

Brian Wood

Clearwater, KS
brianwood1108@gmail.com | (909) 929-1779

<https://www.linkedin.com/in/brian-wood-9370b9274/>

SUMMARY

Experienced Python Developer with a strong background in web application development, backend system development, automation, scripting, and DevOps.

Had expertise in distributed web environments and focused **Django**, **FastAPI**, **SQLAlchemy**

Adept at collaborating with cross-functional teams and mentoring junior developers.

My focus is on building flexible high-performance architectures that scale well and adapt to changing needs.

Now I'm trying to polish and expand Go and Rust skills.

SKILLS

Programming Language – **Python**, Go, C/C++, JavaScript, TypeScript

Framework/Library – **Django**, Flask, **FastAPI**, Tornado, Tensorflow, PyTorch, scikit-learn, Keras, **SQLAlchemy**, Node.js, Verilog, React, Pandas, NumPy, Django REST Framework

Web Technologies & Tool – GraphQL, OAuth, SOAP, WSDL, REST, SAML, Jenkins, Elasticsearch, Hadoop, Kafka, XML, XSD, Selenium, Ansible, Terraform

XML Techs - XML, XSL, XSLT, DOM

Testing – unittest, pytest, mock, Hypothesis

DB & ORM – PostgreSQL, MySQL, Redis, MongoDB, Amazon DynamoDB

Cloud Service – AWS, Google Cloud Platform

Project Management - Jira, Trello

Containers/Orchestration - Docker, Kubernetes, Azure Kubernetes Service (AKS)

CI/CD & Version Control - Azure DevOps, Gitlab, Git, SVN, TFS

Software Architecture - Microservices, AKS

Data Integration & Reporting - SSIS, Power BI

EDUCATION

Bachelor Degree of Computer Science at Columbia University
2010

Apr 2006 – Sep

PROFESSIONAL EXPERIENCE

Sr. Backend Engineer | Pell Software

Nov 2020 – May 2024

➤ **Developed and Deployed Web Applications:**

- Led the development of a scalable web application using Django and React, which improved user engagement by 30%.
- Collaborated with frontend developers to design and implement RESTful APIs with Flask and FastAPI, ensuring seamless integration between the frontend and backend.
- Utilized PostgreSQL and Redis for database management, ensuring efficient data storage, retrieval, and caching mechanisms.
- Implemented user authentication and authorization using OAuth 2.0 and JWT, enhancing the security of the web applications.
- Designed and developed modular and reusable components, adhering to the principles of DRY (Don't Repeat Yourself) and SOLID (Single Responsibility, Open-Closed, Liskov Substitution, Interface Segregation, and Dependency Inversion) design principles.

➤ **Microservices Architecture:**

- Designed and maintained RESTful APIs with Flask and FastAPI, ensuring seamless integration with frontend services.
- Implemented microservices architecture to replace a monolithic application, resulting in improved system scalability and flexibility.
- Implemented inter-service communication using gRPC and RabbitMQ, enhancing the reliability and efficiency of data exchanges.

➤ **Containerization and Orchestration:**

- Employed Docker for containerization, ensuring consistency across development, testing, and production environments.
- Automated deployment processes using Jenkins, Docker, and Kubernetes, reducing deployment time by 50%.

➤ **Testing and Quality Assurance:**

- Conducted unit and integration testing with unittest and pytest, ensuring high code quality and reliability.

➤ **Deployment and Troubleshooting:**

- Deployed applications on AWS and Google Cloud Platform, optimizing performance and cost efficiency.
- Provided technical support and troubleshooting for production systems, ensuring minimal downtime and high availability.

➤ **Collaboration:**

- Utilized Elasticsearch and Kafka for real-time data processing and search functionality, significantly improving data retrieval speed.
- Developed CI/CD pipelines with GitLab and Azure DevOps, ensuring reliable and consistent software releases.

➤ **Mentoring:**

- Conducted code reviews and mentored junior developers, fostering a culture of continuous improvement and learning.

TECHNOLOGY: Python, Django, Flask, FastAPI, Jenkins, Docker, Kubernetes, AWS, Google Cloud Platform, Elasticsearch, Kafka, GitLab, Azure DevOps, unittest, pytest

Sr. Python Engineer | Edvantis

Aug 2015 – Oct 2020

➤ **Developed and Deployed Web Applications:**

- Developed backend systems using Django REST Framework and PostgreSQL, supporting high-traffic web applications.
- Deployed applications on Google Cloud Platform, utilizing services like Compute Engine, Cloud Storage, and Cloud SQL for optimal performance and scalability.
- Employed Terraform for infrastructure as code (IaC), enabling consistent and repeatable deployment of cloud resources.
- Conducted rigorous unit and integration testing with unittest and pytest, ensuring high code quality and reliability.

➤ **Automation and Scripting:**

- Created automated scripts with Python and Selenium to streamline data collection and processing tasks.
- Adopted a microservices security model with OAuth 2.0 and JWT, ensuring secure and authorized access to services.

➤ **Microservices Architecture:**

- Implemented security measures using OAuth, SOAP, and WSDL, protecting sensitive data and ensuring compliance with industry standards.

➤ **Containerization and Orchestration:**

- Managed cloud infrastructure with Ansible and Terraform, achieving a 40% reduction in operational costs.

➤ **Testing and Quality Assurance:**

- Integrated Jenkins for continuous integration and deployment, reducing manual intervention and increasing deployment frequency.
- Conducted unit and integration testing with unittest, pytest, and mock, ensuring high code quality and reliability.

➤ **Deployment and Troubleshooting:**

- Provided technical support and troubleshooting for production systems, ensuring minimal downtime and high availability.

TECHNOLOGY: Python, Django REST Framework, PostgreSQL, Selenium, OAuth, SOAP, WSDL, Ansible, Terraform, Jenkins, unittest, pytest, mock

Full Stack Developer | Iotasol

Jan 2012 – Jun 2015

➤ **Developed and Deployed Web Applications:**

- Developed full-stack web applications using Django and Flask for the backend, and React for the frontend, delivering responsive and interactive user experiences.

- Implemented and maintained RESTful APIs, facilitating smooth communication between the client-side and server-side components.
- **Microservices Architecture:**
 - Designed and implemented microservices architecture with Docker and Kubernetes, improving system scalability and resilience.
 - Integrated Elasticsearch and Logstash for centralized logging and monitoring, enabling quick detection and resolution of issues.
- **Agile Development:**
 - Collaborated with cross-functional teams to deliver projects on time and within budget, ensuring client satisfaction.
- **Testing and Quality Assurance:**
 - Conducted unit and integration testing with unittest, pytest, and mock, ensuring high code quality and reliability.
- **Containerization and Orchestration:**
 - Implemented logging and monitoring solutions using Elasticsearch and Kibana, providing valuable insights into system performance.
- **Deployment and Troubleshooting:**
 - Managed version control using Git and SVN, maintaining a clean and organized codebase.
- **Client Communication and Support:**
 - Developed and maintained technical documentation, facilitating knowledge sharing and onboarding of new team members.

TECHNOLOGY: Python, Django, Flask, React, Docker, Kubernetes, Elasticsearch, Kibana, Git, SVN, unittest, pytest, mock

DevOps Engineer | Ascendix Tech

Oct 2010 – Dec 2011

- **Containerization and Orchestration:**
 - Developed and maintained containerized applications using Docker and Azure Kubernetes Service (AKS), improving deployment consistency.
- **Agile Development:**
 - Collaborated with development teams to streamline workflows and improve productivity, fostering a culture of continuous improvement.
- **Testing and Quality Assurance:**
 - Implemented and managed CI/CD pipelines using Azure DevOps and GitLab, ensuring smooth and efficient software delivery.
- **Automation and Scripting:**
 - Automated infrastructure provisioning and configuration management with Ansible and Terraform, reducing setup time by 60%.
- **Deployment and Troubleshooting:**
 - Managed cloud environments on AWS and Google Cloud Platform, optimizing resource usage and reducing costs.

- Implemented logging, monitoring, and alerting solutions, ensuring high system availability and quick issue resolution.

➤ **Collaboration:**

- Provided training and support to team members on DevOps tools and practices, enhancing overall team capability.

TECHNOLOGY: Docker, Azure Kubernetes Service (AKS), Azure DevOps, GitLab, Ansible, Terraform, AWS, Google Cloud Platform