



Atividade Somativa 2

Nome: Vanessa Milani Ratusznei

Curso: Análise e Desenvolvimento de Sistemas

Link Vídeo: <https://www.youtube.com/watch?v=7FLsQvyw6-8>

Sistema de Matrícula Online da PUCPR: Implementação em Nuvem Azure

Resumo dos Principais Pontos:

Arquitetura e Recursos Essenciais

O desenvolvimento do sistema de matrícula online da PUCPR baseou-se na utilização de recursos de computação em nuvem essenciais para garantir escalabilidade, segurança e disponibilidade da aplicação. Os principais componentes incluem máquinas virtuais (VMs) para hospedar o frontend e o backend da aplicação, Azure SQL Database para o armazenamento das informações de usuários, matrículas, pagamentos e documentos, Azure Functions para a execução de tarefas assíncronas e automatizadas como validações e envio de notificações, Application Gateway com Web Application Firewall (WAF) para controle de tráfego e segurança contra ataques web, e Azure Gerenciamento de Custos com regras de alerta para acompanhar a performance da aplicação e prevenir falhas.

O dimensionamento da arquitetura foi planejado de forma progressiva, iniciando com uma estrutura simples usando VMs de pequeno porte da série B1s, adequadas para ambientes de desenvolvimento e testes. O banco de dados utiliza o plano Basic do Azure SQL, suficiente para simulações com menor volume de dados, enquanto as Azure Functions operam no plano gratuito ou de consumo, garantindo elasticidade e economia. O uso de Application Gateway e alertas monitorados desde o início do projeto reforça a confiabilidade da aplicação, permitindo ajustes conforme o sistema evolui e ganha mais usuários.

Implementação Prática e Desafios

Durante a execução prática do projeto, foram provisionados os principais recursos de nuvem necessários. Duas máquinas virtuais foram criadas, uma para o frontend e outra para o backend, além do Azure SQL Database para armazenar dados relacionados ao sistema. No entanto, instabilidades na conta de estudante impediram a configuração da integração completa entre os recursos. O ambiente apresentou falhas no carregamento das opções ao tentar criar Azure Functions, afetando a automatização de tarefas como validação de documentos ou envio de notificações.

As VMs e o banco de dados ainda não foram completamente integrados, dependendo de ajustes adicionais como configuração de strings de conexão, regras de firewall e permissões de acesso na rede virtual. Apesar desses desafios técnicos, o exercício foi importante para compreender o processo de provisionamento dos recursos e identificar pontos críticos de uma arquitetura em nuvem. O controle rigoroso dos recursos foi



mantido para evitar consumo desnecessário, priorizando instâncias menores e serviços gratuitos ou de baixo custo.

Estratégia de Monitoramento e Gerenciamento

Para o gerenciamento e monitoramento do projeto, foi adotada uma estratégia preventiva baseada em alertas críticos sobre uso de recursos e disponibilidade dos serviços principais. Mesmo com as instabilidades na criação de Azure Functions, o monitoramento nas máquinas virtuais foi configurado com alertas específicos: uso de CPU acima de 80% nas VMs de Backend e Frontend, indicando sobrecarga e risco de indisponibilidade; uso de disco acima de 90% na VM Backend, representando risco de falha de escrita; falha na execução de Azure Functions para identificar problemas em validações e envio de e-mails; e indisponibilidade da VM por status Stopped/Deallocated.

A estratégia de gerenciamento considera que os períodos de matrícula são altamente sazonais, concentrando-se no início e fim de semestre. Os alertas de desempenho permitem detectar picos de uso e reagir rapidamente, evitando impacto nos alunos. O ambiente foi estruturado inicialmente com recursos de desenvolvimento, mas a criação de alertas já nessa etapa é essencial para corrigir falhas antecipadamente e preparar o ambiente para futuras transições.

Segurança e Proteção em Camadas

Para estruturar a segurança dos recursos e aplicações, foi aplicada uma abordagem baseada na proteção em camadas, utilizando recursos nativos da Azure para controle de tráfego, filtragem e monitoramento de ameaças. A principal medida implementada foi a criação de uma política global de Web Application Firewall (WAF), configurada em modo de prevenção, que identifica e bloqueia automaticamente padrões de ataques como injeções SQL e cross-site scripting.

Esta política foi vinculada a um Application Gateway, atuando como ponto de entrada centralizado para o tráfego HTTP/HTTPS das aplicações. Todo o tráfego passa pelo Gateway com WAF, que inspeciona e aplica regras de segurança em tempo real, enquanto o tráfego legítimo segue para as VMs protegidas por NSGs (Network Security Groups) com regras específicas. Pensando no contexto institucional, a proteção de dados sensíveis dos alunos justifica essas medidas mesmo em ambientes de teste, simulando situações reais de segurança.

Redundância, Escalabilidade e Recuperação de Desastres

Diante da criticidade do sistema e da ampliação de escopo que aumentará em até 10 vezes a demanda, foram adotadas estratégias voltadas à redundância, escalabilidade e recuperação de desastres para assegurar alta disponibilidade. Para garantir redundância, foram utilizados Virtual Machine Scale Sets (VMSS), que permitem a criação automática de instâncias adicionais com base em imagens personalizadas da VM principal. Essa abordagem assegura que, em caso de falha, o sistema possa ser restaurado rapidamente com as mesmas configurações, eliminando pontos únicos de falha.

A escalabilidade horizontal é garantida pelo uso de VMSS com autoscale configurado por métricas como uso de CPU, permitindo que o número de instâncias aumente ou diminua conforme a demanda, otimizando



desempenho e custos. Os backups são habilitados diretamente nos serviços utilizados: o Azure SQL Database conta com backup automático garantindo recuperação de dados por até 7 dias, enquanto imagens personalizadas das VMs podem ser restauradas rapidamente em caso de falha.

O plano de backup e disaster recovery (DR) inclui etapas como identificação da falha, acionamento da equipe responsável, restauração da VM a partir da imagem personalizada, verificação dos serviços críticos, testes de validação e retorno à operação. Durante indisponibilidades, ações de contingência como comunicação proativa com usuários e procedimentos manuais de validação de matrícula são ativadas para reduzir o impacto. O plano é testado em ambiente paralelo, simulando falhas sem comprometer a produção.

Considerações Finais

A abordagem adotada equilibra investimento em uma arquitetura segura e escalável dentro dos limites orçamentários disponíveis, utilizando serviços gerenciados que facilitam a manutenção, aproveitando planos gratuitos quando possível e mantendo uma estrutura modular que permite crescimento progressivo. O gerenciamento de todo o ecossistema segue uma abordagem modular e automatizada, com monitoramento constante por meio do Azure Gerenciamento de Custos e alertas bem definidos, permitindo controle de custos enquanto mantém a arquitetura preparada para crescer conforme necessário.

Esta experiência reforça a importância do planejamento e da modularização da arquitetura, permitindo avanços progressivos mesmo diante de limitações técnicas ou orçamentárias, enquanto oferece uma visão prática e integrada da computação em nuvem para compreender como os diferentes componentes trabalham em conjunto para manter o sistema funcional, seguro e escalável.

Durante a elaboração deste trabalho acadêmico (Atividade Somativa 2), a autora recorreu à ferramenta ChatGPT, versão GPT-4 (OpenAI), com o objetivo de aprimorar a redação e a clareza textual. A autora revisou, validou e adaptou todo o conteúdo gerado, assumindo integral responsabilidade pelas informações apresentadas.

Para mais detalhes da arquitetura confira as Respostas das Atividades Formativas:

Respostas da semana 3:

Com base no fluxo macro do sistema de matrícula online da PUCPR, alguns recursos de computação em nuvem se tornam essenciais para garantir a escalabilidade, a segurança e a disponibilidade da aplicação. Entre os principais, destacam-se: máquinas virtuais (VMs) para hospedar o frontend e o backend da aplicação; Azure SQL Database para o armazenamento das informações de usuários, matrículas, pagamentos e documentos; Azure Functions para a execução de tarefas assíncronas e automatizadas, como validações e envio de notificações; Application Gateway com Web Application Firewall (WAF) para controle



de tráfego e segurança contra ataques web; e Azure Gerenciamento de Custos com regras de alerta para acompanhar a performance da aplicação e prevenir falhas.

Para dimensionar a arquitetura, é possível iniciar com uma estrutura simples, usando VMs de pequeno porte, como as da série B1s, que atendem bem a ambientes de desenvolvimento e testes. O banco de dados pode utilizar o plano Basic do Azure SQL, suficiente para simulações com menor volume de dados. As Azure Functions podem operar no plano gratuito ou de consumo, garantindo elasticidade e economia. O uso de Application Gateway e alertas monitorados já no início do projeto reforça a confiabilidade da aplicação, permitindo ajustes conforme o sistema evolui e ganha mais usuários.

A escolha desses recursos se justifica pelo alinhamento com os temas abordados na disciplina e pela possibilidade de configurar uma arquitetura que represente ambientes reais, sem extrapolar os limites de uma conta de estudante. Além disso, essa abordagem oferece uma visão prática e integrada da computação em nuvem, permitindo que o aluno compreenda como os diferentes componentes trabalham em conjunto para manter o sistema funcional, seguro e escalável.

Embora a arquitetura robusta seja desejável em produção, não é necessário implementá-la completamente desde o início. É mais viável começar com uma versão simplificada, validando o funcionamento do fluxo com poucos usuários. Essa estratégia reduz custos e permite que o sistema seja testado e aprimorado antes de escalar. Ao longo do tempo, com base no uso e no desempenho real, a infraestrutura pode ser expandida de forma planejada.

Não há, até o momento, um projeto idêntico institucionalizado que sirva de referência direta. Por isso, recomenda-se a criação de um projeto-piloto, com escopo limitado — como a matrícula de uma única turma ou curso — para validar a arquitetura proposta, levantar feedbacks e antecipar ajustes técnicos antes da implementação em maior escala.

Considerando a importância estratégica do sistema de matrícula para a universidade, o ideal é adotar uma abordagem equilibrada: investir desde o início em uma arquitetura que seja segura e escalável, mas dentro dos limites orçamentários disponíveis. Isso significa escolher serviços gerenciados que facilitem a manutenção, aproveitar planos gratuitos sempre que possível, e manter uma estrutura modular, que permita crescimento progressivo. O planejamento consciente dos recursos é fundamental para garantir o sucesso do projeto sem comprometer a eficiência financeira.

Respostas da semana 4:

Durante a execução prática do projeto de matrícula online na Azure, foram provisionados os principais recursos de nuvem necessários, de acordo com os conhecimentos adquiridos até o momento na disciplina. Foram criadas duas máquinas virtuais (VMs), uma para o frontend e outra para o backend, além do Azure SQL Database, que será responsável por armazenar os dados relacionados ao sistema, como informações dos estudantes, pagamentos e status das matrículas.



No entanto, devido a algumas instabilidades na conta de estudante, não foi possível configurar a integração completa entre esses recursos. O ambiente apresentou falhas no carregamento das opções ao tentar criar Azure Functions, impossibilitando a automatização de tarefas como validação de documentos ou envio de notificações. Essa limitação afetou diretamente a finalização do fluxo completo da aplicação.

Além disso, até o momento, as VMs e o banco de dados ainda não foram integrados. A comunicação entre os componentes depende de ajustes adicionais, como configuração correta de strings de conexão, regras de firewall e permissões de acesso na rede virtual — pontos que estão em andamento e serão retomados assim que a instabilidade for resolvida.

Apesar desses desafios técnicos, o exercício foi importante para compreender o processo de provisionamento dos recursos e identificar, na prática, os pontos críticos de uma arquitetura em nuvem. A aplicação ainda não está sendo executada com sucesso, mas os próximos passos incluem finalizar a integração entre os serviços e testar a funcionalidade completa do sistema.

É importante destacar a necessidade de atenção ao provisionamento, especialmente em contas com créditos limitados. Foi mantido o controle dos recursos para evitar o consumo desnecessário, priorizando instâncias menores e serviços gratuitos ou com baixo custo.

Essa experiência reforça a importância do planejamento e da modularização da arquitetura, permitindo avanços progressivos mesmo diante de limitações técnicas ou orçamentárias.

Respostas da semana 5:

Para o gerenciamento e monitoramento do meu projeto de matrícula online na Azure, optei por adotar uma estratégia preventiva, baseada em alertas críticos sobre uso de recursos e disponibilidade dos serviços principais. Mesmo com instabilidades na criação de Azure Functions, configurei o monitoramento nas máquinas virtuais e planejei os alertas que seriam aplicados também nas funções, assim que possível.

Alertas configurados:

- Alerta de uso de CPU acima de 80% (VM Backend e Frontend)
 - Motivo: Indica sobrecarga no servidor e risco de indisponibilidade.
 - Plano de ação: Analisar processos ativos na VM, escalar recursos temporariamente ou otimizar o código/processos.
 - Tempo de resposta: Imediato (alerta crítico).
- Alerta de uso de disco acima de 90% (VM Backend)
 - Motivo: Risco de falha de escrita no disco, o que compromete o sistema.
 - Plano de ação: Verificar arquivos temporários/logs e liberar espaço, ou expandir o disco.
 - Tempo de resposta: Até 30 minutos.



- Alerta de falha na execução de Azure Function (quando disponível)
 - Motivo: Identifica funções com falha, como envio de e-mail ou validação de documentos.
 - Plano de ação: Verificar logs no Application Insights, corrigir código ou reprocessar a execução.
 - Tempo de resposta: Até 1 hora (alerta de média criticidade).
- Alerta de indisponibilidade da VM (status Stopped/Deallocated)
 - Motivo: Pode indicar erro ou interrupção acidental.
 - Plano de ação: Investigar o motivo do desligamento e reiniciar a VM.
 - Tempo de resposta: Imediato.

Estratégia de gerenciamento do ambiente:

A estratégia adotada foi monitorar indicadores-chave de infraestrutura nas VMs (CPU, disco, disponibilidade) e, futuramente, indicadores de falha lógica nas Azure Functions. Como os períodos de matrícula são altamente sazonais (início e fim de semestre), os alertas de desempenho permitem detectar picos de uso e reagir rapidamente, evitando impacto nos alunos.

O ambiente foi estruturado inicialmente apenas com recursos de desenvolvimento, pois o projeto ainda está em fase de implantação e testes. A criação de alertas já nessa etapa é essencial para corrigir falhas antecipadamente e preparar o ambiente para futuras transições para homologação e produção.

Essa estratégia foi adotada porque permite visibilidade completa da saúde do sistema, além de facilitar ações rápidas e eficazes, mesmo com infraestrutura limitada, como é o caso de uma conta de estudante.

Respostas da semana 6:

Para estruturar a segurança dos recursos e aplicações no projeto de matrícula online, optei por aplicar uma abordagem baseada na proteção em camadas, utilizando os recursos nativos da Azure para controle de tráfego, filtragem e monitoramento de ameaças.

A principal medida implementada foi a criação de uma política global de Web Application Firewall (WAF), configurada em modo de prevenção. Isso significa que, além de identificar padrões de ataques (como injeções SQL, cross-site scripting, entre outros), o WAF atua de forma ativa bloqueando essas requisições maliciosas automaticamente, sem depender de intervenção manual. Essa política foi vinculada a um Application Gateway, que passa a atuar como ponto de entrada centralizado para o tráfego HTTP/HTTPS das aplicações.

Essa configuração garante que:

- Todo o tráfego passe pelo Gateway com WAF, que inspeciona e aplica as regras de segurança em tempo real.



- O modo de prevenção bloqueia tentativas de ataque antes que elas atinjam a aplicação hospedada nas VMs.
- O tráfego legítimo segue para as VMs, que estão protegidas por NSGs (Network Security Groups) com regras específicas, restringindo portas e IPs.

Mesmo sem explorar conceitos mais avançados como criptografia de dados em trânsito ou identidade gerenciada, essa estratégia já representa uma arquitetura robusta e alinhada com boas práticas de segurança em nuvem.

Pensando no contexto institucional, a proteção de dados sensíveis dos alunos é uma prioridade, e medidas como o WAF em modo de prevenção são justificáveis mesmo em ambientes de teste, já que simulam situações reais de segurança. Embora o uso do Application Gateway com WAF represente um custo adicional em relação a arquiteturas mais simples, ele é um investimento de longo prazo, voltado à disponibilidade e integridade dos dados, podendo ser implementado em fases conforme o ambiente evolui.

Em um primeiro momento, foi possível realizar essa implementação sem comprometer os créditos da conta educacional, e ajustes futuros podem ser feitos para otimizar a relação entre custo e benefício, conforme o projeto avança para fases de homologação ou produção.

Respostas da semana 7:

Diante da criticidade do sistema de matrícula online da PUCPR e da ampliação de escopo anunciada — que aumentará em até 10 vezes a demanda — foi fundamental adotar estratégias voltadas à redundância, escalabilidade e recuperação de desastres, assegurando a alta disponibilidade da aplicação.

Para garantir a redundância, optei por utilizar Virtual Machine Scale Sets (VMSS), que permitem a criação automática de instâncias adicionais com base em imagens personalizadas da VM principal. Essa abordagem assegura que, em caso de falha, o sistema possa ser restaurado rapidamente com as mesmas configurações. A escolha por esse modelo se justifica pela sua capacidade de responder de forma automática a variações de carga e por eliminar pontos únicos de falha.

Com a expansão do público-alvo, a aplicação também exige escalabilidade horizontal, o que faz com que o uso de VMSS com autoscale configurado por métricas (como uso de CPU) seja a opção mais adequada. Isso permitirá que o número de instâncias aumente ou diminua conforme a demanda, otimizando o desempenho e os custos.

Quanto aos backups, eles serão habilitados diretamente nos serviços utilizados. O Azure SQL Database já conta com backup automático, o que garante recuperação de dados até por 7 dias em planos básicos. Para as VMs, serão criadas imagens personalizadas, que podem ser restauradas rapidamente em caso de falha, garantindo a integridade das configurações da aplicação. Esses backups serão testados periodicamente para garantir que estejam íntegros e utilizáveis.



O plano de backup e disaster recovery (DR) segue um fluxo simplificado, mas eficaz. Ele inclui etapas como: identificação da falha, acionamento da equipe responsável, restauração da VM a partir da imagem personalizada, verificação dos serviços críticos (como banco e aplicação), testes de validação e retorno à operação. Durante a indisponibilidade, ações de contingência como comunicação proativa com os usuários e execução de procedimentos manuais (como validação de matrícula por e-mail) serão ativadas para reduzir o impacto. Além disso, o plano será testado em ambiente paralelo, simulando falhas sem comprometer o ambiente de produção — o que garante que todos saibam como agir em uma situação real.

Para gerenciar todo esse ecossistema com eficiência e controle de custos, será adotada uma abordagem modular e automatizada, com monitoramento constante por meio do Azure Gerenciamento de Custos e alertas bem definidos. O uso de planos básicos, escalonamento sob demanda e o reaproveitamento de imagens e scripts ajudam a manter o orçamento controlado. Ao mesmo tempo, a arquitetura permanece preparada para crescer conforme necessário, sem desperdiçar recursos.