourceForge.net, for example, had only 60,642 projects with more than one release, and 23,754 with more than one member, out of the a pool of 201,494 software projects in 2009 [9]. It has been estimated that no more than 6% of the projects on SourceForge.net are able to attract a noticeable community of users and developers [9]. In crowdsourcing, however, the firm has a proactive role in attracting contribution from the crowd by anchoring monetary rewards that do not exist in opensource. Hence the subtle difference between the OSS and crowdsourced software requires further investigation of the latter, albeit by drawing on the findings of OSS research.

Finally, with our focus on determinants of success, our work fits within the broader literature of IS success which is widely accepted throughout IS research as the principal criterion for evaluating information systems. Since the seminal work of DeLone and McLean in 1992, there have been a number of studies which apply this model to different domains of IS such as e-commerce [10, 11], e-government [12], knowledge management [13, 14], health-care [15], system implementation [16, 17], opensource [18] and many more. However, as new phenomena emerge within the scope of IS research, so does the issue of defining success and its determinants. In this vein, the literature lacks studies that focus on crowdsourcing and provide a model of success for it.

In sum, our work at the intersection of the three aforementioned areas of IS research focuses on crowdsourcing as a new phenomenon that provides an excellent opportunity to address the gaps that was reviewed and by doing so, furthers our knowledge on success of emerging sourcing arrangements.

## 2. Crowdsourcing: What do we know?

Crowdsourcing represents "the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call" [19]. Although this definition comes from non-academic world and there is not a generally agreed-upon definition for crowdsourcing in literature, it basically reflects the basic concepts of crowdsourcing; the existence of a focal entity, outsourcing, crowd, and open call.

As an emerging phenomenon, most of the studies on crowdsourcing are descriptive and try to explain the phenomenon by showing successful cases [3, 5, 20, 21] or classifying existing models [22-25]. Besides these, there are a few conceptual works that look at the phenomenon from various perspectives. For example, Proulx et al [26] take a perspective of information capitalism in defining crowdsourcing and argue that paradoxically the crowdsourcing can give both a sense of empowerment and alienation to the crowd. Moreover, Wexler [8] looks at crowdsourcing from a sociological perspective and explains how

the crowd has transformed from an irrational entity, which causes problems, to a collectively intelligent entity that can, when managed, solve problems. And finally, Afuah and Tucci [27] look at crowdsourcing from a problem solving perspective and argue that that under certain circumstances crowdsourcing transforms an organization's search for knowledge by spanning its boundaries and enabling access to external resources. However, there is also a growing number of empirical works on crowdsourcing. For example Leimeister et al [28], look at the design issues of competition-based crowdsourcing and Brabham [2] looks at the motivation of participants in a crowdsourcing instance.

As the quick review above indicates, crowdsourcing is the root phenomenon that seeds the research in various disciplines and the literature around it is expanding around various topics. Yet the number of studies in IS that have aspects of crowdsourcing as a focal research question is limited. This might partly be due to difficulty of knitting crowdsourcing instances to the nomological network of IS discipline [4]. In addition, we believe crowdsourcing provides an excellent opportunity for conducting transformational research that delineates the role of technology in transforming organizations and environment [29]. Therefore, we focused on software development via crowdsourcing, which has the IT artifact in its core and we are interested in determinants of success in this method of software development.

In the model of software development with crowdsourcing that we have focused on, there are three key players; the crowdsourcer, i.e. project sponsor or client, the intermediary that acts as a service provider to the client, and the crowd, i.e. the community of developers. The project sponsor defines the project and uses the service of the intermediary to reach out the community of participants in order to have the project accomplished. There are two ways that a project can be accomplished with submissions of individuals from the community. The first way is that each person that participates in the project accomplishes a small task different from that of others and the final outcome is achieved via cooperation and collaboration of these small chunks of work. For example consider the development of Wikipedia in which the submissions of people from the crowd are accumulated above each other to shape the output (See [2] for another example). On the other hand, instead of working collaboratively on tasks, individuals can compete against each other for these small tasks so that the best submission is selected for each part of the broken down project (this is similar to ideas competition case, see [28]). The model of crowdsourced software development that we have chosen and is more common in practice is of the latter form in which participants compete against each other by submitting pieces of work (design, architecture, code, bug report, etc.). The collaborative form of software development is more similar to open-source method. However as was explained, even though crowdsourcing and open source have a lot in common, in contrast to opensource, crowdsourcing entails both a focal entity, which appropriates the outcome and monetary incentivizing mechanisms. It has been shown that the intervention of extrinsic incentives such as monetary rewards can have undermining impacts on intrinsic motivations [30]. This, in turn, can reduce the sample of developers who take part in crowdsourcing projects to those who are more motivated by monetary rewards, compared to contributors to broader set of OSS projects. This attrition in the population of developers can be easier to imagine by considering what Raymond describes as the motivations of developers in opensource movement: "[t]he 'utility function' Linux hackers are

maximizing is not classically economic, but is the intangible of their own ego satisfaction and reputation among other hackers” [31](p. 41).

In the following section we propose a model of success for this method of software development that is based on ideas from IS success and software development areas.

### 3. Success of Crowdsourcing Software Development

The first step in having a success model is to define success and clarify what it refers to in software development by crowdsourcing. An Information System has many stakeholders that each can bring different definition of success into play. Software development with crowdsourcing is process that leads to an IT artifact, i.e. the developed software and in this sense it is subject to evaluation by comprehensive models of IS success that exist in the literature (e.g. [32, 33]). With this approach, the unit of analysis is the crowdsourced software and its success reflects how much the output of crowdsourcing has been successful that can be distinguished in terms of net benefits it can provide. However, we can also focus on the development phase of crowdsourcing and assess how much the development has been successful in yielding the intended deliverables to stakeholders. In this approach, success should be defined for different set of stakeholders than the former approach and different set of criteria should be used to assess the development process. For example in crowdsourcing context, success of a project can be defined as how much the project is completed on time from a manager’s view or it can be as how much it has helped developers to find better jobs from participants’ view. Moreover, as we focus on software development, success should be defined within the limited scope of initiation to release. The model of success should reflect how effective and efficient the whole process has been. At this phase the process should be the unit of analysis and the IT-artifact acts as the glue that binds different players to the process. At post-development phase, on the other hand, the focus of analysis is the IT artifact that is available to users. Hence, we define the success from the perspective of the main stakeholder, i.e. the project sponsor or crowdsourcer.

The success of a software development by crowdsourcing from the perspective of project sponsor is basically the extent to which the process has met the requirements within the desired time frame, cost and quality scope. This concept definition, however, can be operationalized and measured by second order factors. To this aim, we draw on the work of Crownston et al [34] and DeLone and McLean [32] as the basis of our operationalization. At the technical level, success measures can be code quality and documentation quality. These measures are quite applicable to the context of crowdsourcing as developers compete based on code quality. In existing methods of crowdsourcing via intermediaries, through a peer review process, experts and experienced developers evaluate submissions from the crowd. Accordingly, the outcome of this evaluation can be a proxy for user satisfaction as another dimension of success.

After a piece of code is developed by crowdsourcing, the project sponsor has to integrate the code to its existing systems, review the code for security issues, and keep the code up-to-date as the system changes over time [5]. So the use dimension of success can be measured in this sense and it can reflect the amount of rework that is needed for a code to be integrated in the project

sponsor’s system. The last dimension of success can reflect the net benefits in terms of economic gains, or speed of project completion that are realized by crowdsourcing. For the next step, we focus on determinants of success.

The first set of factors that influence the success of development process emanates from project characteristics. We posit that certain project characteristics can enhance success in crowdsourcing method of software development. The characteristics such as modular architecture allow parallel contribution of individuals to be integrated for the single whole. It has been suggested that sophisticated coding problems that make the task challenging can contribute to success of OSS [35]. This can hold true for crowdsourcing as well, since in both methods of development, developers voluntarily select projects to participate and anecdotal evidences suggest that technical sizzle can motivate developers [31]. Moreover, companies who have used crowdsourcing service have claimed that finding the right type of problem and providing appropriate amount of problem detail for developer community have been an issue for them. As Lakhani and his colleagues [5] narrate “...clients discovered that contest participation decreased if they were unclear about what problems they wanted to solve or presented problems that were too complex or vast in scope; in those cases, the TopCoder community struggled to produce an acceptable solution” (p. 10). Since the contributions of the crowd of developers are to build a single whole, projects that can be broken down to small modules that are clear in requirement and have limited interdependencies seem to contribute to success.

Similar to OSS licensing, another important issue in crowdsourcing is that of legal considerations [3]. However, OSS tenets are against proprietary ownership of software code and hence, restrictive licenses have shown to be ineffective in pulling developers, e.g. [18]. In crowdsourcing, on the other hand, the crowdsourcer seeks intellectual property right for its developed software and might see their benefits at risk due to crowdsourcing [5]. Hence, the choice of appropriate type of license for projects and keeping the community informed can be crucial in attracting developers and achieving success. Considering all this, we can infer that:

Proposition 1: Project characteristics affect success of crowdsourcing software development where projects with modular, challenging but not complex structure, and clear scope and license type have greater influence on success.

For any given crowdsourcing provider, there are large numbers of projects that will be in need of contributions from the community. Contrary to traditional managerial methods in which resources are allocated to each project contingently, in crowdsourcing, projects should virtually attract resources themselves. Proposition one suggests that certain project characteristics are necessary to attract resources. Moreover, the composition of community plays a critical role in a project fate. The more people are attracted to a project, the more diversity will shape and newer viewpoints and methods are applied to the project. This is one of the strength points of crowdsourcing that the crowd can even provide the client with a better understanding of the problem or potential solutions [3, 5]. The effect of diversity on success of software project has also been shown in literature [36]. For a team of software developer, diversity can play a crucial role for emergence of new ideas and opinions as well as new ways of doing the task and solving the problems. This, in turn, can

positively affect the success of the project. But one can imagine a maximum point for this after which not only with the increase of diversity the performance does not improve but also it deteriorates due to problems that emerge because of coordination and communication.

Besides diversity, the community should also have the competencies that the project requires. Use of popular languages has been shown to be an effective factor for success since there are readily more developers who can accomplish tasks with common coding languages. Moreover, although the community may be composed of a large number of people, there is a huge variance in terms of expertise among members. The size of community at a crowdsourcing service provider, for example, is large, but the number of people within that community who actively participate in projects is much smaller; the talent pool of this community that can stand at global rankings accounts for 0.5% of the total population [5]. The chance of success will rise if lesser skilled members are cultivated and more professional contributors take part in projects, hence:

Proposition 2: Crowd composition affects success and this effect is higher for diverse crowds with more number of developers who are competent at software development.

In addition, crowdsourcing happens in a social context where individuals interact and communicate to get new information and share knowledge. The importance of relationship management in outsourcing settings is well emphasized, e.g. [6, 37-40]. Relations among developers and between crowdsourcer and community are of utter importance. Cohesion among project team members would lead to more effective communication and learning [41]. On the other hand, trust formation, communication quality, and identification with the project team are factors that have been associated with success of OSS [42] and due to similarity of the development methods we can expect the same to hold true for crowdsourcing.

Another form of relations that count is the one between project sponsors and the community. This has been particularly emphasized in the work of Ågerfalk and Fitzgerald [43] who focus on firms who use OSS as a method of developing their software, which is quite similar to crowdsourcing. They contend that openness, trust, tact, professionalism, transparency, and complementarity are key success factors in open sourcing. As they put it: "...open sourcing is not primarily about commissioning software development to a third party, but rather about engaging in long-term collaborative activities leading to a sustainable ecosystem. Since many of the collaborators in this ecosystem are likely to be the customer's competitors, the collaboration is necessarily done in a spirit of co-opetition... Both customer and community members have a shared responsibility to actively contribute to the development and sustainability of the ecosystem" (p. 404). The collaborative interaction of community and sponsor can also lead to better understanding of problem and enhanced designs. Hence:

Proposition 3: Retention of collaborative relations among developers and between community and project sponsor positively affects success.

So to sum up, we propose that three factors affect the success of crowdsourced projects in software development. The first stem from the characteristics of the project and we posit that certain

kind of projects are more appropriate for crowdsourcing and that task technology fit plays an important role here [44]. The second factor is the crowd composition and how diverse and competent the community of developers is. Since the projects are accomplished by submissions from the crowd, the more readily the it has the knowledge and skill sets that are required for a given project, the more number of submissions with better quality would be expected. Finally the last factor is the relationship among the actors in play. Both studies from outsourcing and from open source software development underscore the importance of maintaining a collaborative relationship among actors in order to achieve success and hence we posit the same would hold for a crowdsourcing instance.

## 4. Research Method

Associate As previously mentioned, in the model of crowdsourcing that we focused on, there is an intermediary that acts as the service provider to various clients, which want to accomplish their projects via crowdsourcing. TopCoder.com is one of those service providers that have established a community of hundreds of thousands of developers. We have chosen TopCoder because it is a successful example of crowdsourcing platform. It has been founded in 2001 and now it offers crowdsourcing service for software development to various clients across numerous projects.

We chose a mixed method to conduct our research project. The first part is qualitative and we intend to collect primary data via interviews from individuals representing all three aspects of the model, i.e. the community, the service provider, and the clients. This will enable us to see the phenomenon from different aspects and analyze it from the viewpoint of each stakeholder involved. We are now in the phase of data collection and analysis. We have conducted 13 interviews; 7 interviews with community members and 6 interviews with TopCoder managers. The interviews have been semi-structured (the complete interview protocol can be provided upon request) and there have been minor modifications to some questions during data collection phase. All interviews have been conducted during a three-day period at the site of an event held by TopCoder and they have been audio recorded. The interviews are from 29 to 122 minutes (with an average of 45 minutes) are now in phase of transcription. We have a model in mind but our strategy is to be open to our data letting it guide us further along the way to shape our model. We still have five interviews in our plan with clients of TopCoder that will be added to our collected data.

The second phase of our research is a quantitative test of our model on the same-targeted sample. We will test our model with a questionnaire that will be answered by subjects from the three perspectives of community, service provider, and clients. As we pass phase one, we can more specifically design this part of our research in details.

## 5. Potential Contribution

Crowdsourcing literature is the focal phenomenon in studies from different disciplines from sociology, e.g. [8, 26], economics, e.g. [45], management, e.g. [27] to IS, e.g. [24]. Yet we believe that as an emerging concept, there is a lot to be explored in this area of research especially in information systems domain.

We have focused on software development via crowdsourcing. In essence, when a company is considering crowdsourcing for

developing a piece of code, it is making the make/buy decision but if two decades ago buy decision would be translated to outsourcing, nowadays there is another option of crowdsourcing on the table for the same chosen strategy. We have a strong body of research in information technology outsourcing in IS. While the finding of such studies can be illuminating for traditional outsourcing arrangement, the new setting of crowdsourcing bears new structure and dynamism that make it dramatically different from outsourcing. In a traditional outsourcing arrangement there are always two key players, client and vendor, and the literature in information technology outsourcing (ITO) or business process outsourcing (BPO) has looked at the relationship between these dyads of organizations. For example a stream of research on ITO looks at determinants of success in a sourcing arrangements by assessing the effects of contractual, e.g. [46, 47] and relational, e.g. [37, 48] forms of governance between client and vendor. In a few studies, the relationship that is being studied goes beyond this dyad and the role of intermediaries are considered, e.g. [7]. But, in line with other studies at this area, the focus remains at the organizational level of analysis. In all these studies, the structure of IOR that is being studied is comprised of collective entities at organizational level.

However, in a crowdsourcing model of software development with the existence of an intermediary, not only individuals as single developers come into play, but also another collective entity emerges, the crowd, that is more of a sociological entity. The existence of individuals, organizations, and the community (or the crowd) in this model, makes this model of interorganizational relationship (IOR) a unique one that has not yet received enough attention in the literature. The structure of this IOR is a complex combination of relationships consisting of three forms: 1) the relationship between the client and the intermediary; 2) the relationship between individuals as single developers with both intermediary and the client; 3) the relationship between the intermediary and the community of developers. While the studies of ITO can cover the organization to organization relationship that exists in this model, there are relationships in this model that go beyond what has been covered by ITO and we believe the findings from this study can shed light on how this relationship works. While the studies of ITO can cover the organization to organization relationship that exists in this model, there are relationships in this model that go beyond what has been covered by ITO and we believe the findings from this study can shed light on how this relationship works.

In addition to structural differences between crowdsourcing and outsourcing, there are differences in what is required to have a sourcing arrangement in the two setting. Simply talking, outsourcing arrangement requires searching, assessing the supplier capabilities, contracting, and monitoring. In this setting, the firm goes beyond its boundary for a specific service or product but after the process of finding the supplier, it will settle down with an option (or multiple options) of a supplier(s), prospecting that it (they) will provide the requirement. In a crowdsourcing arrangement, however, the scenario is changed even though the same thing is sought; when a firm reaches out to the crowd, it does not go through the same process of searching for suppliers, assessing their capabilities, arranging the best legal contracts so that other party is held accountable if it does not provide what it has claimed, and continuously putting resources on monitoring the performance of the supplier. Instead, the crowd provides the service or product that the firm seeks, and the firm

puts some effort into choosing the best option out of what is available. For example consider the case that an R&D problem is outsourced to a scientific lab versus being crowdsourced. In the first case, the firm first seeks the best lab that has the potential to solve the problem, and then the firm goes through a process after which the lab provides the solution. But in case of crowdsourcing, the crowd provides answers to the question and the firm selects the best one from the set of submitted solutions. This is not to say that the latter process is easier or simpler, but it is obviously different from that of an outsourcing arrangement, and the difference lies at the focus of the selection efforts that the firm puts resources on. Hence, although similar, the determinants of success are different in the two settings. We believe that our study can shed light on the determinants of success in this new setting, thus can further our understanding of the broader concept of outsourcing.

Moreover, our study has a software development context that is similar to that of OSS. While there are similarities between the two, there are also subtle differences that make crowdsourcing a unique setting. The findings of crowdsourcing can also feed back the OSS where the role of community comes into play. However, this should happen cautiously as the drivers of community participations may not necessarily be the same in the two settings.

## 6. REFERENCES

- [1] Sieg, J.H., M.W. Wallin, and G. Von Krogh, "Managerial challenges in open innovation: a study of innovation intermediation in the chemical industry". *R&D Management*, 2010. 40(3): p. 281-291.
- [2] Brabham, D.C., "Moving the crowd at threadless". *Information, Communication & Society*, 2010. 13(8): p. 1122-1145.
- [3] Jouret, G., "Inside Cisco's search for the next big idea". *Harvard business review*, 2009. 87(9): p. 43-45.
- [4] Benbasat, I. and R.W. Zmud, "The identity crisis within the IS discipline: Defining and communicating the discipline's core properties". *Mis Quarterly*, 2003: p. 183-194.
- [5] Lakhani, K., D. Garvin, and E. Lonstein, "TopCoder (A): Developing software through crowdsourcing". *Harvard Business School General Management Unit case*, 2010(610-032).
- [6] Oshri, I., J. Kotlarsky, and L.P. Willcocks, *The handbook of global outsourcing and offshoring 2011*: Palgrave Macmillan.
- [7] Mahnke, V., J. Wareham, and N. Bjorn-Andersen, "Offshore middlemen: transnational intermediation in technology sourcing". *Journal of Information Technology*, 2008. 23(1): p. 18-30.
- [8] Wexler, M.N., "Reconfiguring the sociology of the crowd: exploring crowdsourcing". *International Journal of Sociology and Social Policy*, 2011. 31(1/2): p. 6-20.
- [9] Meirelles, P., et al. A study of the relationships between source code metrics and attractiveness in free software projects. in *Software Engineering (SBES), 2010 Brazilian Symposium on*. 2010. IEEE.
- [10] De Wulf, K., et al., "The role of pleasure in web site success". *Information & Management*, 2006. 43(4): p. 434-446.
- [11] Soh, C., M.L. Markus, and K.H. Goh, "Electronic marketplaces and price transparency: Strategy, information

- technology, and success". *Mis Quarterly*, 2006. 30(3): p. 705-723.
- [12] Gil-García, J.R. and T.A. Pardo, "E-government success factors: Mapping practical tools to theoretical foundations". *Government Information Quarterly*, 2005. 22(2): p. 187-216.
- [13] Markus, M.L., "Toward a theory of knowledge reuse: Types of knowledge reuse situations and factors in reuse success". *Journal of Management Information Systems*, 2001. 18(1): p. 57-94.
- [14] Wu, J.-H. and Y.-M. Wang, "Measuring KMS success: A respecification of the DeLone and McLean's model". *Information & Management*, 2006. 43(6): p. 728-739.
- [15] Häyrynen, K., K. Saranto, and P. Nykänen, "Definition, structure, content, use and impacts of electronic health records: a review of the research literature". *International journal of medical informatics*, 2008. 77(5): p. 291.
- [16] Sharma, R. and P. Yetton, "The contingent effects of training, technical complexity, and task interdependence on successful information systems implementation". *Mis Quarterly*, 2007. 31(2): p. 219-238.
- [17] Wixom, B.H. and H.J. Watson, "An empirical investigation of the factors affecting data warehousing success". *Mis Quarterly*, 2001. 25(1): p. 17-32.
- [18] Subramaniam, C., R. Sen, and M.L. Nelson, "Determinants of open source software project success: A longitudinal study". *Decision Support Systems*, 2009. 46(2): p. 576-585.
- [19] Howe, J. The rise of crowdsourcing. *Wired Magazine*, 2006. 14, 1-4.
- [20] Greengard, S., "Following the crowd". *Communications of the ACM*, 2011. 54(2): p. 20-22.
- [21] Huston, L. and N. Sakkab, "Connect and develop". *Harvard business review*, 2006. 84(3): p. 58-66.
- [22] Bonabeau, E., "Decisions 2.0: the power of collective intelligence". *Mit Sloan Management Review*, 2009. 50(2): p. 45-52.
- [23] Cook, S., "The contribution revolution". *Harvard business review*, 2008. 86(10): p. 60-69.
- [24] Doan, A., R. Ramakrishnan, and A.Y. Halevy, "Crowdsourcing systems on the World-Wide Web". *Communications of the ACM*, 2011. 54(4): p. 86-96.
- [25] Haythornthwaite, C. *Crowds and communities: Light and heavyweight models of peer production*. 2009. IEEE Computer Society.
- [26] Proulx, S., et al., "Paradoxical empowerment of producers in the context of informational capitalism". *New Review of Hypermedia and Multimedia*, 2011. 17(1): p. 9-29.
- [27] Afuah, A. and C.L. Tucci, "Crowdsourcing As a Solution to Distant Search". *Academy of Management review*, 2012. 37(3): p. 355-375.
- [28] Leimeister, J., et al., "Leveraging crowdsourcing: activation-supporting components for IT-based ideas competition". *Journal of Management Information Systems*, 2009. 26(1): p. 197-224.
- [29] Agarwal, R. and H.C. Lucas Jr, "The information systems identity crisis: Focusing on high-visibility and high-impact research". *Mis Quarterly*, 2005: p. 381-398.
- [30] Frey, B.S. and R. Jegen, "Motivation crowding theory". *Journal of economic surveys*, 2001. 15(5): p. 589-611.
- [31] Raymond, E., "The cathedral and the bazaar". *Knowledge, Technology & Policy*, 1999. 12(3): p. 23-49.
- [32] DeLone, W.H. and E.R. McLean, "Information systems success: The quest for the dependent variable". *Information systems research*, 1992. 3(1): p. 60-95.
- [33] Seddon, P.B., "A respecification and extension of the DeLone and McLean model of IS success". *Information systems research*, 1997. 8(3): p. 240-53.
- [34] Crowston, K., H. Annabi, and J. Howison, "Defining open source software project success". *Former Departments, Centers, Institutes and Projects*, 2003: p. 4.
- [35] Comino, S., F.M. Manenti, and M.L. Parisi, "From planning to mature: On the success of open source projects". *Research Policy*, 2007. 36(10): p. 1575-1586.
- [36] Liang, T.P., et al., "Effect of team diversity on software project performance". *Industrial Management & Data Systems*, 2007. 107(5): p. 636-653.
- [37] Kotlarsky, J. and I. Oshri, "Social ties, knowledge sharing and successful collaboration in globally distributed system development projects". *European Journal of Information Systems*, 2005. 14(1): p. 37-48.
- [38] Kotlarsky, J., P.C. Van Fenema, and L.P. Willcocks, "Developing a knowledge-based perspective on coordination: The case of global software projects". *Information & Management*, 2008. 45(2): p. 96-108.
- [39] Rottman, J.W. and M.C. Lacity, "Proven Practices for Effectively Offshoring Proven Practices for Effectively Offshoring IT Work". *Mit Sloan Management Review*, 2006. 47(3): p. 56-63.
- [40] Srikanth, K. and P. Puranam, "Integrating distributed work: comparing task design, communication, and tacit coordination mechanisms". *Strategic Management Journal*, 2011. 32(8): p. 849-875.
- [41] Singh, P.V., Y. Tan, and V. Mookerjee, "Network effects: The influence of structural social capital on open source project success". *SSRN eLibrary*, 2008.
- [42] Stewart, K.J. and S. Gosain, "The impact of ideology on effectiveness in open source software development teams". *Mis Quarterly*, 2006: p. 291-314.
- [43] Agerfalk, P.J. and B. Fitzgerald, "" Outsourcing to an Unknown Workforce: Exploring Opensourcing as a Global Sourcing Strategy". *Management Information Systems Quarterly*, 2008. 32(2): p. 385-409.
- [44] Ziggers, I. and B.K. Buckland, "A theory of task/technology fit and group support systems effectiveness". *Mis Quarterly*, 1998: p. 313-334.
- [45] Horton, J.J. and L.B. Chilton. *The labor economics of paid crowdsourcing*. 2010. ACM.
- [46] Gopal, A. and B.R. Koka, "The Role of Contracts on Quality and Returns to Quality in Offshore Software Development Outsourcing". *Decision Sciences*, 2010. 41(3): p. 491-516.
- [47] Gopal, A., et al., "Contracts in offshore software development: An empirical analysis". *Management Science*, 2003: p. 1671-1683.
- [48] Liker, J.K. and T.Y. Choi, "Building deep supplier relationships". *Harvard business review*, 2004. 82(12): p. 104-113.