

# Keller Davis

## ENGINEERING LEAD

Seattle, WA | [keller.davis@gmail.com](mailto:keller.davis@gmail.com) | [LinkedIn](#) | [GitHub](#) | <https://keller.dev>

### PROFESSIONAL SUMMARY

Engineering leader with 18+ years architecting and scaling platforms serving **millions of users** across web, mobile, and cloud infrastructure. Progressed from individual contributor to Engineering Lead managing cross-functional teams and overseas development partners. Led Bible Gateway's evolution through 5 major platform generations, driving complete frontend modernization, pioneering React Native adoption for cross-platform development, and orchestrating EC2-to-Kubernetes infrastructure transformation. Deep expertise in high-traffic system architecture, team development, and delivering business-critical products with measurable impact on user engagement and operational efficiency.

### CAREER HIGHLIGHTS

**18-Year Journey at Bible Gateway** | Progressed through multiple acquisitions and platform generations serving one of the world's largest Bible platforms

- **Led Complete Platform Modernization (2018-2020):** Architected React-based Bible Gateway Version 7 frontend rewrite, eliminating legacy dependencies and achieving 50% initial page load size reduction while serving thousands of concurrent users
- **Pioneered Cross-Platform Mobile Development (2016-2019):** Created Study Gateway React Native prototype and production app, then drove organizational shift establishing cross-platform development standards that increased team productivity
- **Platform Rescue & Rebuild (2011):** Led 4-month project to rebuild core platform after failed Rails migration, successfully launching Bible Gateway Version 5 to replace failing infrastructure
- **Engineering Leadership Growth (2021-Present):** Took on direct management of 3 engineers while assuming ownership of mobile apps and infrastructure operations, mentoring junior developer into primary platform contributor
- **Infrastructure Transformation (2024-2025):** Led migration from EC2 architecture to modern Kubernetes/EKS, implementing GitOps workflows and reducing operational complexity

### PROFESSIONAL EXPERIENCE

**HarperCollins Christian Publishing | Bible Gateway**

**Engineering Lead** | January 2022 - Present

**Senior Web Developer** | October 2011 - January 2022

**Zondervan Publishing | Bible Gateway**

**Application Architect** | November 2008 - October 2011

## KEY PROJECTS & ACHIEVEMENTS

---

### Bible Gateway Web Platform Development

June 2011 - Present

Led **14 years of continuous development** on core Bible Gateway web platform serving **millions of users worldwide**. Drove platform evolution through Bible Gateway Version 5, Version 6, Version 7, and Version 8, delivering major features, performance optimizations, and infrastructure improvements.

**Technologies:** PHP, React, Preact, JavaScript, Nginx, MySQL, Redis, AWS CloudFront, AWS S3, Kubernetes

#### Impact & Achievements:

##### Application Development:

- **Owned platform evolution** across four major generations, each introducing significant architectural improvements
- **Implemented CloudFront header forwarding** enabling authentication and geo-targeting features
- **Configured continuous deployment** across dev, staging, and production ensuring rapid, safe releases
- **Maintained high-traffic platform** ensuring reliability and performance during peak usage periods
- **Balanced innovation with stability** introducing modern features while maintaining 99.9%+ uptime

##### Infrastructure & Operations:

- **Architected deployment standardization** making it significantly easier to create new application deployments for improved testing/QA and rapid new application launch
- **Managed AWS infrastructure** supporting millions of users across global Bible Gateway properties
- **Maintained CloudFront distributions** managing global content delivery with optimized caching strategies and geo-targeting
- **Configured IAM security policies** ensuring secure access management across AWS services
- **Operated DynamoDB tables** supporting user data service at scale
- **Managed AWS WAF rules** ensuring secure, protected access for all users

---

### Bible Gateway Web EU GDPR Compliance Implementation

September 2025

Delivered comprehensive GDPR compliance solution protecting organization from regulatory risk across all EU markets. Architected geographic access control system with intelligent whitelisting, ensuring seamless user experience for allowed paths while maintaining strict compliance boundaries.

**Technologies:** Nginx, PHP, CloudFront, OneTrust/Optanon, Sailthru API, React, Kubernetes, Helm

### Impact & Achievements:

- Implemented dual-layer geographic blocking (CloudFront + Nginx/PHP) ensuring regulatory compliance with zero false positives
- Engineered allowed paths whitelist system enabling critical functionality (unsubscribe, privacy) while maintaining compliance
- Designed custom 403 experience maintaining Bible Gateway brand consistency during access denial
- Integrated OneTrust cookie consent framework with Sailthru email platform, ensuring end-to-end GDPR compliance
- Deployed across all environments (dev/stage/prod) with zero downtime using Kubernetes ConfigMaps

## Kubernetes Infrastructure Modernization

January 2025 - May 2025

Spearheaded complete infrastructure transformation from legacy EC2 deployments to modern cloud-native architecture, reducing operational complexity and positioning platform for future scale. Led technical strategy and hands-on implementation of Kubernetes migration serving millions of users.

**Technologies:** Kubernetes, AWS EKS, Helm, ArgoCD, Terraform, Docker, GitHub Container Registry

### Impact & Achievements:

- **Migrated mission-critical platform** from EC2 instances to EKS, optimizing cost-performance balance
- **Established GitOps practices** with ArgoCD, enabling declarative infrastructure management and audit trails
- **Architected Helm-based deployments** with continuous deployment system via ArgoCD for Bible Gateway and Study Gateway, standardizing release processes
- **Configured Horizontal Pod Autoscaling** ensuring responsive scaling during traffic spikes instead of overprovisioning, saving costs on compute resources

## Bible Gateway Version 8 Platform

May 2024 - Present

Leading complete frontend rewrite modernizing Bible Gateway's web platform with next-generation build tools and state management. Architecting migration to Redux Toolkit while implementing esbuild and Bun for dramatically improved developer experience and build performance.

**Technologies:** React, Redux Toolkit, JavaScript, esbuild, Bun, Modern ES6+, PHP backend, GraphQL backend

### Impact & Achievements:

- **Architecting Redux Toolkit migration** replacing legacy custom state management with modern patterns and improved type safety
- **Implementing esbuild and Bun** achieving faster build times for enhanced developer productivity
- **Updating to modern library versions** eliminating technical debt and security vulnerabilities across dependency stack

- **Maintaining zero-downtime deployment strategy** ensuring continuous availability while rolling out major architectural changes
- **Establishing beta environment** (beta.biblegateway.com) enabling safe feature validation before production rollout
- **Preserving performance gains** from Version 7 while establishing foundation for future enhancements

## Jesus Calling App Subscription Migration Oversight

March 2024 - August 2024

Oversaw team of 3 contracted developers implementing major business model transformation for Jesus Calling mobile app. Managed technical delivery of migration from à la carte purchase system to subscription model, ensuring quality and timeline adherence.

**Technologies:** React Native, iOS, Android, In-App Purchase, Subscription Management, Project Management

### Impact & Achievements:

- **Led cross-functional coordination** between contracted developers, product stakeholders, and business teams
- **Ensured successful business model migration** from individual purchases to recurring subscription revenue
- **Managed technical quality** through code review and testing oversight across iOS and Android platforms
- **Delivered on schedule** meeting critical business timeline for subscription launch
- **Established subscription infrastructure** enabling predictable recurring revenue stream

## Cypress Testing Suite Implementation

July 2023

Architected and implemented modern end-to-end testing infrastructure for Bible Gateway web platform, replacing legacy Selenium-based tests. Established comprehensive test coverage enabling confident deployments and faster feature development cycles.

**Technologies:** Cypress, JavaScript, CI/CD, GitHub Actions, Testing Automation

### Impact & Achievements:

- **Migrated from Selenium to Cypress** improving test reliability and execution speed
- **Established comprehensive test coverage** for critical user flows including search, reading, and account features
- **Integrated with CI/CD pipeline** enabling automated testing on every deployment
- **Improved developer confidence** through fast, reliable feedback on code changes
- **Reduced regression bugs** caught in production through enhanced test coverage

## Continuous Deployment System (kdeploy)

April 2022 - June 2022

Designed and delivered automated CI/CD pipeline **specifically for contractors and non-team members** to commit to Git and have changes automatically deployed to staging and production environments,

eliminating manual deployment bottlenecks and reducing release cycle time. Architected event-driven system handling deployments across multiple environments with queuing and retry logic.

**Technologies:** Python, Ansible, AWS Lambda, AWS API Gateway, AWS SNS, Bitbucket Webhooks

#### Impact & Achievements:

- **Architected webhook-driven pipeline** (Bitbucket to Lambda to SNS to Ansible) enabling automatic deployments on code commits
- **Built scalable queue system** with SNS ensuring reliable deployment execution even during high-frequency commit periods
- **Developed Python orchestration layer** polling queue and triggering Ansible playbooks with comprehensive error handling
- **Eliminated manual deployment steps** reducing human error and freeing engineering time for feature development
- **Enabled contractor autonomy** allowing external developers to deploy without direct team intervention
- **Established foundation for GitOps** practices later expanded in Kubernetes migration

### Engineering Team Leadership and Growth

#### December 2021 - September 2025

Transitioned from senior engineering role to direct leadership position managing **3 engineers** (2 Junior, 1 Senior) plus overseas development partners. Demonstrated leadership through mentorship, expanding scope to own mobile apps and infrastructure after team departures.

#### Impact & Achievements:

- **Hired and mentored junior developer** (December 2021) who became primary contributor to Bible Gateway web platform, demonstrating effective talent development
- **Assumed mobile app ownership** (August 2022) after iOS developer departure, ensuring continuity for critical user touchpoint
- **Took over infrastructure operations** (December 2022) after DevOps engineer departure, expanding technical breadth
- **Managed cross-functional relationships** with overseas contractors, ensuring quality delivery across time zones
- **Drove technical decisions** on platform architecture, technology adoption, and infrastructure strategy impacting millions of users

### Study Gateway Technical Leadership

#### August 2021 - September 2025

Assumed primary technical leadership for Study Gateway platform, driving Laravel development, Kubernetes operations, and mobile app maintenance. Led platform through business model transitions including **2022 Vimeo OTT integration** for video delivery while maintaining billing and authentication in Laravel with SAML-based user authentication.

**Technologies:** Laravel, PHP, Kubernetes, Helm, AWS EKS, Docker, MySQL/Aurora, Redis, Vimeo OTT, SAML

#### Impact & Achievements:

- **Took over technical leadership** ensuring platform stability and feature delivery

- **Led Vimeo OTT platform transition** (2022) integrating video streaming platform while retaining billing and authentication control
- **Implemented SAML authentication** enabling seamless user authentication between Laravel platform and Vimeo OTT
- **Managed production operations** including deployments, database migrations, and infrastructure maintenance
- **Led promotional campaigns** (seasonal sales, homepage updates) driving subscription conversions
- **Maintained mobile app compatibility** during platform transitions and updates
- **Coordinated with product stakeholders** balancing feature requests with technical feasibility

## Expanded Bible Experimental Project

February 2019 - April 2019

Built innovative proof-of-concept exploring enhanced Bible reading experience with dynamic translation choices. Developed experimental UI allowing users to customize passages by selecting from multiple translation options for specific words/phrases, creating personalized Bible text. Integrated text-to-speech capabilities demonstrating future audio features.

**Technologies:** React, JavaScript, AWS Lambda, AWS Polly, AWS S3, DynamoDB, Node.js, Express, Goober CSS-in-JS

### Impact & Achievements:

- **Architected single-page application** with custom routing for Expanded Bible translation exploration
- **Designed innovative UI experiment** enabling user-driven translation customization for individual words and phrases
- **Integrated AWS Polly** for on-demand text-to-speech, pioneering audio features later adopted in production apps
- **Implemented intelligent caching** with DynamoDB reducing Polly API costs while ensuring responsive audio generation
- **Validated experimental approaches** informing future product thinking around customization and user experience

## Bible Gateway Version 7 React Platform

January 2018 - September 2025

Led **complete frontend modernization** of Bible Gateway, serving millions of users. Architected React-based solution eliminating 10+ years of technical debt while achieving 50% initial page load size reduction and improved user experience. **Successfully launched to 100% traffic July 2020** after multi-year development effort.

**Technologies:** React, Preact, JavaScript, Babel, webpack, Modern ES6+, PHP backend

### Impact & Achievements:

- **Achieved 50% initial page load size reduction** through modern JavaScript architecture, directly improving load times and mobile experience
- **Eliminated jQuery and Bootstrap dependencies** replacing with optimized React components and reducing bundle complexity
- **Enhanced Bible Gateway Plus integration** improving subscription user experience and revenue opportunities

- **Delivered enriched sidebar features** without sacrificing performance, increasing user engagement with related content
- **Led cross-functional coordination** ensuring seamless rollout to millions of users with zero downtime

## Bible Gateway Audio App Development

March 2017 - July 2023

Architected and delivered complete audio Bible application for iOS and Android, expanding Bible Gateway's reach into audio-first user experience. Built comprehensive streaming platform with offline support, subscription monetization, and production-grade reliability.

**Technologies:** React Native, iOS, Android, Redux, Redux Saga, Firebase Analytics, Crashlytics, Bugsnag, AWS Kinesis Firehose, AWS Polly

### Impact & Achievements:

- **Launched audio-first mobile experience** creating new user engagement channel and subscription revenue stream
- **Engineered streaming audio player** with position tracking, resume functionality, and background playback for seamless listening
- **Implemented offline downloads** enabling Bible listening without connectivity, critical for international users
- **Integrated in-app purchase subscriptions** (iOS/Android) with receipt validation and restore functionality
- **Managed multiple React Native upgrades** maintaining app stability through framework evolution and platform changes
- **Deployed comprehensive analytics** (Firebase, Crashlytics, Bugsnag, Kinesis) enabling data-driven product decisions

## Study Gateway Mobile App

December 2016 - November 2019

Built cross-platform mobile application for Study Gateway video platform, extending Bible study content to mobile users. **Pioneered React Native adoption** for team starting with December 2016 prototype, establishing technical patterns and cross-platform development standards later adopted across Bible Gateway mobile initiatives. Delivered production app with video streaming, subscription management, and comprehensive content discovery experience.

**Technologies:** React Native, iOS, Android, Redux, React Navigation, Video Streaming, In-App Purchase, Google Analytics, Bugsnag, CodePush

### Impact & Achievements:

- **Created React Native prototype** (December 2016) demonstrating viability and establishing institutional knowledge for cross-platform mobile development
- **Established cross-platform architecture** enabling iOS and Android development with single codebase
- **Launched video streaming platform** on mobile, expanding Study Gateway's addressable market
- **Implemented video position tracking** with resume functionality, improving user experience and content completion rates



- **Built iOS in-app purchase integration** with receipt validation and subscription restore, enabling mobile monetization
- **Developed continue watching feature** surfacing in-progress content and increasing engagement
- **Integrated comprehensive analytics** tracking user behavior and informing product roadmap
- **Utilized CodePush** for rapid bug fixes and feature deployment without app store review delays
- **Informed Bible Gateway mobile strategy** leading to React Native adoption in 2018

## Study Gateway Platform API Recreation

August 2016 - September 2021

**Reverse-engineered external service API** without access to mobile app code or documentation, recreating endpoints in Laravel to ensure zero-downtime mobile app continuity during platform transition. Later inherited complete Laravel platform codebase, assuming ownership of full-stack application including Kubernetes deployments.

**Technologies:** Laravel, PHP, Kubernetes, Helm, AWS EKS, Docker, MySQL/Aurora, Redis, AWS S3, API Design, Reverse Engineering

### Impact & Achievements:

- **Reverse-engineered mobile API** by analyzing network traffic, recreating endpoints with complete compatibility
- **Achieved seamless mobile app transition** with zero user impact or app updates required
- **Inherited and maintained Laravel platform** after initial development by team member, expanding ownership to full application
- **Orchestrated Kubernetes deployments** with Helm charts, implementing modern cloud-native practices
- **Executed database migrations** via Kubernetes CronJobs ensuring zero-downtime schema changes
- **Configured Redis caching** improving application performance and database efficiency

## Production Support and Maintenance

September 2011 - Present

Provided **14 years of operational excellence** maintaining production systems serving millions of users. Managed security improvements, dependency updates, incident response, feature enhancements, and 24/7 availability for business-critical platforms.

**Technologies:** PHP, MySQL, Redis, Nginx, Apache, AWS, Security, Monitoring, Incident Response

### Impact & Achievements:

- **Implemented security improvements** protecting millions of users and maintaining platform integrity
- **Maintained framework and dependency updates** ensuring platform security and compatibility
- **Provided 24/7 incident response** ensuring rapid resolution of production issues
- **Delivered continuous feature enhancements** responding to user needs and business requirements
- **Maintained subscription and payment processing** ensuring uninterrupted revenue collection
- **Balanced new feature development** with operational stability maintaining high availability

## User Data PHP API Service

September 2011 - August 2025



Architected and maintained **mission-critical API service** powering authentication, subscriptions, notes, highlights, and favorites across Bible Gateway ecosystem. Built on AWS DynamoDB serving **millions of authenticated users** with high-availability requirements.

**Technologies:** PHP, AWS DynamoDB, AWS SDK, OAuth2, Braintree, Apple IAP, Google Play Billing, RESTful API, HMAC Authentication

#### Impact & Achievements:

- **Designed RESTful API architecture** serving web, iOS, Android, and internal services with consistent interface
- **Implemented OAuth2 authentication** with Facebook, Google, and Apple Sign-In enabling social login
- **Built multi-provider subscription system** integrating Braintree (web), Apple IAP, and Google Play Billing for cross-platform revenue
- **Developed notes and highlights system** on DynamoDB, enabling core Bible study features at scale
- **Maintained 24/7 availability** for authentication and user data critical to entire Bible Gateway experience
- **Managed security and compliance** including password validation, session management, and data protection

### Bible Gateway CMS Development and Caching System

September 2011 - June 2025

Built Drupal 7-based content management system with sophisticated Memcached integration powering devotionals, reading plans, and landing pages. Optimized performance through multi-layer caching strategy serving high-traffic content.

**Technologies:** Drupal 7, PHP, Memcached, MySQL, Ruby, YAML, Capistrano

#### Impact & Achievements:

- **Architected custom Memcached integration** for Drupal content types, dramatically reducing database load
- **Implemented intelligent cache invalidation** for devotionals ensuring fresh content across all days
- **Built reading plan metadata caching** improving page load times for frequently accessed plans
- **Developed landing page caching** with URL normalization handling diverse traffic patterns
- **Enabled non-technical content updates** through CMS while maintaining performance at scale

### Bible Gateway Mobile Apps (Native iOS/Android)

June 2011 - December 2024

**Launched mobile app initiative** with native iOS and Android applications in 2011, establishing Bible Gateway's mobile presence. Built comprehensive mobile Bible experience, then **led transition to React Native** (2016-2018) enabling cross-platform development efficiency.

**Technologies:** React Native, iOS, Android, Swift, Kotlin, JavaScript, WebView, jQuery/Zepto, React Hooks, Redux

#### Impact & Achievements:

- **Launched native mobile apps** (2011) bringing Bible Gateway to smartphones, expanding user reach

- **Implemented verse selection and favorites** with WebView integration, enabling seamless Bible study on mobile
- **Built scroll-to-verse functionality** with OSIS reference support for precise Bible navigation
- **Developed notes interface** with React Hooks refactoring, modernizing codebase and improving maintainability
- **Led React Native integration** (2016-2018) partnering with Michigan Labs, establishing cross-platform architecture
- **Enabled multi-verse selection** with visual feedback, enhancing mobile Bible study experience

## Bible Gateway Version 6 Platform Launch

June 2011 - July 2020

Led responsive redesign introducing **full user accounts system**, notes, highlights, and e-commerce capabilities. Launched **Bible Gateway Plus subscription tier** (2015) creating new revenue stream. Built Bootstrap-based frontend serving millions of users.

**Technologies:** PHP, MySQL, Bootstrap, jQuery, JavaScript, Responsive Design, Payment Processing

### Impact & Achievements:

- **Architected complete user accounts system** enabling personalized Bible study with notes and highlights
- **Launched Bible Gateway Plus subscription** (2015) creating premium tier and sustainable revenue model
- **Integrated e-commerce platform** enabling direct Bible translation purchases, expanding monetization
- **Delivered responsive design** ensuring optimal experience across desktop, tablet, and mobile browsers
- **Built on Bootstrap framework** accelerating development while maintaining design consistency

## Bible Gateway Version 5 Platform Rebuild

February 2011 - June 2011

**Led platform rescue project** rebuilding core infrastructure in 4 months after Bible Gateway Version 4 Rails platform failure. Made critical architectural decisions under pressure, successfully launching PHP-based solution incorporating best features from failed initiative while ensuring stability for millions of users.

**Technologies:** PHP, MySQL, jQuery, JavaScript, Apache

### Impact & Achievements:

- **Delivered complete platform rebuild in 4 months** rescuing failing infrastructure serving millions of users
- **Made critical technology decision** choosing proven PHP architecture over failed Rails approach
- **Incorporated Version 4 features** (sidebar, resources) salvaging valuable work from abandoned project
- **Ensured backward compatibility** maintaining existing URLs and user expectations during crisis transition

- **Demonstrated leadership under pressure** making decisive technical choices with business-critical consequences

## Bible Gateway Version 3 to Version 4 Platform Migration (Rails)

January 2009 - February 2011

Contributed as core member of **14-person team** on ambitious Ruby on Rails platform rewrite. Project ultimately abandoned due to performance issues, but experience directly informed successful Bible Gateway Version 5 rapid rescue project, demonstrating learning from failure.

**Technologies:** Ruby on Rails, Haml, Sass, jQuery, MySQL, Redis

### Impact & Achievements:

- **Participated in large-scale rewrite** gaining experience with Rails, modern web frameworks, and team collaboration
- **Developed accounts system and features** including topics taxonomy and linked resources
- **Identified performance bottlenecks** contributing to decision analysis that protected business from failed launch
- **Knowledge transfer to Version 5** applying lessons learned to inform rapid 4-month rebuild
- **Demonstrated resilience** pivoting from failed initiative to successful rescue project

## First Bible Gateway iOS App Prototype

June 2009

Created **prototype mobile app** for Bible Gateway during early smartphone era. **Established foundational mobile knowledge** and technical patterns that directly informed successful 2011 mobile app launch.

**Technologies:** iOS, Objective-C, Xcode, Mobile Development

### Impact & Achievements:

- **Pioneered mobile exploration** during iPhone's early years, demonstrating forward-thinking approach
- **Established iOS development capabilities** within organization
- **Created technical foundation** informing successful 2011 mobile app strategy
- **Gained critical experience** in mobile development patterns and platform constraints

---

## TECHNICAL EXPERTISE

### Languages & Core:

JavaScript (ES6+), PHP (7.2-8.x), Python, Ruby, Swift, Kotlin, TypeScript

### Frontend Development:

React, Preact, React Native, Redux, Redux Toolkit, Babel, webpack, esbuild, Bun, HTML5, CSS3, Responsive Design

### Backend & Frameworks:

Laravel, Node.js, Express, Drupal 7, Ruby on Rails, RESTful API Design, GraphQL

<b>Cloud &amp; Infrastructure:</b>	AWS (EKS, Lambda, CloudFront, Route53, S3, DynamoDB, IAM, WAF, Polly, Kinesis, SNS, SQS), Kubernetes, Helm, ArgoCD, Terraform, Docker
<b>Databases &amp; Caching:</b>	MySQL, Aurora, DynamoDB, Redis, Memcached, SQLite
<b>DevOps &amp; CI/CD:</b>	Ansible, GitHub Actions, Bitbucket Webhooks, Git, Nginx, Apache, Fastlane, CodePush
<b>Mobile Platforms:</b>	iOS, Android, React Native, Xcode, CocoaPods, Gradle, WebView, In-App Purchases
<b>Testing &amp; Quality:</b>	Cypress, Selenium, End-to-End Testing, Test Automation, CI/CD Integration
<b>Authentication &amp; Payments:</b>	OAuth2, SAML, JWT, Apple Sign-In, Braintree, Apple IAP, Google Play Billing, Stripe
<b>Monitoring &amp; Analytics:</b>	Firebase Analytics, Bugsnag, Crashlytics, AWS Kinesis Firehose, Google Analytics
<b>Additional:</b>	GDPR Compliance, OneTrust, Security Best Practices, Performance Optimization, System Architecture, Team Leadership, Mentorship, API Design, Reverse Engineering

---

## EDUCATION

---

### Bachelor of Science in Computer Science

Taylor University | Graduated 2006

Studies focused on software engineering, system design and analysis, and computer and network architecture.