Software Engineering in Multicultural Teams

Introduction

Much of today's software is created in software engineering teams, and some teams are made up of engineers from different cultures. As each society has their own way of doing things, sometimes these multicultural teams clash; and, instead of being more productive together, their conflicts result in unproductivity. This case study explores some of the issues that multicultural teams face.

Multicultural Team Issues and Cultural Stereotypes

Software developing teams can struggle to work together because of cultural differences. Some of these issues are a result of inaccurate stereotypes people have of their international teammates. For example, in a workshop conducted by Matthiesen et al. (2020), participants were asked to describe typical personalities of global colleagues based on their experiences. One participant claimed that Indians always say "yes" even when they mean "no" because of their hierarchical orientation. This is a stereotype – and it might cause that person to treat their Indian colleagues unfairly.

Is it possible to go beyond stereotyping, and learn about some of the reasons that might underlie cultural differences? Hofstede offers a theoretical framework for interpreting these and other commonalities between groups of people from similar cultural backgrounds.

Multicultural Team Issues Explained Using Hofstede's Dimensions

Many researchers have analyzed cultural issues through the lens of Hofstede's six cultural dimensions. The power distance index (PDI) is one of the most noticeable dimensions that can cause misunderstandings and conflicts between teams of mixed cultures. Alsenoosy et al. (2018) for example, compared Hofstede's scores for Australia and Saudi Arabia and interviewed requirements engineer practitioners in both countries. Australia has a low PDI (36); on a typical Australian team, we would expect that teammates would be treated more equally and that there would be a looser hierarchy. Saudi Arabia, however, has a high PDI (95). On some Saudi Arabian teams, younger and lower-ranking team members may be expected to submit to older and higher-ranking ones. In a team made up of Australians and Saudi Arabians, members of one culture may offend members of another: the average Australian's tendency to pose questions and make suggestions might be offensive to an average Saudi Arabian, while a Saudi Arabian following cultural norms of information withholding might offend an Australian.

The dimension of uncertainty avoidance (UAI) can have a great impact on multicultural teams as well. Cultures with high UAI like Japan (95), tend to plan out every possible scenario before they get a start on a project. Lower UAI countries like India (40) and the United States (46) like to get a start on projects fairly quickly. This can cause conflict between teams as Japan relies on documentation and planning to complete a project, and the US and India may become impatient with Japan's "excessive" cautiousness (Borchers, 2003). A country like Japan would probably prefer a Waterfall framework for structuring a software engineering project as it involves much planning before any task can take place. Countries like India and the United States, however, might probably thrive more with an Agile framework where there is much less planning and a lot more creating and experimenting.

The language differences between multicultural teams can be a barrier as well and may intersect with the UAI dimension. Though many software engineers speak English, non-native speakers have varying vocabulary and fluency. For those with high UAI, this language barrier can bring a lot of anxiety therefore causing the team to request a substantial amount of documentation (Borchers, 2003). However, the language barrier can also make the documents hard to understand between the team members which can cause misunderstandings (Alsanoosy et al., 2018).

Hofstede’s model can sometimes intersect with stereotypes. The power of the model is that it can help us *explain* differences, as well as to *predict* issues that might arise from a given team composition. Every team is unique, but a wise teammate (or manager) should still anticipate and understand the potential for team conflict.

Discussion Questions

1. Time zones can be a major issue on multi-geography teams. Coming to an agreement of the meeting time for the teams may be a challenge, as one team may have to work at a very strange time in the day. What would be your strategy to communicate and collaborate with people from a distant time zone?

2. Different cultures perceive time in different ways. Some middle eastern cultures, for example, do not value punctuality. Alsanoosy et al. (2018) noted that in a collaboration between Saudi Arabians and Australians, the Saudi Arabians were often later to start meetings and to process documents. Engineers may be frustrated when crossing this "time-perception" boundary.

   What would you do if you worked in a team with members who perceive time differently than you (if they are more relaxed and not punctual, or if they are way more punctual than you)?

3. It is good to learn about other cultures and analyze how they tend to behave in teams before working with people from another country. However, the analysis, along with preexisting bias, can easily turn into an unfair stereotype of the teammates. How do you think you can balance preparing yourself for cultural differences through education, with concerns about avoiding stereotyping?

4. Trust matters. Marutschke et al. (2018) realized that because their subjects did not build relationships and trust within the international teams, they experienced many more difficulties completing the project because of cultural differences. Wang Y. and Zhang M., (2019) realized a related issue where software engineers either were unaware of

biases that hindered their trust of a teammate of a different nationality or were unwilling to admit them. Both of these studies show that it is important to admit the need to intentionally build relationships within global software engineering teams.

Why do you think trust is so important in team projects? How does the issue of trust also tie in with the idea of biases towards other teammates? How might you build trust in a multicultural team?

5. Hofstede's model of culture has six dimensions in total. Take a look at this resource: https://hi.hofstede-insights.com/national-culture. Do you think any of the other dimensions of this model could affect intercultural teamwork? Give an example.

References

Alsanoosy, T., Spichkova, M., & Harland, J. (2018). Cultural influences on the requirements engineering process: Lessons learned from practice. *International Conference on Engineering of Complex Computer Systems (ICECCS)*, *2018-Decem*, 61–70. https://doi.org/10.1109/ICECCS2018.2018.00015

Altin, R., Bektik, M., Eksioglu, N., Koray, C., Oner, O. C., Sadetas, M., Sener, H., Simsek, D., Ma, C.-C., Price, C., & Routh, C. (2009). *Working across time zones in cross-cultural student teams*. 2.

Borchers, G. (2003). The software engineering impacts of cultural factors on multi-cultural software development teams. *International Conference on Software Engineering (ICSE)*, *6*, 540–545. https://doi.org/10.1109/icse.2003.1201234

Jesiek, B. K., Buswell, N. T., & Zhu, Q. (2018). Global engineering competency: Assessment tools and training strategies. *ASEE Annual Conference and Exposition, Conference Proceedings*, *2018-June*. https://doi.org/10.18260/1-2--30559

Jesiek, B. K., & Haller, Y. (2014). *Developing Globally Competent Engineering Researchers: Outcomes-Based Instructional and Assessment Strategies from the IREE 2010 China Research Abroad Program*. 31.

Li, Y., Krusche, S., Lescher, C., & Bruegge, B. (2016). *Teaching Global Software Engineering by Simulating a Global Project in the Classroom*. 9.

Marutschke, D. M., Kryssanov, V. V., & Brockmann, P. (2018). East meets west: Global software engineering course in Japan and Germany. *Proceedings - International Conference on Software Engineering*, 30–34. https://doi.org/10.1145/3196369.3196390

Matthiesen, S., Bjørn, P., & Trillingsgaard, C. (2020). Attending to implicit bias as a way to move beyond negative stereotyping in GSE. *Proceedings of the 15th International Conference on Global Software Engineering*, 22–32. https://doi.org/10.1145/3372787.3390432

Wang, Y., & Zhang, M. (2019). Country Stereotypes, Initial Trust, and Cooperation in Global Software Development Teams. *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*, 152–161. https://doi.org/10.1109/ICGSE.2019.00036