

The past, present and future of standards in software engineering



**ASSIGN
BUSTER**

The Evolution of Standards in Software Engineering The Evolution of Standards in Software Engineering Software engineering is relatively the basic element of developing software programs. The programs created and developed are out to ensure that individuals using technology can find it easy to use the available tools to their benefit. Throughout the decades that have passed, software standards have continued to evolve into what is seen today in many technological programs. This evolution is escalating (Rajilich 2011). This is because engineers are out to find the best program that best suits would-be users of these programs. Through these standards, it is made possible (Saleh 2009). This paper will review the evolution of standard software engineering, and how users are affected by this evolution process. The advancement in technology is making it easier for these organizations to be more effective and more advanced. Standards in software engineering have reached a level where, anything is possible. The creation of programs that seek to identify all the would-be users' needs has revolutionised the way things work in the modern world. Operations are carried out with ease. The effectiveness and precision with which these operations are carried out is growing (Mall 2009). Writing software, to some, is a profession from which they earn their living.

Standards range from local invented, to international standards that help software be accepted globally. These are the de facto standards which enable people to operate software that is acceptable throughout the world. Government entities in the world approve of these standard-setting organizations. Some organizations have recognition that reaches the whole world when it comes to setting software standards. The International

Organization of Standardization is one of the few. It is abbreviated as ISO. It <https://assignbuster.com/the-past-present-and-future-of-standards-in-software-engineering/>

is a representation of many international bodies. This progress has enabled software standards to have global acceptance (Puntambekar 2009).

The evolution of software standards can incorporate the understanding of much more than local industries present in the world. At present, this demand is growing with each passing day. Therefore, soon enough, the future for software engineering will reach its peak (O'Regan 2012). This advancement will enable people to get extra work complete, while taking less time. This is what everybody wants because effectiveness is the aim of this progress. Through the ISO/IEC, many software standards are set and developed to help the global community get through some of the programs that exist.

An example of standard in software is the IEEE standard. It was adopted by the ISO in the year 2007 to create a basis for the architecture of software-intensive systems. Architecture plays an intricate role in the life cycle of any software system. The IEEE standards address many issues. Most of the concerns were not covered by the standards that existed previously. It provided crucial information as to architecture concepts that currently exist (Puntambekar 2009).

In conclusion, to best understand what standards in software engineering are achieving, it is crucial to take a look at the modern world. All that is happening is due to effective and efficient standards that help run software programs. Companies, organizations, and industries run their projects based on these standards. Improvements are touching the lives of modern society in a progressive manner. To ensure this continues happening, global standards continue to be developed. It is, and will always be accountable for the progress of the technological world, and all those who are users of such <https://assignbuster.com/the-past-present-and-future-of-standards-in-software-engineering/>

advances (Chis 2010). It is, therefore, high time people made the choice to embrace such standards. This is for everyone to have a stress free attitude toward their usage in their daily routines.

References

- Agarwal, B & Tayal, P 2009, Software engineering, Cambridge University Press, New York.
- Chis, M 2010, Evolutionary computation and optimization algorithms in software, W. W. Norton, New York.
- Erdogmus, H & Tanir, O 2001, Advances in software engineering: Comprehension, evaluation, and evolution, Springer, New York.
- Madhavji, N & Perry, D 2006, Software evolution and feedback: Theory and practice, Cornell University Press, New York.
- Mall, R 2009, Fundamentals of software engineering, Oxford University Press, Oxford.
- Mens, T & Demeyer, S 2008, Software evolution, PULP, New York.
- O'Regan, G 2012, A brief history of computing, Sage Publishers, London.
- Puntambekar, A 2009, Software engineering, Macmillan Publishers, New York.
- Rajilich, V 2011, Software engineering: The current practice, Hart Publishing, Oxford.
- Saleh, K 2009, Software engineering, Bantam Books, New York.
- Sundar, D 2010, Software engineering, Cambridge University Press, Cambridge.
- Tortora, G & Ferrucci, F 2008, Emerging methods, technologies and process management in software engineering, Hart Publishing, Australia.
- Yang, H 2003, Successful evolution of software systems, Butterworth
<https://assignbuster.com/the-past-present-and-future-of-standards-in-software-engineering/>

Publishers, New Zealand.

Zelkowitz, M 2006, *Advances in computers: Quality software development*, Free Press, New York.

Zielinski, K & Szmuc, T 2006, *Software engineering: Evolution and emerging technologies*, Columbus Dispatch, Columbus.