

Felix Riley-Kay

[enfey](#)  | [felixrk](#)  | [enfey.sh](#)  | [me@enfey.sh](#) 

EDUCATION

University of Nottingham

Expected July 2027

Computer Science MSci (Hons)

Year 3 average: 78% (First)

- **Dissertation:** *Eidolon – Cross-Vendor Translation Validation of GPU Collective Operations in MLIR*
- **Top modules:** Security (88%), Systems & Architecture (87%), Graphics (87%), Cryptography (79%), Compilers (76%)

St John Rigby College

Sep 2021 – Jun 2023

A-Levels: Computer Science, Psychology, Sociology

A, A*, A**

EXPERIENCE

Linkers, Loaders, and ARMv7-A Research Group

Oct 2025 – Jan 2026

Linkers & Loaders study group – highest grade awarded

C, CMake/CTest, Docker, GitHub Actions

- Co-led weekly research discussions analysing foundational literature, contemporary research and open-source implementations of linkers, loaders and executable formats.
- Independently built **Tlalloc**, a C toolchain comprising **meld**, an ARMv7-A ELF linker (~6.8k lines of C), and **lens**, an ELF parser/API.
- Built meld to produce static and PIE executables with PLT/GOT, RELRO, symbol versioning, and relaxation, linking Lua 5.4 and SQLite in under 7 ms each.
- Optimised meld to link 100k functions 2.3× faster than GNU ld, within 8% of single-threaded LLD, and lens to parse ELF files 1.45× faster than GNU readelf.
- Wrote mold-inspired QEMU integration tests, with GitHub Actions CI for ARM32/AArch64 builds and static analysis.

IBM – Industry-Sponsored Team Project

Sep 2024 – May 2025

Frontend Lead, 8-person team, University of Nottingham

IBM watsonx, Next.js, React, Carbon, Firebase

- Led the frontend of a RAG chatbot on IBM watsonx for student onboarding, with Firebase Authentication.
- Defined the team's programming standards and reviewed the Python RAG pipeline (LangChain, Milvus).
- Built a centralised SCSS theming system on IBM's Carbon Design System for a consistent, accessible UI.
- Worked with senior IBM engineers through weekly meetings and code reviews of the team's work.

TECHNICAL SKILLS

Languages: C, C++, C#, Python, Haskell, SQL, Bash, ARM assembly, SCSS

Systems & Build: Linux, ELF, QEMU, CMake/CTest, Ninja, ASan/UBSan, cppcheck, Playwright

Streaming & Games: WebRTC, MediaMTX, Unity, OpenGL, PlayFab

Cloud & Tools: Docker, GitHub Actions, Terraform, GCP, Firebase, Git/GitHub, Jira, Trello, Figma

PROJECTS

Eidolon | *MLIR, LLVM, C++, Z3, NVVM, ROCm, SPIR-V*

Oct 2026 – Present

- Researching and designing an SMT-based translation validator for MLIR GPU collective operations, using symbolic execution to verify semantic preservation across NVVM, ROCm and SPIR-V lowering paths.

Deliverance | *Next.js, TypeScript, WebRTC, MediaMTX, Docker, GCP, Terraform*

- Built a live video radio station for extreme alternative music, streaming 1080p from OBS through MediaMTX to viewers over WebRTC at sub-second latency
- Wrote a WebRTC (WHEP) player from scratch that tracks live status, reconnects dropped streams, and ends server sessions when viewers leave.
- Set up GitHub Actions CI/CD with linting, type checks, Playwright E2E and visual tests, and Dependabot.
- Designed the GCP architecture around egress costs, with Terraform, Postgres/Drizzle, and Stripe.

Reverie | *Unity, C#, Kinematic Character Controller, PlayFab*

88% (top mark in cohort)

- Designed and built a fast, unforgiving first-person parkour game across five levels of rising difficulty.
- Wrote the movement controller on top of KCC, with wall-running, wall-jumps, slides, and buffered jump inputs.
- Decoupled gameplay from presentation (plain-data input, event-driven audio); added PlayFab leaderboards.

TECHNICAL INTERESTS

Compilers & Languages: Compilers (MLIR/LLVM), programming language design, theoretical computer science

Systems: Low-level programming, GPUs and heterogeneous computing, hardware/software co-design, open hardware

Formal Methods: Compiler correctness, memory models and concurrency semantics, memory safety