2021 OkIP International Conference on Software Engineering Research & Development (SERD)

Monday, 15 November 2021 - Thursday, 18 November 2021

MNTC Conference Center & Online

Scientific Tracks

General and Social Aspects of Software Engineering (SE)

Program Analysis, Synthesis, and Repair
Programming Languages and Compilers
Specification and Modeling Languages
Tutoring, Documentation Systems
Software Engineering Standards and Guidelines
Software Economics and Metrics
Empirical Software Engineering
Green and Sustainable SE
Theoretic Approaches of SE
Domain Modeling and Meta-modeling
Aerospace Software and System Engineering
Architectural Analysis and Verification Methods
Enterprise Software, Middleware, and Tools
Quality Oriented Software Architecture
Reverse and Architectural Recovery Methods
Domain-Specific Software Engineering
Software Project Management Issues
Measurement and Empirical SE

Software Design, Testing, Evolution, and Maintenance

Software Design and Design Patterns
Software Evolution and Maintenance
Software Modeling and Design
Software Development(SD) | Process Modeling
Formal Methods in SE | Software Product Lines
Software Engineering Methodologies
Software Reuse and Reverse engineering
Configuration Management and Deployment
Crowdsourcing Software Engineering
Model-Based Software Engineering
Software Testing and Fault localization
Agile Software Engineering and Development
Validation and Verification
Requirement Engineering and Processes
Software Testing | Software Metrics
Evaluation and Analysis Technologies

Formal Methods and Theoretical Foundations

Software Specification/Development/Analysis/Verification
Theoretical Computer Science Fundamentals
Logic Calculi | Formal Languages |Automata Theory
Control Theory | Program Semantics | Type Systems/Theory
Theorem Provers | Programming Language Semantics

Denotational/Operational/Axiomatic Semantics
Sign-off Verification | Human-Directed Proof
Automated Proof | Verifier Verification
Specification Languages | Model Checkers
Visual Specification | Formal Techniques
Systems Level Specification | Reactive Systems Modeling
Model Checking Concepts/Principles/Algorithms/Tools
Linear□time Temporal Logic
Critical Systems Design Verification
Approaches/Tools for Verification/Validation
Requirements Formalization | Formal Specification
Usability/Case-studies/Applications of Formal Methods

Programming Languages (PLs), Systems, and Environments

Programming Languages and Perspectives
Object- Oriented (OO) Design and Analysis
OO technologies, systems and, applications.
Distributed Systems Composition
PLs Practical/Theoretical Investigations
Systems Practical/Theoretical Investigations
Environments Practical/Theoretical Investigations
PLs Requirement/Modeling/Prototyping/Design
PLs Implementation/Generation/Analysis
PLs Verification/Testing/Evaluation/Maintenance
OO Systems Modeling/Testing
New Tools/Techniques Development
Inovative Principles and Evaluations Methods
UML/MDA and AADL
OO Distributed Systems Modeling/Testing

Service Orientation and Human Interactions

Formal Methods and Theoretical Foundations
Service-Oriented Software Architectures
Service-Oriented Requirements Engineering
Middleware for Service-Based Systems
Service Discovery and Composition
Software services | Software visualization
Human-Computer Interaction
Usability Engineering | Gamification
Multimedia in Software Engineering
End-user software engineering
Human and social aspects of SE

AI, Web-Based Environments, and Adaptive Systems

Multi-Agent Systems | Mobile Agents
AI approaches to SE
Agent Architectures & Ontologies
Languages and Protocols
Intelligent CASE Tools and Issues
Mining Software Engineering Repositories
E-Commerce Solutions and Applications
Mobile Commerce Tech. and Applications
Web, Text Mining, and Semantic Web
Autonomic Computing and Adaptive Systems
Automated Software Engineering
Automated Software Design and Synthesis
Mobile applications
Search-based Software Engineering

Emerging SE Technologies and Dependability

Case Studies and Emerging Technologies
Novel Software Tools and Environments
Pervasive SE and Mission Critical Systems
Trust, Reliability, and Survivable Systems
Software Assurance and Dependability
Software Reliability and Security Methods
Engineering of Safety

Distribution, Componentization, and Collaboration

Formal Methods and Theoretical Foundations
Component-Based Software Engineering
Critical and Embedded Software Design
Distributed and Parallel Systems
Real-time Embedded Software Engineering
Cloud Computing | Distributed SE
Distributed and collaborative SE
Workflow Management | Team-Based SD
Computer-Supported Cooperative Work
Middleware, Frameworks, and APIs
Parallel, Distributed, and Concurrent Systems
Aspect-Oriented Software Engineering