

# 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) [songyang.dev](https://songyang.dev)

## EDUCATION

|   |                   |
|---|-------------------|
| <b>National University of Singapore (NUS)</b><br><i>Bachelor of Computing (Computer Science), Honours</i>   | Expected Jun 2026 |
| <ul style="list-style-type: none"><li>• <b>Relevant Coursework:</b> Software Engineering, Data Structures &amp; Algorithms, Operating Systems, Databases, Computer Networks</li><li>• <b>Teaching Assistant:</b> CS1010X Programming Methodology — Python</li></ul> |                   |

## WORK EXPERIENCE

|  |                     |
|--|---------------------|
| <b>Ola Chat</b> (Singapore), <i>Frontend Software Engineer Intern</i>  | Jul 2025 - Present  |
| <ul style="list-style-type: none"><li>• Engineered <b>React</b> mini-games with <b>WebView</b>, ensuring UI consistency through meticulous design reviews and performance optimizations, including lazy loading and <b>TanStack Query</b> caching, improving user experience</li><li>• Refactored <b>CI/CD</b> into a modular two-step architecture, creating reusable <b>GitHub Actions</b> for builds and <b>Aliyun Flow</b> for deployments; improved maintainability and separation of concerns while reducing server load</li><li>• Integrated centralized error monitoring via <b>Sentry</b> into internal npm templates, enabling session replay tracking, Core Web Vitals metrics, and automated Slack alerts — ensuring observability across deployments</li><li>• Refactored legacy URL routing across client, server, and database layers, standardizing URL structure and simplifying <b>nginx</b> configuration to reduce technical debt</li></ul>  |                     |
| <b>Voltade</b> (Singapore), <i>Full-Stack Software Engineer Intern</i>   | Jan 2025 - Jun 2025 |
| <ul style="list-style-type: none"><li>• 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</li><li>• Automated data migration across platforms by building an <b>ETL</b>-style pipeline using <b>Graphile Worker</b> cron jobs to extract data from external sources, transform WhatsApp message data and load into <b>PostgreSQL</b>, ensuring idempotency across 20000+ records</li><li>• Resolved a production bug affecting Outlook inbox users by analysing server logs and identifying <b>Supabase Auth</b> email prefetching behaviour that caused magic links to expire</li><li>• Developed a video-to-report system using <b>React</b> and <b>FFmpeg</b>, optimising speed by extracting key frames in compact format, leveraging OpenAI tool calls for automated student identification, cutting report generation effort by 75%</li><li>• Learnt <b>Vue</b> and <b>Ruby on Rails</b> from scratch within 2 weeks, integrated Meta webhook and API to launch a broadcast feature with tracking metrics, allowing messages to be sent to 1000+ customers per campaign</li></ul> |                     |

|  |                     |
|--|---------------------|
| <b>Podsmart AI</b> (United States), <i>Full-Stack Software Engineer Intern (Remote)</i>  | May 2024 - Aug 2024 |
| <ul style="list-style-type: none"><li>• Devised a usage tracking system using <b>FastAPI</b> &amp; <b>Supabase</b>, eliminating a loophole that allowed free access to transcribed episodes</li><li>• Resolved a critical authorisation issue by implementing <b>role-based access control</b> middleware using <b>JWT</b>, preventing free-tier users from accessing premium features</li><li>• Identified and resolved <b>SQL</b> bottleneck by adding a composite index, optimizing API performance and reducing load time by 60%</li></ul> |                     |

## PROJECTS

|  |                    |
|--|--------------------|
| <b>TypeDash V2</b> ( <a href="#">Repo</a>   <a href="#">Link</a> )   | Jul 2025 - Present |
| <ul style="list-style-type: none"><li>• Developed a real-time type-racing game using <b>React</b> and <b>WebSockets</b> (migrated from <b>Socket.IO</b>), leveraging <b>Redis</b> for centralised game state management and <b>Redis Pub/Sub</b> to broadcast typing progress to users within multiplayer rooms</li><li>• Migrating backend from Express.js to Golang for better concurrency support; implementing a <b>Redis</b>-powered leaderboard system for real-time ranking updates</li></ul> |                    |

|   |          |
|---|----------|
| <b>Exportifellas</b> ( <a href="#">Repo</a>   <a href="#">Link</a> )  | Jan 2025 |
| <ul style="list-style-type: none"><li>• 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</li></ul> |          |

## TECHNICAL SKILLS

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

**Frameworks:** React.js, TanStack, Tailwind CSS, FastAPI, Socket.IO, Firebase, Supabase, Drizzle ORM, Linux, Bash, LLMs