

### **Trabalho final**

**Objetivo:** Desenvolver um Web Services (SOA ou REST) para comunicação entre a aplicação cliente e a base de dados para uma das opções (A ou B) descritas no final deste documento.

#### **Atividades:**

- Definição dos grupos (mínimo dois e máximo três integrantes) e da Opção (A ou B);
- Definição da linguagem/plataforma que será utilizada para desenvolvimento. O objetivo é que tenhamos a maior diversidade possível;
- Desenvolvimento de manual de instalação e configuração da plataforma para desenvolvimento. O objetivo é que possamos divulgar estes manuais e compartilhar conhecimento entre o grupo;
- Desenvolvimento do Web service;
- Entrega do projeto e *dump* da base de dados;
- Apresentação em aula dos serviços desenvolvidos.

#### **Datas importantes:**

**07/06/2019** – Data limite para informar os integrantes, linguagem e opção de trabalho a ser desenvolvida. Para os grupos que escolherem a opção B, esta também é a data limite para entrega do escopo e descrição dos requisitos do sistema.

**14/06/2019** – Data limite para postagem do manual de instalação e configuração do ambiente para desenvolvimento de Web Services para a linguagem escolhida.

**04/07/2019** – Data limite para postagem do projeto e dump da base de dados (arquivo compactado contendo todos os arquivos) no Moodle

**05/07/2019** – Apresentação do projeto em aula

#### **Distribuição da avaliação:**

- Manual de instalação e configuração do ambiente para uso desenvolvimento de Web Services: até 1,5 do valor total;
- CRUD e interfaces gráficas funcionais: até 4,0 do valor total;
- Apresentação após entrega do trabalho (OBRIGATÓRIA): até 1,5 do valor total.

#### **Desafios extras:**

- Cada desafio vale 0,5 pontos extras para N2.
- Somente serão computados os desafios que estiverem completos e funcionais (quando envolverem desenvolvimento).
- Cada grupo poderá escolher até dois desafios para desenvolver, acumulando até 1,0 ponto extra na N2.

**Desafio 1:** utilizar linguagem/plataforma diferente de Java;

**Desafio 2:** incorporar ao trabalho ao menos um consumo de Web Service de terceiro (consulta de cep, receita federal, etc)

**Desafio 3:** utilizar ferramenta de Swagger para documentar o Web Service desenvolvido.

**Desafio 4:** criar uma etapa de autenticação via Web Service (login e senha) antes da tela inicial do sistema.

**Desafio 5:** criar a aplicação cliente em linguagem diferente da utilizada para o backend (serviço).

**No caso de recebimento de cópias de trabalhos, todos os envolvidos receberão automaticamente peso zero na avaliação.**

### Opções de sistema para desenvolvimento

**Opção A:** sistema proposto “Sistema de gerenciamento de cemitério”.

A empresa Cemitério Um Irmão solicitou o desenvolvimento de um sistema para controle interno para permitir o futuro controle da infraestrutura e dos “moradores”. Você foi designado para participar do desenvolvimento do projeto.

Esta parte do escopo deverá permitir o gerenciamento dos jazigos e controle dos pagamentos das taxas de manutenção.

Este sistema deverá contar com um registro de todos os dados referentes a jazigos e taxas a serem pagas em banco de dados.

Requisitos do sistema:

- Cadastro de morador

Nesta opção o usuário cadastra dados do falecido (código, nome, data nascimento, número da certidão de óbito, CPF, data de falecimento, observações, sexo), qual jazigo ele ira ocupar, data do sepultamento, nome e endereço do responsável pelos pagamentos, data para pagamento das taxas e data em que foram realizados o último pagamento (pagamento é anual).

- Cadastro de jazigos

Nesta parte deverá ser realizado o cadastro dos jazigos, quantidade de jazigos, indicação de disponibilidade (ocupado ou não) e valor da taxa de manutenção (valor altera conforme capacidade de cada jazigo). Existem jazigos para um ou dois caixões.

- Pesquisas

Sistema deve permitir pesquisar por nome ou código do sepultado, por jazigo ou por data de falecimento (neste caso usuário digita o número do jazigo e são exibidos os dados do falecido correspondente). Em caso de entrada em jazigo já utilizado, considerar se já está ocupado e lotação máxima do espaço.

- Alterar dados de sepultado

O usuário localiza um falecido e tem a opção de alterar qualquer dado desse inquilino (exceto código do sepultado que é gerado automaticamente no momento do cadastro).

- Exclusão

O usuário deve ter a opção de exclusão tanto de jazigos quanto de sepultados. Um jazigo só pode ser excluído se não houver nenhum sepultado ocupando. Também poderá efetuar exclusão de todos sepultados, neste caso usuário limpa todos os cadastros.

Exclusão de sepultado - o usuário digita o número do jazigo do sepultado ou o nome daquele que deseja excluir (o código deste sepultado não poderá mais ser utilizado para novos cadastros). Sepultado não pode ser excluído se os familiares estiverem em atraso com pagamentos.

- Relatórios

Sistema deve gerar relatórios para: jazigos em atraso com as taxas, quantos jazigos estão ocupados e quantos disponíveis e quais ocupantes de um jazigo.

**Opção B:** proposição de sistema.

O grupo pode propor o escopo do sistema para desenvolvimento. Neste caso devem ser observados alguns elementos mínimos:

Deverá ser encaminhado até **04/05/2019** a proposta de escopo e descrição dos requisitos a serem desenvolvidos (a exemplo da opção A) para aprovação do professor, a ser publicado no Moodle (SOMENTE DEVEM PUBLICAR NO MOODLE AQUELES QUE ESCOLHEREM ESTA OPÇÃO).

Requisitos do sistema:

- Interfaces gráficas para manutenção do CRUD;
- Mínimo de dois tipos de cadastro a serem manipulados, com interação entre os mesmos;
- Mínimo de 5 dados a serem manipulados para cada cadastro;
- Interfaces de consulta e relatório para cada tipo de cadastro;
- Definir no mínimo quatro regras de negócio específicas para manutenção do CRUD;
- Desenvolvimento de banco de dados para armazenamento dos dados.