

Chapter 75

E–Mentoring in Global Software Development Teams: Success Factors to Develop a Common Culture

Ricardo Colomo-Palacios
Universidad Carlos III de Madrid, Spain

Cristina Casado-Lumbreras
Universidad de Complutense de Madrid, Spain

Alok Mishra
Atilim University, Turkey

Pedro Soto-Acosta
Universidad de Murcia, Spain

ABSTRACT

Global Software Development (GSD) teams face communication and coordination problems due to spatial, temporal, and cultural separation between team members. Cultural diversity and cross-cultural management are significant issues among GSD teams. In software development projects, mentoring dramatically reduces the learning curve for novice human resources. Due to the large amount of electronic communication instruments, a remarkable number of different e-Mentoring concepts have emerged, which provides opportunity for mentoring that would not otherwise be possible. This chapter presents key success factors to enable e-Mentoring as a tool to develop a common culture in GSD scenarios. These success factors enable the correct application of mentoring programmes and the use of this to build a common culture in organizations that perform GSD.

INTRODUCTION

Globalization is a business fact, expanded worldwide beyond domestic boundaries, that is creating an interconnected world economy in which companies do their business and compete with each other anywhere in the world, regardless of

national boundaries (Cullen, 1999). Globalization of the world economies has successfully brought significant changes to nearly all industries, and in particular, it includes software development (Smite, et al., 2010). One effect of economy globalization takes place in software development, where software products are developed by teams geographically distributed on a worldwide scale (Yu & Mishra, 2010). Moreover, according to

DOI: 10.4018/978-1-4666-4301-7.ch075

Jaakkola (2009), one of the trends that is changing the characteristics of the software industry very strongly is globalization. In this environment, the software organization, once a close-knit and typically collocated entity, now requires remote collaboration across great distances and often has a multinational and multicultural makeup (Milewski, 2007). As a result of this, software development is considered a globally sourced commodity (Hayes, 2002; Herbsleb, 2007). Jaakkola (2009) stated that globalization has meaningful consequences for the software industry:

- Software companies are establishing branch offices abroad;
- Software companies are offshoring their processes, outsourcing their work, and subcontracting abroad;
- Ownership of companies is becoming global.

These factors require the easy flow of a workforce over geographical borders and also meaningful changes in software organizations, which are becoming multicultural (Jaakkola, 2009).

As a result of these trends, software development is evolving from a single site development to multiple localization team environments (Hernández-López, et al., 2010). Thus, software development evolved in order to adopt some Globalization characteristics; as a result, a new field called Global Software Development (GSD) emerged to cover specific aspects of global distributed software development (Oshri, Kotlarsky, & Willcocks, 2007). The challenges relate to aspects such as economical, technical, organizational, and cultural issues to those arising from different time zones, languages, and geographical locations (Damian & Moitra, 2006a). However, according to Smite et al. (2010), at the same time as development goes global there is an ambition to minimize geographical, temporal, and cultural separation.

This chapter focuses on cultural separation in GSD teams. Moreover, this work presents the success factors to build a common culture by means of the application of e-mentoring in GSD scenarios.

GLOBAL SOFTWARE DEVELOPMENT

GSD involves the development of application software through interactions of people, organizations, and technology across nations with different backgrounds, languages, and working styles (Herbsleb & Mockus, 2003). In the early 1990s, offshoring of software work to development centers in low wage countries pertained to large Western companies such as IBM and SAP who systematically attempted to take a hold of wage differences and resources of a global market (Winkler, Dibbern, & Heinzl, 2008). Now, many Fortune 500 companies produce their business information systems in developing countries (such as China and India) to take advantage of their relatively low-cost labor (Sakthivel, 2007) and large telecommunications and software companies have numerous software development groups around the world (Edwards & Sridhar, 2005). In this scenario, software development outsourcing is an integral part of software development projects (Schümmer & Lukosch, 2009). Moreover, Information systems outsourcing is an indispensable tool in the management of information systems (Leeney, et al., 2011). The nineteen ninety's saw the emergence of the three I's (Ireland, India, and Israel) as the principle locations for significant globally distributed software development to take place (Ashish & Gambardella, 2005). More recently, Eastern Europe, Latin America, and the Far East have been pointed out as feasible and successful GSD locations (Geer, 2006).

In any case, this modern business strategy for developing high quality software in low-wage countries at low cost (Khan, Niazi, & Ahmad, 2011) has also been named offshore software

14 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/mentoring-global-software-development-teams/77770

Related Content

SaaS Requirements Engineering for Agile Development

Asif Qumer Gilland Deborah Bunker (2014). *Software Design and Development: Concepts, Methodologies, Tools, and Applications* (pp. 351-380).

www.irma-international.org/chapter/saas-requirements-engineering-agile-development/77714

XHDLNet Classification of Virus-Borne Diseases for Chest X-Ray Images Using a Hybrid Deep Learning Approach

Srishti Choubey, Snehlata Bardeand Abhishek Badholia (2022). *International Journal of Software Innovation* (pp. 1-14).

www.irma-international.org/article/xhdlnet-classification-of-virus-borne-diseases-for-chest-x-ray-images-using-a-hybrid-deep-learning-approach/311505

Modelling Security Using Trust Based Concepts

Michalis Pavlidis, Haralambos Mouratidisand Shareeful Islam (2012). *International Journal of Secure Software Engineering* (pp. 36-53).

www.irma-international.org/article/modelling-security-using-trust-based/66407

A Case for Using Blended Learning and Development Techniques to Aid the Delivery of a UK Cybersecurity Core Body of Knowledge

David A. Birdand John Curry (2018). *International Journal of Systems and Software Security and Protection* (pp. 28-45).

www.irma-international.org/article/a-case-for-using-blended-learning-and-development-techniques-to-aid-the-delivery-of-a-uk-cybersecurity-core-body-of-knowledge/221921

Security Over the Information Systems Development Cycle

C. Blanco, D. Rosado, C. Gutiérrez, A. Rodríguez, D. Mellado, E. Fernández-Medina, J. Trujilloand M. Piattini (2011). *Software Engineering for Secure Systems: Industrial and Research Perspectives* (pp. 113-154).

www.irma-international.org/chapter/security-over-information-systems-development/48408