KEVIN 'TYLER' COX

Angular Application Developer

(843) 718-4024 kevincox103@gmail.com Charleston, SC

github.com/Ty-lerCox linkedin.com/in/tyler-cox-1715065a/ ty-lercox.github.io/portfolio/

Summary

Angular specialist (Signals, RxJS, NgRx) delivering secure, observable, and automated web applications. Strong in TypeScript, Tailwind CSS, and GraphQL—translate UX specs into responsive, pixel-accurate UIs and optimize Core Web Vitals (LCP/CLS/INP). Eleven years at Expediters International building front-ends, standing up CI/CD (Ansible, GitHub Actions, GitLab Runners, Azure DevOps), and instrumenting the Grafana stack (Prometheus, Loki, Tempo) for both ops telemetry and product event analytics (click/flow funnels). Integrated SSO with Keycloak/Kerberos, established PKI/TLS, and evolved data ingestion from bespoke Kafka jobs to CDC → Kafka with bronze/silver/gold layers. Prototype quickly in Python; ship production services in Java Spring Boot and C#/.NET. Champion feature flags, DORA metrics, and Kanban to improve delivery flow. Built ODEYA.app, an SSR Angular app that auto-curates YouTube playlists from channel collections and indexes shareable playlists for SEO.

Education

Computer Science
Trident Technical College

Technical Skills

Angular (Signals, RxJS, NgRx), TypeScript, Tailwind CSS, GraphQL (queries/mutations, client integration), UI architecture & component design, Responsive Web Design (Flexbox, Grid), Core Web Vitals (LCP, CLS, INP) & web

Work Experience

Expediters International (Expediters) Aug, 2014 - Present

Angular Application Developer

Design and delivery of Angular front-ends and platform tooling across multiple internal applications; emphasis on observability, automation, and secure-by-default practices.

- Built and led Angular front-ends using Signals, RxJS, and NgRx across several logistics/operations applications.
- Styled and shipped responsive UIs with Tailwind CSS (utility-first) alongside semantic HTML and modern CSS (Flexbox/Grid).
- Integrated GraphQL (queries/mutations) alongside RESTful services; streamlined data contracts and normalized client state.
- Implemented product analytics: client-side event tracking for clicks/flows emitted to Prometheus/Loki/Tempo (Grafana stack); created funnels/dashboards and tracked Core Web Vitals to guide iteration.
- Introduced step-centric workflow modeling (e.g., Unified Process) to complement event-driven patterns; improved clarity and traceability of process state.
- Implemented observability with Grafana stack—Prometheus (metrics), Loki (logs), Tempo (traces)—plus SLO dashboards and alerting to reduce detection and resolution time.
- Standardized CI/CD with GitHub Actions and GitLab Runners; used Ansible for repeatable on-prem and cloud provisioning (GCP/Firebase); supported Kubernetes deployments.
- Hardened identity and transport security: Keycloak SSO (OIDC/SAML), Kerberos integrations, internal CA/PKI, and TLS/mTLS with cert rotation policies.
- Modernized data integration from ad-hoc Kafka producers/consumers to database-level Change Data Capture (CDC) feeding Kafka, organized via medallion layers (bronze/silver/gold).
- Optimized performance and SEO using SSR (where applicable), route-level lazy loading, and bundle analysis to improve first-paint and TTI.
- Enabled feature flag rollouts for progressive delivery; instrumented DORA metrics and coached teams on Kanban and velocity/flow tracking via Azure DevOps.
- Delivered APIs/services in Java Spring Boot and C#/.NET; used Python for rapid prototyping prior to hardening.
- Authored Azure DevOps utilities (Python) to create user stories under Features with iteration/assignee/area/story-point fields, improving planning throughput.

Projects

ODEYA.app Feb, 2025 - May, 2025

performance, Java (Spring Boot, JPA/Hibernate), C# (.NET Web API, Entity Framework), Python (prototyping), Kafka & Change Data Capture (CDC), Data medallion layers (bronze/silver/gold), Kubernetes, Ansible, Azure DevOps, GitHub Actions, GitLab Runners, GCP & Firebase, Grafana, Prometheus, Loki, Tempo (ops + product event analytics), Keycloak (OIDC/SAML), Kerberos, PKI/Certificate Authority, TLS/mTLS, RHEL 9, VMs, Feature flags, DORA metrics

Soft Skills

Systems Thinking, Cross-functional Collaboration, Technical Leadership & Coaching, Stakeholder Communication, Problem Solving, Iteration & Rapid Prototyping, Kanban Facilitation, High Ownership & Product Sense

Additional Skills

Observability & SRE Practices, Event analytics & funnels (client-side tracking), Documentation & ADRs, AI-Assisted Engineering (LLMs/ChatGPT CLI), Testing & Quality Gates (unit/E2E in CI), Release & Deployment Strategies

Languages

English

ty-lercox.github.io/portfolio/posts/odeya-overview/
Personal web app that auto-curates YouTube playlists from collections of channels (e.g., cruise, gaming, tech) so new uploads from any channel in a collection flow into a single playlist.

- Implemented server-side rendering so shared playlists are SEO-indexable; optimized for Core Web Vitals with lazy loading and bundle analysis; playlist sharing enables discovery via search engines. Aggregation logic supports many-to-one channel collections; reduced manual playlist upkeep.

Logistics Process Orchestration
Front-end architecture for step-centric workflows replacing purely event-driven UIs in logistics operations.

- Normalized state, clear effects, and deterministic UI flows; fully instrumented for metrics/logs/traces and product event analytics.

Observability Enablement
Service health/SLO dashboards with Prometheus metrics, Loki logs, and Tempo traces; alerting integrated with deployment telemetry.

- Improved issue detection and trace-led debugging across services; consistent dashboards for teams; added client-side event funnels and CWV tracking to inform UI iteration.

CI/CD Blueprint
Reusable pipelines (GitHub Actions/GitLab Runners) and Ansible roles for consistent build/test/deploy across on-prem and cloud targets.

- Cut lead time to change; standardized quality gates and environment promotion.

Data Flow Modernization
Replaced batch/legacy Kafka usage with database-level CDC streaming into Kafka, organized into bronze/silver/gold layers.

- Improved reliability, lineage, and downstream transformations; clearer SLAs and reprocessing.

SSO & PKI Hardening
Unified identity and secure transport across services.

- Keycloak (OIDC/SAML), Kerberos, internal CA/PKI, TLS/mTLS, and certificate lifecycle policies.

Azure DevOps Utilities
Python tooling to create child User Stories under Features with Iteration, Assignee, Area, and Story Points fields.

- Accelerated planning and improved consistency of backlog data.