

# Dylan Burns

San Francisco, CA, 94118 (619) 415-9870 dylanburns1524@gmail.com

<https://github.com/Dylan-Burns>

<https://www.linkedin.com/in/dylan-burns->

## EDUCATION

### Bachelor of Science Computer Science

San Francisco State University, San Francisco, CA

## CERTIFICATIONS

- AWS Certified Solutions Architect - Associate  
Validation #: 4fb961a54957490ea011f82369005b12
- AWS Certified Developer - Associate  
Validation #: cff45391d75b4652ad083d98ce4bffa7
- AWS Certified DevOps Engineer - Professional  
In Progress
- Certified Spring Boot Professional

## EXPERIENCE

08/2023 - present

**Bridge2Partners Inc** San Francisco, CA

Senior Software Engineer

- Improved API response times by 40% across 50M+ daily requests by architecting and deploying Java-based microservices using Spring Boot with optimized RESTful endpoint design and database query patterns, resulting in 99.95% uptime for critical banking operations.
- Reduced production incident resolution time by 30% (from 45 to 15 minutes MTTR) by designing event-driven automation workflows in Appian that processed 100K+ customer requests daily, eliminating manual intervention and improving system reliability.
- Achieved 95% reduction in defect rates and 70% fewer production bugs by implementing comprehensive test automation including unit tests (JUnit/Mockito), integration tests, and contract testing across microservices, raising code coverage from 45% to 92%.
- Reduced P95 latency by 25% under peak loads of 15K requests/second by implementing asynchronous message processing with RabbitMQ and intelligent load balancing across Spring Boot services, maintaining sub-100ms response times.
- Established secure API design standards adopted across 12 microservices by leading technical design reviews and mentoring 3 junior engineers on OAuth 2.0/JWT authentication patterns and API versioning strategies, passing all security audits.
- Cut monthly infrastructure costs by \$12K and improved data retrieval times from 300ms to 50ms by designing efficient caching architecture using Redis with TTL-based invalidation strategies, reducing external API calls by 18%.

06/2023 - 10/2024

**Parkli** San Francisco, CA

Founder

- Founded and led C2C parking marketplace from zero to closed beta as sole technical founder, bootstrapping operations through TestFlight launch while owning product, architecture, and go-to-market.
- Identified asymmetric supply-side opportunity by mapping urban density, transit corridors, and event venue catchments to prioritize launch markets with highest expected liquidity per acquired user.
- Solved two-sided cold-start by seeding host inventory in three Bay Area submarkets before driver acquisition, achieving sub-10-minute average search-to-booking time and validating marketplace clearing at small scale.
- Recruited and onboarded three contract iOS engineers and a part-time designer through founder network, structuring equity-heavy compensation to extend runway while preserving team quality.
- Drove 3,000+ TestFlight beta testers and facilitated \$65K+ in pilot bookings through community-led acquisition in dense urban markets, validating two-sided marketplace economics and repeat-use behavior prior to scaling.

01/2020 - 08/2023

**Helix Affiliates** San Francisco, CA

Full Stack Software Engineer

- Scaled application to 50K+ monthly active users and improved Core Web Vitals by 60% by building full-stack web application using React, TypeScript, and Next.js with server-side rendering and code-splitting strategies, reducing initial page load from 4.2s to 2.1s.
- Reduced complex query times from 800ms to 250ms for 2M+ daily API requests by designing and implementing RESTful and GraphQL APIs with Node.js and MongoDB using optimized indexing strategies and aggregation pipelines.
- Reduced security incidents by 95% and achieved GDPR/CCPA compliance by architecting authentication and authorization system using JWT and OAuth 2.0 supporting multiple providers, implementing data encryption and privacy-by-design principles.
- Improved Lighthouse performance scores from 65 to 95 and reduced render times by 60% by optimizing front-end performance using React.memo, useMemo/useCallback hooks, and virtual scrolling for datasets with 10K+ items.
- Increased mobile user satisfaction scores by 30% and achieved 100% keyboard navigation support by leading accessibility initiative implementing WCAG 2.1 AA compliant components with screen reader compatibility in collaboration with UX designers.
- Reduced storage costs by \$24K annually and improved payload delivery times by 40% by building serverless data processing pipeline using AWS Lambda, S3, and CloudWatch that automated compression for 500GB+ daily uploads.
- Reduced deployment errors by 80% and accelerated release cycles from monthly to weekly by establishing CI/CD infrastructure with GitHub Actions, Docker, and AWS ECS implementing automated testing (Jest, Cypress) and blue-green deployments.
- Reduced production issue detection time from hours to minutes by implementing comprehensive monitoring and observability using Datadog with custom dashboards tracking 20+ KPIs including error rates, latency percentiles, and business metrics.