

Atividades Práticas

Parte 1

Objetivo:

Nossa intenção é praticar os conceitos de computação distribuída através de um modelo arquitetural orientado a serviços. Para isso será necessário criar um protótipo que realize as funcionalidades CRUDs (Create, Read, Update e Delete) de um problema já conhecido.

Etapa 1:

Escolha um problema ou caso de uso que vocês já implementaram em outras disciplinas. Esse caso de uso/problema, tem que apresentar pelo menos **cinco tabelas**. Também é necessário que seu caso de uso/problema apresente pelo **menos três validações de regras de negócio** (Exemplo: Validação de usuário para permissão de update de dado em uma tabela). É importante que você produza/Informe a documentação deste caso de uso.

Etapa 2:

Desenvolva a modelagem de dados desse caso de uso. Lembre-se, o objetivo desta disciplina é praticar conceitos de SD, por isso é importante você economizar trabalho reutilizando atividades realizadas em outras disciplinas.

Etapa 3:

Desenvolver os serviços do CRUD e das regras de negócio. Esses serviços deverão ser capazes de fazer as alterações em banco de dados, se for o caso, e fornecer interface para que o cliente possa consumi-los. Neste momento é que você escolherá uma api para webservice. Você pode usar tanto SOAP como REST, mas.... aconselho fortemente a usar o REST.

Etapa 4:

Desenvolver o cliente que realizará a agregação e consumo dos serviços. Esse cliente pode ser uma aplicação desktop, app mobile ou até mesmo um sistema web que gera a interface web para o usuário.

Observações importantes:

Você pode usar a plataforma ou a linguagem que você quiser, mas lembre-se, a próxima etapa será colocar esses serviços em contêineres (Docker, Kubernetes etc.), então escolha sabiamente antes que tenha que refazer todo o projeto para a próxima etapa. Pesquise qual plataforma/linguagem é mais apropriada (fácil) para fazer isso.

Trabalho será de grupo de até 3 alunos.

Documentação para apresentar:

- código fonte do projeto (pode ser link do github, dando autorização ao professor para acessá-lo, se não for público)

- Documentação explicando passo-a-passo como instalar e rodar o projeto e seus requisitos (também pode estar no github)
- Um vídeo de até 15 min demonstrado o funcionamento do projeto (link youtube)(TODOS os componentes do grupo devem apresentar o vídeo)

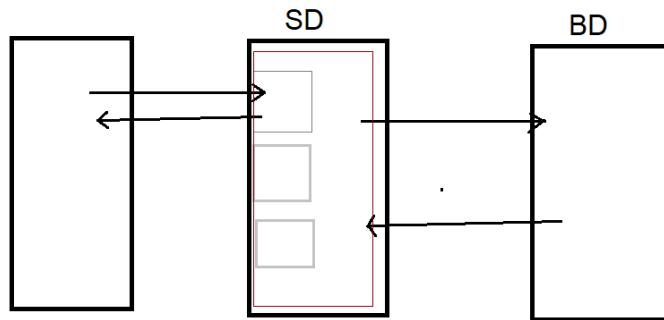
Entrega: 04/06

Atualização:

- a) Conforme combinado, será permitido que a interface do usuário (a que consumirá seus serviços) seja feita através de aplicação de terceiros, como postman, extensões de navegadores etc.(até 3 integrantes)
- b) Caso o grupo decida criar uma página personalizada para o trabalho com interface de usuário para consumir o serviço, então será permitido 4 integrantes no grupo.

Parte 2

Agora você precisa ajustar seu trabalho de forma que você monte a arquitetura de seu SD semelhante à figura abaixo:



Você usará uma arquitetura em camadas separando o BD, interface e regras de negócio em contêineres diferentes. No caso da interface, se vc estiver utilizando uma ferramenta de terceiro, o contêiner é dispensável.

Você tem a opção de utilizar mais contêineres se quiser ou achar apropriado. Também pode utilizar qualquer variação de arquitetura desde que mantenha essa separação de sistemas, bd e interface(se for o caso), como por exemplo, arquitetura de microservices.

Independente da escolha do grupo, deverá ser feita documentação no github junto ao código fonte do sistema, uma página read.me onde tenha o link dos contêineres para download e instrução de como baixar, inicializar, instanciar contêineres, etc, para uso por um leigo.

Com essas alterações a entrega passa para o dia 18/06.

Serão dadas duas notas

Avaliação 1(para nota da primeira entrega): Documentação e verificação aplicação das técnicas estudadas que serão apresentadas no Github.

Avaliação 2(para nota da segunda entrega): Funcionamento das aplicações em testes e a apresentação do vídeo.