

Building Trust through Social Awareness: The SocialCDE Project

Fabio Calefato, Filippo Lanubile
Università degli Studi di Bari Aldo Moro
Dipartimento di Informatica, Via Orabona 4, 70125 Bari (BA)
{fabio.calefato, filippo.lanubile}@uniba.it

Abstract. *Trust is paramount in distributed software development to prevent geographically distributed sites to feel and act like distinct, distant teams. Nevertheless, how to build trust among developers with few or no chances to meet is an open issue. To overcome such a challenge, we hypothesize that increased social awareness may foster trust building in global software teams. Here, we first present SocialCDE, a tool that aims at augmenting Application Lifecycle Management (ALM) platforms with social awareness to facilitate the establishment of interpersonal connections by disclosing developers' personal interests and contextual information. Then, we present two different empirical studies, specifically designed to test our hypothesis.*

Keywords: Trust Building, Distributed Software Development, Social Awareness, Empirical Studies, Human Factors, Social Computing.

1. Introduzione

Benché importante per qualsiasi tipo di team, il trust (fiducia) è un fattore che contribuisce notevolmente al successo o al fallimento di progetti su scala globale [Herbsleb et al., 2005]. Infatti, in tali contesti distribuiti, il trust è fondamentale per evitare che la distanza fisica porti anche a quella psicologica. E' stato osservato che un basso livello di fiducia (a) aggrava la percezione di essere team separati con obiettivi conflittuali, (b) diminuisce la disponibilità a condividere informazioni e cooperare per risolvere i problemi, e (c) influenza negativamente la benevolenza verso gli altri in caso di obiezioni e disaccordi [Al-Ani et al., 2011]. Il trust tra i componenti di un team si sviluppa di solito attraverso una stretta interazione e comunicazione faccia-a-faccia (F2F) poiché essa rappresenta il modo più efficace per stabilire relazioni con gli altri e acquisire consapevolezza (awareness). Purtroppo, l'interazione F2F è proprio l'attività maggiormente penalizzata della distanza nel global software development. Altresì, le compagnie di software sono diventate sempre più distribuite negli ultimi anni, nonostante il fatto che nessuna risposta sia stata fornita alla seguente domanda: *Come si può stabilire o rafforzare il trust tra i*

componenti di team distribuiti su scala globale quando essi hanno scarsa o nessuna possibilità di incontrarsi?

Nel tentativo di fornire una risposta, abbiamo ipotizzato che la condivisione degli interessi personali dei componenti di un team all'interno di uno spazio di lavoro condiviso possa facilitare la creazione di connessioni interpersonali, aumentare la probabilità di interazioni di successo, e contribuire a stabilire fiducia tra i componenti di un global software team. In altre parole, ci aspettiamo che la possibilità di avere accesso alle informazioni condivise sui social media, così come la possibilità di monitorare l'attività dei propri collaboratori sui social network possano funzionare come un surrogato della interazione sociale che avviene naturalmente durante incontri informali faccia a faccia, aumentando così la fiducia reciproca attraverso l'incremento di consapevolezza (awareness). Il concetto di awareness (consapevolezza) proviene dal campo del CSCW e ha avuto una notevole influenza nella campo della ricerca dell'ingegneria del software in quanto fornisce meccanismi per coordinare le attività di gruppo [Mistrík et al., 2010].

Pertanto, abbiamo sviluppato SocialCDE, un tool che estende le piattaforme di Application Lifecycle Management (ALM), aggiungendo il supporto alla social awareness. Nella parte restante di questo articolo, prima descriviamo lo strumento: dopodiché, illustriamo due studi empirici preliminari, al fine di rispondere alla precedente domanda di ricerca.

2. Stato dell'Arte

2.1 Trust

Il trust è un fenomeno complesso da studiare in quanto coinvolge sia i rapporti interpersonali, ossia i problemi culturali tra fiduciario (trustor) e fiduciante (trustee), sia le sfaccettature del comportamento umano, ossia, i tratti personali del carattere. A oggi sono state date diverse definizioni di trust. Una definizione ampiamente utilizzata e concisa è fornita da Jarvenpaa et al. [Jarvenpaa et al., 1998], che definiscono il trust come le aspettative del trustor che il trustee si comporterà come previsto. In altre parole, la fiducia positiva emerge quando le azioni degli altri incontrano la nostra aspettativa; in caso contrario, si genera sfiducia. Altre definizioni di fiducia distinguono tra dimensione *cognitiva* (*cognitive trust*) e *affettiva* (*affective trust*). Per esempio, Wilson et al. [Wilson et al., 2006] definiscono la fiducia cognitiva in termini di aspettative circa la competenza e l'affidabilità degli altri nello svolgimento di azioni importanti che il trustor non può monitorare. Al contrario, la fiducia affettiva si riferisce ai legami emotivi reciproci, le preoccupazioni e l'affetto tra il trustor e il trustee, che spinge quest'ultimo a fare qualcosa per il primo per dovere morale.

Diversi sono stati gli approcci proposti per favorire la crescita della fiducia. L'idea di fondo è che il processo di costruzione della fiducia si sviluppa lungo diverse dimensioni chiamate antecedenti (*trust antecedents*) [Jarvenpaa et al., 1998], cioè le proprietà del trustee che scatenano la valutazione da parte

del trustor nel valutare l'affidabilità degli altri. Secondo Jarvenpaa et al. [Jarvenpaa et al., 1998], l'*abilità* (e.g., competenze, conoscenze), la *benevolenza* (e.g., la cortesia, la disponibilità), l'*integrità* (e.g., la fedeltà, l'adesione alle norme morali), e la *prevedibilità* (e.g., l'affidabilità, comportamenti coerenti) sono le caratteristiche personali di un trustee che facilitano la costituzione del rapporto di fiducia con un trustor. In particolare, le dimensioni di *capacità* e la *prevedibilità* sono valutate per mezzo di un'elaborazione cognitiva delle informazioni personali e professionali. Allo stesso tempo, la valutazione affettiva porta alla costruzione della fiducia lungo le dimensioni di benevolenza e integrità. Infine, per un trustor, è invece la *propensione a fidarsi* il tratto personale del carattere che conta.

2.2 Awareness

La consapevolezza di gruppo, o *group awareness*, è stata definita da Dourish e Bellotti, come "*la comprensione delle attività degli altri, per fornire un contesto alla propria attività*" [Dourish e Bellotti, 1992].

Secondo Gutwin et al. [Gutwin et al., 1996], oltre che riguardo ai propri collaboratori, tipicamente i componenti di un gruppo cercano informazioni anche sulle attività e gli artefatti. Di conseguenza, finora quattro tipi di group awareness sono stati identificati, vale a dire: la *informal o presence awareness* (cioè, chi è online e la loro disponibilità), la *group-structural awareness* (vale a dire, i ruoli di ciascun componente e la struttura interna dei team), la *workspace awareness* (cioè, chi ha modificato un artefatto condiviso e quando), e la *social awareness* (cioè, l'informazione e la comprensione che compagni hanno circa loro connessioni sociali all'interno di un gruppo [Omoronya et al., 2010]).

3. Modello di Ricerca e Ipotesi

Jarvenpaa e Leidner hanno osservato l'evoluzione del trust nei team distribuiti su scala globale in cui l'interazione avviene esclusivamente attraverso la comunicazione mediata da computer [Jarvenpaa e Leidner, 1999]. La loro analisi ha indicato che i team con un basso livello di trust iniziale erano quelli carenti a livello di comunicazione sociale all'inizio del progetto. Al contrario, i team che manifestavano un elevato livello di fiducia al termine del progetto, si focalizzavano inizialmente sulla comunicazione "social", che in seguito diminuiva per dare spazio all'interazione focalizzata sulle attività da compiere e le procedure da seguire. Coerentemente con questa osservazione, noi ipotizziamo che la divulgazione delle informazioni personali, contestualizzate nello spazio di lavoro condiviso, possa aumentare la percezione di similarità e vicinanza tra i componenti di team distribuiti, favorendo in tal modo la quantità di comunicazione sociale scambiata, la manifestazione di un numero più elevato di interazioni di successo, e nuovi legami più forti stabiliti tra individui distanti. Infine, la ricerca da Jarvenpaa et al. [Jarvenpaa et al., 1998] ha dimostrato come la percezione dell'altrui integrità, benevolenza, e propensione a fidarsi percepita sono i trust antecedenti rilevanti per favorire la costruzione di fiducia affettiva. Pertanto, ipotizziamo che (vedi Figura 1):

H1 - Esiste una correlazione positiva tra il livello di social awareness acquisita attraverso i social media e il livello di fiducia affettiva reciprocamente stabilito fra team distribuiti.

Per verificare questa ipotesi attraverso un esperimento, abbiamo bisogno di misurare i livelli di fiducia affettiva e cognitiva. Dal momento che la fiducia è una *sensazione percepita* dagli individui, dobbiamo fare affidamento su dati riportati dai partecipanti all'esperimento.

Treinen e Miller-Frost [Treinen e Miller-Frost, 2006] hanno osservato che lo sviluppo di fiducia reciproca tra siti distanti all'inizio di un progetto è fondamentale per la sua riuscita. Infatti, essi hanno osservato che, durante la fase iniziale di un progetto, la conoscenza personale tra i componenti di un team e la fiducia reciproca sono più importanti della risoluzione dei primi problemi tecnici, dal momento che la fiducia permette di risolvere i problemi in seguito, a distanza (e.g., attraverso conference call), ottenendo così una maggiore efficienza complessiva per l'intero progetto. Pertanto, ipotizziamo che (Figura 1):

H2 - Esiste una correlazione positiva tra il livello di fiducia affettiva reciprocamente stabilita tra team distribuiti e la performance del progetto.

Per quanto riguarda il test di questa ipotesi, riconosciamo che stabilire un rapporto di causa / effetto tra la fiducia e le prestazioni di un progetto sia un compito arduo, giacché molti altri confounding factor – per esempio, il tipo di progetto, le competenze individuali – possono interferire lungo il processo sperimentale.

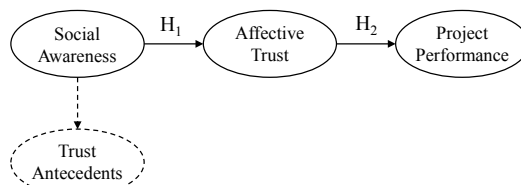


Fig.1 – Il modello di ricerca proposto.

4. Il Progetto SocialCDE

Per Application Lifecycle Management (ALM) si intende un processo continuo di gestione della del ciclo di vita di un progetto attraverso piattaforme che forniscono uno spazio di lavoro condiviso e integrato con un set di strumenti di supporto a tutte le attività di sviluppo software, come per esempio la gestione dei requisiti, la progettazione, la codifica, il testing e gestione del rilascio [Calefato e Lanubile, 2012]. Le piattaforme di ALM più popolari, note anche come ambienti di sviluppo collaborativo (CDE), non supportano affatto la social awareness, oppure, quando disponibile, come nel caso di Jazz e GitHub, il livello di supporto offerto non è sufficiente [Lanubile et al., 2013].

Per far fronte a questa limitazione, abbiamo sviluppato SocialCDE [Calefato e Lanubile, 2013], uno strumento che estende le piattaforme ALM Microsoft Team Foundation Server (TFS) e GitHub rendendo disponibili le informazioni raccolte dalla maggior parte dei più diffusi social network oggi

disponibili (ad esempio, Twitter, LinkedIn). Dal lato client, due sono i plugin disponibili, uno estende Visual Studio, l'altro supporta qualsiasi IDE basato su Eclipse. La Figura 2 mostra i componenti principali del progetto SocialCDE: i plugin gestiscono la visualizzazione di tutti i contenuti sociali raccolte presso i servizi abilitati dall'utente (Figura 3).

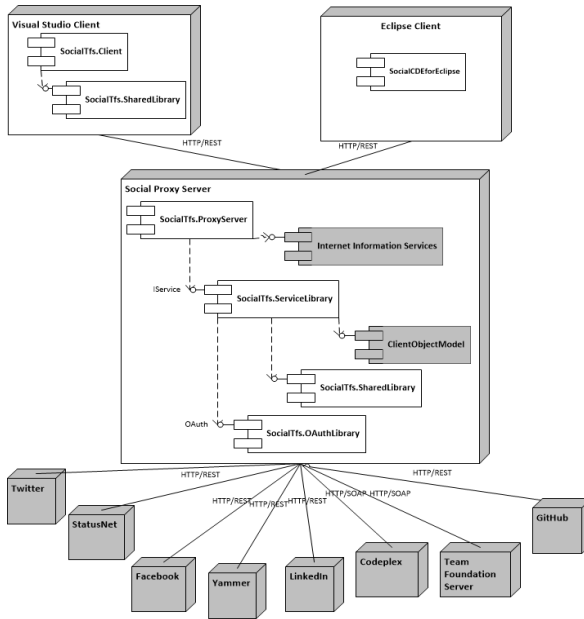


Fig. 2 – Diagramma di deployment delle componenti del progetto SocialCDE.

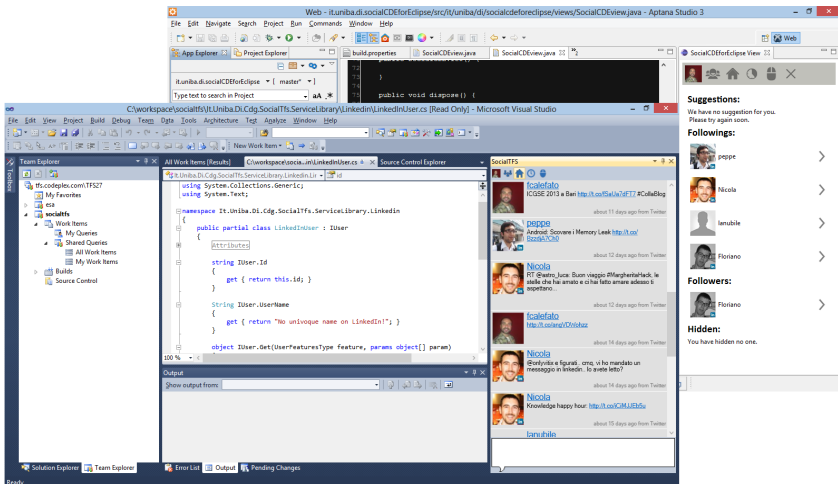


Fig.3 – I plugin di SocialCDE per Visual Studio (sx) e Aptana (dx).

L'altra componente principale, il Social Proxy Server, è un aggregatore che sfrutta le API di microblog aziendali e siti di social networking. Essendo un proxy, tale componente interagisce attraverso protocolli RESTful sia con i plugin lato client sia con le piattaforme ALM. I suoi compiti principali sono il recupero di informazioni riguardo agli account utenti sui servizi di social network (SNS), e ai progetti software dalle piattaforme ALM. Il proxy server ottiene i permessi di accesso attraverso il protocollo OAuth ed effettua il caching dei post, consentendo all'utente di personalizzare le informazioni che è possibile recuperare dai propri account. Inoltre, il proxy può gestire e memorizzare i dati di connessione a GitHub, TFS e CodePlex.

Come accennato prima, il contenuto recuperato dai SNS è caricato e memorizzato nella cache del Social Proxy Server. Quindi, tutte le informazioni richieste da un client sono presentate all'utente finale attraverso una vista all'interno dell'IDE. In particolare, tali informazioni sono mostrate attraverso tre prospettive diverse (dette *timeline*), ossia la *home*, la *iteration*, e la *interactive*. La home timeline replica le timeline disponibili in siti di microblogging come Twitter o Yammer, in quanto popolata dai post dell'utente e di coloro che lo "seguono" (follower). Noi definiamo *statico* questo tipo di follower, perché per ricevere i messaggi di un utente è esplicitamente richiesta l'azione di seguirlo/a, cioè, aggiungerlo/a alla propria *rete di consapevolezza* (o *awareness network*, cioè l'insieme di colleghi le cui azioni sono monitorate da un utente e ai quali sono mostrate le sue stesse azioni). Tuttavia, le *awareness network* sono fluide [de Souza e Redmiles, 2011]: le persone le cui azioni si dovrebbero monitorare e a cui le proprie azioni dovrebbero essere mostrate cambiano nel tempo, a seconda (a) delle attività correnti o (b) della fase di sviluppo del software. Pertanto, oltre la visualizzazione del flusso dei follower statici mostrati nella home timeline, abbiamo anche progettato un tipo dinamico di follower e altre due timeline. A differenza di quelli statici, i follower dinamici non richiedono alcuna azione di follow esplicito, e vengono aggiunti e rimossi automaticamente dalla awareness network quando: i) si verificano dei cambiamenti riguardo ai task di un utente nella iterazione corrente; ii) un artefatto nel repository del progetto è aperto nell'editor dell'IDE. Nel primo caso, la *iteration timeline* aggiunge dinamicamente alla awareness network di un utente ogni componente del team che ha riportato o anche commentato un workitem assegnatogli/le. Nel secondo caso, la *interactive timeline* visualizza dinamicamente i messaggi di chi ha contribuito modifiche alla risorsa aperte e visualizzata al momento.

5. Gli Studi

In questa sezione presentiamo il design di due studi empirici progettati per l'indagine preliminare della nostra ipotesi di ricerca. Il primo (Studio A) è un field study, attualmente in corso, che coinvolge un solo team di sviluppo. Il secondo (studio B) è un esperimento controllato, che si prevede abbia inizio entro la fine dell'estate.

5.1 Studio A

Questo studio coinvolge un piccolo team di sviluppo di 7 componenti che lavorano ad "I Speak Again", un progetto promosso dalla onlus Informatici Senza Frontiere (ISF). Abbiamo selezionato questo progetto perché si adatta ai seguenti requisiti socio-tecnici. In primo luogo, il team è distribuito e non costituito da molto tempo. In particolare, i tre responsabili del progetto sono gli unici che si conoscono da uno o più anni e hanno la possibilità di incontrarsi di persona. Invece, i restanti quattro membri sono completamente distribuiti e hanno iniziato a lavorare per il progetto quando lo studio ha avuto inizio. In secondo luogo, il progetto utilizza GitHub come piattaforma ALM e Aptana come IDE. Infine, i membri del team hanno accettato di connettere uno o più account di social network a SocialCDE.

Poiché alcuni dei soggetti sperimentali coinvolti hanno già lavorato insieme per qualche tempo prima che l'esperimento avesse inizio, abbiamo adottato il seguente design sperimentale di tipo A-B-A. Come mostrato in Figura 4, la fase iniziale A è quella in cui si trovava il team prima di iniziare l'esperimento, quando i componenti sono stati addestrati all'uso del plugin. Inoltre, uno dei team leader ha partecipato a una sessione di demo per dare sostegno agli altri sviluppatori. Prima di iniziare l'esperimento, tutti i soggetti hanno risposto a un questionario preliminare, che si proponeva di raccogliere alcune informazioni personali (e.g., età, sesso, esperienza di lavoro, istruzione), così come misurare la loro propensione a fidarsi degli altri e la quantità di trust stabilità con gli altri componenti del team. Tali misure sono calcolate indirettamente, attraverso gli antecedenti della fiducia.

Al momento, l'esperimento è in fase B, ossia, il team ha iniziato le pratiche di sviluppo agile e le attività collaborative, utilizzando l'IDE Aptana esteso con il plugin di SocialCDE. Questa fase durerà da quattro a sei settimane, anche se imporre scadenze rigide non è fattibile a causa della natura volontaria del progetto. Durante questa fase, i membri del team condurranno regolarmente le loro attività lavorative, usando in più il plugin per leggere e condividere contenuti. I dati di utilizzo saranno automaticamente raccolti dal componente proxy (e.g., il numero di posti condivisi, variazioni nell'awareness network). Alla fine della fase B, un secondo questionario sarà somministrato ai partecipanti, con l'obiettivo di individuare differenze nei livelli di fiducia percepita rispetto agli altri. Infine, durante la terza e ultima fase A, la squadra tornerà a lavorare con l'IDE senza il plugin, come nella fase iniziale. Dopo una settimana di lavoro senza il supporto del nostro tool, uno dei ricercatori condurrà delle interviste semi-strutturate individuali (tramite Skype o faccia a faccia), che saranno registrate e trascritte. Inoltre, sarà anche organizzata una riunione finale con i tre team leader.

Data la natura di field study, saremo in grado di ottenere da questo studio solo alcuni dati per testare l'ipotesi H_1 sulla relazione tra social awareness e fiducia affettiva.

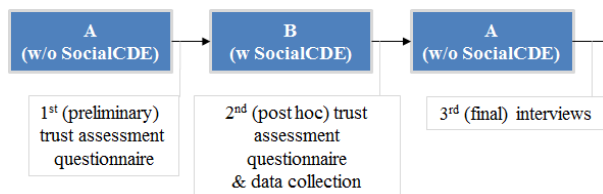


Fig. 3 – Design sperimentale del field study “Studio A”.

5.2 Studio B

E' stato realizzato anche il design di un esperimento controllato che, rispetto allo studio precedente, confronta due progetti differenti, realizzati da due diversi team distribuiti di 5/6 membri. Entrambi i team applicano la metodologia di sviluppo agile SCRUM e realizzeranno un vero prodotto software per un cliente reale. La composizione dei team è la seguente: 3 sviluppatori (stagisti per una grande multinazionale presso PUCRS Università di Porto Alegre, Brasile), 1 team leader (docente), e 1 o 2 product owner (clienti). Come nel caso dello Studio A, anche questo esperimento richiede che i team siano distribuiti e formati di recente. Tuttavia, in questo caso sarà utilizzato TFS come piattaforma ALM e Visual Studio come IDE.

Poiché che saranno disponibili due diversi progetti, useremo un progetto (A), come gruppo di controllo e l'altro progetto (B) come gruppo di trattamento (Figura 5). In particolare, durante la prima fase, ossia, prima di iniziare la fase di sviluppo, il team del progetto B assisterà a una lezione introduttiva che illustrerà come utilizzare il plugin. La lezione di formazione si concluderà con la somministrazione dello stesso questionario pre-esperimento usato nell'altro studio. Dopodiché, durante la seconda fase, il team di progetto A utilizzerà Visual Studio così com'è, mentre il team di progetto B utilizzerà l' IDE esteso con il plugin di SocialCDE per tutta la durata dell'attività di sviluppo (circa 6-8 settimane). Durante questa fase di sviluppo, i membri del team di entrambi i progetti dovranno svolgere le loro attività di lavoro regolarmente. Le uniche attività extra svolte dal team di progetto B attraverso l'uso del plugin riguarderanno la lettura e la condivisione di contenuto sociale con gli altri membri del team. Un secondo questionario post-hoc verrà somministrato al termine di questa fase, ma solo per il gruppo di trattamento. Infine, durante la terza ed ultima fase, al termine della attività di sviluppo per entrambi i progetti, saranno condotte e trascritte interviste semi-strutturate al fine di eseguire analisi qualitativa.

Per questo studio controllato saremo in grado di valutare gli effetti di SocialCDE durante l'intera fase di sviluppo del progetto di test, per poi confrontarli con il progetto di controllo. Pertanto, otterremo dati utili per testare sia H_1 sia H_2 . Tuttavia, rispetto alla seconda ipotesi, considerate il numero limitato di soggetti sperimentali coinvolti, ci sarà necessità di condurre ulteriori repliche dell'esperimento per confermare eventuali differenze riscontrate in termini di performance tra i due progetti.

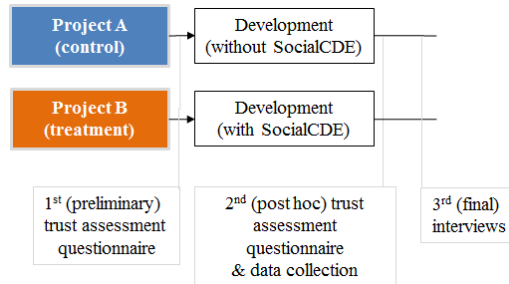


Fig. 5 – Design sperimentale dell’esperimento controllato “Studio B”.

6. Conclusioni e Sviluppi Futuri

In questo articolo, abbiamo descritto SocialCDE, un tool che ha l’obiettivo di estendere le piattaforme ALM attraverso l’aggregazione di contenuto da diversi social media e la condivisione all’interno di uno spazio di lavoro condiviso. SocialCDE è stato progettato per supportare la nostra ipotesi che le informazioni condivise sulle reti sociali possano costituire un surrogato della social awareness ottenuta durante le interazione informali, utile per generare fiducia tra i componenti di software team distribuiti. Infine, abbiamo anche presentato il design sperimentale di due studi preliminari volti a valutare sperimentalmente la nostra ipotesi.

Ringraziamenti

Si ringraziano Nicola Sanitate, Giuseppe Santoro, e Floriano Fauzzi per il loro contributo allo sviluppo del progetto. SocialCDE è parzialmente finanziato dal progetto Intersocial, nell’ambito del Programma di Cooperazione Transfrontaliera Grecia-Italia 2007-2013, e da Microsoft Research attraverso il premio Software Engineering Innovation Foundation (SEIF) Award 2011.

Bibliografia

Al-Ani, B., Wilensky, H., Redmiles, D., and Simmons, E. An Understanding of the Role of Trust in Knowledge Seeking and Acceptance Practices in Distributed Development Teams, Proc. 6th Int’l Conf. Global Soft. Eng. (ICGSE ’11), Helsinki, Finland, Aug. 15-18, 2011, pp. 25-34.

Calefato F. and Lanubile F. Practice: Collaborative Development Environments, in Global Software and IT: A Guide to Distributed Development, Outsourcing, and Supplier Management (C. Ebert ed.), IEEE, Wiley, 2012.

Calefato F. and F. Lanubile F. SocialCDE: A Social Awareness Tool for Global Software Teams. ESEC/FSE 2013, St. Petersburg, Russia.,18-26 Aug. 2013.

de Souza C.R.B. and Redmiles, D.F. The Awareness Network, To Whom Should I Display My Actions? And, Whose Actions Should I Monitor? IEEE Trans. on Sw Eng, 37, 3, pp. 325-340, 2011.

Dourish P. and Bellotti, V. Awareness and Coordination in Shared Workspaces. Proc. Int'l Conf. on Computer Supported Cooperative Work (CSCW '92), Toronto, Canada, Nov 1-4, 1992, pp. 107-114.

Gutwin, C., Greenberg, S., and Roseman, M. Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation. Proc. HCI (HCI'96), London, Aug. 20-23, 1996, pp. 281-298.

Herbsleb, J.D. et al. Global Software Development at Siemens: Experience from Nine Projects. Int'l Conf. Sw Eng (ICSE'05), St Louis, USA, May 15-21, 2005, pp. 524-533.

Jarvenpaa, S.L., Knoll, K., and Leidner, D.E. Is anybody out there? Antecedents of trust in global virtual teams. *Journal of Management Information Systems*, 14(4), pp. 29-64, 1998.

Jarvenpa S.L. and Leidner D.E. Communication and Trust in Global Virtual Teams. *Journal of Organization Science*, 10(6), pp. 791-815, Nov—Dec. 1999.

Lanubile, F., Calefato, F. and C. Ebert, C. Group Awareness in Global Software Eng., *IEEE Softw.*, Mar. 2013, pp.18-23.

Mistrić, I., Grundy, J., van der Hoek, A., and Whitehead, J. Collaborative Software Engineering: Challenges and Prospects. *Collaborative Software Eng.*, 2010, pp. 389–403.

Omoronyia, I., Ferguson, J., Roper, M., and Wood M. A review of awareness in distributed collaborative software engineering. *Software: Practice e Experience*, 40, pp. 1107–1133, 2010.

Treinen J.J. and Miller-Frost S.L. Following the sun: case studies in global software development, *IBM Syst. J.*, 45(4), Oct. 2006, pp. 773-783.

Wilson, J.M., Strausb, S.G, and McEvily, B. All in due time: The development of trust in computer-mediated and face-to-face teams, *Organizational Behavior and Human Decision Processes*, 99(1), Jan. 2006, pp. 16–33.