NANTHEESAN RAVEENTHIRAN

📍 London, United Kingdom | 📞 +44 7411 292986 | ✉️ nantheesanr@gmail.com

🔗 [Portfolio URL]

🔗 LinkedIn | 🔗 GitHub |

EDUCATION

BSc Computer Science | University of Southampton

Classification: **First Class with Honors**

- **Relevant Modules:** Machine Learning, Compiler Design, Theory of Computing, Programming Languages.

A Levels | RR6

- **A*** Further Mathematics, **A*** Mathematics, **A*** Physics

---

TECHNICAL SKILLS

- **Languages:** Java (17+), Haskell, TypeScript, Python, SQL, CSS3/HTML5.
- **Frameworks & Libraries:** React 18, Node.js, JavaFX, Pandas, Scikit-Learn, NetworkX, JUnit.
- **Concepts:** Concurrency, Event-Driven Architecture, Graph Theory, REST APIs, Agile/Scrum.
- **Tools:** Git, Docker, Maven, Vite, Unix.

ENGINEERING PROJECTS

Predictive Scientometrics & Fraud Detection (Dissertation) | *Python, Scikit-Learn, Graph Theory*

- Engineered a machine learning pipeline to predict academic paper retractions.
- Designed a novel "**Meta-Net**" **architecture** to infer semantic relationships between papers using **SBERT Transformers** and cosine similarity, overcoming citation sparsity.
- Implemented complex graph algorithms (**PageRank**, **Betweenness Centrality**) to analyze network topology and identify fraudulent citation circles.

Graph Database Query Language | *Haskell, Alex, Happy*

- Created a domain-specific language (DSL) for querying graph databases, implementing a custom **CEK-style state machine** for the interpreter.

- Built a robust dual-parsing architecture using **Alex** (Lexer) and **Happy** (Parser) with strict schema validation for .n4j files.

Interactive Portfolio Platform | *React 18, TypeScript, Vite*

- Built a performant Single Page Application (SPA) using **Atomic Design** principles and strict TypeScript interfaces for scalability.
- Engineered custom hooks utilizing the **Intersection Observer API** to implement efficient lazy-loading and scroll-triggered animations.

Airport Runway Simulation | *Java, JavaFX, MVC*

- Developed a safety-critical desktop application to calculate aviation parameters (TORA/ASDA) and visualize runway redeclarations in real-time.
- Adopted the Model-View-Controller (MVC) pattern to separate calculation logic from the interactive XML-based GUI

Real-Time Multiplayer Tetris Engine | *Java 17, JavaFX, Multithreading*

- Architected a high-performance desktop game featuring real-time **PvP multiplayer** via **WebSockets** and a custom string-based network protocol.
- Implemented a **non-blocking game loop** using ScheduledExecutorService to decouple game logic from the UI thread, ensuring consistent tick rates regardless of rendering load.
- Designed a thread-safe **Observer Pattern** system to marshal state updates to the JavaFX Application Thread without race conditions.