

Chris Howarth

Software Engineer · BBC

PROFILE

Full-stack software engineer at the BBC with nearly four years of production experience on some of the corporation's highest-traffic live content journeys. I work across the complete delivery stack — React and TypeScript on the client, Node.js middleware and SSR, backend API composition across multiple FABL modules, and Belfrage routing infrastructure — giving me an unusually broad view of how BBC web products are built and delivered at scale. I bring a strong focus on code quality, automated testing, and operational reliability, and actively shape how my team works — leading adoption of AI-assisted development tooling and owning end-to-end experiment delivery through Optimizely.

CORE SKILLS

Languages: TypeScript, JavaScript (expert); Node.js; code-review exposure to Java (Apache Flink/Maven)

Frontend: React, SSR, component-driven architecture, shared design system integration

Backend: API composition, service orchestration, data normalisation, REST API design, middleware

AWS: Lambda, X-Ray, CloudFormation, CDK, CodeBuild, CodePipeline, S3, CloudWatch, Step Functions — application and integration level; infrastructure provisioning exposure via Webcore Platform rotation

Experimentation: Optimizely (end-to-end experiment ownership); feature flagging; progressive rollout patterns

Testing: Unit, integration, snapshot/visual, browser regression; TDD practitioner

Platform: Belfrage routing, CI/CD (Jenkins, CodePipeline), large-scale monorepo workflows

Auth / Identity: CIAM integration — IDCTA, signed-in/signed-out UX, account-gated features; OAuth 2.0 / OIDC at integration layer

AI Tooling: GitHub Copilot advocate; built internal context-generation tooling; authored AI agent decision framework

Practices: Agile delivery, code review, pairing, incident response, technical documentation

EXPERIENCE

Software Engineer — BBC Live · BBC

Aug 2024 – Present

Full-stack engineer on BBC Live, a high-traffic real-time product serving millions of users across Sport, News, Radio, and TV events. Work spans React/TypeScript frontend, Node.js middleware, backend FABL module ownership, and Belfrage routing.

Live experience platform

- Engineered and maintained BBC Live web experiences across middleware and React container layers, delivering resilient real-time page behaviour for millions of concurrent users across Sport, News, Radio, and TV events.
- Composed dynamic page layouts, tabs, and container wiring for real-time content — implementing business logic for page states, fallbacks, conditional rendering, and feature availability across a wide range of live event types.
- Integrated multiple internal APIs and data services to power Live content, event/calendar, comments, and uploader experiences, with robust error handling, guarded failures, and resilient fallback paths.
- Implemented request-context-driven behaviour across service, locale, region, host, and environment dimensions — ensuring correct rendering across BBC's multilingual and multi-platform surfaces.
- Built account-aware user journeys, integrating IDCTA sign-in prompts, availability checks, and personalisation touchpoints across Live features.

- Contributed to internal developer tooling to improve code navigation, change safety, and context indexing within the large shared monorepo.

Experimentation and feature rollout

- Owned end-to-end technical delivery of product experiments using Optimizely — contributing to metric selection and variant design, building experiments, implementing code changes, and monitoring throughout the experiment lifecycle.
- Implemented and maintained feature-flag-driven behaviour to enable safe incremental releases, progressive rollouts, and variant-based iteration across Live journeys.
- Collaborated with product managers on experiment briefs, translating hypotheses into robust technical implementations with appropriate instrumentation.
- Contributed to editorial and technical discussions around AI-generated football content within Live page post streams, helping define the workflow to integrate automated content alongside standard journalist posts.

AI tooling and advocacy

- Active advocate for AI-assisted development within the Live team — promoting adoption of GitHub Copilot and supporting colleagues in integrating AI into their daily workflows.
- Built an internal context-generation tool to improve the efficiency of AI-assisted development, reducing token usage by enabling more targeted, relevant context to be passed to AI agents.
- Authored a decision-guide document helping engineers identify the most appropriate AI agent or workflow for a given task, improving consistency and effectiveness across the team.

Backend module ownership (FABL)

- Owned and maintained a portfolio of production FABL modules across domain, composition, and application layers — including Live Experience, Stream, Post, Media Access Tokens, Programme Playout, Live Calendar, Live Hub, and Live Experience SEO.
- Designed API composition patterns normalising heterogeneous upstream data (Tipo, ARES, EAVIS, iSite, programme metadata and availability services) into stable client-facing response contracts.
- Built orchestration logic for live-page components — streams, media collections, branding, SEO metadata, and calendar data — improving consistency and reusability across BBC digital surfaces.

Routing and platform (Belfrage)

- Owned Live-team route specs and operational configuration in Belfrage for News, Sport, and Live Hub — including canonical routes, redirects, pagination, and app-specific variants across multiple delivery environments.
- Maintained resilience controls: request pipelines, query allowlists, alarm channels, runbooks, and stack/edge targeting across BBC delivery infrastructure.

Quality and observability

- Drove quality via broad automated test coverage (unit, integration, visual/snapshot, browser regression) across a large-scale shared monorepo.
- Contributed to production reliability through observability-aware runtime code, AWS X-Ray instrumentation, and failure recovery patterns.
- Participated in out-of-hours incident response for high-availability live content services.

Graduate Software Engineer · BBC

Sept 2022 – Aug 2024

Joined BBC engineering on a graduate scheme, completing four production rotations across distinct teams before joining BBC Live permanently. Highlights by rotation:

BBC Webcore Platform

- Worked within the team responsible for deploying and maintaining the BBC's primary web frameworks using native AWS technologies.
- Gained hands-on experience with infrastructure as code — working across CloudFormation, CDK, CodeBuild, CodePipeline, Lambda, S3, CloudWatch, and Step Functions in a production environment.
- Contributed to CI/CD pipeline ownership and deployment tooling for high-traffic BBC web products.

BBC Live

- Full-stack live experience work across React, TypeScript, and Node.js middleware — the direct foundation for the current role.

BBC Sport Web Data

- Full-stack data manipulation and presentation in JavaScript and TypeScript, building data-driven sport content features.

BBC Sport Mobile (iOS / Android)

- Native mobile development in Swift and Kotlin, gaining cross-platform experience beyond the web stack.

BEFORE ENGINEERING

Prior to retraining as a software engineer, I ran my own businesses across property development and retail — experience that sharpened commercial judgement, stakeholder communication, and the ability to manage competing priorities under real pressure. That background informs how I approach problems and work with people, and it is part of why I chose engineering as a second career rather than falling into it.

EDUCATION

BSc (Hons) Computing & IT — First Class Honours · [Open University](#)

2019–2022

Distinction grades in 9 of 12 modules. Relevant modules: Algorithms & Data Structures, Web Technologies, Object-Oriented Java, Cisco Networking (CCNA 1–3), Data Management & Analysis.

BSc (Hons) Forensic Psychology — 2:2 · [UCLan, Preston](#)

2002–2005

Relevant modules included Statistical Analysis, IT & Study Skills, and a research dissertation — providing a grounding in rigorous analytical thinking and evidence-based reasoning.

PROFESSIONAL DEVELOPMENT

Ongoing development through BBC Academy training programmes, internal guilds, and self-directed learning. Completed courses include:

- Test-Driven Development — BBC Academy
- Advanced TypeScript — BBC Academy
- Swift for iOS development — BBC Academy
- Threat Modelling — BBC Academy
- Cisco Networking (CCNA 1, 2 & 3) — Open University
- Active member of the BBC Security Champions network, attending bi-weekly sessions run by the internal Security Guild.
- Regular participant in BBC engineering community knowledge-sharing and internal tech talks.

ADDITIONAL

Driving licence: Full, clean UK driving licence

Availability: Available for hybrid working — Salford or London