Shanto Rahman

Ph.D. Student in Electrical and Computer Engineering
The University of Texas at Austin
Cockrell School of Engineering
U.S. Permanent Resident (EB-2)

Personal Website
shanto-rahman
LinkedIn
Scholar
shanto.rahman@utexas.edu

Research Interests

My research interests are at the intersection of **Software Engineering** and **Artificial Intelligence**, with a focus on **Software Testing** to make software systems reliable.

Education

Ph.D. Electrical and Computer Engineering — Aug 2021–Present
University of Texas at Austin (UT Austin)

M.S Software Engineering — Jan 2015–June 2016
University of Dhaka (DU)

B.S Software Engineering — Jan 2011–Dec 2014
University of Dhaka (DU)

Selected Publications (*Full List*)

*Publications include top-tier software engineering venues such as **ICSE**, **OOPSLA**, **ASE**, and **ICST**. >840 citations.*
Badges: ⬤ ⬤ *indicate ACM Artifact Evaluation awards (Available and Evaluated–Functional).*

15. **Shanto Rahman**, Saikat Dutta, and August Shi. Understanding and Improving Flaky Test Classification. In *Object-oriented Programming, Systems, Languages, and Applications (**OOPSLA**)*, Singapore, 2025. ⬤ ⬤

14. **Shanto Rahman**, Sachit Kuhar, Berk Cirisci, Pranav Garg, Shiqi Wang, Xiaofei Ma, Anoop Deoras, and Baishakhi Ray. UTFix: Change Aware Unit Test Repairing using LLM. In *Object-oriented Programming, Systems, Languages, and Applications (**OOPSLA**)*, Singapore, 2025. **Evaluated and Adopted by Amazon

13. **Shanto Rahman**, Bala Chanumolu, Suzzana Rafi, August Shi, and Wing Lam. Ranking Relevant Tests for Order-Dependent Flaky Tests. In *International Conference on Software Engineering (**ICSE**)*, Ottawa, Canada, 2025.

12. Talank Baral, Emirhan Oğul, **Shanto Rahman**, August Shi, and Wing Lam. OptCD: Optimizing Continuous Development. In *International Conference on Software Engineering (**ICSE**) Demo*, Ottawa, Canada, 2025.

11. **Shanto Rahman**, Abdelrahman Baz, Sasa Misailovic, and August Shi. Quantizing Large-Language Models for Predicting Flaky Tests. In *International Conference on Software Testing, Verification and Validation (**ICST**)*, Toronto, Canada, 2024.

10. **Shanto Rahman**, Aaron Massey, Wing Lam, August Shi, and Jonathan Bell. Automatically Reproducing Timing-Dependent Flaky-Test Failures. In *International Conference on Software Testing, Verification and Validation (**ICST**)*, Toronto, Canada, 2024.

9. **Shanto Rahman** and August Shi. FlakeSync: Automatically Repairing Async Flaky Tests. In *International Conference on Software Engineering (**ICSE**)*, Lisbon, Portugal, 2024. ⬤ ⬤

8. Talank Baral, **Shanto Rahman**, Bala Naren Chanumolu, Basak Balci, Tuna Tuncer, August Shi, and Wing Lam. Optimizing Continuous Development By Detecting and Preventing Unnecessary Content Generation. In *International Conference on Automated Software Engineering (**ASE**)*, Kirchberg, Luxembourg, 2023.

7. **Shanto Rahman**, Chengpeng Li, and August Shi. TSVD4J: Thread-Safety Violation Detection for Java. In *International Conference on Software Engineering (**ICSE**) Demo*, Melbourne, Australia, 2023.

6. Nazneen Akhter, **Shanto Rahman**, and Kazi Abu Taher. An Anti-Pattern Detection Technique Using Machine Learning to Improve Code Quality. In *International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, Dhaka, Bangladesh, 2021.

5. **Shanto Rahman**, Md Mostafijur Rahman, and Kazi Sakib. A Statement Level Bug Localization Technique Using Statement Dependency Graph. In *International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE)*, Porto, Portugal, 2017.

4. **Shanto Rahman** and Kazi Sakib. An Appropriate Method Ranking Approach for Localizing Bugs using Minimized Search Space. In *International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE)*, Rome, Italy, 2016.

3. **Shanto Rahman**, Md Mostafijur Rahman, M. Abdullah-Al-Wadud, Golam Dastegir Al-Quaderi, and Mohammad Shoyaib. An Adaptive Gamma Correction for Image Enhancement. *EURASIP Journal on Image and Video Processing (JIVP)*, no. 1 (2016): 35. **Adopted by National Institutes of Health (NIH)

2. **Shanto Rahman**, Kishan Kumar Ganguly, and Kazi Sakib. An Improved Bug Localization Using Structured Information Retrieval and Version History. In *International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, 2015.

1. Md Mostafijur Rahman, **Shanto Rahman**, Minhas Kamal, M. Abdullah-Al-Wadud, Emon Kumar Dey, and Mohammad Shoyaib. Noise Adaptive Binary Pattern for Face Image Analysis. In *International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, 2015. **Best Paper Award

💼 Professional Experience

Graduate Research Assistant, University of Texas at Austin　　　　　　　　　　Aug 2021–Present
　– Worked on software engineering leveraging machine learning models (AI4SE)

Research Intern, Google　　　　　　　　　　　　　　　　　　　　　　　　May 2025–Aug 2025
　– Worked on changelist (CL) culprit prediction

Applied Scientist Intern, Amazon Web Services (AWS)　　　　　　　　　May 2024–Sept 2024
　– Worked on change aware unit test repair using LLM (Published in OOPSLA'25 [14])

Lecturer, Bangladesh University of Professionals (BUP)　　　　　　　　Sept 2017–Jan 2021
　– Taught eight courses •Conducted research •Designed course materials

Senior Software Engineer, Samsung Research　　　　　　　　　　　　　July 2016–July 2017
　– Developed Android and Tizen apps in Java

Software Engineering Intern, Orion Informatics Ltd.　　　　　　　　　　Jan 2014–Aug 2014
　– Worked on a project namely Browser Based Editing (BBE)

🖥 Teaching

University of Texas at Austin (UT Austin)
Teaching Assistant, Department of Electrical and Computer Engineering　　　　　Jan 2025–May 2025
　– ECE 422C: Software Design & Implementation II

Guest Lecturer, Department of Electrical and Computer Engineering　　　　　　　　　Sept 2025
　– ECE 382V: Software Testing in the Era of Nondeterminism

Bangladesh University of Professionals (BUP)　　　　　　　　　　　　　Sept 2017–Jan 2021
Lecturer, Department of Information and Communication Technology (ICT)
　– Taught Courses (total 8 courses)

　　　•ICE 2109: Object-Oriented Programming •ICE 2201: Data Structures •ICE 2203: Database Management System •ICE 3101: Analysis and Design of Algorithm •ICE 3103: Operating System •ICE 3207: Software Engineering •ICE 3206: Software Testing •ICE 4107: Artificial Intelligence

　– Supervised undergraduate and graduate students

🏆 Awards

- **EECS Rising Stars**, MIT   Oct 2025
- **Honorably Invited to NextProf Nexus,** UC Berkeley   Sept 2025
- **Research Credit Award,** OpenAI   Aug 2025
- **CAPS Student Travel Award for ICSE**, ACM SIGSOFT   2025
- **Temple Foundation Graduate Fellowship,** UT Austin   2024-2025
- **Travel Award**, IEEE Technical Committee of Software Engineering (TCSE)   2024
- **Professional Development Award**, UT Austin   2023
- **Cockrell School of Engineering Fellowship**, UT Austin   2021-2022
- **Research Grant**, University Grants Commission (UGC), Bangladesh   2019-2020
- **Research Grant**, University Grants Commission (UGC), Bangladesh   2018-2019
- **Research Fellowship**, Ministry of Information and Communication Technology, Bangladesh   2015-2016
- **Best Paper Award**, International Conf. on Computer and Information Technology (ICCIT), IEEE   Dec 2015
- **Merit Award (Top 2%)**, Kabi Sufia Kamal Hall, University of Dhaka, Bangladesh   2016
- **Merit Scholarship**, Dhaka University Alumni Association   2015

👉 Services

- **Reviewer** - ACM Transactions on Software Engineering and Methodology (TOSEM)   July 2025
- **Reviewer** - Transactions on Software Engineering (TSE)   Apr 2025
- **PC member** - International Flaky Tests Workshop (FTW), co-located with ICSE   2025
- **PC member** - Artifact Evaluation, ICSE   2025
- **Shadow Reviewer** - ICSE, FSE, ASE, ISSTA   2024, 2025
- **Amazon UT Campus Brand Ambassador**   2025
- **Co-organizer** - UT-Cornell Software Engineering Seminar   2024
- **Judge** - Capital of Texas Undergraduate Research Conference   2023
- **Reviewer** - Journal of Information and Software Technology   2020
- **Committee Member** - Academic Curriculum Review Committee, BUP   2020
    - Reviewed and modified undergrad curriculum for the department of ICT
- **Mentor** - National Hackathon on Frontier Technologies, Ministry of ICT   Jan 2020–Feb 2020
    - One of my mentored teams won first place in the waste management category
- **Mentor** - ACM ICPC, NCPC, NGPC   Jan 2019–Jan 2021
    - International and national programming contest
- **Moderator** - BUP Infotech Club (BUPITC)   July 2019–Dec 2020
    - Focuses on arranging programming contest, hackathon, idea contest, workshop
- **House Tutor** - BUP   Apr 2020–Dec 2020
    - Student counseling about their residence
- **Student Advisor** - Dept. of Information and Communication Technology (ICT), BUP   Jan 2018–Dec 2020
    - Student progress monitoring • Student counseling

👥 RESEARCH ADVISING

- Jiaju Wang (MS, UT Austin)
  - Working on evaluating nondeterminism in LLM models
- Nandita Jayanthi (MS, UT Austin)
  - Developed FlakeSync as a Maven plugin
- Bala Chanumolu (MS, GMU); Current: Software Engineer, Amazon Web Services (AWS)
  - Co-authored ICSE'25 [13], ASE'23 [8]
- Emirhan Oğul (BS, GMU)
  - Co-authored ICSE Demo'25 [12]
- Başak Balcı (BS, Özyeğin University); Current: MS, Technical University of Munich (TUM)
  - Co-authored ASE'23 [8]
- Tuna Tuncer (BS, Özyeğin University); Current: MS, Technical University of Munich (TUM)
  - Co-authored ASE'23 [8]
- Nazneen Akhter (MS, BUP; Current: Faculty at BUP)
  - Co-authored ICICT4SD'21 [6]
- Sadia Khan Rupa (BS, BUP; Current: University of Applied Sciences Osnabrück)
  - Worked on developing image classification AI model
- Zannatul Ferdous (BS, BUP)
  - Coached National Hackathon on Frontier Technologies
- Abir Munna (BS, BUP)
  - Coached National Hackathon on Frontier Technologies

🐙 NOTABLE OPEN-SOURCE CONTRIBUTIONS

- International Dataset of Flaky Tests (IDoFT)
  - IDoFT is a public dataset for flaky-test research. I integrated **170 timing-dependent (TD)** flaky tests and **1,900+ order-dependent (OD) related flaky tests**. IDoFT is available https://github.com/TestingResearchIllinois/idoft (My GitHub ID shanto-rahman).
- TSVD4J
  - I present TSVD4J, a Maven plugin for detecting thread-safety violations in Java applications. TSVD4J integrates with any Maven project, runs the project's test suite, and analyzes runtime behavior to expose data races. TSVD4J is evaluated on 12 Java applications, and identified **55 data race pairs** indicative of data race bugs. Compared to RV-Predict, TSVD4J detects more data races with similar runtime, largely due to its explicit tracking of field accesses. I presented TSVD4J at ICSE Demo'23. Repository is https://github.com/UT-SE-Research/TSVD4J.
- OptCD
  - I develop OptCD, a dynamic analyzer for CD pipelines that traces build outputs to expose unused artifacts and wasted work. In evaluation, OptCD enables required changes for **72.0% of unused directories**. OptCD is presented at ICSE Demo'25. I submitted 26 GitHub pull requests to open-source Maven projects (e.g., google/open-location-code, junit-team/junit4, JSQLParser/JSqlParser), with **12 accepted**. https://github.com/software-research/optCD-demo.
- FlakeSync
  - I present FlakeSync, a technique for automatically repairing async-wait flaky tests by introducing synchronization between the critical point and barrier point for a specific test execution. The evaluation is on known flaky tests from a prior dataset, which shows that FlakeSync can automatically repair 83.75% of the async-wait flaky tests. I presented this paper in ICSE'24. I submitted 10 pull requests based on FlakeSync's patches, with **3 accepted pull requests** and none rejected thus far. The repository is https://github.com/UT-SE-Research/FlakeSync.
- FlakeBench
  - I present FlakeBench, a dataset for identifying the root causes of flaky tests via a fine-tuned LLM. **The dataset contains 8,574 tests: 280 flaky and 8,294 non-flaky**. Of these 280 flaky tests, async wait (76), concurrency (37), time (33), order dependent (41) and unordered collection (93) are five different types of flaky tests. I presented this dataset at OOPSLA'25. The repository is https://github.com/UT-SE-Research/FlakyLens/tree/main/FlakeBench.

- FlakyLens
  - I present FlakyLens, a fine-tuned LLM classifier for flaky tests, built on a corrected experimental setup and a realistic dataset (FlakeBench) that we construct. Prior work reported very high accuracy for fine-tuned LLM classifiers, but we show these results were inflated by flawed experimental design and unrealistic datasets. Building on these insights, **FlakyLens improves F1 by 9.54 percentage points over state-of-the-art techniques**. We make our code and models publicly available. The repository is https://github.com/UT-SE-Research/FlakyLens.
- Syn-Bench
  - I present Syn-Bench, a dataset that is used for repairing change-induced test breakage. The main intuition is that code changes but test is outdated, results in spurious assertion failure, and reduced code coverage. Syn-Bench consists of **352 unit tests with assertion failure and 369 unit tests with reduced code coverage** due to changes in the focal methods from 44 projects. I presented this dataset in OOPSLA'25. The repository is https://sites.google.com/view/utfix.
- FlakeRake
  - I present FlakeRake, an automated approach for reproducing Timing Dependent (TD) test failures. FlakeRake outputs configurations that a developer can use to run tests to more reliably reproduce TD-test failures, debug them, and fix them without needing to repeatedly rerun tests. FlakeRake reproduces 136 failures at least once, with **93 reaching with more than 99% reproduction**. In contrast, 10,000× naive reruns reproduce only 115 failures, with none achieving more than 50% failure rate. I implemented FlakeRake for Java, and make it publicly available. I presented FlakeRake in ICST'24. The repository is https://github.com/gmu-swe/flakerake.
- FlakyQ
  - I present FlakyQ, an automated approach for quantizing LLMs to improve the runtime and computational resources needed to predict flaky tests and flaky test categories, allowing them to run effectively in CPUs. FlakyQ **saves 25.4% in prediction time** over all tests, along with a **48.4% reduction in memory usage**. I presented FlakyQ in ICST'24. I make data, code, and scripts publicly available at https://github.com/UT-SE-Research/FlakyQ.

🎤 Presentations (Selected)

Poster	Understanding and Improving Flaky Test Classification, EECS Rising Stars, *Massachusetts Institute of Technology (MIT)*, Cambridge, Oct 2025
Conference Talk	UTFix: Change Aware Unit Test Repairing using LLM, In *Object-oriented Programming, Systems, Languages, and Applications (**OOPSLA**)*, Singapore, Oct 2025
Conference Talk	Understanding and Improving Flaky Test Classification, In *Object-oriented Programming, Systems, Languages, and Applications (**OOPSLA**)*, Singapore, Oct 2025
Guest Lecture	UTFix: Change Aware Unit Test Repairing using LLM, *Software Testing in the Era of Nondeterminism (ECE 382V)*, UT Austin, Sept 2025
Guest Lecture	Understanding and Improving Flaky Test Classification, *Software Testing in the Era of Nondeterminism (ECE 382V)*, UT Austin, Sept 2025
Invited Talk	Changelist (CL) Culprit Prediction in Google system, *Google PhD Intern Summit*, Mountain View, California, Aug 2025
Seminar Talk	UTFix: Change Aware Unit Test Repairing using LLM, *Columbia University*, New York, Aug 2025
Seminar Talk	Understanding and Improving Flaky Test Classification, *UT-Cornell Software Engineering Seminar*, Austin, Texas, Aug 2025
Seminar Talk	UTFix: Change Aware Unit Test Repairing using LLM, *UT-Cornell Software Engineering Seminar*, Austin, Texas, Aug 2025
Seminar Talk	Changelist Culprit Prediction in Google systems, *Google*, Mountain View, California, July 2025
Invited Talk	FlakeSync: Automatically Repairing Async Flaky Tests, *Google TAP Research Meeting*, Mountain View, California, June 2025
Conference Talk	Ranking Relevant Tests for Order-Dependent Flaky Tests, In *International Conference on Software Engineering (**ICSE**)*, Ottawa, Canada, May 2025
Invited Talk	Ranking Relevant Tests for Order-Dependent Flaky Tests, *ECE Outstanding Student Lecture Series*, UT Austin, Feb 2025
Invited Talk	Reproducing Flaky Tests and its Mitigation, *George Mason University*, Virginia, Aug 2024
Seminar Talk	Change Aware Unit Test Repair, *Amazon Web Services*, New York, July 2024

Conference Talk	Quantizing Large-Language Models for Predicting Flaky Tests, In *International Conference on Software Testing, Verification and Validation (ICST)*, Toronto, Canada, May 2024
Conference Talk	Automatically Reproducing Timing-Dependent Flaky-Test Failures, In *International Conference on Software Testing, Verification and Validation (ICST)*, Toronto, Canada, May 2024
Conference Talk	FlakeSync: Automatically Repairing Async Flaky Tests, In *International Conference on Software Engineering (ICSE)*, Lisbon, Portugal, Apr 2024
Seminar Talk	FlakeSync: Automatically Repairing Async Flaky Tests, *UT-Cornell Software Engineering Seminar*, Austin, Texas, Apr 2024
Poster	FlakeSync: Automatically Repairing Async Flaky Tests, *Graduate and Industry Networking (GAIN)*, UT Austin, Jan 2024
Invited Talk	FlakeSync: Automatically Repairing Async Flaky Tests, *ECE Outstanding Student Lecture Series*, UT Austin, Feb 2024
Conference Talk	TSVD4J: Thread-Safety Violation Detection for Java, In *International Conference on Software Engineering (ICSE)*, Melbourne, Australia, May 2023
Invited Talk	TSVD4J: Thread-Safety Violation Detection for Java, *GWGMC Research Symposium*, UT Austin, Feb 2023
Seminar Talk	TSVD4J: Thread-Safety Violation Detection for Java, *UT-Cornell Software Engineering Seminar*, Austin, Texas, Feb 2023
Conference Talk	Appropriate Method Ranking Approach for Localizing Bugs using Minimized Search Space, In *International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE)*, Rome, Italy, Apr 2016
Conference Talk	An Improved Bug Localization using Structured Information Retrieval and Version History, In *International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, Dec 2015
Conference Talk	Image Enhancement in Spatial Domain: A Comprehensive Study, In *International Conference on Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, Dec 2014