

# Objective

Migrate the current on-prem infrastructure to Oracle Cloud to boost efficiency, scalability, reliability, and operational agility.

# CASE STUDY

## Scope

- Migrate infrastructure components from on-premises setups to Oracle Cloud.
- Implement version control and project management tools to enhance development workflows.
- Establish a CI/CD pipeline for streamlined and automated application deployment.

## Solution

- ✓ Conducted a comprehensive migration of the on-premises infrastructure, including servers, databases, and application stacks to Oracle Cloud.
- ✓ Implemented SCM with Bitbucket for robust version control and integrated Jira for effective project management and ticketing.
- ✓ Set up a CI/CD pipeline using Jenkins, facilitating continuous integration and deployment of applications directly to Oracle Cloud.
- ✓ Utilized Docker for containerization of applications, ensuring consistency across development, testing, and production environments.
- ✓ Employed Kubernetes within Oracle Cloud for orchestrating containerized applications, enhancing scalability and manageability.

## Value Added

- Boosted scalability and reliability with Oracle Cloud for efficient growth support.
- Enhanced development speed and collaboration using Bitbucket and Jira integration.
- Streamlined deployments with CI/CD pipelines, reducing errors and time.
- Achieved cost savings by transitioning to cloud infrastructure, lowering hardware expenses.
- Increased operational agility with cloud tools for quicker market response.

## Frameworks & Tools



# Objective

Provide comprehensive DevOps support for client deployments, including regular deployments, preparation of EASE upgrade packages, server upgrades, and patches.

# CASE STUDY

## Scope

- Support regular deployment activities and prepare upgrade packages for the system.
- Build and maintain branches for various deployment stages.
- Perform system upgrades and patches on servers.
- Ensure rigorous documentation and backup processes are in place for system integrity and recovery..

## Solution

- ✓ Managed branch builds and deployment packages, including system upgrades and patches. Conducted server upgrades from the release management system for enhanced functionality and security.
- ✓ Implemented comprehensive server monitoring for URL and disk usage, ensuring system performance is maintained at optimal levels. Verified checkpoints regularly to ensure deployments meet quality standards.
- ✓ Managed the backup of the Release\_UTF database and developed detailed documentation for disk utilization, MongoDB installation scripts, and backup procedures to ensure data integrity and recoverability.
- ✓ Utilized Ansible for deployment automation, streamlining the process and reducing the potential for human error.

## Value Added

- Through regular updates and careful monitoring of server performance and disk usage.
- Utilized Ansible to automate deployments, significantly reducing manual efforts and minimizing deployment times.
- Established comprehensive documentation and backup protocols to ensure easy recovery and continuity in case of system failures.
- Provided consistent daytime support to client's vendors, facilitating smooth operations and quick resolution of issues.

## Frameworks & Tools



# Objective

Built and managed a robust Oracle Cloud Kubernetes cluster for multi-region production and non-production environments with advanced CI/CD, cybersecurity, and cost optimization.

# CASE STUDY

## Scope

- Set up Kubernetes clusters in Oracle Cloud for the US, UK, and Canada across production and non-production environments.
- Develop CI/CD pipelines for applications related to publications and license screening.
- Enable developer self-sufficiency in CI/CD job execution and shift left in SDLC.

## Solution

- ✓ Designed scalable and secure Oracle Cloud infrastructure for diverse environments and regions.
- ✓ Configured Jenkins for automated application deployment across environments.
- ✓ Enabled developers to independently deploy code using automated CI/CD tools, enhancing operational efficiency.
- ✓ Implemented comprehensive monitoring with Oracle Monitoring and Prometheus for optimal performance.
- ✓ Created an environment dashboard for centralized monitoring and management.

## Value Added

- Multi-regional deployment tailored to regional requirements and compliance.
- Streamlined deployment processes with customized CI/CD pipelines.
- Enhanced cybersecurity measures and cost optimization through strategic OCI usage.
- Provided operational documentation and training materials for knowledge transfer.

## Frameworks & Tools



# Objective

Design and implement a scalable and highly available Kubernetes cluster infrastructure on AWS to support the backend for client's application across multiple environments, including development, testing, and production.

# CASE STUDY

## Scope

- Build a robust infrastructure to support the client's mobile application on iOS and Android.
- Set up various environments (Dev, QA, UAT, Demo, Performance, and Production) with a focus on scalability, availability, and security.
- Establish a CI/CD pipeline for seamless deployment of backend applications.
- Implement comprehensive monitoring and alerting systems, and optimize costs.

## Solution

- ✓ Developed a scalable and high-availability infrastructure using AWS services and Kubernetes to cater to multiple backend applications across all environments.
- ✓ Configured a CI/CD pipeline using Jenkins for automated deployments across development, testing, and production environments.
- ✓ Enabled self-sufficient Jenkins jobs for deploying code with specific tags or branches, including script execution and log tracing.
- ✓ Implemented HTTP to HTTPS redirects and cost optimization strategies using AWS Elastic Kubernetes Service (EKS).
- ✓ Established a robust monitoring and alert system using Prometheus and Grafana, and implemented security measures including SAST and DAST integrations.

## Value Added

- Achieved zero downtime deployments with rollback capability for continuous availability.
- Created a unified recon dashboard for environment statuses and code updates from development to production.
- Improved code management with auto-versioning for branches and tags.
- Implemented advanced security measures, including authentication, authorization, encryption, and network security.
- Optimized infrastructure costs using AWS EKS and other cost-effective services without performance sacrifices.

## Frameworks & Tools



# Objective

Support and enhance client DevOps across teams with standardized tools, Windows Docker virtualization, and a DevOps culture for efficiency and innovation in gas analyzer development.

# CASE STUDY

## Scope

- Standardize the use of DevOps tools across different client teams.
- Implement Windows Docker virtualization for various Windows flavors used in Zero
- Reference Modules for isotope instruments in gas analyzers.
- Cultivate a DevOps culture among teams that traditionally did not use DevOps tools.

## Solution

- ✓ Introduced and nurtured a DevOps culture across multiple teams, ensuring an understanding and adoption of DevOps methodologies and practices.
- ✓ Completed various DevOps assignments, including the migration of jobs from TeamCity to Jenkins, which streamlined CI/CD processes and improved deployment efficiencies.
- ✓ Implemented Windows Docker containerization for the Zero Reference Module, enhancing the portability and scalability of applications used in gas analyzers.

## Value Added

- Introduced DevOps practices, boosting collaboration, accelerating development, and improving project agility.
- Standardized DevOps tools like Jenkins, Docker, and Python3 for consistent and efficient outcomes.
- Employed Windows Docker for gas analyzer virtualization, enhancing testing and deployment.
- Transitioned from TeamCity to Jenkins, optimizing CI/CD pipelines for faster turnaround and resource management.

## Frameworks & Tools



Industry BFSI

Domain DevSecOps

Location USA

# Objective

Develop a financial portfolio application using the Serverless Framework on AWS to enhance operational efficiency, reduce infrastructure overhead, and improve scalability.

# CASE STUDY

## Scope

- Develop and deploy AWS Lambda functions using Node.js through the Serverless framework.
- Create and deploy React-based frontend code on AWS S3.
- Integrate AWS Lambda functions with DynamoDB and AWS API Gateway for seamless data flow and API management.

## Solution

- ✓ Implemented React code and deployed it to an AWS S3 bucket for static web hosting, ensuring fast content delivery and reduced server load.
- ✓ Developed Node.js AWS Lambda functions, deployed and managed through the Serverless framework to handle backend processes efficiently.
- ✓ Integrated Lambda functions with DynamoDB for database operations and AWS API Gateway for handling API requests, ensuring a fully managed, serverless architecture.

## Value Added

- **Adaptability:** VAST developers quickly mastered the Serverless framework for efficient solutions.
- **Full-stack expertise:** Proficient in ReactJS, NodeJS, MongoDB, and microservices for scalable Lambda functions.
- **Cost-effective scalability:** Serverless architecture reduces server management costs and scales automatically.
- **Cross-cultural collaboration:** Smooth project execution with remote, cross-cultural teams.

## Frameworks & Tools



DynamoDB



Amazon API Gateway

# Objective

Implemented DevOps for India's top OTT: automated workflows, optimized AWS costs, migrated apps from Azure to AWS, and enhanced CI/CD practices.

# CASE STUDY

## Scope

- Conduct infrastructure assessment and automation.
- Establish DevOps culture across multiple projects.
- Optimize AWS cloud costs and streamline application migration from Azure to AWS.
- Implement centralized log collection, analytics, and alerting systems.

## Solution

- ✓ Assessed existing infrastructure for optimization.
- ✓ Implemented AWS Trusted Advisor for cost control.
- ✓ Automated environment scheduling for cost optimization.
- ✓ Set up AWS CodePipeline for CI/CD.
- ✓ Migrated apps from Azure to AWS seamlessly.
- ✓ Utilized Zabbix for robust monitoring.
- ✓ Established log analytics with ELK for scalable analysis.

## Value Added

- Achieved cost savings with AWS Trusted Advisor and automation.
- Fostered DevOps culture for enhanced collaboration.
- Implemented advanced monitoring using Zabbix and ELK.
- Ensured efficient Azure to AWS migration.
- Integrated multiple tools for streamlined workflows.

## Frameworks & Tools



# Objective

Design, architect, and setup of Kubernetes cluster infrastructure in AWS cloud for multiple environments of Dev, QA, UAT Production in multiple AWS regions using DevOps best practices

# CASE STUDY

## Scope

- Evangelize DevOps practice and Abide by 12 factor App principle.
- Build infrastructure to support a highly scalable and available client's platform for the US and Europe regions.
- Setup of production environment for release deployment with rollback facility
- Setup CI/CD pipeline for deploying frontend and backend applications
- Mitigating Threats to Cyber Security.

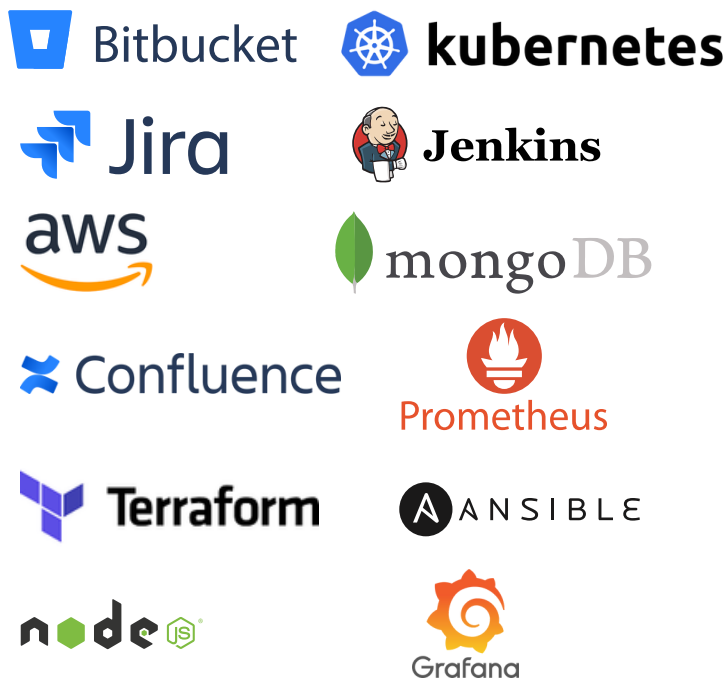
## Solution

- ✔ Infrastructure design, setup, and deployment using DevOps best practices.
- ✔ CI/CD setup for seamless transition of code via tagged auto versioning and monitoring dashboards with logs analytics.
- ✔ Setup of lower environments Dev, QA, UAT, Demo.
- ✔ End-to-End Application Provisioning using IaC (Terraform).
- ✔ SNS Alerts and notifications for infrastructure changes.
- ✔ Setup of the production environment for release deployment with rollback facility
- ✔ Fortnight patching

## Value Added

- Savings on Operation costs, fewer operation cycles, and no miscellaneous charges
- AWS Cost optimization
- DAST - Weekly testing to mitigate security weaknesses and vulnerabilities
- SAST(Snyk) - Weekly testing to build secure applications
- Security Scorecard - to mitigate cyber threats
- Migration of client's Website from AWS (EC2 instance)to WordPress.
- Zero downtime deployments with rollback in case of any failures
- Detailed Monitoring of applications using Prometheus and Grafana

## Frameworks & Tools





# Objective

Discovery of current process and  
Implementation of DevOps Practices.

## CASE STUDY

### Scope

- DevOps Architecture consulting
- DevOps Practice Setup
- Understand the Skills-gap and limiting factor from adoption of DevOps culture.

### Solution

- ✓ Setting of DevOps practices.
- ✓ Create learning paths from top in organization to bottom from start to end for implementing DevOps practices.
- ✓ Develop and curate training content to help NetOps professionals develop the skills and DevOps mindset to implement DevOps culture.

### Value Added

Create a DevOps mindset and culture in the organisation people with various responsibilities ownership

### Frameworks & Tools



# Objective

To create automated cross OS dependent product DevOps deployment processes for Client as well as internal environments

# CASE STUDY

## Scope

- Design an automation approach for both client and internal environments.
- Conduct an analysis of automation tools, particularly Configuration Management (CM) tools.
- Automate the process of product DevOps deployment across different operating systems.

## Solution

- ✓ Implemented automation of VMs post-configuration operations for both Linux and Windows OS, ensuring streamlined setup processes.
- ✓ Automated Windows operations including role installation, domain joining, and software installations using domain user credentials.
- ✓ Automated the installation of third-party vendor software, such as Cisco, and the update processes for various software provided by vendors like Cisco.
- ✓ Enabled multiple VM deployments on VMware platforms for both Linux and Windows, enhancing the deployment capacity and speed.

## Value Added

- Developed automation for Linux and Windows, enhancing consistency and minimizing manual deployment efforts.
- Utilized SaltStack for configuration management and Jenkins for CI/CD, optimizing environment setups and deployments.
- Streamlined post-configuration operations, reducing setup time and complexity.
- Automated updates and installations of critical third-party software, improving service reliability and uptime.

## Frameworks & Tools



# Objective

The client's environment from AWS to Azure and optimize the Total Cost of Ownership (TCO) by establishing DevOps practices, utilizing cost-effective technologies, and ensuring seamless application deployment.

# CASE STUDY

## Scope

- Evaluate Azure services suitable for migration.
- Set up DevOps practices and Azure infrastructure.
- Deploy applications within the Azure environment.
- Replace paid services with open-source alternatives or Azure-integrated services to reduce costs.
- Implement CI/CD pipelines for infrastructure setup and application deployment.
- Implement security features including application and VPN gateways.

## Solution

- ✓ Implemented Azure infrastructure from scratch using Docker technology, tailored to the specific needs of the client.
- ✓ Deployed applications to Azure, ensuring optimal performance and scalability.
- ✓ Achieved cost optimization by integrating open-source tools and adjusting to Azure's native services.
- ✓ Established robust security protocols using Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST).

## Value Added

- Enhanced operational efficiency and agility through comprehensive DevOps practices.
- Leveraged open-source tools and Azure services to cut costs while ensuring high service levels.
- Implemented SAST and DAST for robust application and data security.
- Integrated Prometheus for advanced observability, enabling proactive cloud infrastructure management.
- Utilized Docker and Kubernetes for smooth and scalable deployments on Azure.

## Frameworks & Tools



# Objective

To build and deploy multiple applications as microservices in a Docker-based environment over VMware ESXi, focusing on infrastructure automation, application monitoring, and the introduction of DevOps practices

## CASE STUDY

### Scope

- Assess requirements and provide architectural design for a Docker-based microservices environment.
- Automate infrastructure and application deployment processes, incorporating custom auto-versioning and application monitoring.

### Solution

- ✓ Created a Docker-based environment on VMware ESXi to host microservices, ensuring high scalability and efficiency.
- ✓ Built applications into Docker images, enhancing infrastructure automation.
- ✓ Implemented Bamboo jobs to automate the deployment process and manage multiple environments effectively.
- ✓ Established comprehensive application monitoring to ensure optimal performance and reliability.

### Value Added

- Implemented DevOps practices to boost operational agility and deployment efficiency.
- Developed an environment dashboard for enhanced visibility and control of deployment phases and versions.
- Delivered a cloud-agnostic solution allowing flexibility across any ISP for future scalability.
- Achieved cost savings by transitioning from Windows to Linux servers, reducing licensing and operational expenses.
- Provided comprehensive training and handover to client teams for effective management and scaling of infrastructure.

### Frameworks & Tools



# Objective

Deploying infrastructure in AWS for SAP Business One and automating deployment of SAP Business One with MS SQL, and HANA database for development and production environments.



# CASE STUDY

## Scope

- Set up CI/CD pipeline for deploying Business One with MSSQL & HANA database using Jenkins.
- Setting up Jenkins job to spin up dev environments for SAP B1 with MS SQL and HANA database.
- Enable developer self-sufficiency through Jenkins jobs.
- Setting up a compatibility matrix through Jenkins jobs to support various versions of SAP B1.
- Document automation workflows
- RBAC-based control

## Solution

- ✓ Developed a CI/CD pipeline using Jenkins for SAP Business One deployment.,
- ✓ Automated the provisioning of development environments to increase developer autonomy
- ✓ Implemented RBAC to ensure secure access control.

## Value Added

- Environments provision quickly, boosting speed and agility
- Streamlined processes enhance decision-making with real-time insights.
- Flexible planning eliminates guesswork by only provisioning necessary resources.
- Deployment automation reduces support tickets for Dev/Demo environments
- Developers can independently deploy machines, enhancing their empowerment

## Frameworks & Tools



# Objective

To design, architect, implement, and automate a highly resilient and scalable CI/CD pipeline for AI-enabled products in the Healthcare domain with all compliance

## CASE STUDY

### Scope

- Comprehend the product architecture and document deployment steps.
- Identify suitable Amazon Web Services for deployment and migrate to the appropriate DevOps tools
- Design and implement a resilient CI/CD pipeline using the selected tools
- Provide secure web hosting solutions.
- Conduct training and handover to the client's team.

### Solution

- ✓ Strategized code management with auto-versioning features.
- ✓ Built automated CI/CD pipeline orchestration using various Jenkins plugins.
- ✓ Developed custom scripts for automating the installation and configuration of the Django framework and service management using Supervisor.
- ✓ Maintained environment state and provisioned new environments using Saltstack.
- ✓ Delivered a clean handover with industry-standard documentation and extensive WebEx recordings.
- ✓ Created a customized environment monitoring dashboard with email notifications and alerts for transparent management oversight.

### Value Added

- The CI/CD pipeline was aligned with healthcare compliance requirements to boost security and reliability.
- Automation of pipeline processes cut down on manual errors and sped up deployments
- A customized dashboard was implemented to give management real-time system status and alerts, improving decision-making

### Frameworks & Tools



# Objective

Migration of on-premise Java and .NET-based 3-tier application to AWS. Keeping in view the varied and multi-level Integration abiding by all security and compliances.

# CASE STUDY

## Scope

- Migrate the on-premise application to the Cloud abiding by all security and compliances.

## Solution

- ✓ Successfully migrated and deployed the on-premise application to the AWS cloud.
- ✓ Implemented CI/CD on AWS to streamline development and deployment processes.
- ✓ Enhanced application performance, security, and scalability through service-level implementations.
- ✓ Designed application load balancing to ensure horizontal availability and high resilience.

## Value Added

- **Cost Savings:** Achieved significant cost reductions by migrating to AWS
- **Modernized Legacy System:** Modernized a 20-year-old codebase, enhancing maintainability and scalability
- **Enhanced Performance and Security:** Improved application performance and security, meeting modern standards and requirements.

## Frameworks & Tools



Jenkins



# Objective

Migration of on-premise Java and .NET-based 3-tier application to AWS.

## CASE STUDY

### Scope

- Build a business case for cloud migration.
- Assess the current app infrastructure.
- Adopt a flexible integration model for the cloud environment.
- Address all compliance, security, privacy, and data residency requirements.
- Manage the migration process efficiently.
- Evaluate and select appropriate AWS services for migration.

### Solution

- ✓ Engaged strategically in planning, solutions, and execution stages.
- ✓ Migrated 20-year-old architecture and code to AWS cloud with controlled modernization, such as transitioning RMI service to a serverless architecture
- ✓ Collaborated closely with the client's technical teams (Development & Systems), providing training on the latest technologies applicable.
- ✓ Ensured the application and infrastructure migration to the cloud was secure, scalable, and performed optimally.

### Value Added

- Assisted the organization with adapting to new technologies, and enhancing their technical capabilities
- Achieved direct savings on operational and infrastructure costs through cloud automation
- Provided robust technical support to ensure a smooth transition and ongoing operation.

### Frameworks & Tools

