



Title	Software Engineering in Society
Authors(s)	Kazman, Rick, Pasquale, Liliana
Publication date	2020-01
Publication information	Kazman, Rick, and Liliana Pasquale. "Software Engineering in Society." IEEE, January 2020. https://doi.org/10.1109/MS.2019.2949322 .
Publisher	IEEE
Item record/more information	http://hdl.handle.net/10197/11548
Publisher's statement	© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Publisher's version (DOI)	10.1109/MS.2019.2949322

Downloaded 2026-05-01 23:37:10

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd_oa)



© Some rights reserved. For more information

Guest Editors Introduction: IEEE Theme Issue on Software Engineering in Society

Modern software systems pervade our lives. They are increasingly becoming more open and hyper-connected; they manage large amounts of our personal data and are used to support the lives of individuals and communities, and the functioning of businesses and governments. They are *in* our society and they shape our society.

Software, in addition to its traditional roles, is increasingly playing a fundamental role in addressing major *societal* challenges from improving the quality of healthcare, manufacturing and transportation systems, to ensuring sustainable living, fighting cybercrime, protecting the rights of citizens, and reducing the digital divide. As such, software engineering has a duty to society beyond the narrow technical goals of creating systems that meet their cost, schedule, and quality goals.

Acknowledging the many critical roles embraced by software engineering practitioners in society today, we decided to organize a special issue on this topic. This special theme primarily concerned innovative, inspiring research with a clear impact on software engineering challenges, directions, methods and tools and one which focused on concerns beyond those of traditional software engineering research. We expected research contributions exploring synergies of software engineering with a broad spectrum of disciplines including, but not limited to: Health (e.g., health informatics, software technologies for ageing); Physical Sciences (e.g., computational chemistry, genomic, biotechnologies); Environmental Sciences (e.g., sustainability, urban planning, ecology, climate change); Social Sciences (e.g., ethics, software fairness, regulatory compliance); Management (e.g. socio-technical ecosystems, technical debt, social debt); Economics (e.g., electronic payments, blockchain technologies); Computing and Engineering (e.g., HCI, AI, data science, distributed computing); Security, Safety, and Privacy (e.g., security, safety and privacy preserving software development); Policing (e.g., combating and investigating crime); Manufacturing (e.g., Industry 4.0, smart factory); Engineering emerging cyber-physical systems (e.g., autonomous vehicles, smart cities); The arts (e.g. digital art, performing arts) and crafts (e.g. DIY electronics); Interdisciplinary research (e.g. cognitive science, digital social innovation). In addition, a particular, overarching focus was also on work emerging from research partnerships with communities, NGOs, cultural institutions and the public/private sector, as well as research reflections on the long implications of digital technology interventions on all aspects of society.

For this special issue, we received a total of 18 paper submissions. All submissions went through a rigorous peer review process. Each of them was evaluated by three conflict-free external reviewers. Relying on the reviewers' assessments we made an acceptance or rejection decision about each manuscript. The submissions we received concerned widely varying and complementary topics, which was exactly as we hoped.

One paper was concerned with gender diversity, particularly analyzing its impact to mitigate undesired communication patterns in software communities. One paper provided strategies

that developers can use to maximize security in practice using online communities. Four papers were concerned with how to incorporate ethics and human values into software production and in computer science education. Two papers discussed civic tech practices and the issues emerging over the years. Two papers focused on the development of software for sustainability. Three papers tackled the development of mobile apps and services for the “sharing economy” and for its citizens (e.g., to reduce the digital divide between farmers and the ICT infrastructure). One paper analyzed the social, practical, and methodological challenges of developing cyber-physical production systems in factories. Other papers reported on software patterns to represent human components, characterized developers using their code searches, reflected on knowledge transfer in software engineering, and described the results of a survey to understand the role of software in the public and private sectors.

After a rigorous peer review cycle, out of 18 submissions, 3 submissions were selected for inclusion in this special issue. The first manuscript surveyed 60 software practitioners to understand the correlations between gender diversity and communication anti-patterns found within software communities. One of the main findings of the paper is that although practitioners value communication skills, they seem not to perceive the phenomenon of gender diversity as an important factor that could be employed to mitigate the presence of community smells. The second accepted manuscript describes a collaborative Action Research project that demonstrates how Critical Systems Heuristics (CSH), can be used in the context of requirements engineering to gain a critical awareness of power and politics, to support critical reflection on boundary judgments, and to support requirements professionals in understanding the selectivity of their choices. In the the third manuscript the authors analyzed message exchanges on Stack Overflow to understand how security and secure practices were informally developed and disseminated among practitioners. The study suggests that informal learning through social interaction is the “middle path” for security practices. Sites like Stack Overflow are relevant resources for supporting informal learning, allowing developers to thoughtfully connect with, and tend to, security problems.

Handling this special issue was a great experience for us. This would not have been possible without the hard work of all of our reviewers, who provided important feedback to the authors of the submitted papers. We are also very grateful to the great support received by our editor, Ipek Ozkaya, which was critical for the preparation of this special issue. And we also wish to extend our sincere gratitude to the many authors, reviewers, and chairs of the Software Engineering in Society track at the International Conference on Software Engineering over the past five years. The research presented here builds on the pioneering foundations laid by those contributors.