Zhen (Brian) Guan

zguan@mun.ca · (709) 222-4031
linkedin.com/in/brian-guan • github.com/hatsune-miku

Full Stack Developer

Experienced frontend and backend developer with 2 years of professional work history and 8 years of personal development experience. Proficient in a range of cutting-edge backend technologies:

Frontend Frameworks	Backend Frameworks	Languages	Other
React	SpringBoot	Typescript	Apache Kafka
React Native	SpringSecurity	CSS / SCSS	Google Firebase
Vue	Ruby on Rails	Rust	JoyUI & ElementPlus
Flutter	Express	C++ (C++20)	Vite & Webpack
Tauri		Java (JDK 18)	Amazon Web Services
Electron		Kotlin	PostgreSQL
		C#	Redis

Education

Master in Computer Engineering **Sept 2021 – Aug 2023**
Memorial University of Newfoundland

Bachelor of Computer Science **Sept 2017 – Aug 2021**
Capital Normal University

Work Experience

Full Stack Developer **Jan 2021 – Jun 2022**
Beijing Fengrong Trading Co., Ltd

- **Project: Company Portal Website**
  - Developed the company's web portal utilizing **ReactJS and Ruby on Rails.**
  - Optimized package structure, reduced the cacheless cold loading time by 33%.
  - Deployed and configured the project with **nginx**-based **image cache** and SSL certificate.
  - Implemented backend management system to update carousel pictures, knowledge bases and contact information and to view user visit statistics.
  - Designed CSS animations for UI details to improve user experience.
  - Designed a webform for business cooperation with **data verification on backend**.

Software Developer (Intern) **May 2020 – Jan 2021**
Beijing GSafety Technology Co., Ltd

- **Project: Forest Fire Detection**
  - Implemented RTMP video stream decoding system in **C++ (C++17)** to process and store (**PostgreSQL**) data from DJI drone aerial photography.
  - Translated an OpenCV-based image bright-spot detection algorithm from Python to C++.
  - Servitized the decoding system and hosted a **VueJS**-based video upload & analysis app.
  - Conducted real-world inspection of drone operation scenarios, deployed the decoding system on harsh environment and successfully **tested and verified** it.

- **Project: English Text Emotion Analysis**
  - Developed a multi-thread **Twitter (X) crawler** in Python, gathered 20k+ tweets.
  - Wrote bulk **data sanitization** script for model training, increased the accuracy by 20%+.
  - Designed the emotion analysis command line tool based on **Google Albert**.
  - Containerized tools above with **Docker** for mass deployment and optimized in size.

Recent Projects

Aug 2023 – Now

Multiflow

A multiphase flow transient simulator.

Multiflow is a multiphase flow transient simulator written in **Qt and C++.** This project is developed in response to the requirements of Capital Normal University. Highlights:

- Collaborating with algorithm department to build a platform for Qt for visual modeling, oil well lift method design, model configuration and simulation results presenting.
- Focused on detailed UI design, achieving streamlined functionality and practicality, facilitating user comprehension of the current production system model, operational status and fluid property distribution within the wellbore pipeline.

Feb 2023 – Now

ChatGPTRelay

A chatbot based on OpenAI API with builtin usage control.

Built with **React.js, SpringBoot and JoyUI**, ChatGPTRelay aims to make OpenAI's ChatGPT accessible to everyone, while offering an experience very similar to the official one. Highlights:

- Implemented an API key pool to enhance service reliability.
- Leveraged **Redis** to store high-frequency user token quotas data.

Demo: https://cg.eggtartc.com/#/login (Account: 3000, Password: 123)

AirX

A cross-platform text and file sharing system.

AirX is a project finished by a team of 4. It allows users to **seamlessly copy and paste** files and text between macOS (**SwiftUI**), Windows (**WinUI3**), Android (**React Native**), and Linux (**Rust**) devices over **LAN or Internet**, blurring the boundaries between different platforms. AirX also provides a standalone **cloud storage service** with support for sharing links. Highlights:

- Utilized Google **Protocol Buffers** and self-designed UDP-based protocol for LAN discovery.
- Employed **Apache Kafka** in the backend (**SpringBoot**) for clipboard synchronization.
- Innovatively **combined JWT and AES** with randomized keys to further protect token data.

Demo Video: https://hatsune-miku.github.io/airx.html
Repo: https://github.com/hatsune-miku/libairx

Jan 2023 – May 2023

Memorial Self-service II

A reimplementation for university self-service system.

Memorial Self-service 2 (MSS2) is a revamped version of the university's self-service system, implemented with modern front- (**Vue, ElementPlus**) and back-end (**SpringBoot**) frameworks. Highlights:

- Applied **OAuth2**-based authentication and authorization system with **SpringSecurity** to seamlessly dock with university's existing unified identity verification portal.

Demo: https://jiabao.world/
Frontend Repo: https://github.com/hatsune-miku/MSS-Frontend
Backend Repo: https://github.com/hatsune-miku/MSS