

Guest Editorial

Selected Papers from TCSE 2016

Woei-Kae Chen¹, Alan Liu²

¹ Department of Computer Science and Information Engineering, National Taipei University of Technology, Taiwan

² Department of Electrical Engineering, National Chung Cheng University, Taiwan

wkc@csie.ntut.edu.tw and aliu@ee.ccu.edu.tw

With the theme, “Software Engineering for IoT,” the Twelfth Taiwan Conference on Software Engineering (TCSE 2016) was held in New Taipei city, Taiwan on June 7-8, 2016. TCSE is an annual meeting for the software engineering researchers and practitioners in Taiwan, offering an opportunity for exchanging views, experiences, and skills in different aspects of software engineering. With the recommendation from the organizers of TCSE 2016 and also a call-for-paper of JIT, we received manuscripts for reviews and four papers have been selected in this special issue.

When IoT is concerned, usually people think of IoT devices like sensors and actuators, followed by the topics like wireless network and middleware. In fact, there are many fields that researchers and practitioners can get involved in, from the hardware and software view points and from the system to application levels. Furthermore, the questions on whether such system is locally operated or widely available introduce new possibility of usage. Thus, connecting IoT systems to the cloud becomes an option. No matter what architecture is considered, the primary concern comes back to the quality of the system. This is how the theme “software engineering for IoT” comes into place.

In developing software related to IoT, the practitioners face challenges in heterogeneous system integration and system maintenance. Assuring the quality of the code becomes important. The first three papers focus on the quality of code through analyzing the API usage patterns, handling bad smells, and testing the GUI interaction. The fourth paper introduces a development platform of web services which are generally viewed as a possible extension to bring IoT devices into applications.

The first paper, “A Comment-Driven Approach to API Usage Patterns Discovery and Search,” by Lee et al. shows an automated method in discovering API usage patterns through mining the comments in source code. The authors have implemented their approach as an Eclipse plugin, called Codepus, and have shown promising results in analyzing open source projects for checking its reduced browsing time and better precision.

The second paper, “Enhancing Software Robustness by Detecting and Removing Exception Handling Smells – An Empirical Study,” by Hsieh et al. presents a systematic method of detecting and removing the smells of exception handling in Java code to improve the robustness of software. Generally, it takes a lot of manual effort in handling the smells and improving the code, so an automated tool is in demand. The authors have implemented an Eclipse plugin, called Robusta. They report their method through analyzing an open source project to show how to process the bad smells.

With the growing number of Android applications being created, the complexity caused by GUI interaction introduces a new challenge in testing. Manually tracing the interaction and building a model of such rich interaction become labor intensive. Liu and Chen describe their method of using a crawler in their paper, “A Crawling Approach of Hierarchical GUI Model Generation for Android Applications,” to automatically produce a GUI state model for analysis and testing. With an automated tool, a 100% state coverage can be achieved.

Considering the development of creating web services, Tseng et al. have built their platform, WSRush, to ease the developer in developing web services. The description of the system along with its usage is given in “The Design and Case Study of the WSRush Platform.” They have introduced the system to the classroom usage to examine its usefulness.

The guest editors would like to thank the authors for their contributions to this special issue. We are also grateful for the reviewers who provided constructive comments in many rounds of reviews and Ms. Sharon Chang for handling this special issues for JIT.

Guest Editors



Woei-Kae Chen received M.S. and Ph.D. degrees in Computer Engineering from North Carolina State University in 1988 and 1991, respectively. He is currently a Professor in the Department of

Computer Science and Information Engineering and the director of Software Development Research Center of the National Taipei University of Technology, Taiwan. He is also a member of the SEAT (Software Engineering Association, Taiwan). His research interests include software testing, software engineering, visual programming, and cloud computing.



Alan Liu received the PhD degree in electrical engineering and computer science from the University of Illinois at Chicago in 1994. He is a professor in the Department of Electrical Engineering, National Chung Cheng University in Taiwan. His research interests in artificial intelligence and software engineering include knowledge acquisition, requirements analysis, intelligent agents, and applications in embedded systems, smart home systems, and robotic systems. He is also a member of the IEEE, ACM, TAAI, and SEAT.