

Braylon Dunn

7558 Greenup Lane, Ethel, Louisiana 70730 • 225-421-0403 • braylond770@gmail.com •
linkedin.com/in/braylon-dunn-34b76137b

Senior Software Engineer

Senior Software Engineer with 9+ years of experience building and scaling **full-stack** and **cloud-native** platforms across **web**, **mobile**, and **distributed systems**. I specialize in **JavaScript/TypeScript (React/Next.js, Vue.js, Angular, Node.js)**, **.NET/C#**, and **Python (Django, FastAPI)** with hands-on delivery on **AWS, Azure, and GCP**. I've led migrations to **microservices**, driven **AI/ML** and **RAG** integrations with **LLMs** and **vector databases**, and optimized **SQL/NoSQL** systems for high-throughput, low-latency workloads. I'm comfortable owning everything from **architecture decisions**, **DevOps/CI/CD** and **SRE/observability** to **frontend design systems** and **secure identity flows**. I thrive in **Agile** teams, mentor engineers, and translate complex requirements into practical, user-focused solutions that move key business metrics.

EDUCATION

Western Governors University (WGU)

Bachelor's Degree in Software Development / Computer Science • 05/2014 - 09/2017

SKILLS

Languages & Frameworks: Angular, Ant Design, D3.js, Java, JavaScript/TypeScript, Material UI, .NET/C#, Python (Django/FastAPI), React Native, React/Next.js, React Query, Redux Toolkit, Tailwind, Vue, Webpack & Vite

Back end & APIs: ASP.NET Core (Web API + gRPC), EF Core, GraphQL, Node (Express/Nest), REST

Data & Messaging: Azure Service Bus, design for HA/multi-region-ready data architectures, Kafka, MongoDB, MySQL, PostgreSQL, RabbitMQ, Redis, SQL Server

Cloud & DevOps: AWS (ECS/EKS, CloudWatch), Azure, Docker, GCP (GKE, Pub/Sub), GitHub Actions/Jenkins, Kubernetes (AKS/GKE/EKS), Observability with App Insights, Prometheus, Grafana, SRE practices (SLOs, error budgets, incident response, runbooks), Terraform

Security, Compliance & Identity: audit logging, familiarity with PCI DSS, ISO27001, SOC2 and operating in regulated enterprise environments, GDPR/CCPA, JWT, Key Vault/secret management, OAuth2/OIDC/SSO (Okta, Azure AD), OWASP hardening, RBAC

AI/ML: embeddings & prompt-chaining, FAISS/pgvector, LLM integrations, RAG

Testing & Quality: Cypress, Jest, pact-style contract tests, Playwright, Postman (+ Newman in CI), React Testing Library, Snyk, SonarQube, xUnit

Ways of working: ADRs, Agile/Scrum, Code Reviews, DDD, Design Systems, Forward-deployed/customer-facing engineering with architecture/infra teams, Mentoring, translating between technical and non-technical stakeholders

WORK EXPERIENCE

Bitwarden • Remote (California, United States) • 10/2024 - 07/2025

Senior Software Engineer

- Led delivery of security-sensitive features across **React/Next.js** and **ASP.NET Core** microservices using **OAuth2/OIDC**, **JWT**, and **RBAC**, cutting auth-related incident tickets by roughly one-third and keeping SSO escalations to fewer than 1–2 per quarter for enterprise customers.
- Designed and hardened **GraphQL** and **REST** APIs used by **10+ internal and external clients**, adding strict validation and **OWASP** protections that reduced externally reported API issues and made partner onboarding a same-week activity instead of multi-week.
- Refactored core front-end flows with **Redux Toolkit**, **React Query**, SSR/ISR, and route-level code splitting, lowering LCP on key dashboards from ~3s to nearly 1.5s and improving funnel completion by several percentage points in A/B tests.
- Built high-throughput **.NET** services with **gRPC**, **EF Core**, **SQL Server/PostgreSQL**, **Redis**, and **Azure Service Bus** that reliably processed tens of thousands of messages per day, while reducing message retries and dead-letter volume by about 35%.
- Introduced **Vite** alongside legacy **Webpack** in targeted apps, trimming local rebuild times from 20–30 seconds to under 5 seconds and shrinking production bundles by 15–20%, which noticeably improved both developer feedback loops and user perceived load time.
- Standardized UI with **Tailwind CSS**, shared Storybook components, and selective **Material UI/Ant Design** usage, enabling teams to assemble new admin workflows in hours instead of days and cutting CSS- and layout-related bugs by ~20%.
- Automated **CI/CD** using **GitHub Actions**, **Docker**, and **AKS (Kubernetes)**, moving from largely manual, hour-long release windows to fully automated pipelines that shipped safely in under 15 minutes and supported multiple deployments per day.
- Codified infra with **Terraform** (networking, AKS, storage, identities), reducing environment drift and cutting the time to spin up a new environment from multiple days of manual steps down to a few pipeline runs.
- Deepened observability by wiring **Azure Application Insights** traces, metrics, and logs into **Prometheus/Grafana** dashboards and actionable alerts, lowering MTTR on critical incidents from roughly 90 minutes to about an hour.
- Expanded automated test coverage with **Jest**, **React Testing Library**, **Cypress**, **Playwright**, **Postman/Newman** collections, and contract tests for **GraphQL/REST**, raising automated coverage on customer-critical journeys to ~80%+ and significantly reducing regressions after large refactors.

Sunnova Energy • Remote (Texas, United States) • 11/2020 - 08/2024

Software Engineer

- Delivered customer and partner portals using **Vue.js** and **TypeScript** backed by **.NET** and **Node.js (Express, NestJS)** services, consolidating three legacy UIs into a unified experience and supporting tens of thousands of monthly active users with improved NPS and fewer support tickets.
- Modeled core domains with **ASP.NET Core**, **EF Core**, **SQL Server**, and **MongoDB**, introducing outbox and saga patterns on **Kafka**, **RabbitMQ**, and **Azure Service Bus** that kept integration success rates above 99% across **Salesforce**, **Twilio**, and **SendGrid**.
- Introduced **Python (Django, FastAPI)** services and tooling for ETL, PDF processing, and CI/CD automation, eliminating several hours of manual release validation each week and shrinking lead time for backend fixes and enhancements.
- Prototyped LLM-backed **RAG** search and support workflows with **pgvector**, **FAISS**, and prompt-chaining, enabling internal teams to retrieve answers across thousands of documents and reducing average handling time in pilot queues by

15–20%.

- Led a gradual migration from a monolithic app to **microservices** on **Azure**, containerizing workloads with **Docker** and orchestrating them on **AKS**; this, along with **GitHub Actions/Jenkins** pipelines and **Terraform**, increased successful production releases from 1–2 per month to several per week.
- Tuned front-end performance using SSR, optimized asset loading, caching headers, and CDNs, adding 20+ points to Lighthouse performance scores on key sales flows and contributing to a 5–10% uplift in lead-to-opportunity conversion in marketing experiments.
- Strengthened security posture by implementing **OAuth2/OIDC** SSO through **Okta/Azure AD**, centralized secrets in **Azure Key Vault**, and formalizing audit logging and **GDPR/CCPA**-aligned data retention, helping the team pass multiple enterprise security reviews with zero high-severity findings.
- Built and maintained an automated testing stack using **Jest**, **React Testing Library**, **Cypress**, **Playwright**, **Postman**, and **xUnit**, lifting coverage on critical APIs and UI flows to around 70–75% and reducing P1 production regressions quarter-over-quarter.
- Established platform observability with **Application Insights**, **Prometheus**, and **Grafana**, creating dashboards and SLO/error-budget alerts that cut time-to-detect issues by ~30% and allowed on-call engineers to resolve many incidents before customers noticed.
- Delivered a **React Native** mobile companion for field technicians with offline-first data and secure **OIDC** auth, replacing manual paper workflows and reducing data entry errors by an estimated 20% while giving operations near real-time visibility into jobs.
- Practiced **DDD**, **SOLID**, and **ADRs** across cross-functional squads, mentoring 4–5 engineers on story slicing, technical design, and code review practices, which shortened ramp-up time for new team members and improved consistency across services and UIs.

HCLTech • Hybrid (California, United States) • 03/2018 - 09/2020

Full Stack Developer

- Built enterprise SPAs with **React**, **Angular**, and **Vue** in **TypeScript**, using **Redux**, **RxJS**, and **D3.js** for data-rich dashboards that allowed business users to make decisions from real-time metrics Instead of static reports, cutting some manual reporting cycles from days to same-day self-service.
- Implemented backend APIs in **ASP.NET Core** and **Node.js (Express, NestJS)** with **Entity Framework Core** on **SQL Server/PostgreSQL/MySQL**, tuning hotspots to keep p95 latency under 200ms on critical endpoints even under peak traffic.
- Used **Python (Flask, FastAPI)** to stand up migration tooling and service prototypes, allowing teams to validate API designs with clients in days and reducing rework during integration phases by roughly one quarter.
- Designed event-driven workflows with **Kafka** and **RabbitMQ**, defining schemas, idempotency, and retry patterns that supported tens of thousands of daily events and kept failed-message rates consistently below 1%.
- Delivered multi-cloud solutions across **Azure**, **AWS (ECS, EKS, CloudWatch)**, and **GCP (GKE, Pub/Sub)**, containerizing services with **Docker** and running them on **Kubernetes/Helm**, which unlocked multi-region availability and helped several clients reduce infrastructure costs by 15–25% through autoscaling and right-sizing.
- Created **Jenkins** and later **GitHub Actions** pipelines with security and quality gates via **Snyk** and **SonarQube**, plus **Terraform** for infrastructure as code, moving clients from quarterly or ad-hoc releases to predictable weekly

or biweekly releases with lower change failure rates.

- Migrated selected applications from **Webpack** to **Vite**, employing code splitting, tree shaking, and bundle analysis to shrink JS payloads and push Lighthouse performance scores up by 20–30 points on high-traffic pages.
- Integrated third-party platforms such as **Stripe**, **Twilio**, **SendGrid**, **Salesforce**, and **S3/CloudFront**, enabling clients to process thousands of secure transactions and notifications per month while maintaining high deliverability and low error rates.
- Expanded testing from unit-only to a layered strategy using **Jest**, **React Testing Library**, **Cypress**, **Playwright**, and **Postman** collections, catching defects earlier and reducing production P1 incidents by an estimated 15–20%.
- Mentored junior and mid-level developers, led design reviews grounded in **Design Systems**, **DDD**, and **SOLID**, and partnered with product/architecture stakeholders to align roadmaps, which improved delivery predictability across multiple concurrent projects.