

joseph schaffer

Software engineer in Seattle. At Amazon, I lead the design of a compiled programming language for safe, secure user interfaces, used across Amazon shopping. Outside work, I tinker, build bikes, and make small, designy apps I'd want to use myself.

WORK

Amazon

Seattle

SDE I 2014 SDE II 2016 Senior SDE 2019

2018



8 yrs

A programming language for user interfaces

ROLE	Tech lead
TEAM	10-12 engineers
SCALE	part of ~80% of amazon.com pages
SECURITY	Amazon's preferred XSS defense
FOCUS	language design, type systems, compilers, interop with other languages
STACK	Kotlin compiler; JVM and JavaScript targets
TOOLING	language server, MCP server

I'm tech lead of the team that builds and owns an internal programming language for user interfaces: server-rendered, client-rendered, sparsely hydrated, and native. It's statically typed, compiles to the JVM and JavaScript, and has grown from a safe HTML templating language into a full UI language that lives alongside TypeScript, with a language server for editors and an MCP server for AI agents.

I set its direction with the team: a north star of an approachable language that catches problems in the compiler, so engineers and AI agents alike can build UI quickly and safely across Amazon's polyglot stacks, and a plan each year that moves it closer. I ground those calls in industry analysis, in what its users tell us, and in the compile-time metrics we built to catch regressions before they ship.

Its guarantees are built in. The compiler knows where every value lands in the markup, and its runtime escapes each one for exactly that place, so templates are safe from cross-site scripting by construction. Amazon's security organization names it the first-choice defense in server-rendered HTML. I led the ground-up redesign of its type system: familiar to any TypeScript developer, with types that are sound by construction, and every template's signature inferred.

I designed and built its reactivity, the model React, Svelte, and SwiftUI share: when state changes, only the markup that depends on it updates. The compiler tracks state through the type system, and each template compiles to a server render plus a client bundle that hydrates only its reactive regions. I led five engineers in delivering it, and it runs across thousands of Amazon shopping experiences.

Along the way, I've mentored engineers to seven promotions: five to SDE II, two to senior.

2016

II

2 yrs

A front-end build system and asset pipeline

SCALE	thousands of teams across Amazon shopping	The front-end build tooling behind Amazon's retail pages (similar to Grunt, Gulp, or webpack), paired with the system that ships their assets to the CDN. Every page builds as a matrix of variants — device × locale × language × A/B test — with each combination sliced to carry only what it needs (the key to fast pages before HTTP/2), so each new dimension multiplies the build. As a core contributor, I introduced switching tools that let teams declare variants without building every combination, keeping builds fast as the matrix grows, and designed the plugin framework teams extend it with.
STACK	JVM core (Java, Kotlin), Ruby, JavaScript	

2014

II

2 yrs

A front-end framework and component library

TEAM	10-20 engineers	The foundation of Amazon's retail pages. I worked across it — the carousel on mobile, tablet, and desktop, sweeping refactors, a TV navigation interface for the web — and designed how new versions were released to product teams.
------	-----------------	--

ON THE SIDE

Schaffernation

Seattle

Founder

2026

I

<1 yr

A one-person studio for small, designy apps and tools

Trailer Trash	All the latest trailers, sorted by what's coming, playing, or gone from theaters. trailertrash.app
Vintage NYC Guessr	Guess where historical photos of New York were taken. schaffernation.com/nycguessr
Seattle Sports Today	Is Seattle playing today? Plan your commute accordingly. seattlesportstoday.com

SCHOOL

University of Pennsylvania

Philadelphia

2010

IIII

4 yrs

DEGREE	B.S.E.
MAJOR	Computer Science
MINOR	Mechanical Engineering
YEARS	2010 - 2014

REPERTOIRE

DAILY	Kotlin, TypeScript, React, Next.js, Node, HTML & CSS, Tailwind
ALSO	Rust, Swift, Objective-C, Java, Go
FONDLY	Haskell, OCaml, Coq
MAKING	SolidWorks, Illustrator, the machine shop (mill, lathe, CNC)
OTHERWISE	Typography, saxophone, bicycling, design of every kind, and how things work