Nata Stulova

contact info

nata@stulova.me
web:// **stulova.me**
LinkedIn:// **nata-stulova**

education

PhD in Software, Systems and Computing cum laude
2014–2018
Technical University of Madrid (UPM)

MSc in Artificial Intelligence
2012–2013
Technical University of Madrid (UPM)

BSc in Systems Analysis
2008–2012
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" (NTUU "KPI")

developement

Java, C++, Python, Prolog
♥ bash, git
GitLab, Phabricator
markdown, LaTeX
WordPress, Wix

languages

Ukrainian **native**
English, Spanish **advanced**
German **intermediate**
French, Hebrew **beginner**

awards

Forbes Ukraine 2023 list of women leaders in Ukrainian science

whoami

I am a software engineering researcher, working on applied empirical and formal software analysis projects in industry with over 10 years of prior academic research experience.
Since the start of the full-scale russian invasion into Ukraine of 2022 I spend a share of my time as a volunteer, organizing information campaigns, events and rallies **#StandWithUkraine**

experience

Staff Research Scientist | MacPaw | 2023-current remote / Kyiv, Ukraine

applied software engineering research on Apple app ecosystem > Leading and contributing to research projects in the areas of software engineering, analysis, distribution, combining empirical, formal, and neuro-symbolic tools and techniques. Establishing university partnerships (MIT, NaUKMA) for student research internships. Selected public research projects are published as *[12--15]*.

Research Writer | MacPaw | 2022-2023 remote / Kyiv, Ukraine

research projects audit > Participated in the systematization of the company's internal research projects and worked on several iterations of what has become research.macpaw.com. Rest is NDA.

Senior Researcher | University of Bern | 2020-2021 remote / Bern, Switzerland

code and documentation analysis > As a team leader, project manager, and engineer working with several distributed R&D teams (4-6 people each) on source comments quality analysis, I have: established a collaboration between 4 research institutions for a systematic literature review on code comment quality research trends *[11]*, led data analysis and visualization of an empirical study on developer adherence to coding style guidelines *[10]*, and worked on a comment clone detection tool that found 1300+ API documentation issues in 10 major Java libraries and systems *[9]*.

requirements and documentation engineering > In a team of 4 researchers I have managed the work on tool support for direct in-IDE integration of the source code and non-code software artifacts (code-linked mind maps, Kanban boards, user stories) for use in BDD workflows *[7-8]*.

teaching > at BSc and MSc levels, in person and fully remote: developed from zero a series of practical algorithms and data structures lectures within the Software Skills Lab course (lecture slides and videos, practical assignments, exams), co-supervised MSc and BSc theses, gave lectures on programming languages, software verification, and UI design.

Scientist | Swiss Federal Institute of Technology in Lausanne (EPFL) | 2019-2020 Lausanne, Switzerland

code and documentation analysis> I have worked on natural language processing (NLP) use for augmenting software analyses, establishing and leading a collaboration between 2 research institutions on a project for static detection of code-comment inconsistencies during code change *[6]*.

Research Assistant, Software Engineering | IMDEA Software Institute | 2014-2018 Madrid, Spain

static and dynamic code analysis> I have worked on program specification languages design, and on tools and techniques for specification-based source code analysis and verification. In a team working on Ciao, a dynamic Prolog-based language, its formal specification language of assertions, and its static and dynamic verification frameworks, I have formalized and implemented compiler support for assertions language extensions of dynamic analysis of higher-order function calls *[1]*, several source-to-source translation optimizations to reduce run-time checks overhead by 1-2 orders of magnitude *[2]*, including by incorporating static analysis results *[3-4]*, and worked on a framework for static inference of run-time checks added compute costs *[5]*.

service

Steering Comittees > NLBSE

(Co-)chair > CICLOPS'17, NLBSE'25 tool competition, NLBS'26

Reviewer
journals > TSE, TCSS, EMSE, JOSS, Fundamenta Informaticae
conferences > LOPSTR, ICLP

speaker

2024> panelist and speaker at INSCIENCE'24
2025> guest lecturer at Kyiv School of Economics

driving

Permit category **B** manual

calligraphy

personal exhibition > "Lines of the women's words", works of regional women poets in Ukrainian Skoropys technique, 2021, **Literary Prydniprovia Museum**, Dnipro, Ukraine

volunteering

Secretary, Web master, Event organizer | Ukrainer in Bern | 2022--current remote / Bern, Switzerland

public outreach > (Co-)organizer of scheduled and spontaneous rallies, information campaigns, and collaborations between different Swiss-Ukrainian NGOs. Website maintenance, flyer design, SMM. But I don't repair printers there at least.

Business analyst, Project manager, Web Developer | Ksi Prostir | 2020-2021 remote / Dnipro, Ukraine

digital transformation > developing a website for a Dnipro-based cultural space KsiProstir. I have worked on the initial requirements analysis, after which I had collaborated in the no-code web development and maintenance.

publications

[1] Assertion-based Debugging of Higher-Order (C)LP Programs, **PPDP'14**, *N. Stulova, J. F. Morales, M. V. Hermenegildo*

[2] Practical Run-time Checking via Unobtrusive Property Caching, **ICLP'15**, *N. Stulova, J. F. Morales, M. V. Hermenegildo*

[3] Some Trade-offs in Reducing the Overhead of Assertion Run-time Checks via Static Analysis, **SCP volume 155**, *N. Stulova, J. F. Morales, M. V. Hermenegildo*

[4] Exploiting Term Hiding to Reduce Run-time Checking Overhead, **PADL'18**, *N. Stulova, J. F. Morales, M. V. Hermenegildo*

[5] Static Performance Guarantees for Programs with Run-time Checks, **PPDP'18**, *M. Klemen, N. Stulova, P. López-García, J. F. Morales, M. V. Hermenegildo*

[6] Towards Detecting Inconsistent Comments in Java Source Code Automatically, **SCAM'20**, *N. Stulova, A. Blasi, A. Gorla, O. Nierstrasz*

[7] First-class Artifacts as Building Blocks for Live in-IDE Documentation, **SANER'22**, *N. Patkar, A. Chiş, N. Stulova, O. Nierstrasz*

[8] Interactive Behavior-driven Development: a Low-code Perspective, **LowCode'21**, *N. Patkar, A. Chiş, N. Stulova, O. Nierstrasz*

[9] RepliComment: Identifying Clones in Code Comments, **JSS volume 182**, *A. Blasi, N. Stulova, A. Gorla, O. Nierstrasz*

[10] Do Comments follow Commenting Conventions? A Case Study in Java and Python, **SCAM'21**, *P. Rani, S. Abukar, N. Stulova, A. Bergel, O. Nierstrasz*

[11] A Decade of Code Comment Quality Assessment: A Systematic Literature Review, **JSS volume 195**, *P. Rani, A. Blasi, N. Stulova, S. Panichella, A. Gorla, O. Nierstrasz*

[12] Position Paper: Think Globally, React Locally — Bringing Real-Time Reference-Based Website Phishing Detection on macOS, **STAST'24**, *I. Petrukha, N. Stulova, S. Kryvoblotskyi*

[13] State of the Application Sandboxing on macOS: A Differentiated Replication, **Preprint**, *I. Pastukhova, I. Synytsia, N. Stulova*

[14] Towards Generating App Feature Descriptions Automatically with LLMs: the Setapp Case Study, **FORGE'25**, *Y. Peteliev, I. Synytsia, N. Stulova*

[15] SwiftEval: Developing a Language-Specific Benchmark for LLM-generated Code Evaluation, **FORGE'25**, *I. Petrukha, Y. Kurliak, N. Stulova*