Zhen (Brian) Guan

zguan@mun.ca · (709) 222-4031
eggtartc.com·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 16 years of personal development experience. Proficient in a range of cutting-edge backend technologies:

Frontend Frameworks
React.js
React Native
Vue.js
Flutter
Tauri
Electron
SwiftUI / UIKit

Backend Frameworks
SpringBoot
SpringSecurity
Ruby on Rails
Express
Django

Languages
Java (JDK 18) / Kotlin
Rust
C++ (C++20)
C#
TypeScript
Swift

Other
Apache Kafka
Google Firebase
Vite
Amazon Web Services
Postgres
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 (Part time) **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 **Vue.js**-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.

Notable Recent Projects

***Visit https://eggtartc.com for a clearer view.**

Feb 2024 – Now

NG2 Parental Control

A parental control system with remote monitor & control support.

NG2 is a parental control system developed with **Tauri (Rust + React.js)** and its backend is fully implemented with serverless technologies: **Firebase Cloud Functions + Redis**. NG2 provides a wide range of monitoring functions with parents to limit screen time for their children:

- Maximum minutes per day / per session
- Cooldown time every X minutes
- **Remote** screen lock / unlock, remote message
- Remote shell commands
- Disallowed / unconditionally-allowed timespan for weekends / weekdays / holidays
- Device usage report
- Multiple device management

Time policies are stored in server and changes take effect in real time.

Feb 2023 – Now

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 frontend (**Vue.js, ElementPlus**) and backend (**SpringBoot**) frameworks. Highlights:

- Applied **OAuth2**-based authentication and authorization system with **SpringSecurity** to seamlessly dock with university's existing unified identity verification portal.
- Allowing **complex filters** for each entity while maintaining **highly extensible** code architecture.
- Massive custom CSS for an unique and consistent look and feel.

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

Notable Recent Projects (Continued)

Visit https://eggtartc.com for a clearer view.

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)

Portal 2 Cheat

A tweak for the game "Portal 2" using remote code injection.

Written with **C++ and Win32 APIs**, this cheat tool provides Portal 2 players with some in-game privileges:

- Place portals anywhere
- Place portals on moving platforms
- God Mode switch
- Noclip switch
- Dissolve object
- Spawn cube
- Spawn companion cube
- Reload last checkpoint
- Open target door
- Run any developer commands without enabling the console

Additionally, this tool is based on a huge amount of preliminary work:

- Learning of Source Engine and its command-based system
- **Memory address locating** with Cheat Engine
- Assembly code to **simulate thiscall**

Repo: https://github.com/hatsune-miku/portal2_trainer