Azure Infrastructure Provisioning & DevOps Automation

Introduction

FusionPOS is a next-generation, cloud-native platform designed for unified Point-of-Sale (POS), Inventory, Billing, Appointments, and Reporting across multiple retail chains and e-commerce platforms. This training case simulates the real world provisioning and management of FusionPOS using Infrastructure as Code (Terraform) and DevOps Pipelines (Azure DevOps YAML).

Client Details

Name: Confidential | **Industry:** Ecommerce, Retail, Software | **Location:** USA

Objectives

Participants will:

- Practice **Azure resource provisioning** using Terraform
- Create reusable **Terraform modules**
- Design and implement **Azure DevOps CI/CD pipelines** (YAML)
- Integrate with **Azure services** such as API Management, Cosmos DB, Key Vault, Azure Functions, and more
- Ensure **high availability, security**, and **compliance**
- Implement **DevOps best practices**: automated testing, infrastructure validation, deployment slots, versioning, observability

Technologies

Azure Services to Be Used

- App Service (Web + API)
- Azure Functions
- Azure Cosmos DB (Core API)
- Azure SQL Database
- Azure Storage (Blob Gen2 with SFTP)
- Azure Redis Cache
- Azure API Management
- Azure Front Door

Azure Infrastructure Provisioning & DevOps Automation

- Azure Key Vault
- Azure Service Bus
- Azure Data Factory
- Azure Monitor + App Insights
- Azure Active Directory

Project Description

Infrastructure Setup (Terraform)

Modules to Create

1. Networking (VNet, NSG, Subnets, Private Endpoints)
2. App Service Plan + App Services
3. Function Apps
4. Azure SQL (with Private Endpoint & Geo-redundant backups)
5. Cosmos DB (Multi-region, Session Consistency)
6. Blob Storage (Hot/Cold Tier, SFTP, CMK Encryption)
7. API Management (Standard Tier)
8. Redis Cache (Standard Tier)
9. Front Door (Standard Tier, WAF Enabled)
10. Key Vault (With Managed Identity Integration)
11. Service Bus
12. Data Factory
13. Monitoring & Alerts (App Insights, Diagnostic Logs)

Requirements

- Environments: **Dev**, **UAT**, **Prod** (Active-Passive setup for Prod)
- Configuration Tags: Environment, Department, BillingCode
- Output variables for CI/CD consumption

DevOps Practice (YAML Pipelines)

CI/CD Pipelines

Create the following pipelines:

1. Terraform Deployment Pipeline

- Initialize, validate, plan & apply Terraform templates
- Use Azure Resource Manager Service Connection
- Example trigger: main or env/dev branch

2. App Build & Deploy

- Build Web/API/Function App from GitHub
- Run unit tests
- Deploy to **App Service deployment slots** using AzureWebApp task
- Swap to production on success

3. API Management Publish Pipeline

- Validate OpenAPI specs
- Push APIs to APIM using DevOps extension
- Apply policies (e.g., throttling, format conversion)

4. Monitoring & Alerts Pipeline

- Deploy diagnostic settings for all resources
- Enable App Insights telemetry
- Configure alert rules (e.g., CPU > 80%, Error Rate > 2%)

Functional Use Case Scenarios

Participants will simulate the following:

1. Create a customer appointment and order via API
2. Push transaction data to Cosmos DB
3. Sync order data to Azure SQL via Function App
4. Send confirmation message using Service Bus Queue
5. Render reports using data from Azure SQL + Power BI
6. Secure secrets in Key Vault, accessed by Function App
7. Simulate failover from Primary to Secondary region (Front Door)

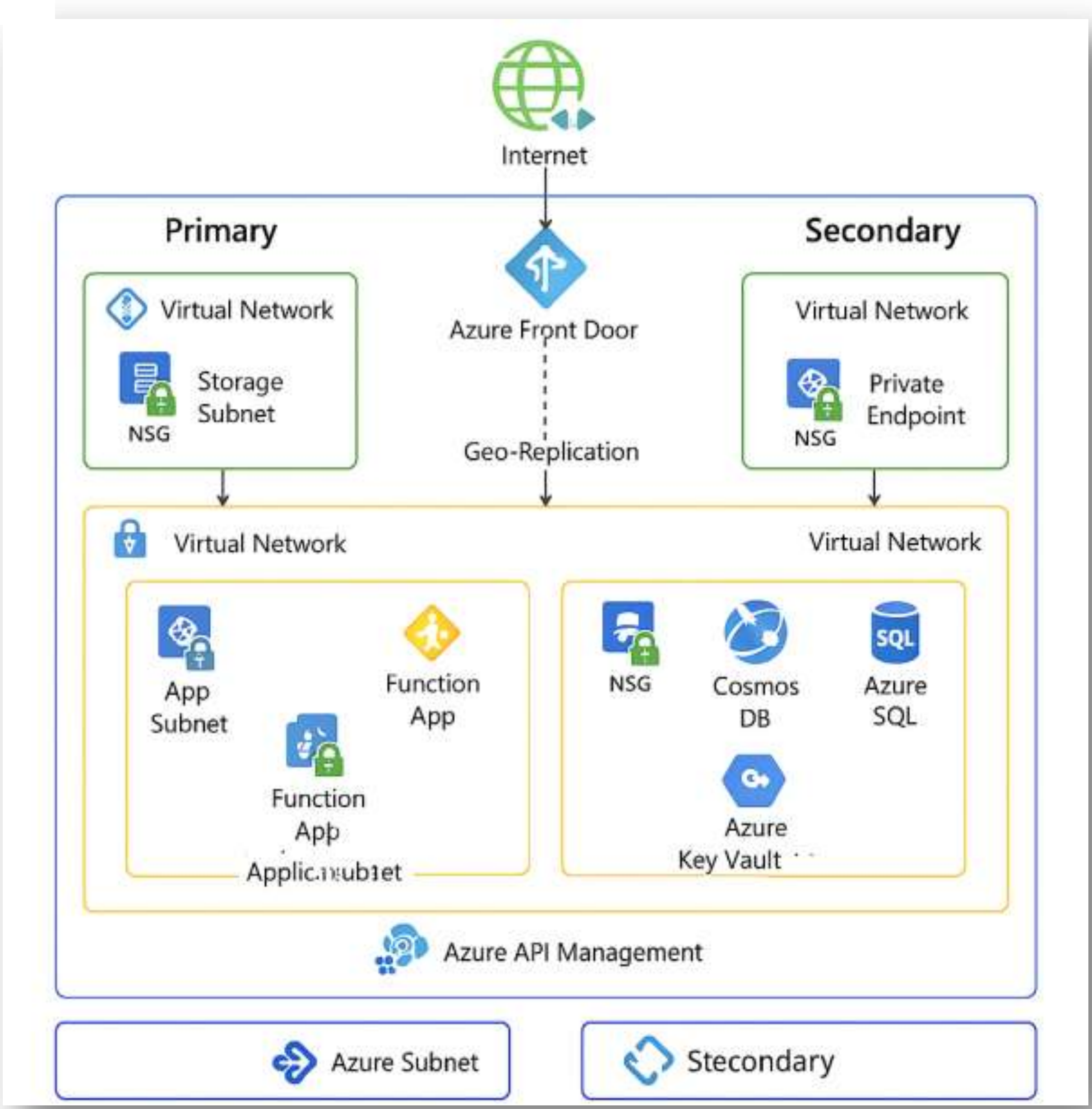
Expected Outcomes

By the end of the case study, participants should be able to:

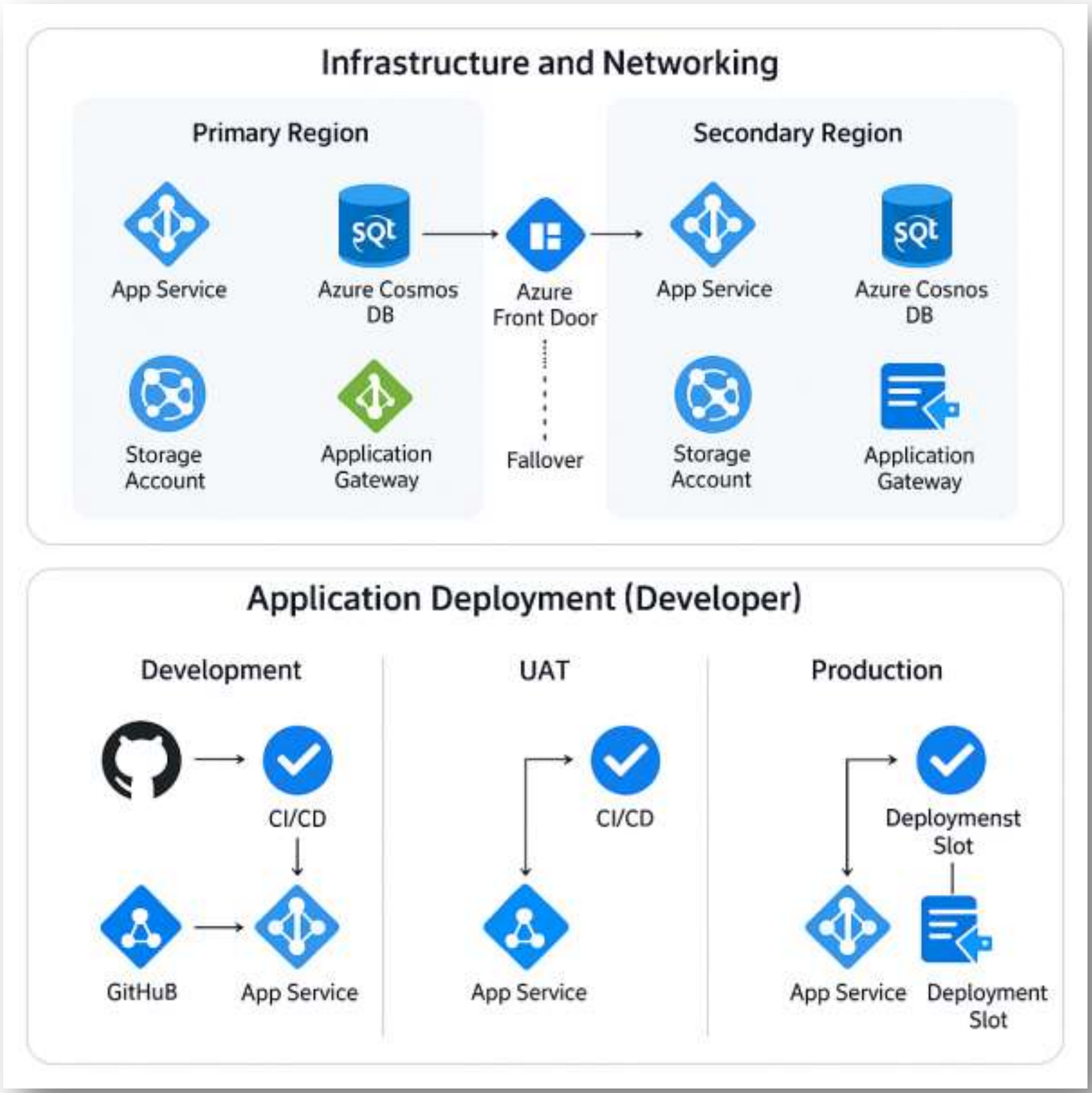
- Build a modular, reusable, and production ready Terraform codebase
- Automate the deployment of FusionPOS into Azure using CI/CD
- Understand security practices: Managed Identities, RBAC, CMKs
- Monitor and troubleshoot applications using Azure-native tools

Azure Infrastructure Provisioning & DevOps Automation

Schematic Diagrams



Azure Infrastructure Provisioning & DevOps Automation


Infrastructure and Networking
Primary Region
App Service
Azure Cosmos DB
Storage Account
Application Gateway
Azure Front Door
Fallover
Secondary Region
App Service
Azure Cosnos DB
Storage Account
Application Gateway
Application Deployment (Developer)
Development
CI/CD
GitHuB
App Service
UAT
CI/CD
App Service
Production
Deploymenst Slot
App Service
Deployment Slot