

Task 1: Proposed Solutions for Leveraging Azure Cloud

1. Hybrid Cloud Architecture:

- Host the public-facing website on the cloud, but maintain sensitive data in an on-premises data center or in a private cloud environment.
- Implement features from Azure services such as Azure ExpressRoute to maintain a secure and private connection between your on-premises infrastructure and Azure, so that never one snippet of sensitive data will traverse the public internet.

2. Azure Security Features:

- Implement Azure's security services, such as Azure Key Vault, for the secure handling of sensitive information like API keys and certificates.
- Use Azure Active Directory (AD) for identity management and access. It also gives role-based access controls, ensuring the right people get access to confidential data.

3. Data Encryption:

- Implement data-at-rest and in-transit encryption using Azure services like Azure Storage Service Encryption and TLS/SSL for web applications.

Task 2: Approach to Governance and Cost Management

a. Resources Governance

- Resource group for Development, Testing, and Production this separates the resource groups to three environments- development, testing, and production. This would enable better management, providing access control and organization.
- Azure Policy-Azure Policy can enforce rules and compliance across all resource groups. Policies will ensure that only approved resources are deployed while configurations aligned to company standards.
- Utilize Azure DevOps for CI/CD pipelines for automating deployments in each environment while keeping things separated.

b. Billing Lifecycle and Cost Management

1. Azure Cost Management and Billing:

- Use Azure Cost Management + Billing to monitor and manage costs appropriately. This tool will be used to give insight into spending patterns and

to cost the bill accurately according to the various departments or projects that are being processed.

- Create budgets for every resource group and allow alerting when consumption is close to budget limits.

2. Tags for Resource Management:

- Append tags to all resources so you can easily track and categorize them: Department, Project or Environment for example. This allows for detailed reporting and cost analysis

3. Separate Subscription:

- Have different Azure subscriptions for the development, testing and production environment. This isolate billing hence easier management.

Task 3: Suggested Tools for Acquisition and Manipulation of Resource Groups in Azure

- 1. Azure Portal:** it offers an interactive graphical user interface for the administration of all the Azure resources; thus, it makes the ride easy for the administrator to access and control resource groups.
- 2. Azure CLI:** for scripting and automation purposes. The Azure CLI allows the development of efficient management of Azure resources with command-line commands.
- 3. Azure PowerShell:** highly apt for automate the administrative tasks and also to manage the resources. With powerful cmdlets, these support the resource management and are of more help to Windows administrators.
- 4. Azure SDKs:** usable for any programming language such as .NET, Python, and Java, for the programmatic access and management of Azure resources which can be put to use in the applications for customized solutions for management.
- 5. Azure Resource Manager (ARM) Templates:** Definition of the infrastructure as code. Ensures a consistent and repeatable deployment of a resource group and all configurations, respectively.