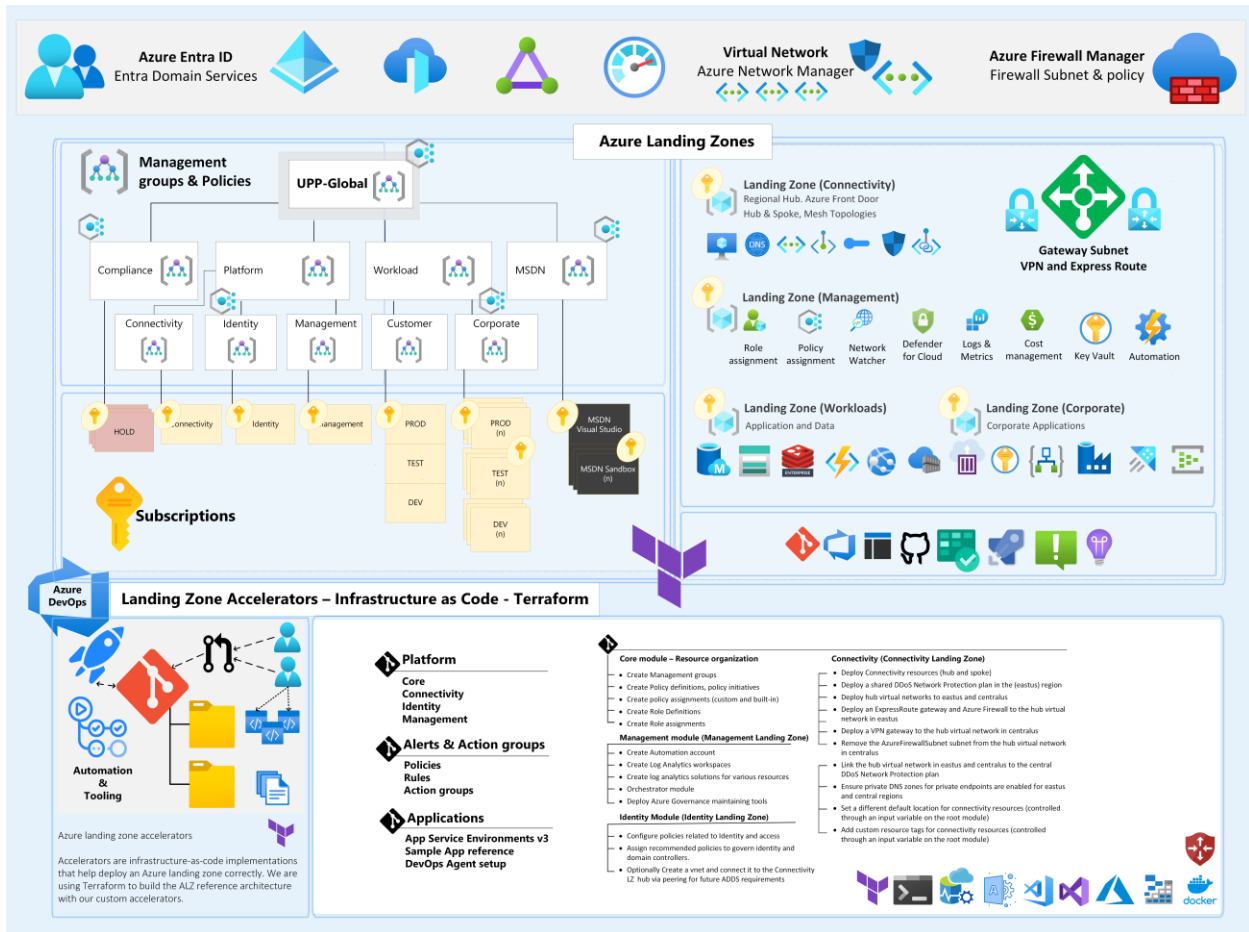


Architecture Blueprint & POC Lifecycle diagrams summarized below.



High-Level Purpose

This blueprint demonstrates the design and implementation of a full enterprise-grade Azure cloud foundation, built to support modern application development teams and onboard corporate workloads. It highlights how the environment was designed to be secure, compliant, automated, scalable, and ready for deploying workloads.

The platform enables:

- Enterprise networking
- Application development environments (dev/test/prod)
- Corporate and line-of-business workloads
- Security and governance baselines
- Identity and access management
- Operational monitoring and management

- Scalable workload landing zones

This aligns with Azure Landing Zone (ALZ), Cloud adoption framework (CAF) principles and Terraform Infrastructure-as-Code to ensure repeatability and enterprise readiness.

What the Architecture Represents

The Azure Landing Zone diagram shows a complete enterprise cloud foundation with:

1. Resource Organization & Governance

- Management Groups (Platform, Workload, Connectivity, Identity, Corporate, MSDN, etc.)
- Azure Policy definitions, initiatives, and assignments
- Role definitions and RBAC assignments
- Subscription design for PROD/DEV/TEST/Management/Connectivity/Identity

2. Identity & Access

- Azure Entra ID
- Entra Domain Services
- Identity Landing Zone
- Conditional Access, MFA, and identity governance policies
- Enterprise Key Management

3. Networking & Connectivity

- Hub & Spoke or Mesh topologies
- Regional hubs
- Azure Firewall & Firewall Manager
- Azure Network Manager
- VPN Gateway / ExpressRoute Gateway
- Shared DNS, Private Endpoints, Route Tables
- Traffic flow and segmentation controls

4. Management, Operations & Observability

- Azure Monitor (Logs, Metrics)
- Application Monitoring
- Network Watcher
- Defender for Cloud
- Automation Accounts
- Diagnostic settings
- Cost Management
- Certificates and Keys Management

5. Workload Landing Zones

- Application landing zones
- Data landing zones
- Corporate application landing zones
- App Service Environments v3
- DevOps agent pools

6. Automation & Tooling

- Azure DevOps Pipelines
- Terraform Landing Zone Accelerators
- IaC modules for:
 - Core (MGs, Policies, RBAC, RGs)
 - Identity
 - Connectivity
 - Management
 - Workloads

Key Azure Services, Tools, and Utilities Used

Below is a consolidated list of the major Azure components required to build this architecture; all which I leveraged as the architect:

Governance & Security

- Azure Policy
- Azure Blueprints / Landing Zone Accelerators
- RBAC & Custom Roles
- Defender for Cloud
- Key Vault
- Management Groups

Identity

- Azure Entra ID
- Entra Domain Services
- Conditional Access
- Privileged Identity Management (PIM)

Networking

- Virtual Networks (Hub/Spoke/Mesh)
- Azure Firewall
- Azure Firewall Manager
- Azure Network Manager
- VPN Gateway / ExpressRoute Gateway
- Private Endpoints
- Azure Private Link
- Azure Monitor Private Link Scopes (AMPLS)
- DNS Zones & Private DNS Resolver, Route Tables & UDRs

Management & Monitoring

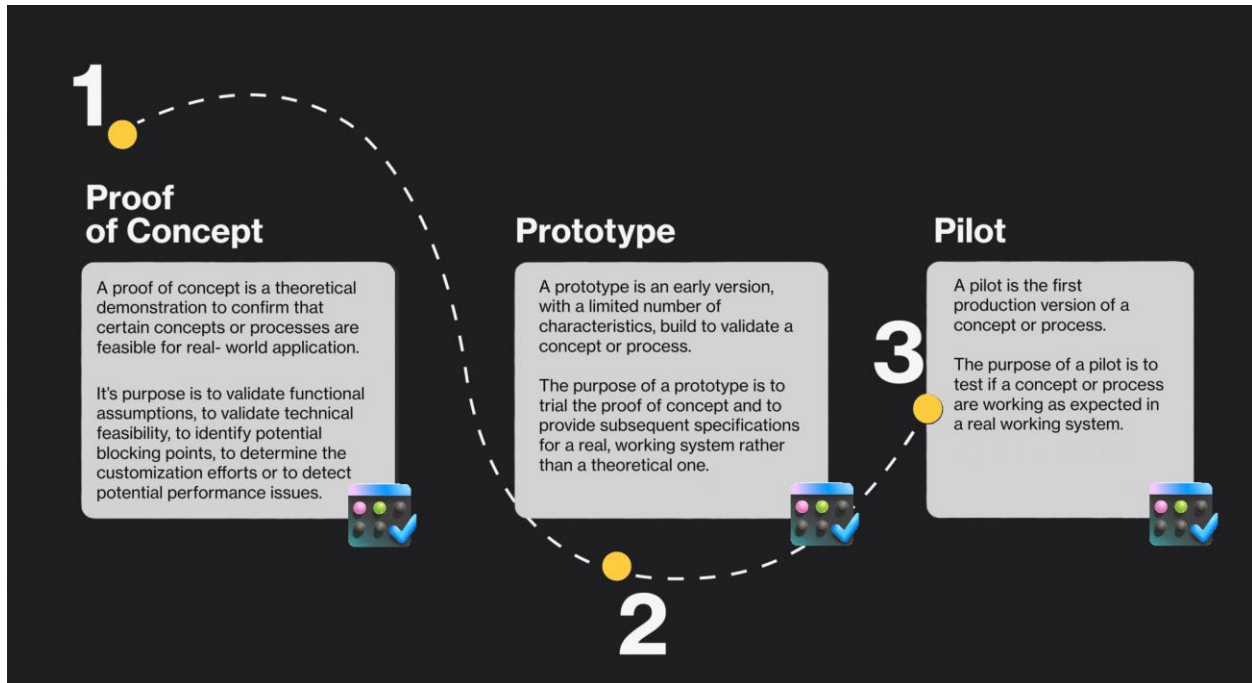
- Azure Monitor (Logs, Metrics, Alerts)
- Log Analytics Workspace
- Application Insights
- Network Watcher
- Automation Accounts
- Action Groups

Application Platform

- App Service Environment v3
- App Services
- Containers
- Azure Front Door
- DevOps Agents
- Corporate application landing zones

Infrastructure-as-Code & DevOps

- Terraform (Landing Zone modules, custom modules, IaC automation)
- Azure DevOps Pipelines
- Git repositories for version control
- CI/CD for platform and workload deployments

POC → Prototype → Pilot Lifecycle

The POC → Prototype → Pilot progression diagram demonstrates the structured, iterative delivery model I followed while building the environment. I began by validating core feasibility, then expanded and refined the design, and ultimately delivered a hardened, production-ready pilot.

This progression directly reflects how the architecture blueprint (detailed above) was implemented. The blueprint provided the foundation with governance, identity, networking, automation, and landing zone structure; and each stage of the lifecycle incrementally brought that blueprint to life. The result is an enterprise-grade cloud platform built through disciplined validation, refinement, and real-world readiness testing.

1. Proof of Concept

I validated:

- Feasibility of Azure Landing Zones
- Terraform automation patterns
- Network topology options
- Policy and governance model
- Identity integration

2. Prototype

I built:

- A functional but limited version of the landing zone
- Initial connectivity hub
- Early Terraform modules
- First workload landing zone
- Monitoring and security baselines

3. Pilot

I delivered:

- A production-ready landing zone foundation
- Fully automated IaC deployment pipelines
- Hardened security and governance
- Operational monitoring and alerting
- Connectivity and identity integration
- Application-ready landing zones