* **[Your Name]**  
  [Your Address] | [City, State, Zip]  
  [Your Phone Number] | [Your Email Address] |

**PROFESSIONAL SUMMARY**

Enthusiastic and detail-oriented entry-level DevOps and AWS Cloud Engineer with a solid foundation in deploying, automating, and managing cloud-based infrastructure. Skilled in using AWS services, including EC2, S3, and IAM, along with popular DevOps tools such as Jenkins, Git, Ansible, Docker, Kubernetes, and Terraform. Possess strong problem-solving abilities and a commitment to continuous learning in cloud and DevOps practices. Proven ability to implement CI/CD pipelines and infrastructure automation, leading to more efficient and reliable deployments.

**TECHNICAL SKILLS**

**Cloud Platforms:**

* **Amazon Web Services (AWS):** EC2, S3, Lambda, RDS, IAM, CloudFormation, CloudWatch, VPC, Route 53, ECS, EKS
* **Cloud Security & Compliance:** IAM roles, policies, security groups, network ACLs, data encryption

**DevOps Tools:**

* **Continuous Integration & Delivery (CI/CD):** Jenkins, Git
* **Configuration Management:** Ansible, Chef (basic knowledge)
* **Containerization & Orchestration:** Docker, Kubernetes
* **Infrastructure as Code (IaC):** Terraform, AWS CloudFormation

**Programming & Scripting:**

* **Languages:** Python, Bash/Shell Scripting
* **Version Control:** Git, GitHub, GitLab

**Additional Tools:**

* Linux/Unix, Jira, Confluence

**PROJECT EXPERIENCE**

**1. Cloud Infrastructure Automation on AWS**

**Technologies Used:** AWS EC2, S3, IAM, CloudFormation, VPC, Route 53

* Deployed and configured scalable, secure AWS EC2 instances to support web applications and backend services.
* Designed and implemented Amazon VPCs with subnets, security groups, and route tables to ensure network isolation and secure data flow.
* Utilized AWS CloudFormation to automate the provisioning of AWS resources, including EC2, S3 buckets, and IAM roles, reducing manual setup time by 60%.
* Configured Route 53 for DNS management to route user traffic to the appropriate resources and support high availability.

**2. Continuous Integration & Continuous Delivery (CI/CD) with Jenkins and Git**

**Technologies Used:** Jenkins, Git, GitHub, Python

* Established Jenkins pipelines for the CI/CD process, automating the building, testing, and deployment of code to development and production environments.
* Integrated Jenkins with Git to automate version control, implemented webhook triggers for continuous build initiation, and reduced deployment time by 40%.
* Developed Python scripts to streamline the deployment process, minimizing human errors and ensuring consistent delivery of updates.

**3. Configuration Management and Automation with Ansible**

**Technologies Used:** Ansible, Linux, Bash Scripting

* Developed Ansible playbooks to automate the setup, configuration, and management of Linux-based environments, resulting in consistent server configuration and reducing errors.
* Automated the installation of required software, system updates, and package dependencies, allowing seamless deployment across multiple environments.
* Reduced configuration time by 75% through automation, enhancing reliability and repeatability of deployments across environments.

**4. Containerization and Orchestration with Docker and Kubernetes**

**Technologies Used:** Docker, Kubernetes, Docker Compose

* Containerized applications using Docker to create lightweight, portable development environments that closely match production environments.
* Deployed and managed Kubernetes clusters to orchestrate and scale containerized applications, ensuring high availability and resilience.
* Created Docker Compose files for multi-container applications, simplifying the setup for complex application environments and accelerating the development process.

**5. Infrastructure as Code (IaC) with Terraform**

**Technologies Used:** Terraform, AWS, Git

* Wrote and maintained Terraform scripts to provision, configure, and manage AWS infrastructure, including EC2 instances, security groups, and VPC configurations.
* Leveraged Terraform’s version-controlled IaC approach, enabling team collaboration and rollback capabilities for infrastructure changes.
* Ensured consistent environments across development, testing, and production, enhancing productivity and reducing time-to-market.

**EDUCATION**

**Bachelor of Science in Computer Science**  
[University Name], [City, State]  
Graduated: [Month, Year]

* Relevant Coursework: Cloud Computing, DevOps Methodologies, Network Security, Software Development, System Administration

**INTERNSHIP EXPERIENCE**

**Cloud Engineering Intern**  
[Company Name], [City, State]  
[Month Year] – [Month Year]

* Assisted in deploying cloud-based solutions using AWS services, including configuring EC2 instances, S3 storage, and IAM for secure access.
* Supported the automation of infrastructure using AWS CloudFormation, gaining hands-on experience in provisioning and managing AWS resources through code.
* Worked closely with senior engineers on performance monitoring and cost optimization strategies, utilizing AWS CloudWatch and AWS Trusted Advisor.

**ADDITIONAL INFORMATION**

* Active member of DevOps and cloud technology communities, participating in [mention relevant events, webinars, or online courses].
* Contributed to open-source DevOps projects on GitHub, focused on building and maintaining CI/CD pipelines and containerized applications.