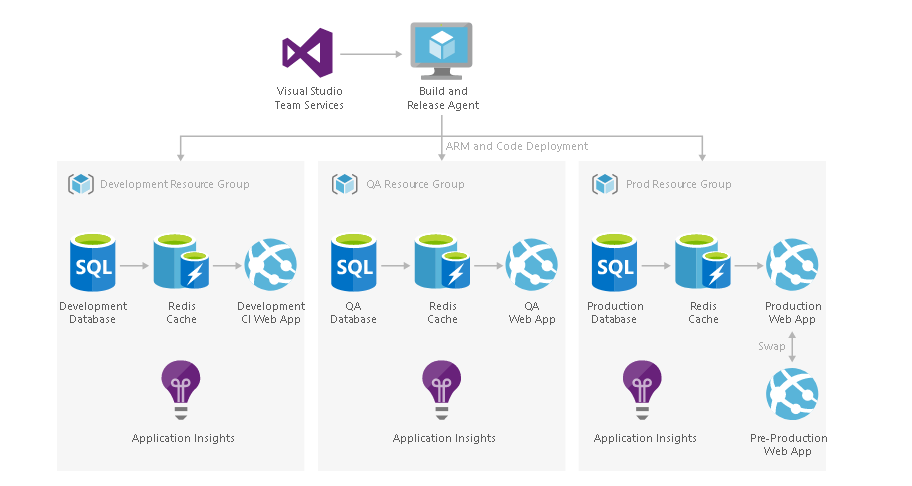
Para este cenário acredito que uma arquitetura voltada a cloud e serviços seria ideal para suprir as necessidades de crescimento e expansão previstas.

Podemos didivir em 2 partes a arquitetura.

A primeira parte sera responsável pelos novos canais:



Essa arquitetura representa uma infraestrutura para desenvolvimento e teste de um sistema estilo PaaS padrão.

Essa solução é criada nos serviços gerenciados do Azure: Azure DevOps, Banco de Dados SQL do Azure, Cache do Azure para Redis e a Application Insights. Esses serviços são executados em um ambiente de alta disponibilidade.

Um aplicativo Web executa o site e é implantado em todos os ambientes.

Com alguns recursos do Azure como os Builds automaticos podemos garantir o funcionamento das aplicações em todos os 3 ambientes(Desenvolvimento, QA e Produção).

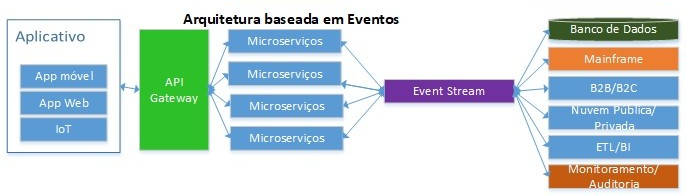
O Banco de Dados SQL do Azure mantém dados para o site. As cópias são implantadas nos ambientes de desenvolvimento, teste e produção.

O Cache Redis do Azure é usado em cada ambiente para melhorar o desempenho do site.

O Application Insights monitora o aplicativo Web durante o desenvolvimento e execuções de teste e, em seguida, monitora o sistema de produção completo quando ele é lançado.

A segunda parte é nas integrações com os sistemas legados que acredito que uma arquitetura baseada em eventos atenderia perfeitamente o cenário.

As principais vantagens desta arquitetura é a simplificação da integração entre sistemas, seja sistemas legados ou sistemas distribuídos e microserviços. A combinação de banco de dados transacionais com sistema baseados em eventos permitem maior desacomplamento entre os componentes, maior distribuição e escalonamento de carga de processamento, resultando em minimização de recurso/custo de infraestrutura. Os eventos evitam a perda de informação e facilitam a auditoria e monitoramento.



Podemos considerar algumas ferramentas para apoio no gerenciamento destes eventos, tais como o Kafka, Azure Service Bus, e entre outros. Em relação a arquitetura de código, podemos considerar a utilização de CQRS, banco de dados NoSQL, Bus Service, Message Queue.

A comunicação entre os sistemas dos novos canais e com os microserviços dos legados via a API de Gateway mostrada no desenho acima.