

# Faith.online Technical Requirements Document (TRD)

**Version:** v1.0

**Date:** August 7, 2025

**Prepared by:** Mike Turner, Fractional CTO

---

## 1. Project Overview

Faith.online is a moderated, Christ-centered digital platform designed for believers, churches, ministries, Christian businesses, and creators. It provides an ad-free, distraction-free, and Gospel-aligned alternative to secular social media platforms, supporting prayer, testimony, connection, events, and discipleship through modern UX.

---

## 2. Technology Stack Overview

### Frontend:

- Framework: Next.js (TypeScript)
- UI: Responsive, mobile-first
- SSR/SSG: Supported via AWS Amplify or CloudFront
- PWA: Installable with offline support (Service Worker)

### Backend:

- Framework: NestJS (Node.js, TypeScript)
- API Type: RESTful (extendable to GraphQL)
- Realtime Messaging: WebSockets (via AWS ALB + ECS or API Gateway + Lambda)

### Database:

- MongoDB Atlas (deployed in AWS region)

- Schema includes: Users, FaithPages, Posts, Comments, Reactions, Moderation, Events, Settings

#### **Authentication & Authorization:**

- JWT-based auth with refresh tokens
- Roles: Believer, Creator, Admin, Superadmin
- RBAC enforced through NestJS guards and route middleware

#### **Cloud Platform:** Amazon Web Services (AWS)

- Justification: Best fit for Ikonic stack, scalability, moderation tooling, SSR support, and MongoDB hosting
- 

### **3. Infrastructure & DevOps**

- **Hosting Services:**
  - Frontend: AWS Amplify or S3 + CloudFront
  - Backend: AWS ECS (Fargate) or Lambda (via SST)
  - Database: MongoDB Atlas (multi-AZ deployment)
- **Security:**
  - AWS Secrets Manager for credentials
  - Role-based access via IAM and API-level restrictions
- **Environments:**
  - dev, staging, production
- **CI/CD:**
  - GitHub Actions + AWS CodeDeploy or CodePipeline

- **Monitoring & Logging:**
    - AWS CloudWatch
    - Application-level logs retained for moderation audits
- 

## 4. Core MVP Modules

Module	Description
Faith Directory	CRUD for FaithPages, claim/verify flow, search + filter
Rhythm Keeper (Basic)	Habit tracker, calendar, shared planning tools
Messaging	Direct messages with safety filters, real-time via WebSockets
User Posts	Text/image posts, testimonies, prayer requests
Reactions	Positive-only (Amen, Bless)
Comments	Threaded replies with moderation queue
Connections System	Connect/unconnect, "My Network" view
Community Stream	Algorithmic or chronological post feed
Moderation	Manual flagging, AI word filter, admin dashboard
Daily Bible Verse	Rotating scripture feed from in-app DB or API
Christian News Feed	Curated RSS imports displayed on dashboard
White-label Branding	Theme color/font/logo customization per FaithPage

---

## 5. Moderation & Safety

- **Filters:**
  - AI word filter for profanity, politics, harmful content
  - Manual flagging by users

- **Admin Tools:**
    - Moderation queue, approve/reject, ban/suspend users
    - Role management and analytics dashboard
  - **Content Types Filtered:**
    - Posts, Comments, Messages
  - **Youth Privacy:**
    - No public profiles under 13, COPPA-inspired limitations
    - All data stored in compliance with GDPR and privacy best practices
  - **Escalation:**
    - Theological disputes escalated to admin team for manual resolution
- 

## 6. Theming & Customization

- Each FaithPage can customize its theme
  - Custom elements: logo, colors, fonts
  - Stored in MongoDB and dynamically applied at render
- 

## 7. Testing & QA

- Unit tests for UI components and API logic
- Integration tests for auth flows, moderation, messaging
- Manual QA during alpha, beta, pre-launch
- Performance testing using Lighthouse

- PWA installable compliance check

---

## 8. Deployment Milestones

Phase	Target Date	Deliverables
Infrastructure Setup	Aug 2025	Auth, DB schema, CI/CD, component library
MVP Core Dev	Sep–Nov 2025	Directory, Messaging, Feed, Rhythm Keeper
QA & Beta Rollout	Nov–Dec 2025	Internal QA, pilot testing, bug fixes
Final Polish	Dec 2025	Mobile optimization, performance tuning
Public Launch	Jan 2026	Live deployment of Faith.online MVP

---

## 9. Open Questions & Assumptions

- Stripe integration is Phase 2 (not included in MVP)
- No headless CMS is planned; Bible/news content will be in-app or RSS
- AWS will host all production infrastructure
- Project cost management will follow weekly milestone billing