Project-

Project 1: Azure Service Deployment and Integration

Objective: Deploy and integrate multiple Azure services to build a simple web application while understanding fundamental Azure concepts.

Solution Steps:

Define Application Requirements: Determine the requirements of your web application, including its functionality, data storage needs, security requirements, and scalability considerations.

Design Application Architecture: Design the architecture of your web application, identifying the Azure services required to fulfill each component's functionality. Consider using a combination of IaaS, PaaS, and SaaS offerings based on your requirements.

Set Up Networking and Security: Configure Azure Virtual Networks to isolate your application resources and ensure network security. Implement Network Security Groups (NSGs) and Azure Firewall to control inbound and outbound traffic. Utilize Azure Active Directory (Azure AD) for identity and access management.

Deploy Virtual Machines and Virtual Machine Scale Sets (VMSS): Deploy Virtual Machines (VMs) to host the backend components of your application, such as databases or application servers. Utilize VMSS for automatic scaling of VM instances based on workload demand.

Implement Azure App Service: Deploy the frontend of your web application using Azure App Service, a fully managed platform for building, deploying, and scaling web apps.

Utilize Azure Key Vault: Securely store and manage sensitive application settings, connection strings, and cryptographic keys using Azure Key Vault. Integrate Key Vault with your application to retrieve secrets at runtime securely.

Implement Azure DNS: Configure Azure DNS to map custom domain names to your web application's endpoints, enabling users to access your application using a user-friendly domain name.

Optimize Storage and Costing: Choose appropriate Azure storage services (e.g., Azure Blob Storage, Azure SQL Database) based on your application's data storage needs. Optimize storage configurations to balance performance, scalability, and cost efficiency.

Testing and Monitoring: Test your deployed application to ensure functionality and performance meet expectations. Set up Azure Monitor to monitor the health, performance, and availability of your application resources. Utilize Azure Application Insights for application performance monitoring and diagnostics.

Documentation and Presentation: Document the deployment process, configuration settings, and integration details of each Azure service used in your application. Prepare a presentation to showcase your deployed web application, highlighting the Azure services utilized and the benefits they provide in terms of scalability, security, and cost-effectiveness.