Unit 4 Seminar Preparation: Activity 2: Risk Identification and Framework Selection

Student Name: Lauren Pechey

Module: Software Engineering Project Management

Main Risks Identified

Recent literature highlights that risks in software development projects are frequently rooted in organisational and process-related factors rather than purely technical shortcomings. Anton and Nucu (2020) identify ineffective risk identification, weak governance structures, and poor stakeholder involvement as persistent challenges in software and enterprise projects. Similarly, Saravanan et al. (2020) emphasise that inaccurate effort estimation and inadequate requirements definition during early lifecycle stages often lead to schedule overruns and cost escalation.

Additionally, insufficient risk monitoring throughout development increases the likelihood that emerging issues remain undetected until later stages, thereby amplifying their impact (Vidoni, Codabux and Fard, 2022). Ethical and human-related risks, particularly those associated with decision-making, accountability, and long-term maintainability, are also increasingly recognised as critical contributors to project failure (Biable et al., 2023).

These risks align closely with the early phases of the Software Development Life Cycle (SDLC), particularly requirements analysis and system design, where deficiencies can propagate into later stages and contribute to technical debt and reduced system quality.

Chosen Risk Framework

To capture and categorise these risks, a Risk Register informed by ISO 31000 principles, as discussed in the Unit 3 Lecturecast, is an appropriate choice. This framework supports continuous

risk management across the SDLC and enables risks to be categorised as technical, organisational, human, or ethical. It also facilitates systematic evaluation through likelihood and impact scoring, ensuring that mitigation strategies are prioritised and traceable throughout the project lifecycle (Anton and Nucu, 2020).

Example Risk and Mitigation (Forum Contribution)

Risk: Underestimation of development effort due to unclear or evolving requirements.

Mitigation: Conduct early and structured requirements validation sessions with stakeholders and apply iterative effort estimation updates at each development milestone to reflect changes in scope and complexity (Saravanan et al., 2020).

Conclusion

This activity demonstrates the importance of integrating estimation techniques with structured risk management frameworks. The use of standardised approaches such as ISO 31000 and risk registers supports proactive identification and mitigation of both technical and non-technical risks. Furthermore, incorporating privacy- and ethics-aware frameworks, such as those advocated by NIST, alongside transparent estimation models like COCOMO, enables more informed decision-making and contributes to improved project planning, governance, and long-term sustainability (Biable et al., 2023; Vidoni et al., 2022).

References

Anton, G. and Nucu, A. (2020) 'Enterprise Risk Management: A Literature Review and Agenda for Future Research', *Journal of Risk and Financial Management*, 13(11), pp. 281.

Biable, S. E., et al. (2023) 'Proposed Ethical Framework for Software Requirements Engineering', *IET Software*, 17(4), pp. 526–537.

Saravanan, T., et al. (2020) *Comparative Analysis of Software Life Cycle Models*.

Vidoni, M., Codabux, Z. and Fard, F. H. (2022) 'Infinite Technical Debt', *The Journal of Systems and Software*, 190, pp. 111336.