

# Thuc Nguyen Phong Pham

University of British Columbia | 3rd Year

**Email:** percypham279@gmail.com | **LinkedIn:** linkedin.com/in/phong-pham-a0292229b **GitHub:** github.com/prcpham-dev

## Experience

---

### Software Engineering Intern – Frontend Focus

Apr 2023 – Aug 2023

*Tinh Van Software*

- Developed responsive web components using React.js, Node.js, TypeScript, Bootstrap, and Tailwind CSS for a client-facing platform.
- Collaborated with backend engineers to integrate and test RESTful APIs using Postman.
- Participated in sprints, design reviews, and deployment planning using Git.
- Gained hands-on exposure to backend infrastructure involving FastAPI and Uvicorn.

### Software Engineering Intern – Full-Stack

Apr 2024 – Aug 2024

*Tinh Van Software*

- Designed and developed a production-ready Retrieval-Augmented Generation (RAG) web app with PDF ingestion and contextual AI chat capabilities.
- Implemented backend services in Python using FastAPI, LangChain, and Google Gemini Pro to support semantic search and response generation.
- Integrated Redis and Celery for asynchronous task queues, improving responsiveness and scalability.
- Built a responsive frontend using React.js and Bootstrap for file upload, chat interaction, and displaying AI-generated insights.
- Deployed on a cloud VPS using Docker, Nginx, HTTPS (SSL), and environment-managed API keys.
- Created RESTful API endpoints for document processing and chat flow, supporting up to 3MB PDF uploads.
- Performed data preprocessing using exploratory data analysis (EDA) and KNN-based filtering.

## Projects

---

### ChatDocument

*FastAPI, LangChain, Redis, Celery, React.js, Bootstrap, Docker, Nginx*

- Developed a Retrieval-Augmented Generation (RAG) app enabling users to upload and analyze PDFs with contextual AI chat.
- Integrated Google Gemini Pro (2M tokens) with LangChain for embeddings, semantic search, and long-context reasoning.
- Built a scalable backend with FastAPI, offloading heavy tasks asynchronously using Redis and Celery.
- Designed a responsive frontend in React.js and Bootstrap for file upload, chat interaction, and insights display.
- Deployed on a VPS with Docker, Nginx, and SSL certificates, securing API keys via environment management.

### InstaSave

*Chrome Extension, JavaScript, MutationObserver*

- Created a browser extension that injects a “Save” button into Instagram posts and stories for quick media downloads.
- Used DOM manipulation and MutationObserver to dynamically detect and modify Instagram’s interface.

- Delivered lightweight functionality to open or save images and videos directly from the browser.

## Shufa Character Downloader

*Selenium, asyncio, Tkinter, ChromeDriver*

- Built a desktop application with a Tkinter-based UI to automate downloading of Chinese calligraphy images.
- Supported search by author, phrase, or style with configurable batch size, wait time, and image count.
- Automated scraping with Selenium WebDriver to fetch and filter results for the selected calligrapher.
- Leveraged asyncio to run multiple download tasks concurrently without blocking the UI.
- Organized images into a structured local directory and displayed progress and logs in the UI.

## Endless Runner

*HTML5 Canvas, JavaScript, LocalStorage*

- Built an endless runner game where a ghost character flips gravity to dodge pillars, inspired by Flappy Bird.
- Implemented physics with gravity inversion, collision detection, and difficulty scaling using vanilla JavaScript.
- Designed game loop with `requestAnimationFrame` for smooth rendering on HTML5 Canvas.
- Added HUD with score, high score persistence via `localStorage`, and interactive pause/restart controls.
- Deployed on GitHub Pages for browser play: Endless Runner.

## SpeedCube

*React.js, Three.js, Tailwind CSS*

- Built an interactive 3D Rubik's Cube simulator with real-time rendering in a React web app.
- Implemented a precision timer, customizable keybinds, and state control (shuffle, reset).
- Designed a modern, responsive UI with Tailwind CSS for smooth cross-device use.

## YoutubeShorts\_ClipMixer

*OpenAI API, AssemblyAI, Tkinter, MoviePy*

- Built a GUI-based tool to automate YouTube Shorts creation from user-provided or AI-generated scripts.
- Integrated APIs for narration, transcription, and captions to streamline video production.
- Assembled and edited clips with MoviePy and FFmpeg, adding background music and rendering final output.
- Provided a Tkinter interface to manage script input, API keys, and video preview/editing.

## Skills

---

- **Languages:** JavaScript, TypeScript, Python, Java, Swift, SQL, HTML/CSS, C++
- **Frameworks/Libraries:** React.js, Node.js, SwiftUI, Tailwind CSS, Three.js, MoviePy, LangChain, FastAPI
- **Tools/Technologies:** Docker, Git, Postman, Redis, Celery, Uvicorn, FFmpeg, REST APIs, OpenAI API, AssemblyAI, Selenium/WebDriver, Nginx