Special Issue on 36th IEEE International Conference on Software Maintenance and Evolution (ICSME 2020)

The Research Track of the 36th IEEE International Conference on Software Maintenance and Evolution (ICSME 2020) was an opportunity for researchers and practitioners to present and discuss the most recent innovations, trends, experiences, and challenges in software maintenance and evolution. The Research track invited high quality submissions describing significant, original, and unpublished results related to software maintenance and evolution topics. We accepted 58 papers for presentation at the virtual conference.

For this special issue, we invited the authors of eight high-quality papers presented at the Research Track of ICSME 2020 to submit an extension of their paper. Each of these extended manuscripts was reviewed by at least three experts to guarantee the quality of the selected paper. For each paper, some of the original reviewers of the ICSME 2020 paper served as reviewers and new reviewers were also invited to provide an external perspective on the work. After the review process, we accepted the following five manuscripts for inclusion in this special issue:

- Quality GateKeepers: Investigating the Effects of Code Review Bots on Pull Request Activities by Mairieli Wessel, Alexander Serebrenik, Igor Wiese, Igor Steinmacher, and Marco A. Gerosa. This paper investigated 1,194 software projects on GitHub and interviewed 12 practitioners to understand the impact of bots on software code review. The results indicate the positive effects of the adoption of code review bots on pull request activities through the transparency and confidence the bot comments introduce.
- Using Contextual Knowledge in Interactive Fault Localization by Ferenc Horváth, Árpád Beszédes, Béla Vancsics, Gergő Balogh, László Vidács, and Tibor Gyimóthy. This paper proposes an interactive fault localization approach which exploits contextual knowledge of the user about the next item in the ranked list to reposition suspicious code entities. The effectiveness and practicality of the approach was evaluated using both simulated users and real users.
- Works for Me! Cannot Reproduce - A Large Scale Empirical Study of Non-reproducible Bugs by Mohammad M. Rahman, Foutse Khomh, and Marco Castelluccio. The paper investigated 576 non-reproducible bug reports and conducted a user study involving 12 developers to understand what leads to non-reproducible bugs and how to avoid them. The study found that links to existing bug reports help improve the reproducibility, and they developed a machine learning method to detect the connected bug reports which can offer complementary information to a non-reproducible bug.
- Test Smells 20 Years Later: Detectability, Validity, and Reliability by Annibale Panichella, Sebastiano Panichella, Gordon Fraser, Anand Ashok Sawant, and Vincent J. Hellendoorn. The paper performed a multi-stage, cross-validated manual analysis of hundreds of test suites written by developers and generated by test generation tools. Through this, they identified six types of test smells. The dataset was then used to benchmark two test smell detection tools. The results reveal a significant gap between the current test smells and real concerns.

- Deja Vu: Semantics-Aware Recording and Replay of High-Speed Eye Tracking and Interaction Data to Support Cognitive Studies of Software Engineering Tasks - Methodology and Analyses by Vlas Zyrianov, Cole S. Peterson, Drew T. Guarnera, Joshua Behler, Praxis Weston, Bonita Sharif, and Jonathan I. Maletic. The paper proposes a record-and-replay method to record all telemetry data (keystrokes, mouse movements, and gaze points) and replay them to map gaze points to the appropriate file, line, and column to perform analysis. The tool implementing the approach demonstrates the mapping correctness and its usefulness in the eye tracking studies in software engineering.

We would like to thank the authors of these papers. We are also grateful for the reviewers who provided detailed and constructive feedback to help the authors improve their papers. Finally, we would like to thank the editorial board of the Empirical Software Engineering journal, who provided the opportunity for this special issue and supported us throughout the process of preparing this special issue.

Kelly Blincoe and Zhenchang Xing
Program Co-Chairs ICSME 2020