Journal of Universal Computer Science, vol. 24, no. 12 (2018), 1677-1679
submitted: 16/12/18, accepted: 20/12/18, appeared: 28/12/18 © J.UCS

Quality and Reliability Engineering: Trends and Future Directions

J.UCS Special Issue

Alok Mishra
(Department of Software Engg., Atilim University, Ankara Turkey
alok.mishra@atilim.edu.tr)

Sunil Kumar Khatri
(Amity Institute of Information Technology, Amity University, Noida, India
skkhatri@amity.edu)

P K Kapur
(Amity Center for Interdisciplinary Research, Amity University, Noida, India
pkkapur@amity.edu)

Uday Kumar
(Luleå University of Technology, Luleå, Sweden
uday.kumar@ltu.se)

Last decade of IT industry has observed an unprecedented growth of software and automation. Computers and software were already an integrated part of every research and development activity, but now these have penetrated into daily life of each human being through smart phones, mobile apps, and e-commerce. Various system and application software have brought the technology very close to ordinary man. Software development process has also gone through a lot of transition and now-a-days a rapid development of software has become a necessity. However, reliability and quality of software needs to be ensured along with faster development. Hence we can expect new dimensions and paradigms of software quality to be developed in the future.

Gone is the time, when software had to be written from scratch. Widespread availability of Application Programming Interface (APIs), Standard Template Library (STLs), reusable components and libraries have made software development faster, however, inter-dependency among heterogeneous components will be a crucial factor affecting software quality. Secure software is another major need of the day. Hence detection and plug-ins of software vulnerabilities especially for critical domains is going to be a prominent direction of future research. Some other significant factors of quality to attract attention of future researchers are usability, maintainability, portability and reusability. Correctness of the software is always first and foremost a requirement of any software and hence testability and its optimization is always a prominent as well as challenging area of research, which can get new solutions through techniques of machine learning and data analytics. To sum up, we can say that

software engineering is going to face the challenges of a fast, secure and reliable development, and maintainable software and researchers need to find solutions for these challenges.

Development of extremely useful embedded systems (e.g. smart phone) has been an instrumental factor in the success of IT-industry. On the other hand, their quality has not been up to the mark for many of such devices. Quality attributes of embedded systems differ from software quality attributes in many ways. Responsiveness, availability and recoverability are attributes which are more relevant for embedded systems, but appropriate metrics for these are yet to be defined correctly. Similarly quality attributes such as inter-operability, usability, portability need to be re-defined from the viewpoint of embedded system.. IT industry has come up with a lot of embedded system based products during the last 10-15 years without paying much attention to the above mentioned quality attributes which is going to be a key research area in near future.

Above mentioned domains can tolerate some degree of compromise in many aspects of the quality of systems. But real time systems and critical systems (like avionics systems, missile control systems, nuclear reactor systems, power grid control systems, etc) cannot be deployed without guaranteeing complete achievements of desired thresholds such as time constraints, precision, reliability, safety and security. Reliability computing, which consists of modeling and predicting, has become of great interest in recent years due to spacious arrays of complex systems and applications in our everyday safety, security and economic welfare. In future, we can expect some useful tools availability for correct and efficient modeling and evaluation of such complex systems.

Technologies such as machine learning, IOT, AI and robotics, IT based health care are going to rule the world in future. Phenomenal growth is expected in these directions. Many new models and techniques will be developed and maintaining a good quality of ever-growing technologies will become increasingly important. Hence, one possible direction of improving quality of such systems can be an incremental quality model. Incremental development processes (such as agile) can be augmented with incremental quality model and some researchers have already initiated research in this direction. Techniques like AI and machine learning are likely to provide big support for semantic analysis of systems. Nevertheless, efficient processing will be a challenge and data mining is more likely to be integrated for quality assessment and assurance in near future.

The guest editors would like to thank the managing editor, Christian Gütl, and the assistant editor, Dana Kaiser, for their assistance and support. Finally, would also like to thank to all authors and reviewers for their contribution towards this special issue.

The paper “A Software Reliability Growth Modeling Framework with Complexity of Path Searching” by Shinji Inoue and Shigeru Yamada introduce their research on a framework for software growth modeling with the difference of the stochastic property of the program path searching time, based on the infinite server queueing theory is proposed. The modeling framework describes the fault-detection phenomenon in the testing-phase generally by applying the infinite server queueing system. The numerical illustrations of the proposed models are given by using actual fault counting-data.

The paper "Software Quality Issues in SCRUM: A Systematic Mapping" by Deepti Mishra and Samia Abdalhamid analyze quality issues in Scrum which is a process framework used to develop complex software. As Scrum is one of the prominent approaches in agile development projects, it is significant to examine the issues of quality in the Scrum. A systematic mapping approach is adopted to answer specific research questions through an objective procedure to identify the nature of quality issues in Scrum studies.

The paper "A Logistic Fault-Dependent Detection Software Reliability Model" by Hoang Pham presents a logistic fault-dependent detection model where the dependent-rate of detected faults in the software can grow much faster from the beginning but grow slowly as the testing progresses until it reaches the maximum number of faults in the software.

The paper "Clustering for Software Remodularization by Using Structural, Conceptual and Evolutionary Features" by Amit Rathee and Jitender Kumar Chhabra analyze various dependency relations individually as well as in pairs, and clustering is carried out using three techniques over these permutations. Based on these analyses, a novel weighted dependency scheme is proposed to integrate three different dependency schemes and the optimal clustering technique is identified.

The paper "A Cross-Device Architecture for Modeling Authentication Features in IoT Applications" by Darwin Alulema, Javier Criado and Luis Iribarne proposes to cross-device architecture that combines the use of modeling techniques to describe and support the development of IoT applications and the inclusion of authentication mechanisms to provide users with a safe access to their personal information and ensure their privacy.

The paper "Dynamic Estimation of Temporary Failure in SoC FPGAs for Heterogeneous Applications" by J. Kokila, N. Ramasubramanian and Ravindra Thamma mainly focused on the reliability of a SoC design. As recent processors are shrinking in size due to the advancement of technology therefore, reliability is an important design parameter along with power, cost, and performance. This work proposes a dynamic thermal and voltage management (DTVM) system which ensures a reasonable level of fault tolerance. The fault tolerance system (FTS) identifies and subsequently can forecast temporary failures at run-time. The temporary failures are dynamically estimated on SoC FPGAs for a class of heterogeneous applications.

The paper "Statistical Usage Testing at Different Levels of Testing" by Kamaldeep Kaur, Sunil Kumar Khatri, Alok Mishra and Rattan Datta demonstrate the process and benefits of applying Statistical Usage Testing (SUT) at different levels of testing. Levels of testing include Unit level, Integration level, System level and Acceptance level.