Collaborative Discussion 1: Factors which Influence Reusability

Refer to the article by Padhy et al. (2018), specifically Table 1, where the authors present a list of factors which they consider influence the reusability of a piece of object-oriented software.

In this collaborative discussion, you are required to prioritise this list, presenting your argument for the priorities assigned.

Peer Response 1:

Hi Helen,

Thank you for sharing your views on assets listed by Padhy et al. (2018). It is valuable to observe your ranking of Requirements Analysis (RA) as first. Given how time and resource-consuming this process can be for organisations, I do see the value of this asset for reusability, and this appears to be validated in existing research (Carillo de Gea et al., 2016). I am still unsure how these assets could be reused between companies as my assumption is that much of RA is company-confidential, however, given the amount of available use cases, there must be opportunities. However, I would like to challenge the low ranking of Test Case (TCTD) reuse. It is widely known as a beneficial asset and with multiple open solutions (Zhao et al., 2020) and measurement frameworks (Carterette et al., 2010), across multiple applications (Mukelabai et al., 2023) and industries.

References:

Carrillo de Gea, J.M., Nicolás, J., Fernández-Alemán, J.L., Toval, A. & Idri, A. (2016) Are the expected benefits of requirements reuse hampered by distance? An experiment. SpringerPlus, 5: 1-26.

Carterette, B., Gabrilovich, E., Josifovski, V. & Metzler, D. (2010). Measuring the reusability of test collections. In Proceedings of the third ACM international conference on Web search and data mining: 231-240.

Mukelabai, M., Derks, C., Krüger, J. & Berger, T. (2023) To share, or not to share: Exploring test-case reusability in fork ecosystems. In 2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE): 837-849.

Zhao, Y., Chen, J., Sejfia, A., Schmitt Laser, M., Zhang, J., Sarro, F., Harman, M. & Medvidovic, N. (2020). Fruiter: a framework for evaluating ui test reuse. In Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering: 1190-1201.

Peer Response 2:

Hi Jordel,

Thank you for sharing your opinion on the assets listed by Padhy et al. (2018). It is valuable to observe your ranking of Requirements Analysis (RA) as first, similarly to our peers. As I have alluded to earlier, given how time and resource-consuming this process can be among teams, there is a lot of value in reusing requirements internally among projects within an organisation, and this appears to be validated in existing research (Carillo de Gea et al., 2016). However, this raises the question on whether this increases the risk of duplication, therefore RA should be reused with caution. However, I would like to challenge the lower ranking of Design Patterns (DP). DPs are industry and application-agnostic, and continue to be applied in innovative contexts (Bellavista et al., 2023) for multiple decades since their release

(Gamma et al., 1995), therefore are now considered common practice. However, it has been found that certain DPs and also the way DPs are implemented and documented may affect software quality, both positively and negatively (Feitosa et al., 2019; Wedyan & Abufakher, 2020).

References:

Bellavista, P., Bicocchi, N., Fogli, M., Giannelli, C., Mamei, M. & Picone, M. (2023) Requirements and design patterns for adaptive, autonomous, and context-aware digital twins in industry 4.0 digital factories. Computers in Industry, 149: 103918.

Carrillo de Gea, J.M., Nicolás, J., Fernández-Alemán, J.L., Toval, A. & Idri, A. (2016) Are the expected benefits of requirements reuse hampered by distance? An experiment. SpringerPlus, 5: 1-26.

Gamma, E., Helm, R., Johnson, R. & Vlissides, J. (1995) Design patterns: elements of reusable object-oriented software. Pearson Deutschland GmbH.

Feitosa, D., Ampatzoglou, A., Avgeriou, P., Chatzigeorgiou, A. & Nakagawa, E.Y. (2019) What can violations of good practices tell about the relationship between GoF patterns and run-time quality attributes?. Information and Software Technology, 105: 1-16.

Wedyan, F. & Abufakher, S. (2020) Impact of design patterns on software quality: a systematic literature review. IET Software, 14(1): 1-17.