

CovidShot

Avaliação Arquitetural

1. Introdução

O propósito deste documento é apresentar a avaliação da arquitetura do software CovidShot, que determina se ela é adequada para o objetivo com que foi projetada. A avaliação foi realizada de acordo com atributos de qualidade previamente definidos.

2. Atributos de Qualidade

2.1. Adequação Funcional

Quanto ao ponto da adequação funcional, deve-se dizer que mediante a separação de funcionalidades no servidor e em suas camadas, a arquitetura CovidShot se adapta as necessidades e funcionalidades essenciais do sistema. Dessa maneira, ao se garantir que tal fator possibilita ainda mais a disponibilidade, a partir do Cliente Servidor, pode-se afirmar que tal característica é encontrada na arquitetura do CovidShot.

2.2. Eficiência de Desempenho

Quanto ao modelo Client-Server e o padrão REST, pode-se dizer que há uma garantia de uma eficiência natural em velocidade, contudo a adequação do sistema ao estilo Em camadas, gera uma visão oposta, já que as requisições precisaram passar por um caminho maior até seu processamento, logo a arquitetura garante uma estabilidade em desempenho, balanceando as desvantagens de um estilo, com as vantagens de outro.

No que tange a persistência, a arquitetura foi moldada para a agilidade em requisições, graças ao Cliente-Servidor, seja pelo tráfego volumoso de dados em bancos não relacionais. Tem-se também a divisão de três fontes de armazenamento para o sistema, adequadas para garantir a velocidade, em um cenário de carga de chamadas.

Por fim, a arquitetura ainda contempla o paralelismo através de mensageria pelas Azure Functions, as quais executam processos longos de traduções em paralelo,

enquanto o sistema permite a execução de outras tarefas, logo tal fator propicia a questão do desempenho.

2.3. Compatibilidade

A arquitetura de software do CovidShot atende a compatibilidade do sistema, visto que sua execução é independente do sistema ou do hardware hospedado, já que sua modelagem permite hospedagem em ambientes Windows ou Linux, até mesmo em contêineres.

Devido à quebra de responsabilidade, os componentes do sistema podem trocar mensagens e serviços com quaisquer outros sistemas, já que suas implementações são protegidas por contratos de interface, por meio do domínio da aplicação. Ou seja, com tal nível de abstração a compatibilidade depende apenas da expansão e não da mudança.

Além disso, mediante a divisão em servidores e nas camadas, há um esforço extra no quesito intercomunicação e, portanto, a compatibilidade com outros sistemas torna-se mais fácil, além de que com a divisão em camadas garante-se que a comunicação só precisa ser realizada com a camada de aplicação.

2.4. Usabilidade

A usabilidade abordada pela arquitetura, além dos tópicos principais de usabilidade apresentados na interface do usuário do sistema, que permite a execução de diversas atividades simultâneas, com diversos clientes, gerando uma eficiência comprovada para o CovidShot.

2.5. Confiabilidade

A arquitetura definida estabelece uma disponibilidade de 99.8%, visto que toda sua infraestrutura será hospedada dentro do Azure, que garante por contrato essa margem de erro de 0.2% de falhas, que serão tratadas pela arquitetura em caso de falha, como a hospedagem em outro ambiente, em outra região, com outro país.

A tolerância à falha é englobada pela arquitetura que garante o tratamento de toda exceção gerada no domínio, e informada dentro de logs do Azure, para que as decisões para problemas extremos sejam tomadas. E por fim, a recuperação de

dados está englobada em backups hospedados no Azure, em diversas regiões, prevenindo desastres.

Além disso, a estruturação em cliente-servidor possibilita-se que sejam separados as funções do sistema de modo a estar sempre disponível as requisições do cliente, além disso, com o uso da separação em camadas, a arquitetura do CovidShot possibilita a verificação de falhas mais facilmente.

2.6. Segurança

Com a separação da arquitetura em diversos fragmentos há a possibilidade de adicionar níveis de segurança diferentes entre as camadas. O sistema possui os estilos REST e Em Camadas que facilitam a segurança do sistema, gerando apenas requisições autenticadas e diversas camadas de verificação e validação de usuário e permissão.

Além disso, os dados sensíveis do usuário serão armazenados no banco de dados SQL do Azure, que possui ao nível de implementação e serviço uma proteção contra SQL Injections e validação de localização dos usuários. Por fim, a criptografia dos dados sensíveis que só serão descriptografados no momento de leitura do sistema, pela interface.

2.7. Manutenibilidade

A arquitetura divide o sistema em camadas, que possuem suas próprias responsabilidades e desconhecem as demais camadas. Sua comunicação depende apenas de contrato de interfaces, ou seja, o acoplamento é isolado. Logo, a manutenibilidade é garantida apenas em um ponto e, além disso, dentro de cada camada, sua responsabilidade isolada, permite que cada funcionalidade realize apenas uma operação.

2.8. Portabilidade

O sistema possui sua portabilidade já que, devido a sua linguagem escolhida, após a compilação, sua execução pode ser feita em qualquer máquina em cloud ou on-premise.

3. Conclusão

Conclui-se então que a arquitetura desenvolvida para o sistema CovidShot foi avaliada e está de acordo com o objetivo para o qual ela foi publicada. Nesse sentido, a avaliação foi realizada de acordo com atributos de qualidade definidos: adequação funcional, eficiência de desempenho, compatibilidade, usabilidade, confiabilidade, segurança, manutenibilidade, portabilidade e, a partir das análises, a arquitetura se mostrou satisfatória e completa para o que havia sido proposto e esperado.