



Seven Wang

Objective:

Software engineer


Career planning:

System Architect/CTO

 October 13, 1994 >

 Chengdu, Sichuan >

 Tianjin University of
Technology >

 Software engineering >

PERSONAL RESUME

Professional Skills

- C# (.net/.net core) , Java, Python 7+ years experience
- Relational database skills like MySQL/SQL Server and MongoDB, Elasticsearch non-relational database skills
- HTML, CSS, Javascript, React technology skills, Proficient in Restful API design
- Experience in asp.net core mvc and other server-side programming frameworks
- Familiar with AOP, IOC, ORM underlying framework design principles
- Experience in Kafka, Redis and other middleware
- Distributed microservice architecture Experience
- Familiar with SaaS, PaaS, FaaS, Experience in developing and designing low-code platforms
- Experience in Git, Docker, Jenkins, Sonarqube, ELK, Automation Test, Cypress, NewRelic, Azure
- Passionate about research and sharing
- High standards and strict requirements for code cleanliness and readability

Work Experience

2023.01 - 2025.02 Seismic

Position: Senior Software Development Engineer

Responsible for the development and maintenance of the core business of the Page/HomePage team

2017.11 - 2023.01 Beisen

Position: Development/leader assistant

Responsible for the development and maintenance of the team's basic components and core business

PROJECT INTRODUCTION

Page Editor

Project Description

Content management system, and the Page maintained by the Page team is one of the content. By providing a series of content-related widgets and a powerful page layout editor, users can freely drag and drop different widgets to complete a Page.

Main responsibilities: System architecture design, business development of core functions, architecture explanation, and technical sharing.

These technologies are also used in projects

React, .NET 8.0, WebApi Restful APIs, microservices architecture, Jenkins, Docker, Azure cloud platform, K8s, Fabric, Ansible, AWS Secret Manager, SQL Server, Dapper, Redis, Local Cache, Kafka, NewRelic, Snowflake DBT, LaunchDarkly, OpenAI, PyTorch, Transformer Model, Faiss.

Learning Student Progress Calculation (Core Business)

Project Description

This project is the core business module that I am responsible for optimizing in the Beisen learning team. Due to the large number of resources involved in employee learning scenarios and a large number of queries, the basic service Cassandra server is under high pressure, and the computing business is complex and difficult to maintain. Historical logic stacking leads to customer learning computing Error rate is high. In this context, a set of progress calculation modules based on Redis/Kafka/high expansion resource calculation model are redesigned.

After redesign and optimization, 800 concurrent, 10 minutes execution, 90% response time is 1.436s, 897 QPS. The service memory usage is reduced by 60%, the interface response speed is increased by 100%, and the throughput is doubled. (apr. 22, 2022 goes online)

PROJECT INTRODUCTION

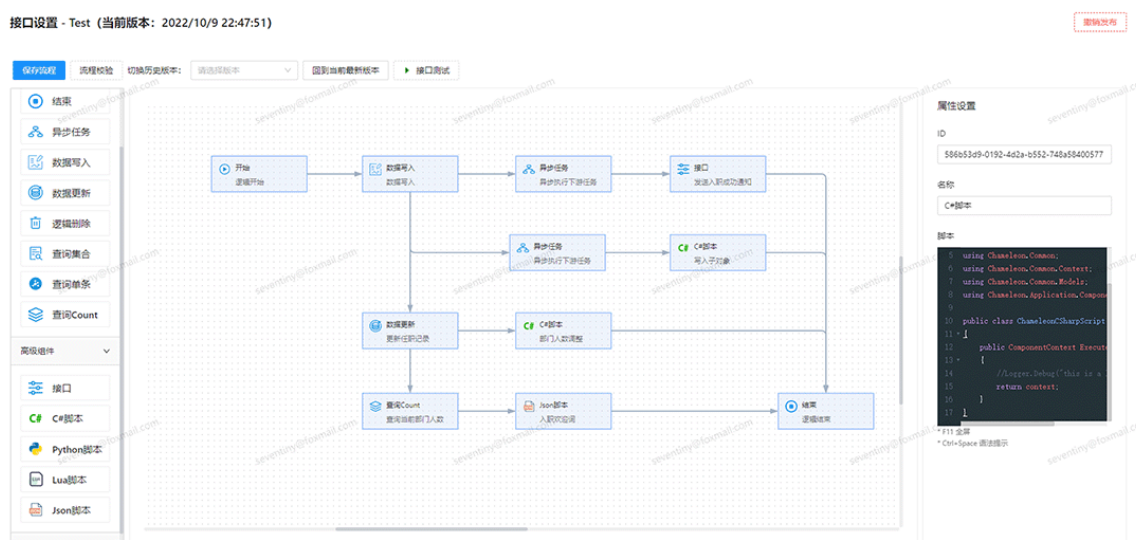
Chameleon Platform as a Service

Project Description

Chameleon is a flexible and easy-to-use PaaS (No/Low Code Development) platform. With its powerful modular features, it makes building enterprise-level applications easy. By easily combining different business modules on the Chameleon platform, you can quickly build enterprise-level applications with complete functions, greatly improving development efficiency and lowering technical thresholds.

Quickly configure forms, lists, functions, workflows, pages and other modules to easily meet business needs. Built-in user/organization/identity/role/page permission control/message backlog modules, available out of the box. A process-based configuration method provides rich business components and expands complex business through flow charts. Multi-tenant, database level isolation, safe and reliable. Support expansion in the form of microservices or microapplications. Comprehensive project documentation and development manuals.

The technologies adopted include: .NET 8.0, WebAPI Restful, React/Type-script/Antd, Postgresql, Redis, MongoDB, FreeSql, Jenkins, Docker, Roslyn Dynamic Compilation.



PROJECT INTRODUCTION

Bamboo.Spring lightweight AOP/IOC/DataPipeline framework

Project Description

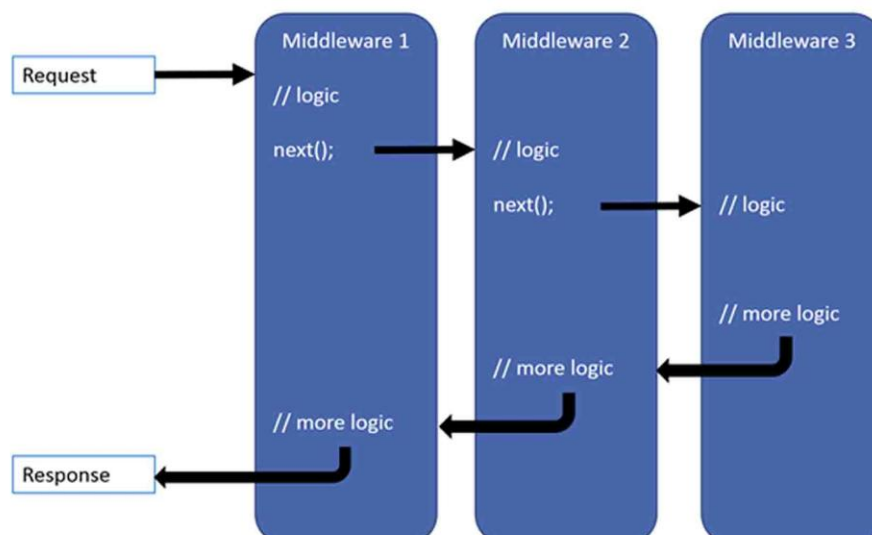
project blog : <https://www.cnblogs.com/7tiny/p/9657451.html>

open source address: <https://github.com/sevenTiny/Bamboo>

This project is a lightweight AOP/IOC framework developed by myself. The AOP part uses Emit to generate an intermediate language to realize dynamic proxy, and supports interface/implementation mode and abstraction/inheritance mode. On this basis, the ability of AOP is used to expand the data pipeline, and support various projects to connect with highly scalable pipeline middleware.

Technical Characteristics

1. AOP dynamic proxy (Emit implementation)
2. Support for injecting method aspects in the form of annotations
3. Built-in IOC container, use built-in pipeline to control the life cycle, support construction injection and property injection
4. The pipeline model is built internally to support runtime injection of middleware like netcore, and rich middleware can be provided to the pipeline model in a flexible pluggable manner.



PROJECT INTRODUCTION

Bamboo.Configuration Distributed Remote Configuration Framework

Project Description

open source address: <https://github.com/sevenTiny/Bamboo>

This project is a distributed remote configuration component developed by myself, which solves the problem of synchronous remote configuration of multiple service nodes in the microservice scenario. The framework supports registering different data sources as remote configuration centers (mysql, git, interfaces, json files, xml files, etc.). Support remote configuration hot update, local mode and other features.

Technical Characteristics

1. Support database, git service, json, xml file as remote configuration center data source
2. The abstract access layer supports the expansion of various data sources and has strong scalability
3. Supports different configuration formats such as json files, xml files, etc.
4. The configuration file is automatically mapped to the strongly typed entity model, which is convenient for the client to use
5. Synchronize remote configuration changes in a timed pull mode
6. Support local mode, support local configuration of a node, suitable for different scenarios