Software projects are complex and varied in nature. Consequently, finding the reasons for the failure of software projects can be just as complex. Lehtinen et al. (2014) analyse which factors mainly contribute to project failure, and note that half of the causes for software project failures were "bridge causes", which are actions or behaviours that occur across various stages of a software project and across various departments. These causes were namely having a weak task backlog due to a lack of information and poor prioritisation, a lack of cooperation between employees, and an insufficient allocation of resources for testing. I thus argue that these three categories are the main contributors to failed software projects.

As evidence of how these factors caused project failure, I make reference to two case studies: the Federal Aviation Administration (FAA) Advanced Automation System (AAS), and the U.S. Air Force Expeditionary Combat Support System (ECSS).

The FAA AAS was a software project which aimed to modernise air traffic control systems: the idea was conceived in 1983, with an initial budget of $2.5 billion and an initial deadline of 1996. However, this budget increased significantly along with development time- estimates in 1994 reached $7 billion and it was said that some parts of the project were behind by 8 years (Hilburn et al., 2021). Ultimately, the system never materialised. A key reason for failure was a poor testing process- testing milestones were frequently skipped or done to a lower standard, and departments responsible for testing did not cooperate well, resulting in the duplication of testing effort in some cases (Barlas, 1996). Additionally, time and cost tracking were not done consistently, leading to much larger costs being realised (Gibbs, 1994).

The U.S. Air Force ECSS was a software project which was intended to allow the Air Force to track all of its physical assets and ERP tasks. However, after $1 billion spent on the project and less than 25% of the intended functionality being achieved, the project was cancelled (Centre for Public Impact, 2017). A key reason for given for the failure of this project was a lack of communication and alignment between involved parties (namely developers and end-users, along with other inter-organisational groups). Another contributing factor was the constant loss of project knowledge due to high churn rates of programme managers and programme executive officers, and a lack of organisational awareness of the project led to increased friction in its attempts to be adopted, which also decreased its overall fit (Calleam Consulting, 2022).

From the above examples, it can be seen that testing failures and misalignment between groups are large contributors to project failure. Although not as visible but equally as important, these examples highlight how poor top-level management also contributes to information loss, ambiguity, and improper planning, which would manifest as a weak task backlog for engineers.

**References**

Barlas, S. (1996) Anatomy of a Runaway: What Grounded the AAS. IEEE Software 13(1): 104-106. DOI: http://doi.org/10.1109/MS.1996.476294

Baumann. (2021) Lessons Learned from the Waste Management ERP Failure. Available from: https://www.panorama-consulting.com/waste-management-erp-failure/ [Accessed 12 March 2022].

Calleam Consulting. (2022) US Department of Defense – U.S. Air Force. Available from: http://calleam.com/WTPF/?p=4914 [Accessed 12 March 2022].

Centre for Public Impact. (2017) The US Air Force’s Expeditionary Combat Support System (ECSS). Available from: https://www.centreforpublicimpact.org/case-study/air-forces-expeditionary-combat-support-system-ecss [Accessed 12 March 2022].

Gibbs, W. (1994) Software's Chronic Crisis. Scientific American 271: 86-95. DOI: http://doi.org/10.1038/scientificamerican0994-86

Hilburn, T., Squires, A., Davidz, H. & Turner, R. (2021) Federal Aviation Administration (FAA) Advanced Automation System (AAS). Available from: https://www.sebokwiki.org/wiki/Federal\_Aviation\_Administration\_(FAA)\_Advanced\_Automation\_System\_(AAS) [Accessed 12 March 2022].

Lehtinen, A., Mäntylä, V., Vanhanen, J., Itkonen, J. & Lassenius, C. (2014) Perceived causes of software project failures – An analysis of their relationships. Information and Software Technology 56(6): 623–643.