# Nikita Bobko

Amsterdam, NL | jobs@nbobko.com | github.com/nikitabobko | linkedin.com/in/nikitabobko | nbobko.com

- EU work authorization: Dutch national
- US work authorization: Requires sponsorship
- UK work authorization: Requires sponsorship
- Open to: relocation / remote

## Personal Projects

**AeroSpace - a tiling window manager for macOS** *May 2023 – Present*

The author and the maintainer of the project

https://github.com/nikitabobko/AeroSpace

- More than 20k stars on GitHub
- Top 100 Homebrew casks by installs
- 60 Contributors
- Written by hand. The project predates the AI era
- The project suggests a novel approach for managing Virtual Desktops on macOS
- Multithreaded. The project makes use of Swift 6 Strict Concurrency mode to catch concurrency bugs

## Experience

**Independent: AeroSpace** *Amsterdam, NL*

OSS Maintainer *Jun 2026 – Present*

- AeroSpace maintenance
- Conference talk at KotlinLeeds: Things I would change in Kotlin Coroutines design

**JetBrains, Kotlin Language Evolution** - 3 years and 9 months *Amsterdam, NL*

Senior Software Engineer and Language Designer *Aug 2022 – May 2026*

- Language designer and author of various Kotlin language proposals:
  - KEEP-0416 Collection Literals
  - KEEP-0326 Non-local break and continue
  - KEEP-0391 KMP Kotlin-to-Java direct actualization
  - KEEP-0443 CoroutineContext context parameter in suspend functions
  - Multishot continuations design doc
- Delivered the major visibility change of the auto-generated `copy` method of data classes with private constructors: KT-11914. It was one of the most popular requests in the Kotlin issue tracker. It is a big breaking change in the Kotlin ecosystem.
- Investigated the potential integration points between Kotlin and OpenJDK Project Loom. The result of the work was issue proposals (KT-78467, KT-79151), design document and the discussion on the OpenJDK mailing list (June, July, August) that identified the missing functionality in Java's ScopedValues API.
- Improved and refined various aspects of the expect/actual language feature for its stable release. That includes designing the language feature, narrowing down its scope and writing the implementation.
- Co-lectured a Kotlin course at Constructor University Bremen in spring 2026. Lectures' topics: Threads, Async programming, Coroutines, Continuations, Sequences (Lazy vs Eager), Android, Compose
- Guest lecturer at several European universities (TU Delft, University of Warsaw, CTU Prague). Taught the basics of Kotlin and more advanced topics like data-flow analysis and flow-sensitive typing.
- Implemented the fake-overrides subsystem in the K2 kotlin-reflect library. The fake-overrides subsystem is the backbone of the popular `KClass.members` API in the kotlin-reflect library. Fake-overrides subsystem answers such fundamental questions as: "which function overrides which other functions".
- Owned the non-local break and continue language feature: KT-1436.
- Owned the data object language feature: KT-4107

**JetBrains, Kotlin IDE plugin and internal infra** - 2 years and 4 months *Apr 2020 – Dec 2020 (Moscow, Russia)*
Senior Software Engineer *Jan 2021 – Aug 2022 (Amsterdam, NL)*

- One of the key figures behind the one-year effort of Kotlin IDE plugin migration from the Kotlin compiler repo to the intellij-community repo. Both are big open-source git repositories with different policies, different CI infrastructures, build systems, and release cycles.
  - I set up and automated the new TeamCity CI infrastructure from the ground up.
  - I have designed infrastructure enabling synchronized release cycles for the Kotlin compiler and IDE plugin despite their separate repositories.
  - I authored one of the commits with the most lines removed in the Kotlin repo.
- Another important outcome was the implementation of the Kotlin JPS plugin manager - a subsystem that makes sure that the right version of the Kotlin compiler is used in JPS (JetBrains' internal build tool, similar to Maven and Gradle) and Maven projects. KTIJ-11633

**JetBrains, RubyMine team** - 1 year and 9 months *Moscow, Russia*
Intern, Junior Software Engineer *Jul 2018 – Apr 2020*

- Delivered the MVP of an open-source Ruby dynamic types analyzer tool. The tool captures types at runtime and provides hints to the IDE later.
- Later, switched over to the main project in the team - RubyMine. Primarily worked on the static analysis, symbolic execution, and type inference engines inside the IDE.

**Russian Academy of Sciences** - 11 months *Moscow, Russia*
Software Engineer *Jul 2017 – Jun 2018*

- Worked on a C# static code analysis tool using the Roslyn analyzer. Added support for arrays in the existing symbolic execution engine.

## Public Speaking

**KotlinLeeds 2026 | Things I would change in Kotlin Coroutines design** YouTube link *Oct 2026*

**Podlodka podcast | Window managers (in Russian)** YouTube link *Mar 2026*

**KotlinConf 2025 | Swift concurrency for curious Kotliners** YouTube link *May 2025*

## Education

**Moscow State University** *Moscow, Russia*
Bachelor's degree in Applied Math and Computer Science *Sep 2016 – Aug 2020*

## Technical Skills

- Languages: Kotlin, Swift, Java, Shell, Rust, Haskell, Nix, Ruby, C#, C, SQL
- Software Engineering: Language design, API design, Concurrency, Distributed systems, Local-first software, Reactive programming, Functional programming
- Kotlin: Kotlin Multiplatform, Kotlin Coroutines
- Programming language theory: Continuations, Data-flow analysis, Flow-sensitive typing, Mutable value semantics
- Java / JVM: Virtual Threads, Java memory model, JMH, Java binary compatibility
- Apple Platforms: Swift Fearless Concurrency, Swift Actors, macOS Accessibility API
- Developer Tools & Systems: Git, Version control in general (jj, Pijul, etc.), Monorepos, Vim, CLI design, Unix/Linux, Homebrew, Tiling Window Managers
- Android: Compose UI Framework, Storage Access Framework (Scoped storage)
- Build & Infrastructure: TeamCity, GitHub Actions, Homebrew packaging, CI/CD
- Build systems: Gradle, Maven, Bazel, Swift Package Manager, Xcode
- Open Source: maintenance, community/issue management
- Cybersecurity: Personal security hygiene, Asymetric/Symmetric encryption, pass unix password manager, SSH, GPG, git-crypt

## Languages

Russian (Native), English (Fluent), Dutch (A2 - B1)