

Performance Evolution of Software Engineering Development and Project Management: A Critical Evolution Software Engineering Process

Gurudev Sawarkar¹ and Dr. Dipesh Rajput²

PhD Scholar, Department of Computer Science and Engineering¹

Associate Professor, Department of Computer Science and Engineering²

Swami Vivekananda University, Sagar, MP, India

Abstract: *Among the many pressing concerns in the realm of computers, software project development is among the highest. System development life cycle is a part of this (SDLC). A key goal of the software development life cycle (SDLC) is to reduce the likelihood of errors while improving the quality of the final result. Without a well-defined set of steps, the software development process is a pretty complicated affair. Method established to standardize and streamline software development. The introduction of the SDLC (Software Development Life Cycle) existence. What we have here is a methodical and organized approach to the process of creating software. Using the SDLC as a guide, entails a wide variety of steps and processes that must be completed before the final programmer is released. Software comes in many forms. Types of software development life cycles, each with their own benefits and drawbacks, are commonly employed in the software development process. Disadvantages Five of these software development life cycle (SDLC) models, including the waterfall model, the v-shaped model, and the prototype model, are presented in this study. Existing models are compared using a spiral and an iterative structure.*

Keywords: SDFC, Waterfall Model, Spiral Model.

REFERENCES

- [1]. R. Conradi and A. Fuggetta, "Improving software process improvement," IEEE Software, vol.19, no.4, pp.92–99,2002.
- [2]. R. L. Glass, Facts and Fallacies of Software Engineering. Addison-Wesley Professional,2002.
- [3]. S. T. Acuna, N. Juristo, A. M. Moreno, and A. Mon, A Software Process Model Handbook for Incorporating People's Capabilities. Springer-Verlag,2005.
- [4]. T. DeMarco and T. Lister, People ware: productive projects and teams. Dorset House Publishing Company,1999.
- [5]. Y. Dittrich, C. Floyd, and R. Klischewski, Social thinking-software practice. The MIT Press, 2002.
- [6]. M. Grechanik and D. E. Perry, "Analyzing software development as a non cooperative game," in IEE Seminar Digests, vol.29, 2004.
- [7]. H. Van Vliet, "Editorial: Signs of a thriving journal," Journal of Systems and Software, vol. 86, no.1, p.1, 2013.
- [8]. R. Charette, "Why software fails," IEEE Spectrum, vol. 42, no. 9, pp.42–49,2005.
- [9]. D. Hartmann, "Interview: Jim Johnson of the Standish Group," Infoqueue, Aug, vol.25,2006.
- [10]. C. Jones, Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies. McGraw-Hill Osborne Media, 2009.
- [11]. J. E. Tomayko and O. Hazzan, Human Aspects of Software Engineering. Firewall Media, Dec. 2005.
- [12]. G. M. Weinberg, The psychology of computer programming. Van Nostrand Reinhold New York,1971.
- [13]. H. Robinson and H. Sharp, "Collaboration, communication and coordination in agile software development practice," in Collaborative Software Engineering. Berlin Heidelberg: Springer,2010, pp.93–108.

- [14]. S. Biffi, A. Aurum, B. Boehm, H. Erdogmus, and P. Grunbacher, Value-based software engineering. Springer,2005.
- [15]. Standish Group, “The chaos report,” Available on-lineat <http://www.projectsmart.co.uk/docs/chaos-report.pdf>,1995.