

Summary - Collaborative Discussion 1: Project Failures Study

Throughout this discussion, we examined the multifaceted nature of software project failure. Initially, I aligned with Agrawal et al. (2024), identifying knowledge-based errors, organisational influence, and cognitive load as primary drivers. However, the dialogue with peers has significantly refined this perspective, shifting the focus from individual deficits to systemic and epistemic pressures.

Common failure modes such as inadequate planning, unclear requirements, and poor communication were consistently identified across case studies like the NHS NPfIT and Therac-25 (Verner & Sarwar, 2021). Yet, the discussion revealed that these are often symptoms of deeper issues. As debated with Doug, the "knowledge gaps" responsible for these failures are paradoxically linked to information abundance. In modern development, errors often stem not from a lack of information, but from cognitive saturation (Helgesson et al., 2019). This reframes "poor planning" not just as a procedural oversight, but as a failure to manage the cognitive capacity of the team.

This synthesis suggests that organisational influence plays a defining role. It creates the pressure that intensifies cognitive load, but it also holds the key to mitigation. As noted by Ben and Sergei regarding leadership, effective governance must go beyond simple resource allocation. It requires active knowledge management to prevent the "brain drain" or saturation that leads to critical oversight (Sallai & Pepper, 2025).

Ultimately, the consensus is that failure emerges from the interplay of human limits and organisational structure. Resilience requires a holistic approach: rigorous requirements validation and risk management must be paired with strategies to protect the technical team from cognitive burnout—potentially through AI support—ensuring that project planning accounts for human psychological limits as much as technical milestones.

References

- Agrawal, T., Walia, G.S. and Anu, V.K. (2024) 'Development of a Software Design Error Taxonomy: A Systematic Literature Review', *SN Computer Science*, 5(7), p. 467. DOI: <https://doi.org/10.1007/s42979-024-02797-2>.
- Belabbes, M.A., Ruthven, I., Moshfeghi, Y. and Pennington, D. (2023) 'Information overload: a concept analysis', *Journal of Documentation*, 79(1), pp. 144–159.
- Helgesson, D., Engström, E., Runeson, P. and Bjarnason, E. (2019) 'Cognitive load drivers in large scale software development', in 2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE). Piscataway, NJ: IEEE, pp. 91–94.
- Nizam, A. (2022) 'Software project failure process definition', *IEEE Access*, 10, pp. 34428–34441.
- Sallai, D., & Pepper, A. (eds.) (2025) Navigating the 21st Century Business World: Case Studies in Management. London: LSE Press.
- Verner, C.M., & Sarwar, D. (2021) 'Avoiding project failure and achieving project success in NHS IT system projects in the United Kingdom', *International Journal of Strategic Engineering (IJSE)*, 4(1), pp. 1–22.