Multicloud Architecture Document

Multicloud Architecture

1. Component Breakdown cloud Service Purpose Component Frontend AWS BackendAPI AWS   S3 + CloudFront Serves static web assets (React/HTML/CSS) to endusers storage AWS S3 EC2 (or Lambda + API Gateway) Hosts application logic, orchestrates requests Persists usergenerated files (images, docs) Database Azure AzureSQLDatabase Houses structured data such as user profiles, orders Monitoring Azure AzureMonitor / ApplicationInsights Centralized logging, tracing, performance metrics

2. Datas Control Flow Usex Frontend The browser requests the React/HTML bundle from AWSS3/cloudFront Frontend BackendAPI Frontend makes HTTPS calls to the AWSBackendAPI for business actions (e.g., POST/order_).BackendAPI storage For file uploads the API stores objects in AWSs3 and returns the file URL to the client.BackendAPI Database CRUD queries are executed against AzureSQL Database over a secure comection. The API may cache reads and sanitise inputs to avoid SQLinjection. BackendAPI Monitoring Logs, traces, and custom metrics are streamed to AzureMonitor/ApplicationInsights for observability across both clouds.

3 Security& GovernanceTransport Security:All traffic (client frontend, frontend backend, backend Azure SQL) is encrypted via TLS Identity & Access:AWSresources use IAM roles with leastprivilege policies. AzureSQL uses managed identities AAD accounts for auth. Network Controls:M - BackendAPI SG/NACL only allows outbound to AzureSQLs public endpoint or via VPN/PrivateLink. - AzureSQL firewall limits inbound IPS to the AWSVPCCIDR. Secrets Management: store DB credentials or comection strings in AWSSecretsManager or AzureKeyVault; never hardcode. Compliance: Centralized logging in Azure Monitor helps meet audit requirements (e.g., ISO27001, soc2). How to Use This Document. Place multi_cloud_architecture.md and multi_cloud_diagram.png in the same repo folder. GitHub will render the Markdown with the embedded diagram automatically. Feel free to expand the explanations or swap services to fit your exact usecase.