

Prof. Dr. Raphael Gomes ([raphael.gomes@ifg.edu.br](mailto:raphael.gomes@ifg.edu.br))

### Projeto Final – Sistema de revisão textual



#### Prefácio

A revisão textual consiste em organizar o texto de forma a adequá-lo às normas gramaticais requisitadas propondo melhorias para a clareza e qualidade do que foi produzido. Entre os tipos de revisão mais utilizados no Brasil estão:

- **Revisão gramatical:** consiste na leitura do texto e sua adequação aos critérios gramaticais, ortográficos e sintáticos da norma culta da língua portuguesa.
- **Adequação às normas da ABNT:** revisa-se o texto de forma a garantir que atenda às normas da Associação Brasileira de Normas Técnicas (ABNT).
- **Textual-gramatical:** inclui tanto a revisão gramatical quanto a análise crítica do texto, apontando aspectos de construção textual que levem a um texto mais claro.
- **Edição/formatação:** prepara, de acordo com as normas editoriais, um texto ou uma seleção de textos, com ações de padronização de fonte, inserção de índice, formatação etc.

Cada tipo de revisão requer um tratamento distinto e, dessa forma, é cobrado de forma diferente.

#### Definição do Trabalho

O trabalho consiste em implementar um sistema para orçamento e cobrança de revisões textuais. Para tal, também é realizado o gerenciamento dos dados do cliente solicitante e do ofertante do serviço.

O cabeçalho em todas as páginas deve constituir de um menu com as seguintes opções:

- Apresentação
- Serviços
- Central do cliente
- Contato

Além disso, o nome do cliente deve aparecer, caso ele já tenha realizado login ou um link para a página de cadastro de cliente, caso seja um novo usuário. A verificação e manutenção de login deve ser realizado usando sessão.

A descrição de cada página é apresentada abaixo:

1. **Apresentação:** página com descrição geral sobre o ofertante dos serviços (empresa ou pessoa física), formas de pagamento, link para redes sociais, etc.
2. **Serviços:** descrição dos serviços de revisão oferecidos, formas de cobrança e valor.  
Crie uma enumeração `BillingOption` para representar a opção de cobrança. Considere como opções iniciais a cobrança por página (o orçamento é dado pela quantidade de páginas do arquivo), por palavras (o orçamento é dado pela quantidade de palavras do arquivo) ou por lauda<sup>1</sup> (o orçamento é dado pela quantidade de laudas do arquivo).

1 Uma lauda equivale a um texto com 1.250 caracteres (incluindo os espaços).

Crie uma classe `Service` cujos atributos são a descrição do serviço, características do serviço, formas de cobrança aceitas e, para cada forma de cobrança aceita o valor cobrado. Os serviços disponíveis devem ser carregados dinamicamente através da leitura em um banco de dados.

3. **Central do cliente:** esta página oferece as seguintes opções:

- *Manutenção de dados do cliente:* visualização e alteração dos dados cadastrais e remoção de usuário.
- *Orçamento:* deve ser apresentado um formulário com a opção do cliente enviar um arquivo PDF do documento que deve ser revisado, além de selecionar o serviço e a opção de cobrança requisitada. Após o envio do arquivo deve ser apresentada uma página com valor a ser cobrado. O mesmo deve ser calculado de acordo com o serviço/forma de cobrança requisitados e os metadados do arquivo enviado<sup>2</sup>. A página de orçamento deve exibir uma *radio list* com todas as formas de cobrança disponíveis, sendo que a forma de cobrança inicialmente selecionada deve ser mostrada selecionada. Deve ser apresentado também um link para a requisição do serviço. Note que apesar de o cliente selecionar o orçamento usando um determinado tipo de cobrança ele pode optar por outro neste instante caso o valor orçado seja menor.
- *Requisição do serviço:* após a realização do orçamento, o cliente poderá solicitar a realização do serviço. O arquivo não deve ser enviado novamente, ou seja, deve ser armazenado durante o orçamento em uma área segura (o diretório não pode estar acessível pela URL). A página de requisição consiste em um formulário em que o usuário seleciona a opção de pagamento. Após a escolha, ele é encaminhado para a página de pagamento apropriada. Ao menos a forma de pagamento por boleto bancário deve ser implementada<sup>3</sup>. O boleto deve ser armazenado no espaço do cliente para caso ele queira solicitar a reimpressão. Será concedido um bônus ao grupo que implementar outras formas de pagamento (cartão de crédito, PayPal, etc.).

4. **Contato:** página com a descrição dos dados públicos do administrador do site (Nome, e-mail, telefone comercial, etc.)

Além das páginas visíveis aos clientes, o sistema deve oferecer uma parte de administração, acessada pela URL `/admin`. A página inicial desta parte consiste em um formulário de login do administrador. Caso o login seja realizado com sucesso ele é encaminhado para uma página onde pode gerenciar seus dados cadastrais, os dados dos serviços e os dados das formas de cobrança.

### O que deve ser entregue

- Código-fonte do projeto (via Moodle). Crie um único arquivo zip contendo o projeto do Eclipse. Inclua na pasta `WEB-INF/lib` todas as dependências do projeto.

### Datas

- 18/07/2018 (envio via Moodle até 19h): Entrega e apresentação do trabalho. Neste dia o trabalho será apresentado por grupo. A ordem de apresentação será definida por ordem chegada. A nota do trabalho será utilizada como nota da Prova 02. Além da apresentação do que foi desenvolvido, cada aluno será questionado sobre conceitos teóricos.

### Critérios para pontuação

A nota final do trabalho será dada considerando a nota do código-fonte com peso 3 e a nota da apresentação com peso 7.

- 2 *Dica:* use o framework Apache Tika para extrair os metadados do arquivo. Para informações sobre o uso consulte: <http://tika.apache.org/> e [https://www.tutorialspoint.com/tika/tika\\_extracting\\_pdf.htm](https://www.tutorialspoint.com/tika/tika_extracting_pdf.htm).
- 3 *Dica:* use a biblioteca Bopepo para a geração de boletos. Para informações sobre o uso consulte: <http://www.jrimum.org/bopepo/>, <http://www.jrimum.org/bopepo/wiki/Componente/AcervoDeTemplates> e <https://gist.github.com/gilmatrix/9161455>.

Os critérios considerados na pontuação do código-fonte são descritos na tabela a seguir:

<b>Critério</b>	<b>Descrição</b>	<b>Peso</b>
Prazo	Avalia se foi entregue no prazo estabelecido. Não serão aceitos trabalhos entregues fora do prazo.	Não se Aplica
Compleitude	Avalia se foi entregue tudo o que foi especificado ou, no caso de provas, se foi especificado tudo o que a questão pedia. Para cada item faltante, perde-se (2,0/ total de itens) pontos.	4
Consistência	Avalia a consistência entre as ideias apresentadas. Por exemplo, em um ponto do trabalho fala-se algo e em outro ponto nega-se este algo sem qualquer justificativa. Para cada ideia inconsistente perde-se 0,25 ponto até que se atinja 1,0 (um) ponto.	1
Qualidade do projeto	Avalia se o projeto do sistema foi bem elaborado, considerando aspectos como manutenibilidade, escalabilidade e isolamento. Considere o uso dos padrões de projeto apresentados. Quando aplicado a texto, avalia se o fluxo de ideias está bem elaborado e se a redação está clara para permitir um bom entendimento.	4
Formato	Verifica se o código foi bem documentado (comentado) e se boas práticas de codificação foram adotadas. Para códigos em Java utilize como referência as convenções sugeridas pela Oracle: <a href="http://www.oracle.com/technetwork/java/codeconventions-150003.pdf">http://www.oracle.com/technetwork/java/codeconventions-150003.pdf</a>	1

Os critérios considerados na pontuação da apresentação são descritos na tabela a seguir:

<b>Critério</b>	<b>Descrição</b>	<b>Peso</b>
Prazo	Avalia se foi entregue no prazo estabelecido. Não serão aceitos trabalhos entregues fora do prazo.	Não se Aplica
Domínio do conteúdo	Avalia se o aluno domina o conteúdo necessário para o desenvolvimento do trabalho, assim como para sua melhoria.	4
Capacidade de responder questionamentos	Avalia se o aluno é capaz de articular os argumentos para fornecer resposta a um determinado questionamento.	3
Clareza na explicação	Avalia se os argumentos dados pelo aluno são apresentadas de forma clara e objetiva.	1
Capacidade de extensão	Verifica se o aluno é capaz de estender o trabalho que foi produzido visando incluir novas funcionalidades e/ou melhorar o projeto desenvolvido.	2

### Comentários Gerais

- Devem necessariamente ser formados 10 grupos, sendo que a quantidade mínima de pessoas por grupo é 1.
- Cada grupo deve eleger um representante que ficará responsável por realizar a postagem dos documentos no Moodle. Os demais membros do grupo não deverão realizar a submissão do trabalho. Essa atividade não confere pontuação adicional ao responsável.

***People often say that motivation doesn't last. Well, neither does bathing – that's why we recommended it daily.***

Zig Zigler