



Jeff Offutt, PhD

Professor, Department of Computer Science

Education

PhD, Computer Science, Georgia Institute of Technology

Key Interests

Software Engineering | Software Testing | Software Engineering Education | Usable Security | Human-Computer Interface | Computer Science Education | Software Analysis

CONTACT

Phone: N/A | Email: offutt@gmu.edu

Website: <https://cs.gmu.edu/~offutt/>

SELECT PUBLICATIONS

- › N. Li & J. Offutt. Test oracle strategies for model-based testing. *IEEE Transactions on Software Engineering*, 43(4), 372-395 (2017).
- › K. Baral & J. Offutt. An empirical analysis of blind tests. *12th IEEE International Conference on Software Testing, Verification, and Validation (ICST)*, (2020).
- › J. Offutt, B. Lindstrom, & K. Baral. Teaching an international distributed discussion-based course. *The 15th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)*, (2019).

Research Focus

My research generally looks at issues of software quality. Software is not just ubiquitous in our lives, it is increasingly at the core of running our civilization. We depend on software to perform our work, for entertainment, for shopping, to satisfy government and societal requirements, for communication, our transportation, and almost every aspect of our lives. Quality problems, of any type, add friction by reducing what we can do and by increasing the difficulty of achieving our goals. This friction appears as lost money, lost time, and lost safety. Most of my research focuses on assessing the ability for software to behave as intended or desired (software testing), increasing the ability for software to be intuitively and efficiently used by humans (usability and human-computer interface design), and improving the ability for engineers to design high quality software (education, analysis, and maintenance). My philosophy is that software engineering research should address problems that practicing software engineers are currently facing, and lead to solutions that can, in the near to medium term, be applied in real settings.

Current Projects

- Software Test Oracles: I am trying to increase the automation of software testing, both to find more problems before they are inflicted on users, and to reduce the human cost of testing.
- Increasing Usable Security: I am exploring techniques to decrease the human cost of using software security protocols (such as password authentication) without reducing the overall security. Unusable security is not secure.
- K-12 Computing Education: I am studying how people learn to program, especially in elementary education, by focusing on developing computational thinking skills at an early age.
- Minimal Mutation: Mutation testing is arguably the most effective way to test software before releasing it, however, it is also the most expensive. We are developing a theory for how to dramatically reduce the cost of mutation testing.