

PHASUWUT CHUNNAPIYA

Full Stack Developer & AI Developer

Phone: 0638102102

Email: phasuwut.job@gmail.com

Line: [phasuwut](#)

Address: Pak Kret Sub-district, Pak Kret District, Nonthaburi 11120.

LinkedIn: [LinkedIn.com/phasuwut-chunnapiya](https://www.linkedin.com/phasuwut-chunnapiya)

Website : www.phasuwut.com

SUMMARY

Full Stack Developer with over 7 years of experience building scalable web applications, APIs, and AI-powered systems. Experienced in developing end-to-end solutions using modern frameworks such as React, Next.js, NestJS, and FastAPI. Strong background in Artificial Intelligence, Computer Vision, and Large Language Models, with hands-on experience in object detection, multimodal retrieval, and RAG-based systems. Passionate about designing intelligent systems that combine modern AI with robust backend architectures.

EDUCATION

Master of Science Program in Computer Science.

- King Mongkut's University of Technology North Bangkok | 2021-2024 | GPA 3.85

Bachelor of Science Program in Computer Science.

- King Mongkut's University of Technology North Bangkok | 2016-2019 GPA 2.71

EXPERIENCE

TechFlow Solution, Co., Ltd. April 2025 - Present

(Full stack developer)

• Commerce Marketplace Platform

- Developed and scaled a full-featured e-commerce marketplace platform enabling seamless interactions between buyers and sellers, leveraging Next.js, NestJS, and TypeORM within a modular and scalable architecture. Led both frontend and backend development, ensuring high performance, maintainability, and clean system design.
- Engineered reusable and responsive UI components using Ant Design, and Tailwind CSS, delivering a consistent, accessible, and user-centric experience across devices.
- Designed and implemented core marketplace systems, including secure authentication (registration/login), user profile management, seller onboarding, and comprehensive store management dashboards. Architected a subdomain-based multi-tenant storefront system (e.g., store.domain.com) using dynamic routing and Cloudflare DNS, enabling scalable and isolated storefronts for each merchant.
- Built end-to-end onboarding and compliance workflows, integrating with DBD company data to auto-populate official business information. Developed a robust multi-step KYC verification system with document handling, status tracking, and admin moderation (approve/reject), significantly improving merchant verification efficiency.
- Implemented location-aware features using Google Maps API, including address autocomplete, interactive map visualization, and pin-based selection. Designed backend services to manage geolocation data (latitude/longitude) for accurate merchant location handling.
- Developed advanced product and discovery systems, including product lifecycle management, category structuring, search functionality, and search history tracking, enhancing product discoverability and overall marketplace usability.

Full Stack Developer specializing in developing scalable web platforms and APIs using React.js, Next.js, Node.js ecosystem, and modern web technologies. Experienced in delivering digital solutions across real estate, automotive, banking, insurance, and marketing industries, including property management platforms, automotive promotion systems, campaign websites, and enterprise corporate platforms. Strong background in API-driven architecture, microservices, analytics integration (GA4, UTM), chatbot systems, and lead generation platforms that support business growth and customer engagement.

- **Real estate client**

- Developed a responsive real estate website and REST APIs using Next.js, Node.js, and LoopBack to manage property listings and related content.
- Delivered the project over a one-year development cycle, continuously improving features based on client feedback.
- Implemented key pages including About Us, Career, Case Report, Search, Procurement, Inspiring Stories, and Land Listings, with dynamic data integration via APIs.
- Integrated Google Maps with a custom map-drawing feature that allows users to define multi-sided property boundaries for land listings.
- Implemented multi-language support (Thai/English) to improve accessibility for a wider audience.
- Integrated Google Analytics 3 and Google Analytics 4, along with UTM tracking, to monitor traffic and analyze marketing performance.
- Developed cron jobs for background processing to offload heavy tasks and improve overall system performance.
- Implemented user consent management tools to ensure compliance with privacy policies and data regulations.
- Developed automated email notifications for customers and internal teams when users submit lead forms.
- Integrated Google reCAPTCHA on lead and promotion forms to prevent spam and bot submissions.

- **Automotive industry client (China)**

- Developed a responsive automotive website using Next.js, Laravel, Bootstrap, and Tailwind CSS.
- Delivered websites for five car models (two in 2023 and three in 2024) to support product promotion and customer engagement.
- Integrated backend APIs to enable test drive registration and appointment management for customers.
- Connected with the existing showroom database to synchronize and manage test drive appointment data.
- Built dynamic vehicle model and specification pages that retrieve and display detailed information via APIs.
- Developed a chatbot system using Dialogflow, LINE Messaging API, and Facebook Messenger API to assist users with inquiries about car models, promotions, and test drive bookings.
- Implemented LINE Rich Menu to provide quick access to FAQs and key actions, improving user interaction and engagement.

- **Automotive industry client (Japan)**

- Developed a responsive automotive website using Next.js, Nest.js, Prisma, Tailwind CSS, and Mantine.
- Designed and implemented a microservices architecture to improve system scalability and maintainability.
- Developed promotion and promotion detail pages to display marketing campaigns and product offers.
- Implemented and integrated custom APIs to fetch and display promotion data dynamically.
- Integrated LINE Tag Conversion via Google Tag Manager to support marketing tracking and campaign performance analysis.

- **Insurance Industry client**

- Developed a responsive website for an insurance industry client using Next.js with API integration for dynamic content management.
- Implemented key features including search functionality, bookmarks, article listing, and article detail pages, retrieving content from backend APIs.
- Built essential pages including About Us, custom 404 page, and Cookie Consent System to improve compliance and user experience.
- Designed and implemented a global layout with a reusable Navbar to ensure consistent navigation across the website.

- **Eyewear industries client**

- Developed a responsive website with API integration for an eyewear industry client using Next.js, Sequelize, Tailwind CSS, Lottie, and GSAP.
- Implemented key pages including Search, Awards, Career & Career Detail, Promotions, and Promotion Detail, dynamically retrieving data from APIs.
- Built drop-lead forms to collect user information and store it in the database for marketing and customer engagement.
- Implemented essential pages including Contact Us, Privacy Policy, and custom 404 page.
- Enhanced user experience with interactive animations using Lottie and GSAP.
- Integrated Google Analytics 4 to monitor website traffic and analyze user behavior.

- **Coffee shop campaign client**

- Developed a responsive campaign website and APIs using Next.js and Tailwind CSS.
- Implemented a homepage showcasing product bag designs with a reward condition popup.
- Designed a five-step reward claim form including bag selection, name input with profanity filtering, quiz participation, user details, and Facebook sharing.
- Integrated Google Analytics 4 and Facebook Pixel to track user interactions and measure campaign performance.

- **Gaming campaign client**

- Developed a responsive campaign website and APIs using React.js, Node.js, Express.js, and Tailwind CSS.
- Implemented a homepage displaying campaign participation details, reward information, and FAQ sections.
- Built a multi-step drop-lead form with 12 questions to collect participant information and improve user engagement.
- Integrated a database to store user responses and connected with a third-party service for data validation and redeemable game code generation.

- **Banking client (Bank A)**

- Developed and delivered 4 responsive static websites for a banking client using HTML, CSS, JavaScript, jQuery, Tailwind CSS, GSAP, and Bootstrap, improving user experience and digital service accessibility.
- Credit Card Website: Implemented responsive pages to display credit card details, benefits, and related financial services.
- Second-Hand Property Website: Developed user authentication features (login/register) to allow users to manage property listings and account information.
- Funeral Service Website: Built pages for activity search, obituary listings, and event registration, including multi-step forms, QR code donation verification, and features for condolence messages and online wreath submissions.
- Funeral TV Website: Designed responsive pages to display condolence messages and event schedules, enabling users to easily access funeral event timelines and information.

- **Banking client (Bank B)**
 - Developed responsive static websites for a banking client using HTML, CSS, JavaScript, jQuery, Tailwind CSS, and GSAP.
 - Implemented key pages including Who We Are, Team Member Details, Strategic Partnerships, Partner Profiles, News & Blog, and Splash Page.
 - Built Contact Us pages with lead capture forms and a Recognition section to display company awards and achievements.
 - **Glass Packaging Industry client**
 - Developed a responsive corporate website using HTML, CSS, and JavaScript to present company information and industry updates.
 - Implemented key pages including About, Contact Us, and News to improve content accessibility and communication with customers.
 - Designed a global layout and reusable components (e.g., footer) to maintain consistency across the website.
 - **Special Projects**
 - Developed a LINE OA chatbot to introduce the organization and its products, helping the sales team provide information to customers more efficiently.
 - Built web-based object detection applications using React.js and TensorFlow.js for real-time visual recognition experiments.
 - **Content Management**
 - Managed and updated website content for a beverage sales platform using the Drupal CMS.
-

Adesso Technology Co.,LTD | July 2020 - July 2021

(Full stack developer)

- **Video Streaming Platform Client**
 - Developed a responsive video streaming platform using React.js, Node.js, Express.js, GraphQL, and Sequelize.
 - Implemented core features including video browsing, user authentication, and subscription management to support content access control.
 - Integrated GraphQL APIs for efficient data querying and delivery between frontend and backend services.
 - Designed and implemented a responsive user interface using Bootstrap, ensuring compatibility across desktop and mobile devices.
 - **E-Commerce Book Platform Client**
 - Developed a responsive e-commerce web application for online book sales using Next.js, Node.js, Express.js, Sequelize, and Material-UI.
 - Implemented key features including product listing pages, product detail views, and user authentication system.
 - Built a responsive UI with Material-UI to provide a consistent and user-friendly shopping experience across devices
-

The Siam Cement (Ta Luang) Co., Ltd. | May 2019 – August 2019 (Internship Full Stack Developer)

- **Warehouse Management System (WMS)**
 - Developed a Warehouse Management System (WMS) to support inventory tracking and warehouse operations using HTML, CSS, JavaScript, jQuery, and Bootstrap.
 - Implemented features for inventory management, data handling, and user interaction to streamline warehouse workflows.
 - Designed a responsive user interface to improve system usability and operational efficiency for warehouse staff.
-

TECHNICAL SKILLS

- **Artificial Intelligence & Machine Learning & Computer Vision**
 - Large Language Models (LLMs), RAG Systems, Vector Search (FAISS, Qdrant), Vision-Language Models (CLIP, BLIP), Multimodal AI, Semantic Search & Retrieval, Speech-to-Text, Summarization & Conversational Q&A Systems, Object Detection, Multi-Object Tracking, Instance Segmentation, Object Detection, Multi-Object Tracking, Instance Segmentation, Video & Image Analysis.
 - **Programming Language**
 - TypeScript, JavaScript, Python, HTML, CSS, SCSS, PHP, Solidity, Java, XML, C, C++, and jQuery.
 - **Frontend Development**
 - React.js, Next.js, Vite, Laravel, Tailwind CSS, Ant Design, Mantine, Material-UI (MUI), Bootstrap, React Native, and Expo.
 - **Backend Development**
 - Nest.js, Node.js, Express.js, FastAPI, Laravel, and LoopBack 4.
 - **DevOps & Cloud**
 - AWS(EC2), DigitalOcean(Droplet), AWS(s3), Cloudflare, Nginx, Jenkins, Docker, Heroku, and PM2.
 - **Vector Database**
 - FAISS, Qdrant
 - **Relational Database**
 - MySQL, and PostgreSQL.
 - **Unit Test**
 - Jest, and React Testing Library.
 - **Messaging & APIs**
 - Kafka, Dialogflow, LINE Messaging API, Telegram Bot API, Facebook Messenger API and Line Messaging API.
 - **Version Control**
 - Git, GitHub, GitLab, and BitBucket.
 - **Analytics Tools**
 - Google Analytics, and Google Tag Manager.
 - **Development Tools**
 - Postman, Grafana, Jmeter, Figma, Adobe XD, Trello, and MATLAB.
 - **Content Management Systems**
 - Drupal.
 - **Network**
 - MikroTik, and RouterOS.
 - **IoT**
 - Arduino.
-

PROJECT

• Master's Degree Projects

- **Electric Vehicle Charger Socket Detection:** Developed computer vision models to detect and classify multiple types of electric vehicle charger sockets using Python, PyTorch, YOLOv5–YOLOv8, OpenCV, Albumentations, and NumPy, enabling automated identification of charging connector types from images.
 - **EV Charger Socket Detection Web Application:** Designed and implemented a web-based system for detecting EV charger socket types, integrating Next.js and Tailwind CSS for the frontend with YOLOv6/YOLOv8 and TensorFlow models for real-time image-based detection.
 - **Indoor Furniture Detection:** Developed object detection models to classify various indoor furniture types such as chairs, sofas, and tables using Python, YOLO, and OpenCV, enabling automated visual recognition of furniture objects.
 - **Flower, Pet, and Emotion Detection:** Built image classification and object detection models using CNN architectures including GoogleLeNet and YOLOv2 with MATLAB, capable of recognizing flowers, pets, and facial emotions from image data.
 - **Movie Review Sentiment Analysis:** Developed a natural language processing model to analyze and classify movie review sentiment using Python, Scikit-learn, Pandas, and NumPy, enabling automated sentiment prediction from textual data.
 - **Blockchain-based Lottery Trading Platform:** Designed and developed a decentralized web application and backend APIs for lottery trading using React.js, Redux, Web3, Solidity, and Heroku, enabling smart contract-based lottery transactions.
 - **Heart Disease Prediction:** Developed a machine learning model to predict heart disease risk using Python, Scikit-learn, Pandas, and NumPy, applying data preprocessing, feature selection, and classification techniques.
 - **People Detection System:** Built a real-time people detection system using Arduino and sensor modules, capable of detecting human presence and triggering automated responses based on sensor input.
-

• Bachelor's Degree Projects

- **Warehouse Management System (WMS):** Developed a warehouse management system for SCG Saraburi using HTML, CSS, JavaScript, jQuery, and Bootstrap to manage inventory and warehouse operations.
 - **Face Mask Detection:** Developed a computer vision model to detect and classify face mask usage from images using Python, YOLOv3, and OpenCV, enabling automated recognition of masked and unmasked faces for real-time safety monitoring.
 - **Numerical Methods Web Applications:** Developed web-based applications for solving numerical methods problems using JavaScript and Ajax.
 - **Barcode Scanner Application:** Developed an Android barcode scanning application using Java and Android Studio.
-

• Personal Projects (2026)

- **AI Document Question Answering System with RAG and LLM:** Developed a document question answering system that enables users to upload PDF files and query their content using natural language. Implemented document processing with PyPDFLoader and text chunking using RecursiveCharacterTextSplitter, then generated semantic embeddings with HuggingFaceEmbeddings (BAAI/bge-m3) and stored them in a FAISS vector database for efficient similarity retrieval. Relevant document chunks are retrieved and provided as context to selectable Large Language Models (GPT-4o-mini, GPT-4.1, GPT-5) to generate accurate answers based on the document content. Built the backend using FastAPI for document upload, indexing, and question-answering APIs, and developed a React-based interface that allows users to submit questions, select LLM models, control response length, and view answer sources linked to the original document.

- **AI Resume Chat with LLM and RAG (Portfolio Assistant):** Developed an AI-powered assistant that answers recruiter questions about my resume and portfolio using Next.js, TypeScript, OpenAI API, LLMs, and vector search. The system crawls content from all portfolio pages through the sitemap, extracts and cleans the text, and splits it into chunks before generating embeddings using OpenAI text-embedding-3-small, which are stored in a vector database for fast semantic retrieval. When a user submits a question, the query is converted into an embedding and matched with the most relevant content using top-k retrieval, after which a selectable LLM (GPT-4o-mini, GPT-4.1, GPT-5) generates a summarized answer grounded only in the retrieved context. The response also includes source references and relevance information, allowing users to trace the answer back to the original portfolio content for better transparency and reliability.
- **AI Video Search and Scene Analysis System:** Developed a video analysis application that allows users to upload videos and automatically analyze their content. The system extracts frames from videos, detects objects using YOLOv8, and generates visual embeddings with CLIP to represent each scene. These embeddings are stored in a FAISS vector index, enabling fast search and retrieval of relevant scenes based on objects or visual similarity. Built backend services with FastAPI for video upload, timeline generation, frame preview, and scene navigation, and integrated LLMs (GPT-5, GPT-4o-mini, GPT-4.1) to automatically summarize video timelines. A Next.js interface was developed to let users easily search scenes, preview frames, and jump directly to important timestamps in the video.
- **Real-Time Smart Home Surveillance System for Parcel Delivery and Fence-Climbing Intrusion Detection:** Developed a real-time smart home surveillance system using YOLOv9-seg for instance segmentation and ByteTrack for multi-object tracking to monitor parcel deliveries and detect fence-climbing intrusions. The system tracks detected objects across video frames and analyzes their movement relative to predefined boundary lines (e.g., fence areas). When a tracked object crosses the boundary, the system automatically triggers a Telegram alert notification, enabling homeowners to receive immediate real-time warnings and respond quickly to potential security events.
- **Visual Question Answering System with YOLO, CLIP, ViT, BLIP, BLIP Caption and LLM:** Developed an end-to-end Visual Question Answering (VQA) system that allows users to upload an image and ask natural language questions about the scene. The system uses YOLO for object detection, CLIP for image-text similarity reasoning, Vision Transformer (ViT) for image classification, and BLIP for automatic image captioning, and integrates the extracted visual information with selectable Large Language Models (LLMs) to generate context-aware answers. The system supports multiple LLM options including GPT-4o-mini, GPT-4.1, and GPT-5, allowing users to compare responses from different models. Built the backend using FastAPI for model inference and API services, and developed an interactive React-based web interface where users can upload images, select the LLM model, visualize detected objects and bounding boxes, and receive AI-generated explanations about the image content.
- **Multimodal Semantic Search Chat System (FastAPI, Qdrant, CLIP, BLIP, Typhoon OCR):** Developed multimodal semantic search chat system enabling cross-modal retrieval across text and images using a dual-database architecture (PostgreSQL for structured chat data and Qdrant for vector similarity search). Leveraged SentenceTransformers for multilingual text embeddings and CLIP for unified image-text representation, enhancing image understanding through BLIP-based captioning and Typhoon OCR for text extraction with translation. Designed a hybrid search pipeline combining semantic similarity from both text and image modalities with ranking and filtering to improve retrieval relevance and accuracy. Built scalable backend services using FastAPI and integrated with a Next.js frontend to support real-time chat interaction and efficient semantic search.

- **Electric Vehicle Charger Socket Semantic Visual Search System with YOLO, CLIP, and FAISS:** This project extends the previous works, “**Electric Vehicle Charger Socket Recognition and Semantic Retrieval Model using OpenCLIP (ViT-B/32)**” and “**Electric Vehicle Charger Socket Detection**” by combining semantic text understanding and object detection into a two-stage pipeline where, in Stage 1, a natural language query such as “A red car plugged into an AC Type 2 charging port” is encoded using a pretrained CLIP model and matched against image embeddings through FAISS to retrieve broadly relevant images based on general visual features, meaning the system may primarily focus on obvious elements like “a red car” because the pretrained model does not strongly understand specific Electric Vehicle Charger Socket types, and therefore the initial results might emphasize car color rather than the exact charger socket; then in Stage 2, refinement is performed in two parts: first, a custom-trained CLIP model specialized for Electric Vehicle Charger Socket re-ranks the candidate images to prioritize those containing the correct plug type, and second, YOLO object detection is applied to the top-ranked results to explicitly detect and label charger types with bounding boxes and confidence scores, ensuring that the final output not only matches the general scene but also correctly identifies the specific Electric Vehicle Charger Socket type.
- **Electric Vehicle Charger Socket Recognition and Semantic Retrieval Model using OpenCLIP (ViT-B/32):** Built a model that can recognize different types of Electric Vehicle charger sockets and support similarity search using OpenCLIP (ViT-B/32). The model converts both images and text descriptions into numerical vectors (embeddings) that represent their visual meaning. By placing images and text in the same vector space, the system can classify five socket types (AC Type 1, AC Type 2, DC CCS Type 1, DC CCS Type 2, CHAdeMO) and also match user text queries with relevant images.
- **Multilingual Video Understanding and Event Summarization System (Thai-English Timeline Intelligence):** Built an end-to-end AI video analysis system that accepts a video file as input and performs temporal segmentation based on a configurable Seconds Per Clip parameter (e.g., every 4 seconds). The system samples frames at fixed intervals using OpenCV, generates segment-level descriptions in English using BLIP-2, summarizes the overall video using FLAN-T5, and translates the outputs into Thai using NLLB-200, producing bilingual timeline descriptions and summaries (description_en, description_th, summary_en, summary_th). The system exposes its functionality through FastAPI, which handles video upload and returns structured JSON results. The frontend is built with React for user interaction and timeline visualization, while Tailwind CSS is used for UI styling and layout design.
- **Multimodal Semantic Retrieval (Video & Image Search) :** Extended and integrated previous projects (**Text-to-Video Semantic Search** and **Text-to-Image Semantic Search**) into a unified multimodal semantic retrieval platform capable of searching across both videos and images using natural language queries. Leveraged CLIP (ViT-B/32) developed by OpenAI to generate shared embeddings for text, video keyframes, and images within the same vector space, enabling cross-modal semantic similarity search with FAISS and accurate timestamp alignment for video playback. Integrated language translation preprocessing (Thai → English) to improve embedding alignment and retrieval accuracy, as CLIP performs more effectively with English text inputs. Developed RESTful APIs using FastAPI that return structured JSON responses containing media_id, timestamp (for videos), similarity score, and media URL, and built a responsive frontend using React.js and Tailwind CSS for real-time result visualization.
- **Parcel and Person Multi-Object Tracking:** This project extends the previous works, “**Parcel and Person Instant Segmentation**” results to build a multi-object tracking system using BoT-SORT and ByteTrack, assigning a unique ID to each detected parcel and person to enable consistent tracking across video frames.
- **Parcel and Person Instant Segmentation:** Builds an AI model to detect and segment parcels and people in images or videos using YOLOv8-seg, YOLOv9-seg, YOLOv11-seg, YOLOv26-seg, and RF-DETR-seg. It builds a unified dataset by collecting free public data, merging multiple sources, and manually correcting labels for the two classes (parcel and person). All models were trained on RunPod cloud GPUs using the same dataset and hyperparameters to ensure a fair performance comparison.

- **AI Meeting Transcription, Summarization & Q&A System (RAG + LLM):** Developed an end-to-end meeting intelligence system that processes uploaded audio/video by splitting recordings, transcribing speech using Whisper, and generating structured Thai summaries and action items with GPT-4o-mini, while storing transcript embeddings using OpenAI text-embedding-3-small in Qdrant for semantic retrieval; the system enables context-aware Q&A by retrieving relevant transcript segments, returns source-backed answers with traceable references, tracks processing status in real time, and provides a React-based interface for reviewing transcripts, summaries, audio segments, and downloadable reports.
- **Multilingual Semantic Video Event and Action Search Engine and API:** Built a FastAPI-based video semantic search system that enables natural language queries over video content by segmenting videos into fixed-duration clips, extracting representative frames with OpenCV, generating semantic embeddings using CLIP (ViT-B/32) in PyTorch, and indexing them in FAISS for efficient cosine similarity search; the system processes queries such as "dog running in park" or "car accident at intersection," performs multilingual translation when necessary to improve embedding alignment, matches results against stored clip embeddings using similarity scoring, groups matched segments by video, and returns structured JSON responses with video filenames, URLs, and precise timestamp intervals (start_time, end_time) for direct navigation to relevant moments without watching the entire video.
- **Image-to-Image and Video Similarity Search System:** Built an end-to-end visual retrieval system that accepts a query image and performs cross-media similarity search across both images and videos. The system encodes the uploaded image into a semantic embedding using CLIP (ViT-B/32) in PyTorch and compares it against pre-indexed image files and extracted video keyframes stored in a FAISS inner-product index. For videos, the system aggregates matched frame timestamps, merges temporally adjacent segments into consolidated intervals (start_time, end_time), and ranks results based on similarity scores. Integrated BLIP to automatically generate descriptive captions for the query image to improve interpretability and contextual understanding. The backend is implemented with FastAPI, providing endpoints for index construction and search execution, returning structured JSON responses containing media paths, similarity scores, and timestamp ranges. The frontend is developed using React and Axios, enabling configurable search parameters (top_k, similarity threshold, caption length) and interactive video playback with direct navigation to detected relevant segments.
- **Text-to-Video Semantic Search:** Built an end-to-end text-to-video semantic search system that allows users to locate relevant moments in videos using natural language queries. The system samples video frames at fixed intervals with OpenCV and converts them into semantic embeddings using CLIP (ViT-B/32), storing the vectors in a FAISS index with metadata linking each embedding to its original video path and timestamp. The FastAPI backend processes JSON-based search requests, translates multilingual queries into English for improved embedding alignment, and converts the query into the same semantic space as the indexed frames to enable meaning-based retrieval rather than keyword matching. FAISS performs high-speed similarity search to return ranked results with similarity scores, video URLs, and precise timestamps. A responsive frontend built with React.js and Tailwind CSS visualizes results in real time, allowing users to instantly navigate to specific scenes without watching entire videos.
- **Text-to-Image Semantic Search Module:** Built an end-to-end text-to-image semantic search system that enables users to search for images using natural language instead of keyword matching by leveraging CLIP (ViT-B/32) to encode both text and images into a shared embedding space and indexing precomputed image embeddings with FAISS for efficient similarity search. Developed a FastAPI backend to process queries and return matched image URLs with similarity scores, along with a React.js frontend for real-time search and dynamic result visualization.
- **Thai Mango Variety Detection:** Built an object detection model to detect and classify five Thai mango varieties (ChokAnan, KhiaoMoragot, MunDuanKao, NamDokMaiSeeThon, PhetBanLat) using YOLOv9, YOLOv10, and YOLOv26 by collecting and labeling a custom image dataset, preparing the data in YOLO format, and training the models for multi-class detection.

• **Personal Projects (2025)**

- **Electric Vehicle Charger Socket Instant Segmentation & Tracking:** Extended the **Electric Vehicle Charger socket instance segmentation** work by adding multi-object tracking to keep consistent socket IDs across video frames using ByteTrack. Integrated RF-DETR-Seg/YOLOv12-Seg with ByteTrack and generated tracked socket masks with consistent IDs across frames.
- **Electric Vehicle Charger Socket Instant Segmentation:** Built and trained instance segmentation models (YOLOv8-Seg, YOLOv12-Seg, RF-DETR-Seg, Mask R-CNN) to identify and segment multiple EV charger socket types in real time. Collected and annotated data, performed preprocessing (resize/normalize, label checks) and augmentation (flip/rotate/brightness), then trained and evaluated each model to compare accuracy and inference speed and select a practical model for use.
- **Electric Vehicle Charger Socket Detection & Tracking:** Built a computer vision system that detects EV charger sockets from video footage and tracks each socket over time with a consistent ID, ensuring the same object is not confused, duplicated, or switched with others, using RF-DETR/YOLOv8 for detection and ByteTrack for object tracking, along with data preparation and model training for practical deployment.
- **Electric Vehicle Charger Socket Detection (React Native + Expo, ONNX):** Built a React Native (Expo) mobile app to detect and classify EV charger socket types in real time from the camera and gallery images. Trained a YOLOv8 detection model and deployed it via ONNX for on-device inference, implementing image preprocessing and an in-app overlay for bounding boxes, labels, and confidence scores.
- **Electric Vehicle Charger Socket Detection API:** Built a high-performance REST API using NestJS and ONNX Runtime for real-time image-based EV charger socket detection and classification. Supported multiple connector types (AC_Type_1, AC_Type_2, DC_CCS_Type_1, DC_CCS_Type_2, CHAdeMO) and returned structured JSON outputs including bounding box coordinates (x, y, width, height) and confidence scores for seamless system integration.
- **Object Detection and Segmentation Auto-Labeling (GroundingDINO, SAM, YOLOv8, YOLOv8-seg) :** Developed code for dataset labeling using two approaches: automatic and manual annotation. Automatic labeling used GroundingDINO for text-guided object detection and Segment Anything (SAM) to generate segmentation masks, which were converted into YOLO-format annotations for training YOLOv8 (object detection) and YOLOv8-seg (instance segmentation) models. For manual labeling, an initial set of images was annotated and used to train a YOLO model; the trained model then generated labels for additional images, which were reviewed, corrected, and refined to improve dataset quality before retraining.
- **Deployment Automation: Jenkins + Docker + Cloudflare DNS Sync + Alerts:** Designed and maintained a CI/CD deployment system using Jenkins and Docker to automate build, test, and deployment workflows. Managed domain and DNS configuration on Cloudflare, and developed an automation to detect server IP address changes and automatically update Cloudflare DNS records to keep the website online. Integrated notification alerts to Discord and Telegram for both deployment status (success/fail) and server IP/DNS update events.
- **AI Chatbot for Document Queries on LINE:** Developed a LINE-integrated AI chatbot that answers user questions based on uploaded or connected documents. Utilized n8n for workflow orchestration and ChatGPT for natural language understanding and response generation, enabling users to interact with document knowledge through conversational queries on the LINE platform.
- **Document Q&A System using LangChain, LLMs, and Python:** Developed a document question-answering system using Python and LangChain to ingest and process content from websites and PDF files, with semantic retrieval powered by embeddings and FAISS. Integrated both local LLMs via Ollama (LLaMA 3.2) and cloud-based models (GPT-4o) to generate accurate, context-aware responses based on retrieved document content.

- **MCP (Model Context Protocol) Tools for API Integration:** Designed and implemented a modular tool using Python, FastMCP, and httpx to fetch and process structured data from personal REST API endpoints, including academic papers, projects, and work portfolios. The tool operates via standard input/output (stdio) transport to ensure seamless integration with automated systems and other MCP-compatible modules. Integrated with Claude (Anthropic) as an MCP client, allowing natural language interaction with the tool via Model Context Protocol.
-

- **Personal Projects (2024)**

- **Home Network Notification Alerts with Discord Bots and Telegram Bot:** Developed Discord chatbot to send real-time home network alerts using Discord Bots, Telegram Bot and RouterOS. The chatbot integrates with home network infrastructure to monitor and inform users of important events such as public IP updates, home VPN access.
 - **Home Network Notification Alerts with LINE Chatbot:** Developed LINE chatbot to send real-time home network alerts using Line Notify and RouterOS. The chatbot integrates with home network infrastructure to monitor and inform users of important events such as public IP updates, home VPN access.
-

PUBLICATIONS

- **International Journals**

- **Enhancing Automatic Electric Vehicle Charging: A Deep Learning Approach with YOLO and Feature Extraction Techniques**, (Scopus Journal Q1, Frontiers) Doi: [10.3389/fcomp.2025.1505446](https://doi.org/10.3389/fcomp.2025.1505446).

- **International Conferences**

- **Efficient Video Object Detection of Indoor Furniture and Home Appliances**, 2024 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), DOI: [10.1109/BlackSeaCom61746.2024.10646241](https://doi.org/10.1109/BlackSeaCom61746.2024.10646241).
 - **Efficient Video Object Detection for EV Charging Plug Type Identification in Real-World Scenarios**, 2024 19th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), DOI: [10.1109/iSAI-NLP64410.2024.10799466](https://doi.org/10.1109/iSAI-NLP64410.2024.10799466).
-

CERTIFICATION

- **Certification (2024)**

- In the 19th International Joint Symposium on Artificial Intelligence and Natural Language Processing, 2024.

- **Certification (2021)**

- Version Control with Git, coursera.org, June 16, 2021.
- Server-side Development with NodeJS, Express and MongoDB, coursera.org, June 16, 2021