# KEE SONG YANG

+65 8715 7381    contact@songyang.dev    github.com/s0ngyang    linkedin.com/in/keesongyang    songyang.dev

## EDUCATION

**National University of Singapore (NUS)**                                                                                              Expected May 2026
*Bachelor of Computing (Computer Science), Honours*
- **Relevant Coursework**: Software Engineering, Data Structures & Algorithms, Operating Systems, Databases, Computer Networks
- **Teaching Assistant**: CS1010X Programming Methodology — Python

## WORK EXPERIENCE

**Ola Chat** (Singapore), *Software Engineer Intern*                                                                                    Jul 2025 - Present
- Engineered **React** mini-games with **WebView**, ensuring UI consistency through meticulous design reviews and performance optimizations, including lazy loading and TanStack Query caching, resulting in higher user engagement
- Migrated **CI/CD** to a modular two-step flow by creating reusable GitHub workflows for artifact builds and **Aliyun Flow** for deployments, cutting server load and improving maintainability and separation of concerns
- Optimized deployment process of a monorepo by analyzing complexity vs. downtime tradeoffs; selected a remove-and-replace method over symlink-based deployment, ensuring maintainability while keeping downtime insignificant (<0.2s)

**Voltade** (Singapore), *Software Engineer Intern*                                                                                     Jan 2025 - Jun 2025
- Spearheaded end-to-end design of AI curriculum planner adopted across 28 preschool branches, showcasing attention to detail by iteratively refining features through weekly stakeholder reviews, live demos, and continuous feedback loops
- Automated data migration across platforms by building an **ETL**-style pipeline using **Graphile Worker** cron jobs to extract data from external sources, transform WhatsApp message data and load into **PostgreSQL**, ensuring idempotency across 20000+ records
- Resolved a **Supabase Auth** production bug affecting Outlook users by analysing server logs and identifying email prefetching behaviour that caused magic links to expire
- Developed a video-to-report tool using **React** with **TanStack, FFmpeg** to scale and extract frames, and **OpenAI GPT** tool calls to identify students, reducing teachers' time spent per report by 75%
- Learnt **Vue** and **Ruby on Rails** from scratch within 2 weeks to launch a broadcast feature with metrics tracking leveraging WhatsApp Cloud API to send messages to 500+ customers per campaign

**Podsmart AI** (United States), *Software Engineer Intern (Remote)*                                                                     May 2024 - Aug 2024
- Built an intuitive audio player in **React** with a mobile-responsive design, enhancing user experience through features like progress bar markers for key topics, synchronised playback with transcript timestamps, and direct navigation to identified entities
- Devised a usage tracking schema using **FastAPI**, **Supabase** with **PostgreSQL** to enforce user monthly podcast summarisation limits, eliminating a loophole that allowed unlimited access to transcribed episodes
- Resolved a critical authorisation flaw by implementing **role-based access control** middleware using **JWT**, preventing free-tier users from accessing premium features
- Optimised API performance for episode page by adding **composite index** to SQL table, sped up page loading by 60%

## PROJECTS

**TypeDash V2 (Repo | Link)**                                                                                                           Jul 2025 - Present
- Developed a real-time type-racing game using **React** and **WebSockets** (migrated from **Socket.IO**), leveraging **Redis** for centralised game state management and **Redis Pub/Sub** to broadcast typing progress to users within multiplayer rooms
- Migrating backend from Express.js to Golang for better concurrency support; implementing a **Redis**-powered leaderboard system for real-time ranking updates

**Exportifellas (Repo | Link)**                                                                                                         Jan 2025
- Ideated a plaintext playlist extraction feature using the **Spotify API** in an open-source fork, deployed for internal use in NUS Raffles Hall to streamline song voting process for all performances

## TECHNICAL SKILLS

**Languages**: JavaScript, TypeScript, SQL, Python, Java
**Frameworks**: React.js, TanStack, Tailwind CSS, Hono.js, FastAPI, Socket.IO, Firebase, Supabase, Drizzle ORM
**Technologies**: Linux, Bash scripting, LLMs, CI/CD