RELIABLE WEB APP PATTERNS FOR .NET

Prasanth Balakrishnan

AGENDA

Introduction

Key Objectives

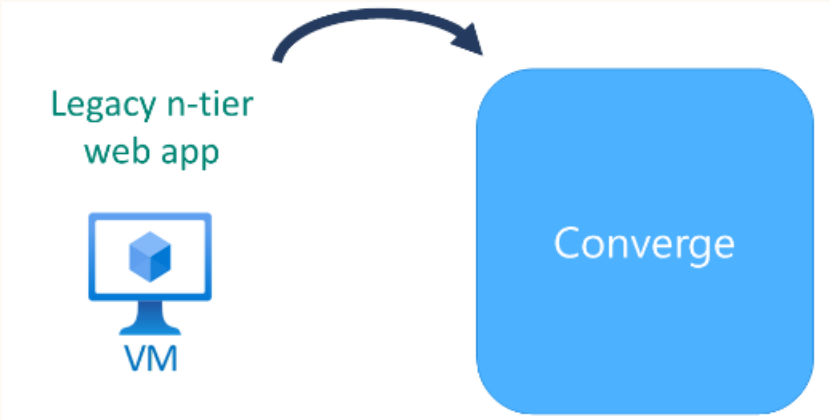
Main Pillars

Implementation

Q & A

INTRODUCTION

RWA is a set of best practices built on the Azure Well-Architected Framework that helps developers successfully migrate to the cloud and set a foundation for future modernization in Azure.



KEY OBJECTIVES

- Set of principles to help developers successfully migrate web applications to the cloud.
- Follow Azure Well-Architected Framework principles
- Low-cost, high-value wins
- Minimal code changes to:
    - Meet security best practices
    - Apply reliability design patterns
    - Improve operational excellence
- Cost-optimized environment(s)
- Business-driven service level objective

MAIN PILLARS

RELIABILITY

Application should meet commitments made to the customers

SECURITY

Application should maintain confidentiality and integrity

COST OPTIMIZATION

Application achieves business goals staying within budget.

OPERATIONAL EXCELLENCE

Undertake processes to keep the

PERFORMANCE EFFICIENCY

Application should scale to meet user demands in an efficient manner.

RELIABILITY

- Key Features
  - Resilience
    - Auto-recover from failures
    - Application should continue functioning after failure
  - Availability
    - Workloads should be accessible
- RWA Recommendations
  - Design for Failure
    - Retry Pattern
    - Circuit Breaker Pattern

SECURITY

- Key Features
  - Confidentiality
  - Integrity
  - Availability
- RWA Recommendations
  - Managed Identity
  - Identity based authentication
  - Secret Management
  - Endpoint Security

COST OPTIMIZATION

- Key Features
  - Balance business goals with budget
  - Reduce unnecessary expenses
- RWA Recommendations
  - Choose correct resources
  - Optimize workloads
  - Dynamically allocate and deallocate resources

OPERATIONAL EXCELLENCE

- Key Features
  - Processes that keep an application running in production
  - Deployments must be reliable and predictable
- RWA Recommendations
  - Automated deployments
  - Repeated Infrastructure
  - Monitoring

PERFORMANCE EFFICIENCY

- Key Features
  - Ability of workloads to scale to meet user demands efficiently
- RWA Recommendations
  - Design for Scaling
  - Cache Aside Pattern
  - CQRS Pattern

PERFORMANCE EFFICIENCY

- Key Features
  - Ability of workloads to scale to meet user demands efficiently
- RWA Recommendations
  - Design for Scaling
  - Cache Aside Pattern
  - CQRS Pattern

Q & A

REFERENCES

- Microsoft Learn – Reliable Web App Pattern for .NET
- Microsoft Learn – Azure Well-Architected Framework
- Microsoft Learn Assessments – Azure Well-Architected Review
- GitHub – Reference Implementation of Reliable Web App Pattern for .NET
- YouTube videos – The Reliable Web App Pattern for .NET

THANK YOU

Prasanth Balakrishnan
iam.prasanth@gmail.com