Cham Bandara

Phone: 0452122690 | Email: cham.bandara@outlook.com | Location: Bellbird Park, QLD 4300
LinkedIn: linkedin.com/in/cham-bandara | Portfolio: chamb.vercel.app | GitHub: github.com/nchamb

Education

Master of Information Technology (Software Development Major, Data Science Minor)
University of Southern Queensland | Jan 2024 – Dec 2025

Bachelor of Computing (Software Engineering)
Curtin University | 2020 – 2023

Technical Skills

Programming Languages: Python, Java, JavaScript, C++, C, C#, .NET
Frameworks & Libraries: ReactJS, NodeJS, HTML5, CSS, XML, SOAP, NextJS
Data Management & Analysis: SQL, MongoDB, GraphQL
Data Science Tools: Pandas, Scikit-learn, R, PyTorch, TensorFlow, RapidMiner
Machine Learning: Supervised and Unsupervised Algorithms
Deep Learning: Convolutional Neural Network, Recurrent Neural Network (LSTM), Transformer, Coordinate Attention Residual
Project Management: Jira, GitHub
CI/CD and Cloud Services: Jenkins, Git, Docker, Kubernetes, Azure, AWS, Google Cloud
System Administration: Windows Server, Domain Controller, DNS, Active Directory, Microsoft IIS
Relevant Coursework: Data Structures and Algorithms, Software Engineering Testing, Cyber Security, Design Patterns, Statistics for Quantitative Researchers, Machine Learning, Data Mining for Business Analytics and Cyber Security, Blockchain Fundamentals, Intelligent Multimedia (Computer Vision & Audio Analysis), Artificial Intelligence for Business

Software Engineering Project Experience

Titans Tickets | titans-tickets.com
Tech Stack: JavaScript, ReactJS, NodeJS, ExpressJS, MongoDB, Docker, Email and Text Services, QRScanner

- Designed and developed a **full-stack ticket selling platform** for a sporting event clients with ~1000 attendees per year, supporting seamless ticket purchases and digital validation of tickets using QR scanner.
- Built a **client-facing portal** with user authentication using mobile number with OTP (OAuth2) to enable ticket selection and real-time email confirmations with dynamic QR codes.
- Engineered a **QR scanning admin platform** for event staff to validate tickets using device cameras; verification synced with the backend in real-time to mark entries as "entered" and prevent re-entry using same ticket.
- Architected and consumed **RESTful and GraphQL APIs** to manage ticket inventory and user data.
- Implemented **caching strategies** with React Query for fast data fetch and low-latency UI rendering.
- Ensured security by integrating **JWT-based authentication** for both users and admins, with role-based access control.
- Deployed using **Docker and Kubernetes**, with CI/CD pipelines on **Vercel for two separate frontends and Render** for backend which is scalable and reliable hosting.
- Monitored system performance and optimized database queries for high-traffic periods during pre-event peaks.

Avanoa Web Application | avanoa-frontend.vercel.app
Tech Stack: JavaScript, ReactJS**,** WebGazer.JS, Sass, NodeJS, ExpressJS, MongoDB, Azure, EmailJS

- Alternative communication platform for motor disability people using their eye movements.
- By using the **WebGazer library** tracking eye movements to interact with the functionalities of the application.
- The application is based on **JavaScript** following the **MERN stack.**
- Project was developed following **the Agile framework** and as **MVP versions** for an actual client as a final-year project.
- Relevant documentations such as **SRS**, **Technical Investigation Summary**, **Software Architecture Specification** and **Project planning** were made.

Research Content Management System | GitHub

Tech Stack: ASP.NET Core 8.0, Entity Framework Core (SQL Server), AutoMapper, JWT, Swagger, xUnit, C#

- Research Content Management System (RCMS) designed for managing academic research papers, tags, comments, users, and metadata.
- Developed secure RESTful APIs with role-based authentication and authorization using JWT.
- Implemented modular CRUD operations for Research, Tags, ResearchTags, Comments, and Users with EF Core as ORM.
- Integrated Swagger for interactive API exploration and automated documentation.
- Applied xUnit with EF Core InMemory database for unit and integration testing to ensure reliability.
- Followed Agile methodology for iterative development and delivery.
- Comprehensive technical documentation including Software Architecture, SRS, and setup guides were produced.

Peer-to-Peer Application

Tech Stack: C#, ASP.NET, MVC

- **Peer-to-Peer desktop application** to post jobs in **Python** and execute jobs of each peer automatically.
  1. **ASP.Net MVC Web server WPF** (.net framework) with **multi-threaded** (remoting server)
  2. **ASP.Net Core Web Application**

Data Science Project Experience

Depth of Anesthesia (DoA) Index Design | GitHub

Tech Stack: Python, scikit-learn, pandas, NumPy, Jupyter Notebook

- Developed a novel Depth of Anesthesia (DoA) index using EEG-derived features, targeting enhanced intraoperative patient monitoring in clinical settings.
- Conducted feature selection, model tuning, and performance evaluation using Pearson correlation, R^2, and Bland-Altman plots to validate model robustness on withheld testing data.
- **Supervised machine learning techniques**: Linear Regression and Random Forest to model BIS-like index from EEG features (x1–x7), ensuring the model output ranged from 0 (deep anesthesia) to 100 (awake).
- **Unsupervised learning algorithms:** K-means clustering and Gaussian Mixture Models on a separate dataset of 4965 EEG segments for data clustering, labeling states A/B to infer depth of anesthesia without supervision.
- Integrated both supervised and unsupervised findings to design a **hybrid ensemble model** combining linear regression and random forest using weighted averaging predictors for enhanced predictive accuracy and resilience.

Chest X-ray Multi-Label Classification | GitHub

Tech Stack: Python, PyTorch, scikit-learn, NumPy, Google Colab (TPU/GPU), Jupyter Notebook, ChestX-ray14

- Developed and trained deep learning models (DenseNet-121 & custom CNN) for multi-label classification of chest X-ray images using 100,000+ samples for comprehensive medical image analysis.
- Implemented a custom 4-block CNN architecture with SE attention mechanisms, batch normalization, and channel recalibration, optimized for extracting radiology-specific features.
- Fine-tuned DenseNet-121 pretrained on ImageNet with a custom classification head; addressed class imbalance using BCEWithLogitsLoss with positional weighting, improving performance across all disease classes.
- Applied advanced preprocessing techniques (CLAHE, Gamma correction, ColorJitter) and extensive data augmentation (RandomCrop, Flip, Rotation) to enhance robustness and generalization.
- Optimized training pipeline with differential learning rates, AdamW optimizer, and ReduceLROnPlateau scheduler; deployed on Colab TPU/GPU with dynamic batching for efficient high-throughput processing.

Environmental Sound Classification | GitHub

Tech Stack: Python, PyTorch, librosa, scikit-learn, NumPy, UrbanSound8K, Jupyter Notebook, Google Colab (TPU/GPU

- Developed and optimized deep learning models (CNN, LSTM-RNN, CAR-Transformer) for ESC using the UrbanSound8K dataset, achieving high accuracy in noisy, real-world environments.
- Extracted and engineered advanced audio features (MFCC-40, log-Mel spectrograms, Δ & Δ^2 features) to enhance temporal and spectral representation.
- Applied attention mechanisms and Transformer-based architectures (CAR-Transformer with Coordinate Attention & Residual Convolutions) to improve robustness, interpretability, and global context modeling.
- Implemented preprocessing and augmentation pipelines (time shifting, pitch shifting, noise addition, time stretching) to address imbalance, increase diversity, and improve model generalization.
- Conducted performance benchmarking across multiple architectures (CNN, CRNN, Transformer, Generative augmentation) on UrbanSound8K cross-validation folds with detailed model inspection.
- Evaluated classification performance using rigorous training, validation, and testing protocols with metrics including accuracy, precision, recall, F1-score, and confusion matrix analysis.

Languages

English, Sinhala

References

1. **Dr. Zhi Chen**

Lecturer in Computer Science, University of Southern Queensland
Research Officer in in Artificial Intelligence, Computer Vision
School of Electrical Engineering & Computer Science, University of Queensland
Email: zhi.chen@unisq.edu.au | Phone: +61416669800 | Portfolio: https://uqzhichen.github.io/