01 Jan 2005

Collaborative Global Software Development and Education

Xiaoqing Frank Liu
Missouri University of Science and Technology, fliu@mst.edu

Recommended Citation

X. F. Liu, "Collaborative Global Software Development and Education," *Proceedings of the 29th Annual International Computer Software and Applications Conference*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.
The definitive version is available at https://doi.org/10.1109/COMPSAC.2005.56

Collaborative Global Software Development and Education

Xiaoqing (Frank) Liu
Department of Computer Science
University of Missouri – Rolla
fliu@umr.edu

Abstract

In this position paper, challenges and issues with collaborative global software development and education are discussed. Suggestions are made to improve and strengthen software engineering education to adapt to it.

1. Background

With the need of reduced software product development cost and time and improved software quality, software products are increasingly developed via collaborations across people, organizations, and countries [1, 2]. Although software implementation tasks may be moved from countries with high labor costs to countries with low costs relatively easily due to internet and common programming environments, tasks requiring intensive customer interaction, such as requirements analysis and specification, software design, and verification and validation, are hard to migrate. Therefore, software developers across countries must collaborate effectively to develop software products globally.

In global software development, geographically and culturally diverse groups come together to develop complex software products. However, analysts, developers, and testers with diverse technical and cultural backgrounds, expectations and goals in geographically distributed locations often have a difficult time to collaborate. Many conflicts exist in tasks in each phase in a global software development life cycle, such as requirements analysis, priority setting, task assignment, system integration, schedule, resource allocation, time management in multiple time zones, and communication. Due to many conflicts, global software development projects are often more complex to manage than the most complex in-house projects [2]. Conflict should be managed and software development activities should be coordinated effectively in many geographically distributed locations across countries so that software products can be developed on time, within budget, and in high quality.

2. Collaborative Global Software Education

Currently global software development results in declined demands for programming professionals and declined enrollment in many computer science programs in developed countries, and a software technology boom in several developing countries, such as India and China. In order to deal with the challenges in developed countries, software engineering education process needs be improved and strengthened. Some of actions listed below may help to address the issue.

1. Enhance software engineering programs to incorporate collaborative global software development processes.
2. Develop collaborative software engineering programs by establishing a partnership among universities and industrial software companies to equip students with advanced software environments and tools.
3. Strengthen and expand curricula on software development tasks which are hard to migrate, such as global software project management, global software process, requirements analysis, software design, and verification and validation.
4. Develop collaborative software engineering programs by establishing a partnership among software engineering programs across countries.
5. Enhance and develop software certificate programs
6. Enhance distant learning software engineering programs

References

1. M. Bass and D. Paulish, "Global software development process research at Siemens", Proc. of Intl Workshop on global software development, Edinburgh, Scotland.

2. N. Zamiska, "Quality lures software outsourcing", the Wall Street Journal, May 5, 2005.