

BRANDON EASTWELL

brandoneastwell@gmail.com | +44 7479271984 | github.com/brandoneastwell | linkedin.com/brandoneastwell

Software Engineer with a First-Class Computer Science degree and experience building backend services, APIs and full-stack web applications using TypeScript, Python, Node.js and PostgreSQL through ambitious personal projects. I'm naturally curious and excited about the challenge of solving difficult technical problems. I love learning new technologies by building products and solving everyday issues with them that people can use.

EDUCATION

University of Essex - BSc Computer Science (First Class Honours) September, 2021 - August, 2024

- Graduated with First-Class Honours in Computer Science
- Awarded Dean's list for excellence in years 2021 - 2024
- Modules: Data Structures and Algorithms, Databases, Software Engineering, Web Development

TECHNICAL SKILLS

- **Languages:** TypeScript, JavaScript, Python, Java, SQL
- **Backend & Databases:** Node.js, Express, NestJS, Django, REST APIs, PostgreSQL, Prisma
- **Frontend:** React, React Native, Next.js, HTML, CSS, TailwindCSS
- **Cloud & Tools:** Git, Claude Code, Docker, AWS (S3, EC2), CI/CD, GitHub Actions, Jest, Jira
- **Engineering:** Test-Driven Development (TDD), AI-assisted development, REST API design

EXPERIENCE

Web Developer (Contract) | Eastys Accessories March, 2025 - January, 2026

- Built a responsive ecommerce storefront for a family-run business using React and TypeScript, creating reusable components for product listings, navigation and other shared interface elements.
- Integrated the storefront with the eBay API to retrieve product and inventory data, keeping product availability aligned with the business's existing eBay store.

PROJECTS

Outdoor Reflections – Offline First Hiking Journal June, 2026 - September, 2026

outdoor-reflections.vercel.app | github.com/BrandonEastwell/outdoor-reflections

- Built and docker deployed a NestJS and PostgreSQL API supporting authentication, mixed-media journal storage, offline synchronization, password recovery and AI-assisted writing.
- Engineered a synchronisation workflow that validates every entry of ownership and edit timestamps using a LWW policy, preventing stale device data from overwriting newer reflections.
- Secured email/password and Google OAuth sign-in by implementing short-lived JWTs, bcrypt-hashed refresh sessions and HttpOnly cookies, allowing sessions to be verified without storing usable credentials.
- Developed an AI-generated writing suggestion service by integrating Groq with contextual prompts and schema-constrained output, preventing invented memories and unrelatable responses.
- Built 48 passing integration, unit and end to end tests across 12 suites, covering database transactions, authentication, cookies, token refresh, OAuth callbacks and synchronization failures.

Geospatial Photography Discovery Platform

March, 2025 - August, 2025

github.com/BrandonEastwell/photography-app

- Designed a Django REST-style backend for a photography platform, modelling users, profiles, sessions, photos, cameras, lenses, and geographic coordinates with PostgreSQL/PostGIS to support discovery features.
- Solved complex photo-discovery requirements by implementing time-based filtering, EXIF matching, PostGIS distance annotations, and weighted fallback ranking, returning useful results even with limited data.
- Developed secure API endpoints with JWT middleware, ownership validation, password controls, session refresh, login-attempt tracking, and protected upload and deletion endpoints.
- Built a photo-processing and EXIF normalising pipeline that saves image details such as GPS location, camera, lens, ISO, shutter speed, and focal length while storing uploaded files in AWS S3.

DJ Set Finder Chrome Extension

January, 2025 - March, 2025

github.com/brandoneastwell/ra-set-finder

- Built a full-stack Chrome Manifest V3 extension using React and TypeScript that scrapes Resident Advisor event lineups and surfaces up to five relevant long-form YouTube sets per artist.
- Implemented event and artist-level caching with local storage, reducing redundant scraping, API requests, and YouTube quota consumption.