

# KEE SONG YANG

+65 8715 7381 [contact@songyang.dev](mailto:contact@songyang.dev) [github.com/s0ngyang](https://github.com/s0ngyang) [linkedin.com/in/keesongyang](https://linkedin.com/in/keesongyang) [www.songyang.dev](http://www.songyang.dev)

## EDUCATION

### National University of Singapore (NUS)

Expected May 2026

*Bachelor of Computing (Computer Science), Honours*

- **Relevant Coursework:** Software Engineering, Object-Oriented Programming, Database Systems, Information Security, Network Systems, Operating Systems, Data Structures & Algorithms, AI & ML
- **Teaching Assistant:** CS1010X Programming Methodology — Python and Java

## WORK EXPERIENCE

### [Ola Chat](#) (Singapore), *Software Engineer Intern*

Jul 2025 - Present

- Built and deployed responsive, multi-brand static sites with **Vite** and **PNPM**; automated CI builds with domain-specific assets and streamlined artifact deployment to **Aliyun ECS**, simplifying **nginx** setup across 3 domains
- Built interactive **React**-based mini-games for **WebView** and **gRPC** integration within a Flutter app; applied lazy loading, image optimization and **TanStack Query** caching to enhance load speed and runtime performance
- Implemented a pre-push shell script to run build on changed subfolders in a monorepo, catching build errors before deployment

### [Voltade](#) (Singapore), *Software Engineer Intern*

Jan 2025 - Jun 2025

- Led the end-to-end design and launch of an AI curriculum planner adopted by 200 users across 28 preschool branches by running weekly customer meetings, live demos, and continuous feedback loops
- Resolved a **Supabase Auth** production bug affecting Outlook users by identifying email prefetching behavior causing expired magic links; replaced magic link authentication with OTP-only flow after analyzing Supabase auth limitations and gathering user feedback
- 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%
- Automated data migration across platforms by building an **ETL** pipeline using **Graphile Worker** CRON jobs in TypeScript to extract data from external sources, transform data through validation and cleaning steps and load into **PostgreSQL** database, ensuring idempotency across 20000+ records
- Learnt **Vue** and **Ruby on Rails' MVC architecture** from scratch within 2 weeks to launch a WhatsApp campaign automation using **ActionCable**, **Redis-backed Sidekiq workers** and WhatsApp Cloud API, enabling real-time CRM messaging to 500+ clients with delivery metrics tracking

### [Podsmart AI](#) (United States), *Software Engineer Intern (Remote)*

May 2024 - Aug 2024

- Built an intuitive audio player in **React** in a mobile responsive layout, enabling UX through progress bar markers for key topics, synchronized playback with transcript timestamps, and direct audio jumps to identified entities
- Devised a usage tracking schema using **FastAPI**, **Supabase** with **PostgreSQL** to precisely track and enforce user monthly podcast summarisation limits, eliminating a loophole that allowed unlimited access to transcribed episodes
- Resolved a critical authorization 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, which sped up page loading by 60%

## PROJECTS

### TypeDash V2 ([Repo](#) | [Link](#))

Jul 2025 - Present

- A type-racing game built with **React** and **WebSockets** (previously **Socket.IO**) to broadcast live typing progress in multiplayer rooms and **Redis** as the source of truth for game state
- 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](#))

- Built a plaintext playlist extraction feature using the **Spotify API** in an open-source fork, deployed for internal use in NUS Raffles Hall, streamlining the song voting process for all future performances

## TECHNICAL SKILLS

**Languages:** JavaScript, TypeScript, SQL, Python, C, Java

**Frameworks:** React.js, Next.js, TanStack, Tailwind CSS, Hono.js, FastAPI, Socket.IO, Firebase, Supabase, Drizzle ORM, gRPC

**Technologies:** Linux, Bash scripting, Docker, LLMs, CI/CD, Protobuf