

DESCRIÇÃO DO TRABALHO PRÁTICO

(esse documento pode sofrer alterações, acesse através do Google Classroom para garantir que está com a versão mais recente)

Objetivo

- Desenvolvimento de um serviço disponível via web (REST API).
- Linguagem/framework de livre escolha.
- Front-end opcional.

Equipes

- Mínimo de 3 integrantes.
- Escopo do trabalho será proporcional ao tamanho da equipe.
- Após a Entrega 1, as equipes não podem mais serem modificadas.

Especificação do trabalho

- Cada equipe tem a liberdade para escolher os requisitos da aplicação e detalhá-los como achar melhor.
 - Ideias aqui <https://www.programmableweb.com/apis/directory>
 - Não será permitido utilizar o tema “Cartão de Crédito” ou similar no trabalho
- A equipe precisa especificar os requisitos que seu serviço irá disponibilizar, o escopo de especificação é livre.
 - Exemplo em português de serviço
<https://documenter.getpostman.com/view/4141833/RWTsqyN/>
- As equipes que tiverem dificuldade para definir um escopo de trabalho, devem procurar o professor ANTES da primeira entrega.

Características mínimas (p/ equipes de 3 a 4 pessoas)

- *Obs: são elementos mínimos que o design do serviço deve apresentar.*
- O serviço deve apresentar funcionalidades (requisitos) que envolvam pelo menos 3 entidades (coleções) relacionadas.
 - Deve ser possível inserir, ler e editar instâncias de cada entidade.
 - Pelo menos 1 entidade deve ter uma relação de agregação/composição com outra entidade, e a leitura da entidade principal deve retornar a sub-coleção da outra entidade.
- O serviço deve apresentar funcionalidades que envolvam pelo menos 1 transação/processo/procedimento qualquer relacionado às entidades.
 - Além da funcionalidade transacional, deve haver uma funcionalidade para leitura do histórico e/ou acumulado de transações/processos/procedimento.
- As funcionalidades do serviço devem apresentar, associada a entidades ou transações, pelo menos 2 regras de negócios distintas a serem verificadas.
- A interface deve ser implementada em padrão REST API com respostas em formato JSON. Não precisa implementar autenticação/autorização.
- A documentação da API deve ser clara e detalhada o suficiente para abranger todas as funcionalidades do serviço e atender as características mínimas.
 - Exemplo de documentação
<https://documenter.getpostman.com/view/4141833/RWTsqyN/>
- O serviço deve usar um banco de dados na retaguarda (escolha livre da equipe).

PROJETO DETALHADO DE SOFTWARE
QXD0058 – 2022.1 – PROF. CAMILO ALMENDRA



- O serviço precisa retornar códigos de erro coerentes em caso de acesso a endpoints inexistentes, entidades inexistentes ou operações não permitidas por conta de lógica de negócios.
- O design do serviço deve seguir as melhores práticas de separação de interesses, distribuição de responsabilidades entre componentes e controle de dependências.
- O design do serviço deve utilizar algum padrão de acesso a persistência de dados.

Entregas

- **Entrega 1 – 14/06**
 - Especificação das funcionalidades (requisitos funcionais) do serviço. **Documento em pdf** contendo:
 - Título do projeto (nome do serviço)
 - Lista de membros da equipe (nomes completos)
 - Descrição do sistema (resumo)
 - Lista de requisitos funcionais
 - Especificação do design do BD. **Documento em pdf¹** contendo:
 - Entidade e seus campos com descrição
 - Relacionamentos entre entidades com descrição
 - Implementação completa (API, objetos de negócio e BD) de uma entidade:
 - Inserir nova instância
 - Ler uma instância
 - Editar instância
 - Listar instâncias
- **Entrega 2 – 05/07**
 - Especificação da funcionalidades junto com a documentação do serviço
 - Documentação de todos os endpoints
 - Especificação final do design do BD
 - Implementação de todos os endpoints das entidades e das transações

¹ Pode ser o mesmo documento com a especificação das funcionalidades.