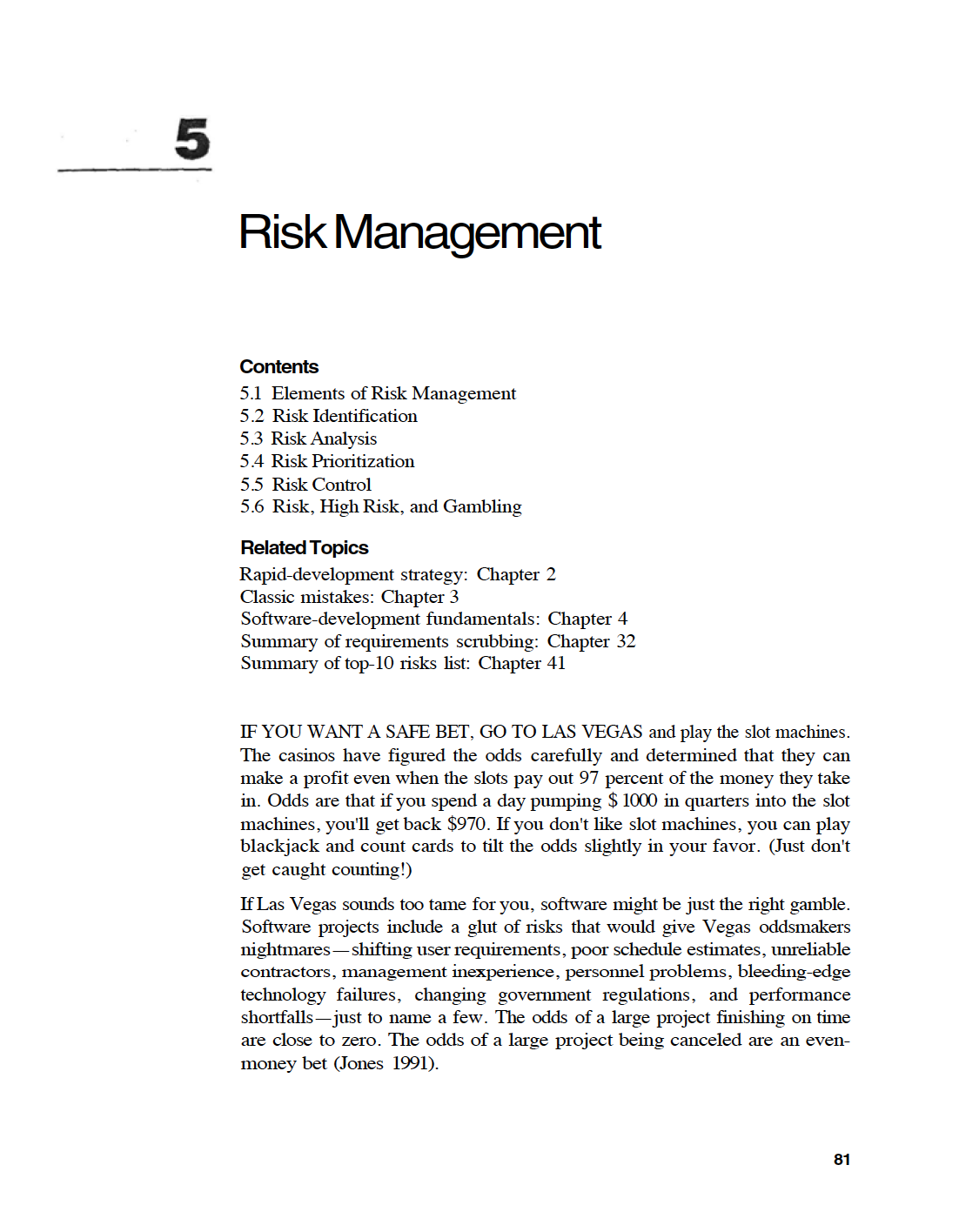
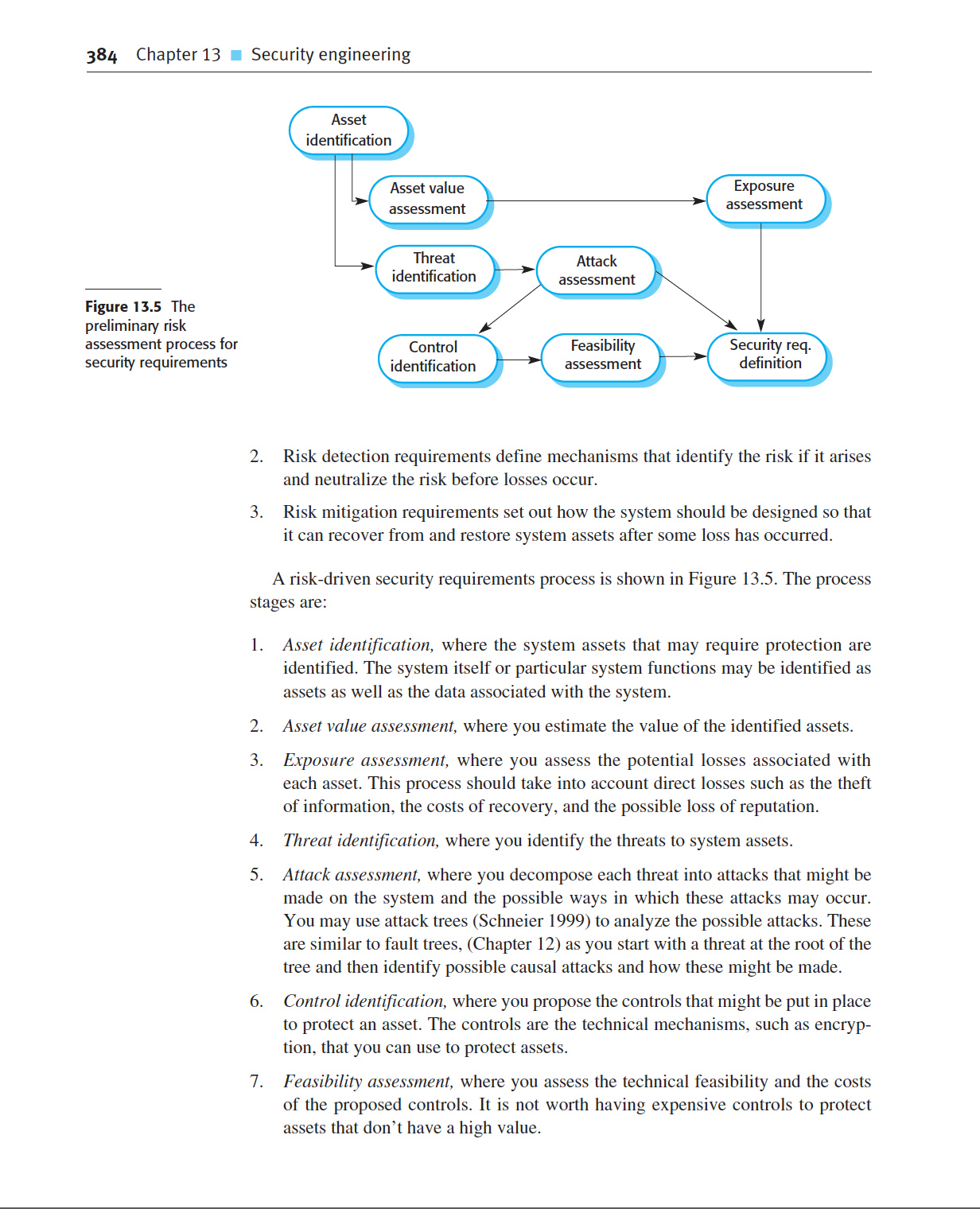
McConnell, S. (1996). *Rapid Development: Taming Wild Software Schedules*. Microsoft Press.



Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson Education Limited.



Ahmad, J., et al. (2022). Quality Requirement Change Management’s Challenges: An Exploratory Study Using SLR. IEEE Access, 10, 127575–127588. https://doi.org/10.1109/ACCESS.2022.32127575

Ahonen, J., & Savolainen, P. (2010). Software engineering projects may fail before they are started: Post-mortem analysis of five cancelled projects. Journal of Systems and Software, 83, 2175-2187. https://doi.org/10.1016/j.jss.2010.06.023

Anton, A. I., & Potts, C. (1998). The use of goals to surface requirements for evolving systems. In Proc. 20th International Conference on Software Engineering (pp. 157-166).

Berry, D. M. (2002). The inevitable pain of software development, including of extreme programming, caused by requirements volatility. In Proc. Eberlein and Leite.

Bhatti, M. W., Hayat, F., Ehsan, N., et al. (2010). A methodology to manage the changing requirements of a software project. In Proc. 2010 Int. Conf. Computer Information Systems and Industrial Management Applications (CISIM) (pp. 319-322). IEEE.

Bohem, B. W. (1991). Software risk management: Principles and practices. IEEE Software, 8(1), 32-41.

Cao, L., & Ramesh, B. (2008). Agile requirements engineering practices: An empirical study. IEEE Software, 25(1).

Dingsøyr, T., & Conradi, R. (2002). A survey of case studies of the use of knowledge management in software engineering. International Journal of Software Engineering and Knowledge Engineering, 12(4), 391-414.

Eberlein, A., & Leite, J. (2002). Agile requirements definition: A view from requirements engineering. In Proc. Int. Workshop Time-Constrained Requirements Engineering (TCRE’02) (pp. 4–8).

El Emam, K., Holtje, D., & Madhavji, N. H. (1997). Causal analysis of the requirements change process for a large system. In Proc. Int. Conf. Software Maintenance (pp. 214–221).

Ernst, N. A., Borgida, A., Jureta, I. J., & Mylopoulos, J. (2014). Agile requirements engineering via paraconsistent reasoning. Information Systems, 43, 100–116.

Gotel, O., & Finkelstein, A. (1994). An analysis of the requirements traceability problem. In Proc. First International Conference on Requirements Engineering (pp. 94-101).

Inayat, I., Salim, S. S., Marczak, S., Daneva, M., & Shamshirband, S. (2015). A systematic literature review on agile requirements engineering practices and challenges. Computers in Human Behavior, 51, 915–929.

International Organization for Standardization. (2018). ISO/IEC/IEEE 29148:2018 - Systems and software engineering — Life cycle processes — Requirements engineering.

Jayatilleke, S., & Lai, R. (2017). A systematic review of requirements change management. Information and Software Technology. https://doi.org/10.1016/j.infsof.2017.09.004

Jun, L., Qiuzhen, W., & Lin, G. (2010). Application of agile requirement engineering in modest-sized information systems development. In Proc. Second World Congress on Software Engineering (WCSE), vol. 2 (pp. 207–210). IEEE.

Lehtola, L., Kauppinen, M., & Kujala, S. (2004). Requirements prioritization challenges in practice. In Proc. 5th Int. Conf. on Product Focused Software Process Improvement (pp. 497-508).

Leung, H., & Fan, Z. (2002). Software cost estimation. In Handbook of Software Engineering (pp. 1–14). Hong Kong Polytechnic University.

LinkedIn. (n.d.). How do you verify/validate software changes? Retrieved from https://www.linkedin.com/advice/0/how-do-you-verify-validate-software-changes-skills-software-testing

McConnell, S. (1996). Rapid Development: Taming Wild Software Schedules. Microsoft Press.

Rajper, S., & Shaikh, Z. A. (2016). Software development cost estimation: A survey. Indian J. Sci. Technol., 9(31).

Ramesh, B., & Jarke, M. (2001). Toward reference models for requirements traceability. IEEE Transactions on Software Engineering, 27(1), 58-93.

Ramesh, B., Cao, L., & Baskerville, R. (2010). Agile requirements engineering practices and challenges: an empirical study. Information Systems Journal, 20(5), 449–480.

Salesforce. (2024). What are customer expectations, and how have they changed? Retrieved from https://www.salesforce.com/resources/articles/customer-expectations/

Shafiq, M., Zhang, Q., Akbar, M. A., Khan, A., Hussain, S., Amin, F.-e, Khan, A., & Soofi, A. (2018). Effect of Project Management in Requirements Engineering and Requirements Change Management Processes for Global Software Development. IEEE Access, PP, 1-1. https://doi.org/10.1109/ACCESS.2018.2834473

Sommerville, I. (2016). Software Engineering (10th ed.). Pearson Education Limited.

Souza, R. R. G. (2015). Inappropriate Software Changes: Rejection and Rework. Journal.

Visure Solutions. (2024). Requirements Change Management: Definition, Process | Complete Guide. Retrieved March 13, 2024, from https://visuresolutions.com/requirements-change-management/

Weigers, K., & Beatty, J. (2013a). Change happens. In Software Requirements (3rd ed., ch. 28, p. 477). Microsoft.

Weigers, K., & Beatty, J. (2013b). Change impact analysis. In Software Requirements (3rd ed., ch. 28, p. 484). Microsoft.

Weigers, K., & Beatty, J. (2013c). Documenting the requirements. In Software Requirements (3rd ed., ch. 10). Microsoft.

Weigers, K., & Beatty, J. (2013d). Enhancement and replacement project. In Software Requirements (3rd ed., ch. 21, p. 398). Microsoft.

Weigers, K., & Beatty, J. (2013e). Enhancement and replacement project. In Software Requirements (3rd ed., ch. 21, p. 401). Microsoft.

Epicflow. (2024). Changing requirements in project management: