

Prabhas Gyawali

Dhapakhel, Lalitpur 23 | +977 9868699025 | prabhashgyawali@gmail.com | www.linkedin.com/in/prabhas-gyawali | www.prabhasgyawali.com.np

Profile

Completed my Bachelor's in Computer Engineering, with a focus on software development, DevOps, systems engineering, cloud platforms, and emerging technologies. Demonstrated strong interest in Artificial Intelligence (AI) and DevOps on cloud platforms like AWS and Microsoft Azure, in machine learning, and in developing foundational knowledge in areas like neural networks and AI-driven problem-solving. Gained hands-on experience through academic projects and practical implementations of machine learning algorithms and other engineering principles, further solidifying my passion for this field. In DevOps, I learned different tools to automate the software development lifecycle effectively and efficiently on different cloud platforms.

Education

Bachelor in Computer Engineering | July 2024 *Advanced College of Engineering and Management, Kathmandu, Nepal*

- **Key Coursework:** Data Structures & Algorithms, Operating Systems, Computer Networks, Database Management Systems, Artificial Intelligence, Distributed Systems.
- **Major Project:** Brain Hemorrhage Classification and Detection using UNet and YOLO.

SLC (+2) in Science | May 2019 *Prasadi Academy, Lalitpur, Nepal*

SEE | July 2017 Janjyoti Secondary Boarding School, Bardiya

Skills & Abilities

Technical Skills:

- Operating System: Windows, Linux, RHEL
- RHEL: Disk Management, Log Management
- Version Control: Git and GitHub
- Frontend: HTML, CSS, javascript
- Database: MongoDB, MySQL
- AI: DL, ML
- Programming language: C++, Python, Bash scripting, YAML, Next.js
- DevOps Tools: Docker, Kubernetes(loadbalancing, blue green deployment, canary deployment, gateway api, helm), Jenkins, IaaS (Terraform), Ansible, Argo CD
- Reverse Proxy Tools: Nginx, Caddy, Apache HTTP, Server
- Cloud: Microsoft Azure, AWS (IAM, EC2, S3, Route 53, Load Balancer, EKS)

- Networking(Hardware):AP, Switches, Firewalls
- Networking: VLAN, SSID, Software Firewall, Connection Type, VPN

Soft Skills:

- Communication
- Problem-Solving
- Adaptability
- Leadership

PROJECTS:

Project 1: Brain Hemorrhage Classification & Detection (Major Academic Project)

- Developed a deep learning model using **UNet** for image segmentation and **YOLO** for real-time detection.
- Focused on AI-driven problem-solving in the healthcare sector to assist in rapid medical diagnosis.

Project 2: Deployment of a Three-Tier Application on AWS Using Kubernetes

Technologies: AWS EC2, IAM, Docker, AWS ECR, Kubernetes (EKS), Ingress Controller, Application Load Balancer

Project Overview:

Deployed a containerized three-tier Node.js application on AWS using Kubernetes (EKS), focusing on scalable architecture, service communication, and high availability.

Key Responsibilities & Achievements:

- Wrote custom dockerfiles for all tiers (frontend, backend, database) that lacked existing Docker images.
- Built and pushed Docker images to AWS Elastic Container Registry (ECR) for reliable access by Kubernetes deployments.
- Deployed the full application stack on AWS EC2 instances using Amazon EKS (Elastic Kubernetes Service).
- Authored a multi-tier deployment.yaml file to define Kubernetes Deployments and Services for each layer of the application.
- Configured a Kubernetes Ingress Controller to expose the application to the internet and manage routing rules.
- Integrated an AWS Application Load Balancer (ALB) with Ingress to distribute incoming traffic evenly across the backend services.

- Used IAM roles and policies to securely grant EKS and EC2 instances access to ECR and other AWS services.
- Designed internal Kubernetes Services to enable reliable communication between different tiers.
- Followed best practices for fault tolerance and scaling within EKS, including readiness probes, resource limits, and replica management.

Result:

I have successfully deployed a scalable, containerized three-tier Node.js application on AWS using Kubernetes, ensuring continuous communication between services, secure access through IAM roles and high availability through proper resource management and load balancing. The setup visualized effective use of cloud-native tools and Kubernetes best practices for real-world application delivery.

Project 3: End-to-End CI/CD Pipeline for Laravel Web Application Deployment

Technologies: Docker, Jenkins, Git & GitHub, Nginx, Composer, Docker Compose, Ubuntu Linux

Project Overview:

Implemented a fully automated CI/CD pipeline to build, test, and deploy a Laravel-based web application using containerized infrastructure and Jenkins orchestration.

Key Responsibilities & Achievements:

- Integrated Git with GitHub for source code management.
- Configured SSH key-based authentication to securely pull and push code from private repositories.
- Created custom Dockerfiles for PHP (Laravel) and Nginx to containerize the application stack.
- Configured and maintained Docker Compose to orchestrate multi-container setup, including Laravel app, Nginx, and MySQL.
- Established a Docker bridge network for secure and isolated inter-container communication.
- Utilized Composer for Laravel dependency management inside containers.
- Ensured environment variables and Laravel configuration were securely injected into containers.
- Installed and configured Jenkins on Ubuntu to automate the build and deployment pipeline.
- Wrote a custom Jenkins pipeline script i.e Jenkinsfile to:
 - Clone the Laravel project
 - Build Docker images
 - Run composer install
 - Bring up services via Docker Compose
- Trigger automated deployment upon code changes
- Configured Nginx as a reverse proxy to serve the Laravel application and route incoming traffic to the appropriate container.
- Followed infrastructure as code and modular deployment practices.
- Emphasized repeatability, scalability, and team collaboration through version-controlled pipeline definitions and containerized environments.

Result:

I have successfully achieved a fully automated and reliable CI/CD workflow for a Laravel web application using Jenkins and Docker, enabling consistent builds, secure deployments, and faster delivery cycles. The pipeline ensured service modularity, repeatability, and scalability while aligning with DevOps best practices for infrastructure automation and team collaboration.

Project 4: Infrastructure Provisioning and Deployment of Java Web Application Using VMProjects

Technologies: Git, Vagrant, Oracle VirtualBox, Nginx, Redis, RabbitMQ, Tomcat, Maven, MySQL

Project Overview:

Provisioned a production-grade multi-tier infrastructure using Vagrant and VirtualBox to deploy a Java web application. The setup included dedicated virtual machines for Nginx, Tomcat, MySQL, Redis, RabbitMQ, and Maven, ensuring service isolation, scalability, and maintainability. Secure Git operations were integrated using SSH authentication, enabling efficient version control and streamlined deployment.

Key Responsibilities & Achievements:

- Automated the provisioning of multiple virtual machines using Vagrant to simulate a production-grade environment on a single host.
- Configured dedicated VMs for each service:
 - Nginx as a reverse proxy and load balancer
 - Tomcat to deploy and host the Java web application
 - MySQL for persistent data storage
 - Redis as an in-memory cache to optimize data retrieval
 - RabbitMQ as a message broker for asynchronous communication
 - Maven for Java project build lifecycle management
- Used key-based SSH authentication for secure Git operations (pull/push) and repository access.
- Used Git for version control and ensured continuous code integration and collaboration.
- Used Redis to cache frequently accessed data, significantly reducing database load and response time.
- Achieved a modular and maintainable deployment setup, demonstrating understanding of service isolation, scalability, and high availability concepts.

Result:

I have successfully provisioned and deployed a modular, service-isolated Java web application environment across multiple virtual machines, replicating real-world production infrastructure. The project improved deployment efficiency, reduced database latency through Redis caching, and demonstrated strong competency in infrastructure automation, service orchestration, and DevOps best practices.

Project 5: Enterprise Network Infrastructure Setup – YAJ Tech Pvt. Ltd.

Technologies: Fortinet Fortigate Firewall, Huawei Managed Switches, Ruijie Wireless Access Points (APs), Star Topology, VLAN Segmentation, Site-to-Site VPN (Silverlining), Network Protocols (DHCP, NAT, DNS)

Project Overview:

Designed, configured, and deployed a complete enterprise-grade network infrastructure for YAJ Tech Pvt. Ltd. across a multi-floor office environment. The architecture utilized a core Fortinet Fortigate firewall paired with a ground floor master switch to manage traffic routing, department-level VLAN isolation, site-to-site VPN connectivity, and campus-wide wireless coverage via floor-level Ruijie access points.

Key Responsibilities and Achievements:

- **Core Security & Edge Routing:** Deployed and configured a Fortinet Fortigate Firewall as the central security gateway, enforcing security policies, NAT rules, and inter-VLAN routing for all internal and external network traffic.
- **Star Topology & Core Switch Hierarchy:** Architected a star topology network layout, connecting the primary ground floor Huawei core switch directly to the Fortigate firewall to serve as the distribution hub for all secondary floor-level switches.
- **VLAN Segmentation & Access Control:** Implemented multiple VLANs across Huawei switches to logically segregate network traffic for different corporate departments, enhancing security, managing broadcast domains, and enforcing strict access control lists (ACLs).
- **Wireless Network Deployment:** Configured and deployed Ruijie Access Points across all floor switches, providing seamless, high-density Wi-Fi coverage mapped to designated SSID subnets and VLANs.
- **Encrypted Site-to-Site VPN:** Established a secure VPN connection with Silverlining on the Fortigate firewall, enabling encrypted, low-latency communication for remote infrastructure and cross-site services.

Result:

Successfully delivered a secure, scalable multi-floor network infrastructure for YAJ Tech Pvt. Ltd. with zero single-point-of-failure bottlenecks in traffic distribution. The implementation achieved robust departmental security isolation through VLANs, safe cross-site data access via the Silverlining VPN tunnel, and reliable high-speed wireless connectivity across all office floors.

Project 6: End-to-End CI/CD Pipeline with Kubernetes Deployment on Multi-VM Setup

Technologies: Git, GitHub, Jenkins, Docker, Kubernetes (kubeadm), Oracle VirtualBox, Vagrant, Nginx, Node.js, MongoDB, Ubuntu Linux

Project Overview:

Designed and implemented a complete CI/CD pipeline to automate the build, testing, and deployment of a Node.js application using Jenkins and Kubernetes. The application was deployed on a multi-node Kubernetes cluster provisioned across multiple virtual machines using Vagrant and VirtualBox.

Key Responsibilities and Achievements:

Infrastructure Provisioning and Cluster Setup:

- Provisioned three Ubuntu-based virtual machines using Vagrant, each configured with specific roles (Jenkins controller, application worker, and database/nginx worker).
- Installed and configured Docker, kubeadm, and kubelet on all VMs.
- Initialized a Kubernetes cluster using kubeadm and securely joined worker nodes to the master.
- Created a Dockerfile to containerize the Node.js application.
- Developed a Docker Compose file for local multi-service development (Node.js backend, MongoDB, and Nginx).
- Build and push docker images to DockerHub for later deployment via Kubernetes.
- Installed Jenkins on the master VM and configured it to manage CI/CD jobs.
- Wrote a custom Jenkins pipeline script i.e Jenkinsfile to:
 - Cloning the application from a GitHub repository.
 - Building the Docker image and pushing it to DockerHub.
 - Deploying the updated application to the Kubernetes cluster using kubectl.
- Configured GitHub webhooks to trigger Jenkins jobs automatically on code push events.
- Authored Kubernetes manifests (deployment.yaml, service.yaml) to deploy application components in pods.
- Set up internal Kubernetes services for communication between Node.js, MongoDB, and Nginx.
- Configured an Nginx Ingress Controller and NodePort service to expose the application externally.
- Implemented SSH key-based authentication for secure access between Jenkins and the virtual machines.
- Applied basic Kubernetes RBAC to limit access to deployment resources from Jenkins.
- Installed Prometheus and Grafana for container and resource monitoring.
- Set up Jenkins email notifications to alert on build failures or deployment issues.

Result:

I have successfully built a fully automated, scalable, and containerized deployment pipeline across a simulated production environment. The setup demonstrated practical knowledge of DevOps lifecycle processes, service isolation, infrastructure-as-code, container orchestration, and CI/CD best practices.

Project 7: Morpiq Platform – High-Availability Multi-Server Infrastructure

Technologies: Docker, Kubernetes, Drone CI, Ansible, Grafana, Loki, Caddy, Python (Backend), Next.js (Frontend), Ubuntu Linux(hostinger, aws)vps

Project Overview:

Designed and deployed a high-availability, multi-tier cloud-native infrastructure for the Morpiq platform across a multi-server setup. The deployment featured a Next.js web application and a Python-based backend API, orchestrated using multi-stage Kubernetes manifests, automated CI/CD pipelines via Drone CI, server configuration via Ansible, and progressive traffic management using Caddy reverse proxy and canary deployment strategies.

Key Responsibilities and Achievements:

- **Container Build & Pipeline Automation:** Configured **Drone CI** to automate code testing, multi-stage Docker builds, and image pushing upon repository updates, streamlining the release workflow.
- **Infrastructure Automation & Server Management:** Automated system configuration, dependency provisioning, and multi-node server management across target environments using custom Ansible playbooks.
- **Microservices & Container Orchestration:** Packaged Next.js frontend and Python API backend services into multi-stage Docker images and orchestrated deployments across multi-node Kubernetes clusters.
- **Progressive Traffic Routing & Load Balancing:** Configured Caddy as an edge load balancer and reverse proxy to manage automatic SSL/TLS termination and route external traffic smoothly across services.
- **Canary Deployment Implementation:** Executed a canary deployment strategy within Kubernetes to route a small percentage of production traffic to new releases, verifying application stability before initiating full rollouts.
- **Unified Observability & Monitoring:** Integrated Grafana and Loki to build an end-to-end log aggregation and visualization pipeline, allowing real-time tracking of cluster health, traffic metrics, and container logs.

Result:

Successfully established a fully automated, resilient multi-server infrastructure for the Morpiq application that enabled zero-downtime releases via Drone CI automation and Kubernetes canary deployments. The setup significantly reduced manual deployment overhead, accelerated release cycles through Ansible, and provided complete visibility into system health and application logs via Grafana and Loki.

Other projects:

Hydrojet.ae - Enterprise Water Jetting Service Website

HydrojetTechnology.com - Corporate Technology Website

FitAxis

Bizzconnect Nepal and Dubai

Morphiq (CRM, ERP, Accounting)

ZaikaX

Experience

Fullstack and DevOps Engineer | NineNeurons | June 2025 – Present

- Leading DevOps initiatives for e-commerce web applications, focusing on self-hosted Supabase and database migrations.

Junior DevOps & Network Engineer | YAJ Tech Pvt. Ltd. | March 2025 – June 2025

- Managed network infrastructure and automated software lifecycles on cloud platforms.
- Commended for strong collaboration and commitment to team-based engineering projects.

AWARDS & CERTIFICATIONS

Internship Completion Certificate: DevOps & Network Engineering, YAJ Tech Pvt. Ltd.

Teaching Assistant: Madan Smarak Secondary School (Leadership & Communication focus).

REFERENCES

- Dr. Durga Prasad Sangraula
Principal
Advanced College of Engineering and Management
durga.sangraula@acem.edu.np
- Er. Roshani Ghimire
HOD
Advanced College of Engineering and Management
roshanighimire@acem.edu.np

- Er. Ajaya Shrestha
Professor
Advanced College of Engineering and Management
ajaya.shrestha@acem.edu.np

