Social Computing for Software Engineering: a Mapping Study

Amalia Ardini, Mahmood Hosseini, Amen Alrobai, Alimohammad Shahri, Keith Phalp, Raian Ali
{aardini, mhosseini, aalrobai, ashahri, kphalp, rali}@bournemouth.ac.uk
Bournemouth University, UK

Abstract

There is a continual growth in the use of social computing within a breadth of business domains; such as marketing, public engagement and innovation management. Software engineering research, like other similar disciplines, has recently started to harness the power of social computing throughout the various development phases; from requirements elicitation to validation and maintenance and for the various methods of development and structures of development teams. However, despite this increasing effort, we still lack a clear picture of the current status of this research. To address that lack of knowledge, we conduct a systematic mapping study on the utilisation of social computing for software engineering. This will inform researchers and practitioners about the current status and progress of the field including the areas of current focus and the geographical and chronological distribution of the research. We do the mapping across a diversity of dimensions including the activities of software engineering, the types of research, the characteristics of social computing and the demographic attributes of the published work. Our study results show a growing interest in the field, mainly in academia, and a general trend toward developing designated social computing platforms and utilising them in mainly four software engineering areas; management, coding, requirements engineering, and maintenance and enhancement.

Keywords: Social Computing, Software Engineering, Systematic Mapping

1. Introduction

A quality software product is the outcome of a good software development process, in which the collaboration amongst stakeholders is designed to work properly and in a sustainable manner. One aspect that assists in building a working collaboration is the establishment of solid communication, coordination, and awareness amongst stakeholders. The research in various disciplines, such as Computer Supported Cooperative Work [31] Global Software Engineering [33], Cooperative and Human Aspects of Software Engineering [34], and Social Software Engineering [2], has investigated various ways to build and maintain such work collaboration and support the software engineering process. The emergence and wide popularity of social media and Web 2.0 technology encouraged the research on utilising these techniques in the context of software development.

Social interaction, which is viewed as a core component of software engineering, takes place amongst various stakeholders and developers. The use of social computing in software engineering, viewed as a highly interactive activity, is advocated as a rich means to increase the efficiency of interaction in terms of clarity and speed of communication, situational awareness, documented and easily searched interaction, and community forming. Using social computing in the context of software engineering is challenging and requires an investigation on when and how to conduct it and with whom, i.e. it requires an engineering process itself. Commercial social computing falls short in serving the diversity of software development activities and development styles and teams. For example, a popular de facto social network is too generic to serve as an efficient communication medium between end users and developers. This has motivated researchers to investigate how to develop social techniques that are expressly tailored to the peculiarities of software engineering.

Although the field has attracted a wide range of researchers and practitioners, we still have no clear picture of its current state. Systematic mapping studies are powerful tools to explore the extent to which the research has been conducted and applied, and also investigate the distribution of that research with regards to certain criteria [50], [40], [5]. To address this lack of information, this paper conducts a systematic mapping study, which focuses on the area of software engineering aided by the use of social computing. Our search for literature was focused on papers which were written between the years 2006 and 2013 leading to a final collection of 139 papers. We analysed each of the selected papers against a set of mapping criteria, including the demographics as well as the content of the research. We report a detailed classification and analysis of the results and answer several intriguing points on the current state of the research in this field.

The paper is structured as follows. In Section 2, we introduce the domain of our systematic mapping study. In Section 3, we describe the protocol we followed for searching and analysing the research in the area. This includes the research questions of our mapping study. In Section 4, we extract the data and produce a descriptive statistics from the collected papers. In Section 5, we interpret the statistics we produced in Section 4 to answer each of the research questions raised in Section 3, and we also discuss the limitations of our study. We conclude the paper in Section 6.

2. Social Computing for Software Engineering

Current advances in software engineering consider the social aspects amongst stakeholders during the development process as a first class concept to accommodate. The social aspects of software engineering concern the relationships amongst various stakeholders, including the end users. These relationships should be accommodated in both the development process and the system-to-be in which the designed software operates [2, 8].

The accommodation of social aspects in the software and development process benefits from software tools and environments. This has stimulated the application of social computing and the use of Web 2.0 in this context. Early studies have investigated the use of social computing in facilitating discussions [25] and knowledge sharing [4]. Further studies have also shown that such facilitation is not only useful for supporting discussions and knowledge sharing, but also for building awareness [12] and aiding team coordination [41].

Research conducted in this area has started to gain attention in the software engineering community, as this study is going to demonstrate. The research explores the general concept of utilising social computing to facilitate and support a range of social aspects in software engineering [44, 59]. Now we can also observe practical applications of the concept through creating social tools to support certain software engineering activities, such as requirements elicitation, development and validation [36, 32, 29].

Despite this growing interest in the area, we still lack a clear picture on what and how current research has been established. To collect and recognise studies on social computing for software engineering, we conducted the systematic mapping study presented in this paper. This study is meant for researchers and practitioners to estimate the current state of the research and to identify areas which are still under-researched as well as the distribution of the research across various criteria, such as the geographic distribution, chronological distribution, types of affiliation, and publication venues. It also analyses each selected paper to classify it against criteria related to the content itself, such as the social computing technique being followed and the software engineering activity being aided.

3. Systematic Mapping Protocol

A systematic mapping is a process of identifying, categorising, and analysing existing literatures that are relevant to a certain research topic. The result of a systematic mapping will provide a structured report based upon categorisations of the existing literatures, which is often able to illustrate a visual summary that portrays the mapping relationship between the literatures and the categories [50].

In this study, our method mainly relied on the foundation work in [50] and it was also informed by other foundation and systematic mapping studies presented in [40], [5], and [46]. We looked at the area of the use of social computing for supporting the various activities of software engineering. By social computing, we not only mean the commercial, publicly-available social media and applications, but also tools which are specifically designed to facilitate interaction in software engineering. Therefore, the notion of social computing, as used in this paper, refers to software-mediated approaches that facilitate interactions amongst individuals and organisations. Our systematic mapping study was performed in two stages: the preparation of the study and the conduct of the study. We describe these stages in the next two sections respectively.

3.1. Preparation of the Study

In this phase, we defined the research questions, scope, search criteria, links amongst the papers, search for the papers, identification of the study content, and creation of the map from the selection criteria. The

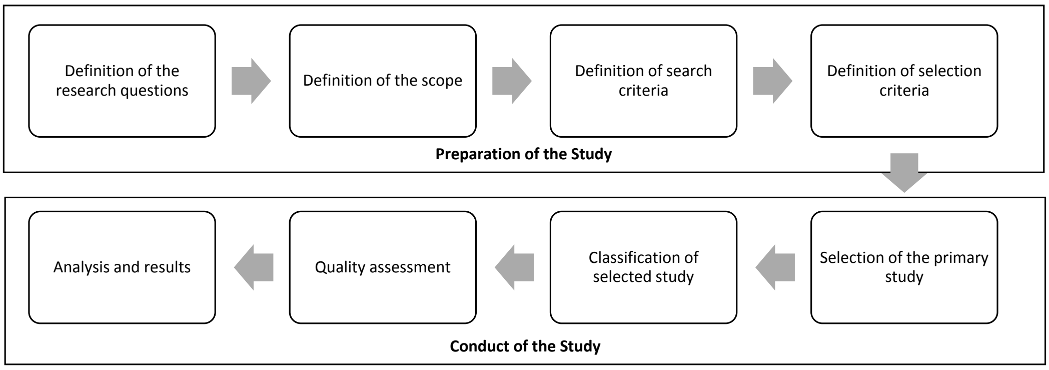

Definition of the research questions
Definition of the scope
Definition of search criteria
Definition of selection criteria
Preparation of the Study
Selection of the primary study
Classification of selected study
Quality assessment
Analysis and results
Conduct of the Study

Figure 1: Systematic Mapping Process

mapping process is best described as illustrated in Figure 1.

3.1.1. Definition of Research Questions

Our mapping study was focused on addressing the following research questions (RQ):

- **RQ1**. What are the areas of software engineering assisted by the use of social computing?
- **RQ2**. What are the types of research used to conduct the study?
- **RQ3**. For any empirical study, what are the forms of study adopted to conduct the research?
- **RQ4**. What are the facets and aspects of social computing which were used to support software engineering and how?
- **RQ5**. What are the research communities that conducted the research in the area and what are the characteristics of publications used to publish the studies?
- **RQ6**. What is the geographical and chronological distribution of the research in the area?
- **RQ7**. What are the social computing blocks which are presented in the research and how?

3.1.2. Definition of Scope

To frame the above research questions, we adopted the PICO criteria suggested by [51]. It helps to define the preliminary scope of the research questions and decide the research initial boundaries, and guide the course of the research.

- **Population**. The use of social computing in general and its use for software engineering in particular are new concepts. Thus, we choose to widen the population and enable broad exploration of the existing studies. Consequently, we ensure that the relevant areas of study are included with less strict criteria on the type of study. Therefore, the population included a wide range of peer-reviewed articles including empirical researches, preliminary studies, and proposed theories at various stages of maturity.
- **Intervention**. In order to provide an appropriate mapping, there are certain processes that need to be executed. These include searching for literatures within the population, extracting information from collected papers, defining a set of classifications, creating the map, and analysing the results to answer the research questions. Two of the researchers made the search and analysis and a third researcher acted as a referee when a disagreement occurred. At the end of this process, an additional check was done by a fourth researcher to ensure accuracy.
- **Comparison**. Our mapping study compares various dimensions of both social computing and software engineering. The dimensions on the software engineering side are the phases of the development. We chose a wide range of dimensions on the social computing side. This includes the techniques used, the connectivity and interaction style and the structure of the team.
- **Outcomes**. The collected papers must represent a wide coverage of studied areas in the field of social computing for software engineering. This is to ensure the validity and the objectivity of this mapping study. We also chose to perform categorisation of the existing studies that create the map. Finally, an analysis of the map was performed to draw insights about the field, including the trends and areas which are still under-researched.

3.1.3. Definition of Search Criteria

We applied less restrictive search criteria in order to retrieve a wider range of literature for this emerging topic. Having more studies provides assistance in recognising the trend within this particular topic and its prime focus. Our tracking of the main conferences and workshops in the area enabled us to identify an initial set of key research groups and their scopes. Then we began a manual search by tracking those research groups, hence tracking a wider set of dissemination places (journals, conferences and workshops). This helped us in identifying further key researchers which were both cited by, and also citing, the initial set of papers.

Our search indicated that early research in the field goes back to 2006, which was decided to be the start year for our search process. We observed a set of well-reputed journals in both software engineering and information systems which published papers in the area of our study. The main journals which we searched in were:

- Information and Software Technology (Elsevier)
- Journal of Systems and Software (Elsevier)
- Information Systems (Elsevier)
- Software and System Modeling (Springer)
- Information Systems Frontiers (Springer)
- ACM Transactions on Software Engineering and Methodology
- IEEE Transactions on Software Engineering

For conference papers, and to increase the number of studies, we included studies published not only in the main track but also as a new and emerging idea, short or position papers, demos, and doctoral symposiums. In the beginning, we covered this initial set of conferences:

- International Conference of Software Engineering (ICSE)
- International Conference on Advanced Information Systems Engineering (CAiSE)
- IEEE Requirements Engineering Conference (RE)
- Conference on Foundation of Software Engineering (FSE)
- Conference on Future of Software Engineering Research (FoSER)
- Conference on Computer Supported Cooperative Work (CSCW)
- International Conference on Global Software Engineering (ICGSE)
- Conference on Human Factors in Computing Systems (CHI)
- Conference on Designing Interactive Systems (DIS)
- Conference on Software Engineering Research and Practice (SERP)
- International Conference on Information Systems (ICIS)
- International Conference on Human-Computer Interactions (INTERACT)

Furthermore, we identified that there is a range of workshops strongly related to the topic of our study. Some of these workshops were standalone and others were co-located with a main conference. In the beginning, we particularly searched the following workshops:

- SSE (Social Software Engineering) Workshop
- Workshop on Web 2.0 for Software Engineering (Web2SE)
- Workshop on Cooperative and Human Aspect of Software Engineering (CHASE)
- Workshop on Requirements Engineering for Social Computing (RESC)
- Workshop on Requirements Engineering for Systems, Services, and Systems-of-Systems (RESS)
- Workshop on Wikis for Software Engineering (WIKIS4SE)
- Workshop on Sharing and Reusing Architectural Knowledge (SHARK)
- Workshop on Software Engineering within Social Software Environments (SENSE)

Whilst searching for eligible studies, we recognised that major digital repositories such as Scopus, ScienceDirect, DBLP, ACM Digital Library, and IEEEXplore provided a collection of published studies from various journals, conferences and workshops. Consequently, we exploited these repositories to look for the relevant existing studies not necessarily published in the aforementioned venues. We also used automated search using Google Scholar. When using the search engines, we determined a set of keywords for each discipline of the study, software engineering and social computing, and used combinations of these keywords:

- Software Engineering: Management, Requirements, Coding, Design, Maintenance, Metrics, Development, Validation, Testing, and Software.
- Social Computing: Blog, Microblogging, Crowdsourcing, Wiki, Instant Messaging, Dashboard, Social Media, Social, Support, Global, Networking, Knowledge, Socio-Technical, Sharing, Awareness, Collaborative, Communication, Coordination, Distributed.

We emphasise that the results of our manual search through the proceedings of the predetermined set of journals, conferences and workshop as well as the results obtained via automated search engines led to another set of papers which cite or were cited by those results. For each candidate paper, a scanning through the titles and abstracts of the papers was done to take an initial decision on the relevance of the paper. When there was a doubt about the relevance, we had to scan through the introduction and the conclusion of the paper. However, if the scan did not confirm the relevance, we had to do a final examination through the content itself.

3.1.4. Definition of Selection Criteria

We decided a set of inclusion criteria aiming to increase the possibility of having relevant and reliable results, and minimise the possibilities of any harm to the subject:

1. Literatures (e.g. books, papers, technical reports) that described a study in the area of the use of social computing for software engineering.
2. When similar studies were reported in several papers/literatures, only the latest publication was considered, unless there was a significant difference between these studies.
3. Several studies reported in one paper were treated separately.

Exclusion criteria were applied on the papers which met the search and inclusion criteria. Our exclusion criteria were:

1. Unaccredited literatures.
2. Literature that was unavailable or the availability was only in the form of abstract, tutorials, posters or presentation material.
3. Duplicated studies.

3.2. Conduct of the Study

After the search was completed and a number of literatures were collected, we initiated the main mapping study in three steps: (1) selecting the primary study, (2) classifying selected studies, and (3) determining quality assessment. The aim of these steps was to investigate and assess the collected studies for their relevance and qualification towards the objective of this mapping study.

3.2.1. Selection of the Primary Study

In the selection process, we first screened the titles and abstracts to take a preliminary decision about the relevance. We implemented the inclusion and exclusion criteria to the filtering process. Unless they were clearly irrelevant, the literatures were kept to ensure the comprehension of the potential primary studies. The review of papers found through keywords-based search led to another set of relevant papers which were scanned again through the same two stages process. Based upon both stages of the filtering process, we found a total of 351 literatures from various sources including journals, conferences, workshops, etc., on the first stage. Then, on the second stage, we narrowed down the number of literatures into 269 papers.

To conduct further examination, the value of these collected papers were identified through several stages, including content review, results comparison, and if necessary, consultation. Content review was used to determine the content of the papers related to a set of classifications, where it was applied to each of the collected papers. We assigned two reviewers to conduct the content review. Results comparison was used to compare the assessment outcome between both reviewers. If there was a disagreement in the values of the outcome, the two reviewers discussed their points of view until they reached a consensus. Whenever a consensus was not reached, a third reviewer was involved to take decision. The results from the entire content review process limited the number of relevant papers to 139 papers. The classification process was applied to this final set of papers. The list of these papers and their classification could be found at **`http://goo.gl/URKM3x`**.

3.2.2. Classification of Selected Study

Each of the collected studies was classified into several categories, with the aim of providing sets of results to address the research questions. In the following, we discuss the various dimensions which we used in the analysis of the papers.

Software Engineering Activities. We derived a list of software engineering activities by reviewing several classification systems and bodies of knowledge focusing only on software engineering. This includes the Computing Classification System [1], Graduate Software Engineering [30], IEEE Taxonomy classification [35], and SWEBOK [61]. We ended up with the following, perhaps widely-accepted, list:

- Requirements/Specification: A set of needs and demands of stakeholder(s) that should be fulfilled by the system. The development of requirements consists of elicitation, analysis, specification, and validation

- Design: Overall layout that describes implementation, concerned with architecture, tools, techniques, strategies, display, and pattern. It conforms to the earlier defined specification
- Coding: Process of translating requirements, specifications and designs into a set of standards arranged systematically using tools and techniques (including refactoring)
- Software/Program Verification: Evaluating product development through reviews, walkthroughs, inspections, and testing to ensure it meets the specifications and business requirements
- Distribution, Maintenance, and Enhancement: The administrative process of product deployment, feedback collection, maintenance supply and enhancement support
- Metrics: Quality measurement standards used to assess efficiency, functionality, usability, reliability, performance, availability, progress, design, and process of a product
- Management: Control over knowledge of an entire production process involving planning, organising, coordinating, communicating and monitoring

Social Computing Techniques. This dimension represents the techniques used to facilitate interaction. We adopted the types of techniques from [60] as following:

- Email/alert: intermediary tools to exchange information and/or notification, in order to build awareness amongst users
- Blog/microblog: intermediary applications to document information that is accessible by other users, thus allowing information exchange amongst users
- Wiki: web-based documentation of information developed collaboratively by a community of users
- Tag: a form of social bookmarking using a web-based interface that provides services to share and organise bookmarks collaboratively
- Feeds: they provide updates from particular sources, usually from web page(s), which is similar to the notification concept
- Social networking: it depicts the concept of virtual community made by social actors and a set of relationships that connects the actors
- Crowdsourcing: a virtual space occupied by online communities that provides an information exchange supported by the power of the crowds
- Instant messaging: real time exchange messages amongst online users over the internet
- Dashboard: a web-based panel that provides organised information in the most convenient way for the user
- General concept: applicable if the general concept of social techniques were described without specifying any particular techniques

Social Computing Design. This category identifies whether the social computing tool was expressly tailored for the software engineering activity or adapted from publicly available tools or a public one:

- Designated: specific social computing tools or applications that were originally designed to support certain software engineering activities
- Public: public social media, usually involving the World Wide Web and open for any user, e.g. blog, microblog and social network
- Adapted: adaptation of existing public tools into advanced designated tools in order to meet the needs for software engineering

Social Interaction Multiplicity. The multiplicity of social interaction is another dimension which we considered in our mapping study. Social interaction is one the following kinds:

- One-to-one: interaction that occurs between an individual user with another individual user, e.g. emails, and instant messaging
- One-to-many: interactions that occur between a user and a group of users where usually there is an individual user that acts as an intermediary, e.g. mailing list and announcement board
- Many-to-many: interaction that occurs amongst groups of users where the messages are broadcast, in order to achieve the fastest arrival at the person of interest, e.g. Wikis and Forums
- Mixture: a combination of social interactions that was facilitated by a tool or application where it can connect users directly, within, and amongst groups

Social Location. The use of social computing enables collaboration amongst developers within an organisation or across organisations and within a region or across regions. This category is to identify the location of the different stakeholders or teams in software development process who are utilising the social computing technique.

- Localised: when the social computing technique is used within the same regional location. Tools or applications implemented to mediate the collaboration are used to facilitate a smaller and centralised group

- Distributed: when the social computing technique is used to facilitate cross-region communication. In this category, tools or applications are implemented to mediate the collaboration and used across different locations, countries and even continents to support bigger and distributed groups
- Mixture: when the social computing technique is used both to support smaller centralised groups and to facilitate bigger distributed groups
- Not acknowledged: when social computing techniques are used in an environment, but they do not specify the scope in terms of the region. This means the location does not make a difference for the utilisation of the tool and the approach

Honeycomb framework. [39] described the Honeycomb framework as a guide to identify the elements attributed to social media. We used this framework to analyse the tools or applications that provide support for software engineering activities. The Honeycomb framework provides the following dimensions to describe a social medium:

- Identity: represents the ability given to users to identify themselves
- Conversations: represents the ability of users to communicate with each other
- Sharing: represents the ability of users to exchange, distribute and receive contents
- Presence: represents the ability of a user to know whether other users are available for interaction and allow others to know the same thing for the same user
- Relationship: represents the ability of users to connect and relate to each other
- Reputation: represents the ability of users to know the social standing of themselves and other users and build that reputation
- Groups: represents the ability of users to form communities

Types of Research. To classify the types of research followed in each paper, we adopted the classification suggested by [64] and we summarised it as follows:

- Evaluation: describes an investigation of an existing technique or standards (analyse, assess, and evaluate) in practice to acquire an understanding of a problem
- Validation: describes an investigation of a novel method or technique that has never been implemented in practice
- Solution: proposes new techniques to solve a problem where the technique itself can be either new or a significant extension of an existing technique. It is supported by examples and solid arguments
- Philosophical: describes the nature of background and knowledge research in presenting a new conceptual framework that implies a new point of view
- Opinion: describes the author's opinion, values, and preferences without introducing new research results, designs or any conceptual framework
- Experience: describes the author's personal experience in conducting a practice

Forms of Study. Empirical studies are those studies that derive data and insights based upon actual and objective observations or experimentations. We classified the forms on study based on [5] as following:

- Case study: Descriptive explanation of empirical inquiries performed in an in-depth study of a particular subject within its real-life context
- Laboratory study: A study using controlled conditions and variables to detect, identify, evaluate, and establish the nature of particular variables in order to investigate an intended result
- Observation: Investigational method conducted systematically to observe behaviour of certain objects without influencing or interfering with the object
- Experiment: Evaluation method to determine if changes applied to certain variables would affect changes in another variable. This study is generally applied to a conditioned environment to ensure its investigation quality
- Literature review: Informative report upon critical analysis of prior researched studies in a particular subject that describes, summarises, evaluates, classifies, compares, and clarifies
- Document analysis: Performing a content systematic examination of documents in order to acquire understanding in a particular subject
- Non-empirical: Describing the researcher's persuasive arguments without data validation obtained from human perception or judgement

Distribution of Researchers. This refers to the affiliation of the individuals who performed the study. It can be from either industry or academia. Thus, we also consider the combination of both implementers as mixed researchers.

Types of Publication. This category refers to the types of publication. We recognised that the collected studies are accredited through several means of publications. Thus, we categorise the types of publications as journals, conferences, workshops, academic technical reports, and doctoral symposiums.

Publication Year. Publication year refers to the year in which the paper was published. This is to identify the trend of researches in this particular topic.

The Origin of the Affiliation. The origin of the affiliation refers to the countries where the study implementers conducted their researches in their relation to certain affiliation. This is to identify the demographic spread of the interest on research in this particular topic.

3.2.3. Quality Assessment

To address an appropriate assessment, we identified the influential aspects of a study that indicate the reliability and trustworthy value. Results derived from the review of a study act as validation where it provides supports and evidence for the assessment. Thus, it can also build the standards required for further studies.

A quality assessment is conducted more often in a systematic literature review and less in a mapping study. However, while we do not use it in the mapping study, we prepared a set of questions inspired by [26] to assess the quality of our collected studies. The important quality dimensions for this study are that the papers are peer reviewed and appear in relevant event.

4. The Analysis and Results

This section presents the finding results and analyses on the categorisation from the existing studies, thus addressing each of the seven research questions (RQ) presented in Section 3, respectively.

4.1. What are the areas of software engineering assisted by the use of social computing?

The first research question is aimed to identify the areas of software engineering that benefited from social computing. Within each of these areas, a more detailed identification of the types of social computing techniques was also conducted. This was performed to investigate the distribution use of social computing techniques in software development projects, in order to assist collaborative work in each of the software engineering areas.

Figure 2 illustrates the descriptive statistics based upon the number of papers that applied or advocated the use of social computing on each of the areas of study in the domain of software engineering. We identified that most of the papers have more than one association with the types of social techniques. This many-to-many relationship applies also for the other statistics reported in the rest of the paper. This explains why the total in some figures and tables is above or below 139, the total number of papers reviewed.

As it can be observed, even though there are some differences in the amount of utilisation of social computing in each area, we can still say that the use of social computing is indeed emerging and started to be seen useful in supporting the whole software development life cycle. This is especially true in areas requiring heavy interaction amongst software engineers and other stakeholders, specifically, in management, coding, requirements and maintenance and evolution. The utilisation is less popular in areas such as metrics and verification probably due to the highly formal nature of those activities requiring limited amount of human communication and relying on standards and automated processes.

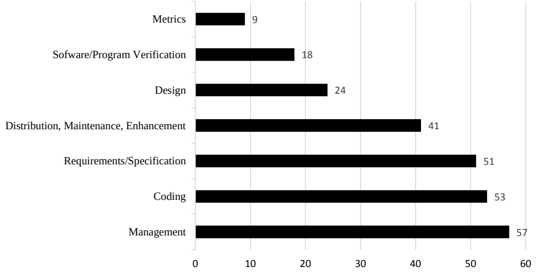


Figure 2: Areas of Software Engineering Supported by Social Computing

We also investigated the distribution of the use of social computing techniques within each area of software engineering. Table 1 charts the descriptive statistics of the relationship between the uses of social techniques and the various activities in software engineering.

The results showed that at least six out of ten different social computing techniques were applied or advocated to be helpful to each area of software engineering. The results showed an immense study of several social techniques, including general concept, instant messaging, email/alert, and wiki. It shows the degree of recent necessity for both synchronous and asynchronous communication. Although the high number of email/alert was anticipated, the results surprisingly showed a significant

	Coding	Design	Distribution, Maintenance, Enhancement	Management	Metrics	Requirements/ Specification	Sofware/Program Verification
Blog/Microblog	10	4	11	4	1	6	3
Crowdsourcing	10	1	4	2	2	5	3
Dashboard	5	4	4	8	0	6	1
Email/Alerts	12	4	11	15	1	9	4
Feed	1	0	2	1	0	1	1
General Concept	10	7	4	16	3	12	3
Instant Messaging	13	7	9	13	3	13	4
Social Networking	9	2	7	10	1	5	4
Tag	2	0	6	3	0	3	0
Wiki	6	5	9	12	2	10	0

Table 1: Social Techniques Vs. Software Engineering Activities

use of instant messaging and wikis as well. Some of these applications reused commercial social computing platforms which are publicly available. Others used expressly tailored tools or generic tools adapted to the specific software engineering task they were meant to facilitate.

4.2. *What are the types of research used to conduct the study?*

This research question is aimed to identify and analyse the types of researches used to conduct existing studies. The values were retrieved from the collected papers and were categorised based upon the types of research category where we also allowed a paper to be associated to more than one type of research. This explains why the accumulated number of papers in Figure 3 is larger than 139, the total number of papers reviewed. This applies also for other figures and tables. The figure illustrates the descriptive statistics of the distribution types of research used in the collected papers.

The results showed a significant difference between both solution and evaluation types on one hand and the other types of research on the other. This indicates a high degree of interest in the use of social computing to provide solutions for supporting software engineering, and the evaluation of the existing or proposed social tools to elevate their use to support software engineering.

On the other hand, we have found only 11 studies that were of the validation type. We refer to validation as the studies that validate previously evaluated approaches in practice. The low number of validation studies might be due to the fact that most of the existing studies are relatively new or works in progress. The majority of papers are published in either conferences or workshops as shown in Section 3.2. This is another indicator that the field is still growing and needs some extra time to get solid results.

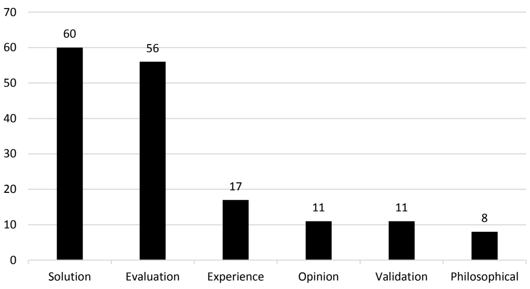


Figure 3: The Distribution of the Types of Research

4.3. *For any empirical study, what are the forms of study adopted to conduct the research?*

From the set of papers that we collected, regardless of the conducted types of research, the majority of papers were in the form of empirical study. Within these empirical studies, an analysis on the form of study was applied to identify the actual approaches used by the researchers. We also realised that the same study could be associated with more than one form of study, hence the total number exceeds 139, the total number of papers reviewed. Figure 4 illustrates the descriptive statistics of these identified forms based upon the analysis of the 139 collected papers. It also illustrates the number of non-empirical studies as a comparison criterion.

The results show that the majority of the empirical studies conducted an investigation by using case studies. Case studies are found to be a leading approach to examine the extent to which an idea would work in a real-life context. Moreover, the high number of the solution and evaluation types of research used within the collected papers (see Section 3.2), might also have influenced the high use of case studies as a form of study. Most of the solution and evaluation studies proposed and evaluated an approach, even with preliminary results and at an early stage, by the use of a case study.

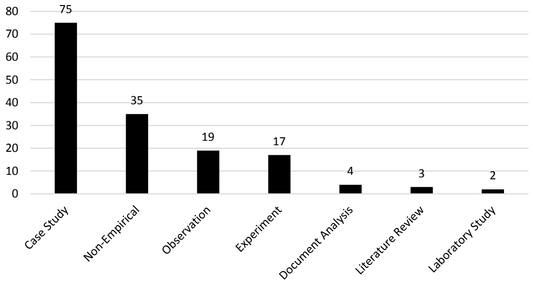


Figure 4: The Distribution of the Forms of Study

4.4. *What are the facets and aspects of social computing which were used to support software engineering and how?*

This research question provides the analysis and identification of the use of social techniques with their relation to social connectivity, interaction, and location.

4.4.1. Social Computing Techniques

We refer to the social techniques based upon our social computing techniques category (see Section 3.2). Values for this category were derived from the collected papers, and the analysis showed that a single paper could be associated with more than one social technique, hence the total number is above the total number of papers reviewed. Figure 5 provides the descriptive statistics retrieved as described in the collected studies to illustrate the distribution of the social techniques used to support software engineering.

The results show that the general concept, email/alerts, and instant messaging are the top three social computing techniques investigated by the collected studies. The high percentage of the general concept might indicate a general tendency to consider social computing useful for supporting software engineering. This has brought researchers to propose an accepted concept for utilising social computing, instead of pointing out certain more specialised social techniques.

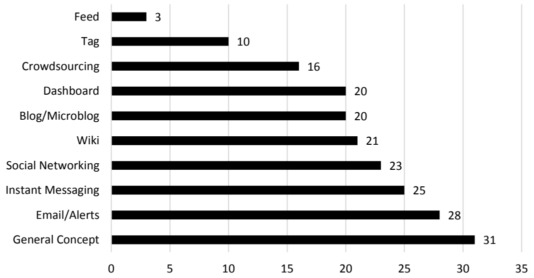


Figure 5: The Distribution of the Use of Social Techniques

On the other hand, more investigations on email were perhaps due to the fact that it is one of the earliest communication methods [19], wherein some researchers have debated its functionalities [49]. Furthermore, the use of instant messaging was highly discussed in the collected papers and this might be due to its flexibility and simplicity in supporting software engineering, particularly in distributed environments where real-time response is needed [48].

4.4.2. Social Computing Design

The aim of analysing the use of social computing techniques in relation to their design is to identify the built-in approaches in supporting software engineering. These approaches varied from designated, public, and adapted design (see Section 3.2), in which a paper from the collected studies could adopt more than one type of design. Figure 6 illustrates the descriptive statistics of the overall design approaches in the collected studies.

56% of the collected studies discussed the possibilities, implementations, and enhancements of designated tools to support software engineering. This demonstrates a high level of investigations into specialised tools, mostly examined by case studies (see Section 4.3), that could be used to support various areas of software engineering. The result also indicates a number of visible intentions of the investigation towards the possibilities of utilising and/or adapting public and general purpose tools to support software engineering.

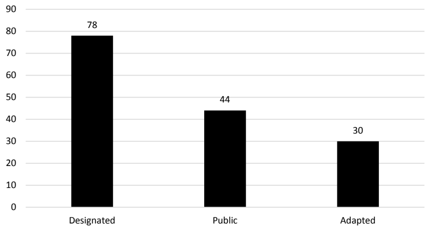


Figure 6: The Distribution of Social Computing Design

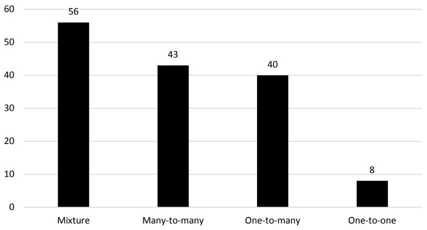


Figure 7: The Distribution of Social Interaction

	Adapted	Designated	Public
Blog/Microblog	7	5	9
Crowdsourcing	3	7	6
Dashboard	1	19	0
Email/Alerts	5	9	14
Feed	0	2	1
General Concept	7	25	8
Instant Messaging	5	10	11
Social Networking	4	9	10
Tag	3	7	2
Wiki	6	9	7

Table 2: The Design Styles of Social Techniques

	Many-to-many	Mixture	One-to-many	One-to-one
Blog/Microblog	4	10	8	1
Crowdsourcing	11	2	2	1
Dashboard	2	9	8	1
Email/Alerts	2	17	11	1
Feed	0	2	1	0
General Concept	8	18	3	4
Instant Messaging	1	19	5	0
Social Networking	11	12	2	0
Tag	3	3	7	0
Wiki	11	9	7	0

Table 3: Social Techniques Vs. Interaction Styles

The results in Table 2 show that dashboards and the general concept of software-mediated social interaction were highly investigated in the subject of designated tools. On the other hand, the use of email/alerts and instant messaging were highly investigated in the subject of public connectivity. Furthermore, the use of blog/microblog and wikis were highly discussed as the most adapted tools utilised to support software engineering. It might be due to the high potential of the blog/microblog and wikis to be adapted to support software engineering.

4.4.3. Multiplicity of Social Interaction

We categorised the social interaction multiplicity into four kinds (see Section 3.2). We derived the values from the collected papers. We identified that more than one interaction is possible to be associated with a paper. Figure 7 shows the descriptive statistics of the connections offered by social techniques related to the social interaction.

Of the collected studies, 40% of the papers described the use of social computing techniques to have a mixture of approaches. This might indicate that any single means of approach is considered insufficient. Another obtained result is the low number in the use of one-to-one interaction. This might indicate that the use of social computing within software development is more encouraging to an open group interaction instead of a direct one-to-one interaction.

The distribution of the use of social computing techniques related to social interaction is illustrated in Table 3. The results reveal the current application of social techniques related to their ability to bridge the interactions between stakeholders. They show that the mixture interaction has the highest application for both emails/alerts and general concept. The results also identify emails/alerts as the most utilised tool in facilitating a one-to-many interaction, and wikis for a many-to-many interaction.

4.4.4. Social Location

Analysis of the use of social computing techniques related to social location is to identify how social techniques facilitate collaboration by considering geographical distance. Recent popularity in global software de-

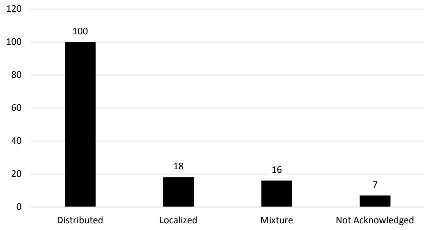
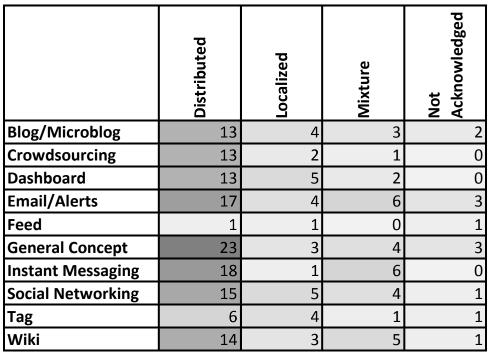


Figure 8: The Distribution of Social Location



	Distributed	Localized	Mixture	Not Acknowledged
Blog/Microblog	13	4	3	2
Crowdsourcing	13	2	1	0
Dashboard	13	5	2	0
Email/Alerts	17	4	6	3
Feed	1	1	0	1
General Concept	23	3	4	3
Instant Messaging	18	1	6	0
Social Networking	15	5	4	1
Tag	6	4	1	1
Wiki	14	3	5	1

Table 4: Social Techniques Vs. Social Location

velopment has strengthened the investigation of social techniques that could facilitate work collaboration regardless of the distance. This has encouraged the analyses for the category of social location (see Section 3.2). Figure 8 illustrates the descriptive statistics of how the social techniques facilitate collaboration based upon social location.

As shown, 72% of the studies described the use of social techniques to facilitate collaboration in a distributed environment. This percentage increases to 83% when combined with the percentage that has a mixture of locations. The high percentage of distributed locations indicates that most of the social computing techniques applications identify the importance of support for collaboration in a distributed environment. Furthermore, this result also confirms the change within the recent trend in software development method from localised environment to distributed environment.

The lower number of studies in localised locations might be due to less necessity for the localised environment to take advantage of social computing for collaboration. Localised environment enables the stakeholders to have a face-to-face meeting any time they feel the need to collaborate in the domain of software engineering. This is not a trivial observation as certain other domains would still require software-mediated interaction regardless of whether the users are in the same physical work environment or not. Still, several studies pointed out the benefits of utilising social computing even in a localised environment.

The result from Table 4 shows an overall distribution on the capability of social techniques to support collaboration based upon the geographical distance. Instant messaging and email/alert were pictured as the two highest social techniques used to facilitate collaboration in a distributed environment. A more interesting indication based upon the result is the high number of the general concept in supporting distributed location. This could mean that researchers have considered that the future environment for software development will occur in a distributed way; therefore, they have started to introduce the concept of social techniques to mediate this situation.

4.5. What are the research communities that conducted the research in the area and what are the characteristics of publications used to publish the studies?

Studies on the use of social computing to support software engineering have gained researchers' interest not only from academia, but also from industry and a combination of both affiliations. Researchers from academia, as shown in Figure 9, published 72% of the papers that we collected. The number rises to 84% when it is combined with the mixed affiliations. This number is calculated from the researchers that published the studies in the collected papers, regardless of the number of publications they made. The high number of publications authored by the academics in contrast with those of industry, indicates that the benefit of utilisation of social computing is still not widely recognised in industry.

We identified several types of publication in which the collected studies were published. Figure 10 illustrates the descriptive statistics of the distribution of the collected studies related to the methods of publication.

The results show that a significant number of publications were made through conferences (59%) and workshops (33%). A limited number of publications were found in journals (4%). This result indicates that research in this particular topic is still considered as new and emerging [43, 47]. Ideas and interests were still in the introduction and proposal phase through workshops, thus a more mature study towards research in this area was reported through conferences.

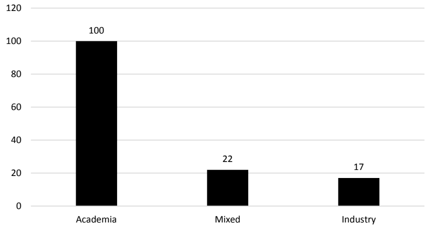


Figure 9: The Distribution of the Affiliations of Researchers

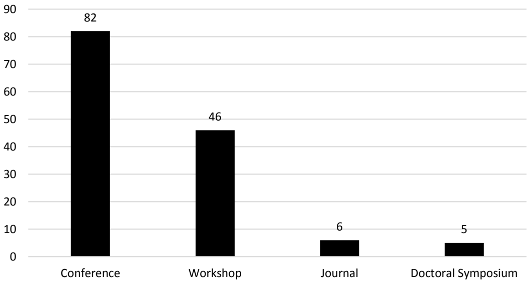


Figure 10: The Distribution of the Types of Publication

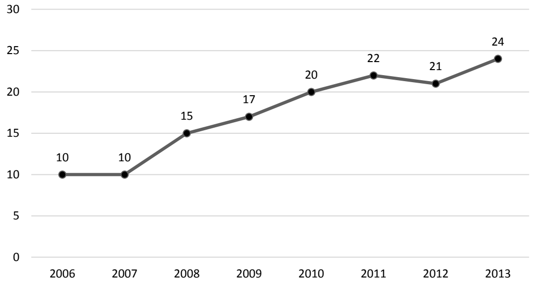


Figure 11: The Distribution of Papers Published Per Year

4.6. *What is the geographical and chronological distribution of the research in the area?*

4.6.1. *Paper Publication Per Year*

As we explained in Section 3, we started collecting and analysing the existing studies published between 2006 and 2013, included. We then started to analyse the results and generate the map. Thus, we believe that there might be other related studies published after that date until this study is completed. However, we analysed the number of published studies in the collected papers for each year as illustrated in Figure 11. The result shows a significant increase in publication in the period of 2006 to 2011. The field is gathering a constant and stable attention looking at the number of papers in 2012 and 2013.

Figure 12 shows the distribution of the published papers related to the types of publication. It illustrates an overall increase in the number of papers published in conferences and workshops, while papers published in doctoral symposiums and journals have maintained a low number. The appearance of studies published in doctoral symposiums might indicate a shift [62], improvement [22], or expansion [28] of studies in this topic, which might stimulate more studies in the future.

4.6.2. *Demographic Spread of Researchers*

Considering the novelty of this topic, we include the analysis of the location where the collected papers were investigated or published. We would like to identify which country makes a significant contribution and has a larger community based upon the total number of resided researchers. Therefore, we analysed two distributions related to the demographic spread of the papers and the demographic spread of the researchers. We

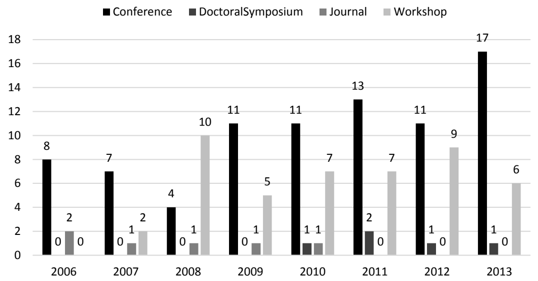


Figure 12: The Distribution of Published Papers Related to the Types of Publication

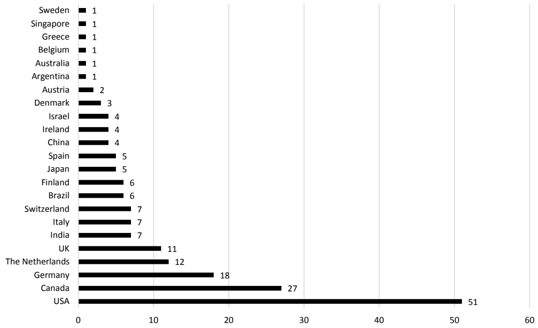


Figure 13: The Distribution of Published Papers Related to the Origin of Affiliation

identified the collected papers and we drew the connection of each paper to the countries based upon the affiliation of the researchers. This means that a paper can be connected to more than one country, only if the researchers came from different affiliations in different countries. Figure 13 illustrates the distribution of the analysis and shows that researchers based in the USA affiliations have contributed the largest amount of studies within the time period of our paper collection.

37% of the collected papers were investigated or co-investigated by researchers in the USA, followed by 19% and 13% of the collected papers by researchers from Canada and Germany, respectively; whereas the rest of the papers were fairly distributed amongst the rest of the participant countries. The results show that this particular topic is highly researched in America and

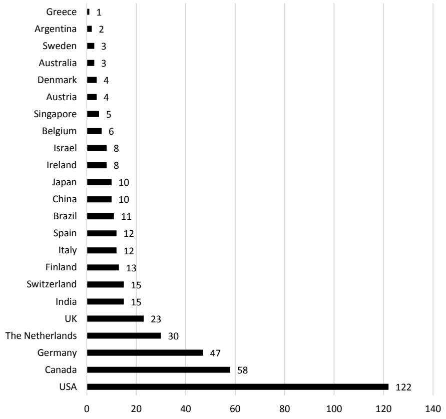


Figure 14: The Distribution of Researchers Based on Their Country of Affiliation

has started to gain popularity amongst several countries in Europe and Asia.

On the other hand, we also calculated the number of researchers that resided within each country. This number will indicate the population of researchers that show interest in this particular research topic, as illustrated in Figure 14. In comparison to the total authors that we identified in the collected papers, the results show 29% of the authors resided in the USA. It also shows a higher number of the total researchers that came from Canada with 13% compared to Germany with 11%.

4.6.3. Yearly Publication Related to Demographic Spread of Researchers

Earlier analysis showed that several countries have contributed highly in this particular topic. We select the top five countries to further analyse the papers published within these countries based upon the number of publications for each year, as seen in Figure 15. From these five countries, the results show that the USA mainly leads in the number of researches. It also shows that the USA was the only country that contributed not only highly but also consistently within each year.

4.7. What are the social computing blocks which are presented in the research and how?

Studies on this topic have introduced various emerging social aspects in social techniques. The Honeycomb framework [39] is intended to identify the social aspects embedded in a social computing technique (see Section 3.2). Figure 16 pictures the descriptive statistics of

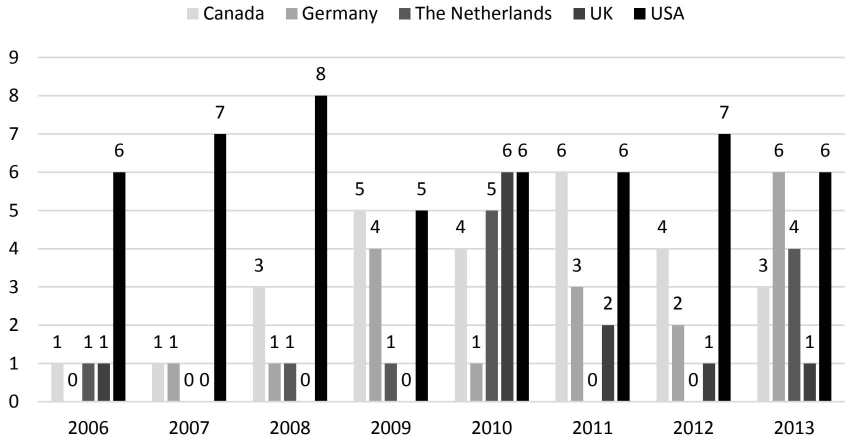


Figure 15: Published Papers Vs. the Origin of Affiliations

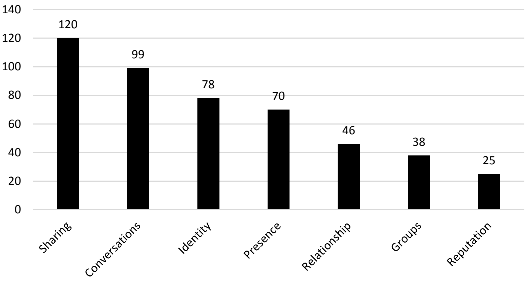


Figure 16: The Support of the Blocks of Honeycomb Framework

the social aspects in comparison with the Honeycomb framework.

The results show that several blocks (sharing, conversations, identity and presence) in the Honeycomb framework have been mainly supported within the social techniques used in software engineering. This might indicate that these blocks represent the main social aspects that are required for bridging collaboration amongst software development stakeholders. Other blocks (relationship, groups, reputation) seem to be less attractive for such work environments.

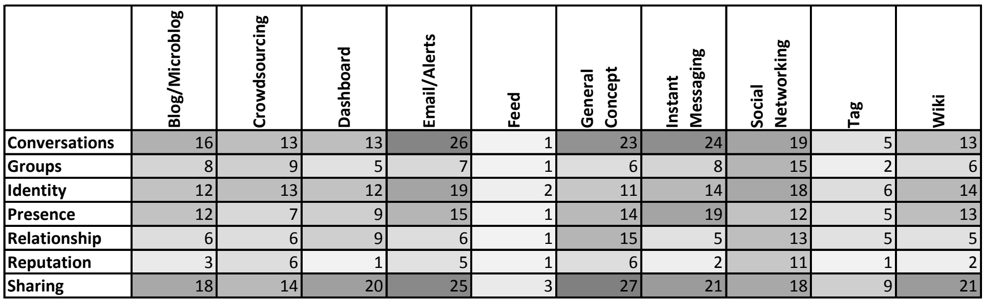
We then investigated how these blocks appear together with the social techniques in the collected studies. The result is shown in Table 5. The results show that emails/alerts and instant messaging were both highly represented in the conversations block. Emails/alerts also represent a higher portion of the sharing block. It also shows that there is a less significant representation of reputation and relationship blocks within social computing techniques, even the ones in the general concept. However, compared to other social techniques, social networking represents a fair distribution of the Honeycomb blocks.

5. Discussions

In this section, we go deeper in the analysis of our findings to answer our research questions presented in Section 3.1.1. We will also describe the limitations of this study.

5.1. RQ1: What are the areas of software engineering assisted by the use of social computing?

As illustrated in Figure 2, we identified the top four areas of software engineering supported by social com-



	Blog/Microblog	Crowdsourcing	Dashboard	Email/Alerts	Feed	General Concept	Instant Messaging	Social Networking	Tag	Wiki
Conversations	16	13	13	26	1	23	24	19	5	13
Groups	8	9	5	7	1	6	8	15	2	6
Identity	12	13	12	19	2	11	14	18	6	14
Presence	12	7	9	15	1	14	19	12	5	13
Relationship	6	6	9	6	1	15	5	13	5	5
Reputation	3	6	1	5	1	6	2	11	1	2
Sharing	18	14	20	25	3	27	21	18	9	21

Table 5: Honeycomb Framework Vs. Social Techniques

puting. It showed that the area of management has attracted most research into the use of social computing. We identified that social computing in the area of management is highly used in the form of project management tools [23]. Thus, it is used for managing tasks [38], schedules [7, 53], risk [11], knowledge [16], requirements [20], and dependencies with other teams or projects [33, 17].

On the other hand, in the areas of requirements/specification, social computing is mainly used for gathering feedback from stakeholders [55, 13]. It is also used for eliciting requirements [42, 57], communicating and notifying changes of requirements, resolving ambiguities and achieving shared understanding of the requirements [33].

In addition, in the area of coding, social computing is also used for sharing and discussing programming code [45], performing Q&A related to building code [63] and increasing awareness towards coding activities [56]. Thus, it helps to raise awareness of the bugs amongst the developers [54], assist responses towards bugs [52], and provide debugging history [56] in the area of distribution, maintenance, and enhancement.

The top areas that were supported by social computing showed an indication of a higher necessity for intense collaboration. This might be due to the position of these early stages that require more consideration, thus ensuring the success of the rest of the development project. Other than these top areas, the use of social computing is also indicated with every other area of software engineering.

However, the type and the degree of support required in each area are relative to its need for collaboration, as shown in Table 1. It illustrates the high degree of support for software engineering provided by social techniques, and it shows the areas that have the potential to be supported by certain social techniques.

Both areas of requirements/specification and coding were highly supported by synchronous communication methods (e.g. instant messaging), in which there exists a higher need for fast and direct communication. In order to collect requirements, and later translate them into specifications and then into code, frequent communication is required amongst the stakeholders. Moreover, both areas were also supported by asynchronous communication methods (e.g. email and wiki) to keep record of the interaction, and as a result enabling information tracking in the future. These would allow software to be built effectively with high efficiency.

The area of management requires both synchronous and asynchronous communication. Mainly, synchronous communication is used for tactical decisions during the development while asynchronous is used for strategic decisions and those requiring reviews. Asynchronous is also used for purposes related to documenting the process of decision making and the decisions made as an outcome of it. Techniques like tags, feeds, blogs and crowdsourcing do not seem to be attractive to management probably for reasons related to the type of information required of the need of more controlled communication at this development stage

However, a more interesting result was also shown in Table 1. Together with email, instant messaging and wiki were both highly used in providing support for software engineering. As shown in Section 4.4.2, these techniques were not only used by their public form [22, 6] but they were also adaptable [27, 58] and highly designated [15, 38] for supporting software engineering.

5.2. *RQ2: What are the types of research used to conduct the study?*

Based upon the descriptive statistics result that addresses the second research question (see Section 3.1.1), 43% of the studies proposed a solution (either new or improved tools) to facilitate collaboration amongst stakeholders by utilising social computing techniques.

This shows a high interest from researchers in building tools that could mediate collaboration in software development. It also illustrates a high interest in an in-depth investigation of existing tools. Furthermore, it strengthens the assumption that research in software engineering is problem driven and empirically performed [21].

However, a high number of papers relating to solution and evaluation, together with a low number in validation papers, show an indication that research in this particular topic is new and emerging indeed. We could not also find more validation papers that confirmed the implementation of a technique used to support software engineering.

Thus, we observe that various proposed tools, utilising different types of social techniques for different software engineering activities, were consolidated and evaluated only recently. This is supported by the fact that most of the studies were still published in conferences or workshops while few started to appear in journals (see Section 4.5).

5.3. RQ3: For any empirical study, what are the forms of study adopted to conduct the research?

As described earlier, studies in software engineering were mostly solution driven and were performed empirically, confirmed by a 75% of empirical studies from the total studies of the collected papers (see Section 4.3). Amongst these studies, we presented an analysis that identified a 54% use of case study. This supports the previous analysis on the high use of solution and evaluation types of research as discussed in Section 4.2.

Most researchers conducted a case study to evaluate the implementation of the tool or application that they proposed in a real-world situation. This might be due to the reason that case studies provide the necessary environment to perform an in-depth investigation that includes the analysis of the interpersonal relationships within and amongst groups using the proposed tool or application.

A wide range of different case studies were conducted, from small [22] to large [10] companies or from single [37] to multiple teams [18]. These cases were used to understand the interactions, behaviours, and patterns that appear amongst people with social techniques, the detailed relationship amongst them, and the main factors that could support team development in an effective and collaborative way. It might be also due to the nature of case studies that delivers real results without any interference from the researchers.

Furthermore, considering their high percentage, researchers might view case studies as preliminary studies to investigate complex issues related to the research subject, in which the results were to be used in further advanced research or even perhaps supporting validation research.

5.4. RQ4: What are the facets and aspects of social computing which were used to support software engineering and how?

Analysis results from Section 4.4.1 show that the use of a traditional means of communication is still dominating the social techniques. Email is still highly utilised even when other social techniques have emerged and shown their capability to provide similar support. However, based upon the result, the general concept shows a higher number than email. This might indicate the necessity for a new breed of social techniques that could provide advanced support for software engineering.

The need for this new breed is confirmed by the analysis of the social technique design. Results of Section 4.4.2 show a high number of researches in designated tool or application for supporting software engineering. 56% of the studies suggested and described the use of designated techniques. This shows that facilities used for collaboration in software engineering require tools or applications that were specifically built for this purpose, even though results also show several utilisation of the public and the customised tools or applications.

Furthermore, we also analysed how the design styles were distributed across the areas of software engineering (see Figure 17). Both areas of requirements/specification and management showed the highest advocating the use of designated tools or applications. This might indicate that these areas require different designated approaches, relative to the type of the development environment. The use of social computing does not appear to be compelling for the areas of metrics and software verification. This is probably due to the relatively focused and centralised interaction these areas require.

Several business-related factors have recently caused extreme changes in the environment of software development; from it being concentrated in a single location to it being distributed geographically [56]. This has added the challenge for coordinating work across sites with the consideration of nationality, language, and cultural barriers. Therefore, the need to provide tools or applications that can facilitate distributed communication and coordination is called for.

Both results from Sections 4.4.3 and 4.4.4 identify the range of interactions and locations that the social techniques could support. Results from the analysis of

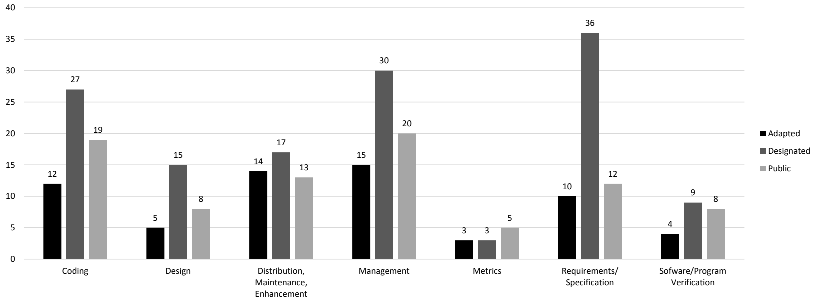


Figure 17: The Distribution of Social Connectivity Related to Areas of Study

social interaction show that most of the research suggested a mixture of interactions. This indicates that a single approach of interaction is insufficient to facilitate collaboration in software engineering. The mixture of interactions might provide more choices to enhance collaboration with the most suitable approach available.

However, an interesting result was shown through the low number of one-to-one interactions. This might suggest that current software development has encouraged group interaction instead of direct interaction. We identified that even when a development process requires direct relationship amongst stakeholders, current collaboration enables this relationship through a means of clustered information exchange that is also available for other stakeholders [9, 54] thus confirming the group interaction.

Moreover, results from the analysis of social location show that a significant 72% of researchers endorsed the use of social computing in a distributed environment. Amongst these, the various social techniques offered different facilitation in supporting distributed collaboration (shown in Table 4). This result has confirmed the current trend that is occurring in software engineering shifting towards development in a distributed environment.

In addition, the results also show that the highest number of investigations (22%) on the use of social computing is conducted through the general concept. This might be due to the attempt to identify and propose a suitable approach for future facilitation through social computing in supporting software engineering. This has also strengthened the support of software development in a distributed environment.

5.5. RQ5: What are the research communities that conducted the research in the area and what are the characteristics of publications used to publish the studies?

The growth of the software industry has always been compatible with the growth of academics in computing [24]. This implies that there is cooperation between academics and industries in bringing numerous advancements of software into life. However, the analysis of the communities that conducted the collected studies (see Section 4.5) indicates a significant difference of interest between academics and industrial researchers. This might be due to the novelty of this particular research topic, which may not yet have fully caught the interest of industrial researchers.

Furthermore, many industries have not yet recognised the approaches and benefits offered by social aspects to be emerged in daily collaboration [39]. Some would even assume that the use of social computing might obstruct work performance [3].

Furthermore, the lack on researchers from industrial community might also be due to the limited use of tools or applications that utilises social computing in supporting real world software engineering. However, we identified the importance of the industrial sector in becoming a part of the study especially in providing a supporting environment for case studies (see Section 4.3) and validation researches (see Section 4.2).

In Section 4.5 we also presented the distribution of research publications, which shows that most of the collected studies were published through conferences. A significant percentage of 59% supports the assumption that studies in this particular area are still considered new and emerging, yet starting to gain maturity.

Moreover, we also identified some researches that were published by academics in the doctoral symposium (see Figure 10 and Figure 12). We assume that these initial studies might lead to a shift or branching of study within this particular area, where their appearance has caught the interest of researchers in academia. Thus, it might enrich the completeness of studies in the future.

5.6. *RQ6: What is the geographical and chronological distribution of the research in the area?*

Results show a constant and growing interest of the researchers, indicated by the increasing number of papers published from 2007 to 2011 and then relatively stable number in 2012 and 2013. The number of papers in 2011, 2012 and 2013 is around 20 papers per year which would be a medium size (see Figure 11). We observed that certain research groups have become specialised in this area and are committed to take the field further. A good portion of the papers published each year is authored by members of those groups. The groups are mainly based in the USA, Canada, Germany, the Netherlands and the UK (see Figures 13 and 14). It also appears that a community of researchers working on the area has been formed and it is growing. This includes the community of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE) and the International Workshop on Social Software Engineering (SSE).

We also presented the analysis that includes the top five countries that contributed the most published papers. From 2006 to 2013, we identified that the USA and Canada contributions were not only the highest, but also the USA contribution was the most consistent within each year (see Figure 15). We also identified the interest from Germany starting from 2007, when their contribution began to increase. On the other hand, the Netherlands also showed a strong leap in publishing research papers in 2009 - 2010, although it was less consistent in the annual contribution.

In spite of the growing interest, there is still a limited number of journal publications in the area (see Figure 12). This is perhaps an indicator that the field is still in the consolidation stage, especially if we observe the relatively high number of conference papers (some in Tier A conferences). Also, the fact that doctoral theses are being developed around the topic is an indicator that it would continue growing in the future.

5.7. *RQ7: What are the social computing blocks which are presented in the research and how?*

In the studies we collected, the main objective of utilising social computing is to provide facilitation for work collaboration in software engineering. However, finding a suitable combination of social techniques to facilitate particular areas of software engineering is challenging. It is necessary to understand which type of social relationship is required by a particular area of software engineering. Furthermore, it is also important to identify the social aspects within a social technique to understand the fundamental features offered by the social technique.

We presented the descriptive statistics of the social aspects, based upon the Honeycomb framework, that were investigated regarding their use within software engineering (see Section 4.7). The analysis result showed that current social aspects used for assisting collaboration were still relying on the basic needs for communication, which is sharing and conversation. However, the aspects of presence and identity both appeared as emerging needs. This confirms the assumption that every effective software team requires intense communication, coordination, and awareness to work effectively and efficiently, in order to build a good quality product. Thus, the social aspects as described in the results could provide the approaches for facilitating these needs.

5.8. *Limitations of the Study*

We realise that there are limitations in our mapping study.

- We put a date restriction for the collection of published literatures which we considered would be sufficient to identify the early research within this topic. Still, it is possible that there are other studies beyond this date that were not included. However, the relatively high number of studies we included, 139 papers, should be sufficient to draw a picture, which is generalisable enough, of the current state of research. This is especially true given that the use of social computing in a business context, including software engineering, is relatively new and it is unlikely that we have missed much by including studies published starting from 2006.
- We have only presented a set of criteria which classify the research on social computing for software engineering. There could be other criteria which would help a wider understanding of the domain, such as the kind of applications used for the case studies and the limitations of those studies. However, this requires an in-depth analysis of the literature which requires a systematic literature review rather than a mapping study.

- The papers were selected and reviewed by two researchers and a third referee was involved when there was a disagreement between the two researchers. A fourth reviewer made an additional check to ensure accuracy in the reporting. Obviously, the background of the researchers affects the entire process. This is especially true for papers which were in the grey area with regards to some of the criteria. We believe that this is almost the case with every empirical study. We tried to minimise the effect by having a thorough discussion and looking iteratively at the literature which defines the criteria.

6. Conclusions

In this paper, we have explored the research on using social computing for software engineering. That is, on how social computing may be used, or is suggested, to enhance various aspects of the software engineering process, and ultimately bring gains to software engineering practice.

Hence, our initial aim was to draw a picture of the current state of the research in this emerging area, to help researchers and developers identify what has been established so far, to understand those areas which have seen particular emphasis and similarly, what aspects or areas are still under-researched or warrant greater scrutiny. To this end, we conducted a systematic mapping study, which reviewed a total of 139 papers, having selected and analysed them following a protocol designed according to good practice in the empirical research field [50], [40], [5], and [46].

Our findings show a reasonably constant level of attention and a growing level of maturity in the last few years. Importantly, we also observed that the advanced features of social computing are still not fully explored within the context of software engineering. A relatively large amount of the papers are visionary and position papers, which could be seen as an indicator of the novelty, timeliness and attractive nature of this field. We found that there is an adoption of the general (social computing) concept in some papers but without a discrete specification of how it is used, i.e. what social techniques are best to use and for what software engineering activity. We also noted that there is a tendency to design bespoke social computing to fit software engineering and be independent from the commercial social computing platforms, which were not expressly tailored to the peculiarities of software engineering. We also analysed the demographics of the papers and noticed a good degree of interest in social computing for software engineering from industry.

As a future work in this direction, we intend to investigate, in depth, more specific questions about this field. For example, we will explore the range of domains of applications used for the case studies and see whether the use of social computing appears to be more appropriate to certain domains and what drives that decision. Similar analysis will be undertaken with respect to the structure of the development team and the characteristics of the members. Another important question to be answered is whether the proposed usage of social computing for software engineering would indeed lead to a better quality artefacts and what management procedure to apply to maximise this possibility. Hence, we hope, by such analyses, to help to shed light on the most promising directions for future research in social computing for software engineering, so that our efforts to use social computing to bring enhancements to the software engineering process are directed most fruitfully.

Acknowledgement

The research was supported by an FP7 Marie Curie CIG grant (the SOCIAD Project), and by Bournemouth University through the Fusion Investment Fund (the BBB, the BUUU and the VolaComp projects).

References

[1] ACM (Association for Computing Machinery), 1998. The 1998 ACM Computing Classification System. http://www.acm.org/about/class/ccs98-html#D.1.0

[2] Ahmadi, N., Jazayeri, M., Lelli, F., and Nesic, S., 2008. A Survey of Social Software Engineering. In: ASE Workshop of the International Conference on Automated Software Engineering, 15 - 16 September 2008.

[3] Ali-Hassan, H., Nevo, D., Kim, H., and Perelgut, S., 2011. Organisational Social Computing and Employee Job Performance: The Knowledge Access Route. In: 44th Hawaii International Conference on System Sciences 4 - 7 January 2011.

[4] Babar, M. A., 2008. The Application of Knowledge-Sharing Workspace Paradigm for Software Architecture Processes. In: Proceedings of the 3rd International Workshop on Sharing and Reusing Architectural Knowledge (SHARK), Leipzig, Germany, 10 - 18 May 2008.

[5] Bailey, J., Budgen, D., Turner, M., Kitchenham, B., Brereton, P., and Linkman, S., 2007. Evidence relating to Object-Oriented Software Design: A Survey. In: 1st International Symposium on Empirical Software Engineering and Measurement, 20 - 21 September 2007.

[6] Bach, P. M., Jiang, H., and Carroll, J. M., 2008. Sharing Usability Information in Interactive System Development. In: Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), Leipzig, Germany, 13 May 2008.

[7] Begel, A. and DeLine, R., 2009. Codebook: Social Networking over Code?. In: Companion Volume of 31st International Conference on Software Engineering (ICSE), Vancouver, Canada, 10 - 18 May 2009.

[8] Begel, A., DeLine, R., and Zimmermann, T., 2010. Social Media for Software Engineering. In: Proceeding of the Workshop on Future of Software Engineering Research (FoSER) on the 18th ACM SIGSOFT International Symposium on Foundations of Software Engineering, Santa Fe, USA, 7 - 11 November 2010.

[9] Biehl, J. T., Czerwinski, M., Smith, G., and Robertson, G. G., 2007. FASTDash: A Visual Dashboard for Fostering Awareness in Software Teams. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 2007, California, USA, 28 April - 3 May 2007.

[10] Bjarnason, E., Wnuk, K., and Regnell, B., 2011. Requirements are Slipping Through the Gaps - A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development. In: Proceedings of the 19th IEEE International Requirements Engineering Conference, Trento, Italy, 29 August - 2 September 2011.

[11] Blagojevic, V., Codenie, W., Dedecker, J., Gonzales-Deleito, N., Deleu, J., and Boucart, N., 2009. Murphy: A Web 2.0 Approach for Proactive Risk Management in Hardware/Software Co-Design. In: Companion volume of 31st International Conference on Software Engineering (ICSE), Vancouver, Canada, 10 - 18 May 2009.

[12] Calefato, F. and Lanubile, F., 2012. Augmenting Social Awareness in a Collaborative Development Environment. In: Proceedings of 5th International Workshop on Co-operative and Human Aspects of Software Engineering (CHASE), Zurich, Switzerland, 2 June 2012.

[13] Careno, L. V. G. and Winbladh, K., 2013. Analysis of User Comments: An Approach for Software Requirements Evolution. In: Proceedings of the 35th International Conference on Software Engineering (ICSE), San Fransisco, USA, 2 June 2012.

[14] Cataldo, M. and Herbsleb, J. D.. Communication Patterns in Geographically Distributed Software Development and Engineers' Contributions to the Development Effort. In: Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), Leipzig, Germany, 13 May 2008.

[15] Cataldo, M., Shelton, C., Choi, Y., Huang, Y., Ramesh, V., Saini, D., and Wang, L., 2009. CAMEL: A Tool for Collaborative Distributed Software Design. In: Proceedings in the 4th IEEE International Conference on Global Software Engineering (ICGSE), Limerick, Ireland, 13 - 16 July 2009.

[16] Clerc, V., 2008. Towards Architectural Knowledge Management Practices for Global Software Development. In: Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), Leipzig, Germany, 13 May 2008.

[17] Dabbish, L., Stuart, C., Tsay, J., and Herbsleb, J. D., 2012. Social Coding in GitHub Transparency and Collaboration in an Open Software Repository. In: Proceedings of the 2012 ACM Conference on Computer Supported Cooperative Work (CSCW), Seattle, USA, 11 - 15 February 2012.

[18] Damian, D., Lanubile, F., and Mallardo, T., 2008. On the Need for Mixed Media in Distributed Requirements Negotiations IEEE Transaction on Software Engineering, 43 (1), 116 - 132.

[19] Damian, D., Marczak, S., Dascalu, M., Heiss, M., and Liche, A., 2009. Using a Real-Time Conferencing Tool in Distributed Collaboration: An Experience Report from Siemens IT Solutions and Services. In: Proceedings in the 4th IEEE International Conference on Global Software Engineering (ICGSE), Limerick, Ireland, 13 - 16 July 2009.

[20] Damian, D., Marczak, S., and Kwan, I., 2007. Collaboration Patterns and the Impact of Distance on Awareness in Requirements-Centred Social Networks. In: Proceedings of the 15th IEEE International Requirements Engineering Conference (RE), New Delhi, India, 15 - 19 October 2007.

[21] Day, R. E., 2007. Kling and the "Critical": Social Informatics and Critical Informatics: Research Articles. Journal of the American Society for Information Science and Technology, 58 (4), 575 - 582.

[22] Dittrich, Y. and Giuffrida, R., 2011. Exploring the Role of Instant Messaging in a Global Software Development Project. In: Proceedings of the 6th IEEE International Conference on Global Software Engineering (ICGSE), Helsinki, Finland, 15 - 18 August 2011.

[23] Donker, H. and Blumberg, M., 2008. Collaborative Process Management and Virtual Teams. In: Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), Leipzig, Germany, 13 May 2008.

[24] Ellis, H. J. C., Moreno, A. M., Mead, N. R., and Seidman, S. B., 2003. Industry/University Software Engineering Collaborations for the Successful Reeducation of Non-Software Professionals. In: 16th Conference on Software Engineering Education and Training, 20-22 March.

[25] Erickson, T., Kellog, W. A., Laff, M., Sussman, J., Wolf, T. V., Halverson, C. A., and Edwards, D., 2006. A Persistent Chat Space for Work Groups: The Design, Evaluation and Deployment of Loops. In: Proceedings of the 6th Conference on Designing Interactive Systems (DIS), University Park, PA, USA, 26 - 28 June 2006.

[26] Fernandez, A., Insfran, E., and Abrahao, S., 2011. Usability Evaluation Methods for the Web: A Systematic Mapping Study. Information and Software Technology, 53 (8), 789 - 817.

[27] Fitzpatrick, G., Marshall, P. and Philips, A., 2006. CVS Integration with Notification and Chat: Lightweight Software Team Collaboration. In: Proceedings of the 2006 ACM Conference on Computer Supported Cooperative Work (CSCW), Canada, 4 - 8 November 2006.

[28] Forster, S., 2013. Investigating the Collaborative Process of Process Modeling. In: Proceeding of the 25th International Conference on Advanced Information Systems Engineering (CAiSE) Doctoral Consortium Valencia, Spain (CEUR Workshop Proceeding), 20 June 2013.

[29] Garrido, P. J., Vizcaino, A., Andrada, J., Monasor, M. J., and Piattini, M., 2012. DPMTool: A Tool for Decisions Management in Distributed Software Projects. In: Proceedings in the 7th International Conference on Global Software Engineering (ICGSE) Workshops, Porto Alegre, Brazil, 27 - 30 August 2012.

[30] Graduate Software Engineering, 2009. Curriculum Guidelines for Graduate Degree Programs in Software Engineering. Integrated Software & Systems Engineering Curriculum (iSSEc) Project. `http://www.gswe2009.org/fileadmin/files/GSwE2009_Curriculum_Docs/GSwE2009_version_1.0.pdf`

[31] Grudin, J. (1994). Computer-supported cooperative work: History and focus. Computer, 27(5), 19-26.

[32] Hattori, L. and Lanza, M., 2009. An Environment for Synchronous Software Development. In: Proceedings of the 31st International Conference on Software Engineering (ICSE) Companion Volume , Vancouver, Canada, 16 - 24 May 2009.

[33] Herbsleb, J. D., 2007. Global Software Engineering: The Future of Socio-Technical Coordination. In: Future of Software Engineering (FOSE), 23 - 25 May 2007.

[34] Hazzan, O., & Tomayko, J. (2004). Human aspects of software

engineering: The case of extreme programming. In Extreme Programming and Agile Processes in Software Engineering (pp. 303-311). Springer Berlin Heidelberg.

[35] IEEE, 2013. IEEE Computer Society Taxonomy Structure. IEEE Computer Society.

[36] Jakobsen, M. R., Fernandez, R., Czerwinski, M., Inkpen, K., Kulyk, O., and Robertson, G. G., 2009. WIPDash: Work Item and People Dashboard for Software Development Teams. In: Proceeding of the 12th IFIP TC 13 International Conference on Human-Computer Interaction (INTERACT), Uppsala, Sweden, 24 - 28 August 2009.

[37] Johri, A., 2011. Look Ma, No Email!: Blogs and IRC as Primary and Preferred Communication Tools in A Distributed Firm. In: Proceedings of the 2011 ACM Conference on Computer Supported Cooperative Work (CSCW) Hangzhou, China, 19 - 23 March 2011.

[38] Jorgensen, N. M., Hale, M. L., and Gamble, R. F., 2011. SEREBRO: Facilitating Student Project Team Collaboration. In: Proceedings of the 33rd International Conference on Software Engineering (ICSE), Waikiki, Honolulu, USA, 21 - 28 May 2011.

[39] Kietzmann, J. H., Hermkens, K., McCarthy, I. P., and Silvestre, B. S., 2011. Social Media? Get Serious! Understanding the Functional Building Blocks of Social Media. Business Horizons, 53 (3), 241 - 251.

[40] Kitchenham, B., 2007. Guidelines for Performing Systematic Literature Reviews in Software Engineering. United Kingdom. EBSE Technical Report EBSE-2007-01

[41] Knauss, E., Brill, O., Kitzmann, I., and Flohr, T., 2009. SmartWiki: Support for High-Quality Requirements Engineering in a Collaborative Setting. In: Proceedings in the 31st International Conference on Software Engineering (ICSE) Workshop on Wikis for Software Engineering (WIKIS4SE), Vancouver, BC, Canada, 19 May 2009.

[42] Kukreja, N., 2012. Winbook: A Social Networking Based Framework for Collaborative Requirements Elicitation and WinWin Negotiations. In: Proceedings of the 34th International Conference on Software Engineering (ICSE), Zurich, Switzerland, 2 - 9 June 2012.

[43] Kwan, I. and Damian, D., 2011. The Hidden Experts in Software-Engineering Communication (NIER Track). In: Proceedings of 33rd International Conference on Software Engineering (ICSE), Waikiki, Honolulu, USA, 21 - 28 May 2011.

[44] Lanubile, F., 2009. Collaboration in Distributed Software Development. In: Software Engineering Journal: International Summer Schools, 5413, 174 - 193.

[45] Lotufo, R. A., Korbes, A., Machado, R. C., and Ramos, R. G., 2009. Adessowiki On-line Collaborative Scientific Programming Platform. In: Proceedings of the 5th International Symposium on Wikis and Open Collaboration (WikiSym), Orlando, FL, USA, 25 - 27 October 2009.

[46] Mujtaba, S., Petersen, K., Feldt, R., Mattsson, M., 2008. Software Product Line Variability: A Systematic Mapping Study. In: 15th Asia Pacific Software Engineering Conference (ASPEC), August 2008.

[47] Murphy-Hill, E., 2012. Continuous Social Screencasting to Facilitate Software Tool Discovery. In: Proceedings of the 34th International Conference on Software Engineering (ICSE), Zurich, Switzerland, 2 - 9 June 2012.

[48] Niinimaki, T. and Lassenius, C., 2008. Experiences of Instant Messaging in Global Software Development Projects: A Multiple Case Study. In: Proceedings in the 3rd IEEE International Conference on Global Software Engineering (ICGSE), Bangalore, India, 17 - 20 August 2008.

[49] Pagano, D. and Bruegge, B., 2013. User Involvement in Software Evolution Practice: A Case Study. In: Proceedings of the 35th International Conference on Software Engineering (ICSE), San Fransisco, USA, 2 June 2012.

[50] Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M., 2008. Systematic Mapping Studies in Software Engineering. In: 12th International Conference on Evaluation and Assessment in Software Engineering, 26 - 27 June 2008.

[51] Petticrew, M. and Roberts, H., 2006. Systematic Reviews in the Social Sciences: A Practical Guide.

[52] Rigby, P. C. and Storey, M., 2011. Understanding Broadcast Based Peer Review on Open Source Software Projects. In: Proceedings of 33rd International Conference on Software Engineering (ICSE), Waikiki, Honolulu, USA, 21 - 28 May 2011.

[53] Roy, B., Graham, T. C. N., and Gutwin, C., 2012. DiscoTech: A Plug-In Toolkit to Improve Handling of Disconnection and Reconnection in Real-Time Groupware. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW), Seattle, USA, 11 - 15 February 2012.

[54] Sarma, A., Maccherone, L., Wagstrom, P. A., and Herbsleb, J. D., 2009. Tesseract: Interactive Visual Exploration of Socio-Technical Relationships in Software Development. In: Proceedings of the 31st International Conference on Software Engineering (ICSE), Vancouver, Canada, 16 - 24 May 2009.

[55] Savio, D., Creighton, O., Anitha, P. C., and Patil, A., 2012. Visualizing Requirements in Distributed System Development. In: Proceedings of the 2nd IEEE International Workshop on Requirements Engineering for Systems, Services, and Systems-of-Systems (RESS), Chicago, IL, USA, 25 September 2012.

[56] Sengupta, B., Chandra, S., and Sinha, V., 2006. A Research Agenda for Distributed Software Development. In: Proceedings of the 28th International Conference on Software Engineering (ICSE), Shanghai, China, 20 - 28 May 2006.

[57] Seyff, N., Graf, F., and Maiden, N., 2010. End-User Requirements Blogging with iRequire. In: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE), Cape Town, Africa, 2 - 8 May 2010.

[58] Singer, L. and Schneider, K., 2011. Supporting the Cooperation of End-User Programmers through Social Development Environments. In: Proceedings of the 2nd International Workshop on Web 2.0 for Software Engineering (Web2SE), Waikiki, Honolulu, Hawaii, 21 - 28 May 2011.

[59] Singer, L. and Schneider, K., 2012. Influencing the Adoption of Software Engineering Methods Using Social Software. In: Proceedings of the 34th International Conference on Software Engineering (ICSE), Zurich, Switzerland, 2 - 9 June 2012.

[60] Storey, M., Treude, C., Deursen, A., and Cheng, L., 2010. The Impact of Social Media on Software Engineering Practices and Tools. In: FSE/SDP Workshop on Future of Software Engineering Research 7 - 11 November 2010.

[61] SWEBOK, ca. 2013. SWEBOK V3 Review. Available from: http://computer.centraldesktop.com/swebokv3review/ [Accessed 19 October 2013]

[62] Todoran, I., 2012. StakeCloud: Stakeholder Requirements Communication and Resource Identification in the Cloud. In: Proceedings of the 20th IEEE International Requirements Engineering Conference (RE), Chicago, IL, USA, 24 - 28 September 2012.

[63] Treude, C., Barzilay, O., Storey, M., 2011. How Do Programmers Ask and Answer Questions on the Web?. In: Proceedings of 33rd International Conference on Software Engineering (ICSE), Waikiki, Honolulu, USA, 21 - 28 May 2011.

[64] Wieringa, R., Maiden, N., Mead, N., and Rolland, C., 2006. Requirements Engineering Paper Classification and Evaluation Criteria: A Proposal and a Discussion. Requirements Engineering, 11(1), 102-107.