**SURVEY - Instrumento de pesquisa**

**Avaliação da Documentação Gerada pelo Amazon Q (SARSB)**

**Título (formulário): Avaliação da Documentação Automatizada do Amazon Q — Projeto SARSB (ANA)**

**Descrição resumida (texto introdutório visível para o respondente):**

Você está convidado a participar de uma pesquisa que avalia a documentação gerada automaticamente pelo Amazon Q para métodos Java no sistema SARSB. A pesquisa mede três dimensões — Completude, Utilidade e Veracidade (CHT) — e deve levar de 10 a 25 minutos, dependendo do número de métodos atribuídos para revisão. A participação é voluntária e as respostas serão usadas apenas para pesquisa acadêmica.

**CONSENTIMENTO (Primeira página)**

Leia e aceite antes de continuar

Objetivo: Avaliar a qualidade da documentação gerada automaticamente pelo Amazon Q para métodos Java do sistema SARSB. Os resultados serão usados ​​para pesquisa acadêmica em engenharia de software.

Confidencialidade: Suas respostas serão anonimizadas. Nenhum código ou dado confidencial será publicado.

Voluntariado: A participação é voluntária e você pode se retirar a qualquer momento.

Contato: murillo.ed1402@gmail.com (Murillo Carvalho, Univ. de Brasília).

Pergunta de consentimento (obrigatória):

Li as informações e concordo em participar. ☐ Sim ☐ Não (Se não → encerrar a pesquisa)

**SEÇÃO A — Perfil do participante (uma página)**

1. ID do participante (texto) — use um código exclusivo (ex.: ANA\_DEV\_01).

2. Função (escolha única): Analista / Desenvolvedor / Testador / Arquiteto / Outro (texto).

3. Anos de experiência em Java (escolha única): <2 / 2–5 / 5–10 / >10.

4. Familiaridade com SARSB (escolha única): Nenhuma / Baixa / Média / Alta.

5. Você já utilizou ferramentas de IA para código ou documentação antes? (escolha única) Sim / Não.

Se sim: Quais ferramentas? (texto)

6. Tempo médio diário gasto na manutenção de código (horas) (número).

**INSTRUÇÕES DE ADMINISTRAÇÃO (internas, não mostradas aos respondentes)**

Tarefa: Cada participante avalia até 10 métodos. Os métodos são amostrados usando amostragem estratificada proporcional entre camadas (Config, Modelo, Repositório, Recurso, Serviço, Utilitário, etc.). Atribua métodos aleatoriamente aos participantes, garantindo a cobertura entre as camadas.

Repetição de bloco: O próximo bloco (Seção B) se repete para cada método atribuído. Limite o total de métodos por participante a 8–10 para evitar fadiga.

Anexos de arquivo: Forneça o trecho de código (assinatura + corpo do método), comentários originais (se houver) e a documentação gerada (Javadoc e/ou Markdown) ao participante no próprio formulário (ou por meio de um link por método). Cada bloco de método deve incluir esses elementos.

**SEÇÃO B — Avaliação em nível de método (repetir para cada método atribuído)**

Metadados do método (somente exibição):

ID do método: M\_METHODID

Caminho do arquivo: br.gov.ana.sarsb/...

Camada: Serviço/Repositório/...

Assinatura do método: (mostrar assinatura)

Corpo do método: (mostrar trecho de código)

Documentação gerada: (mostrar texto Javadoc e saída Markdown)

Instruções para este bloco: Para cada método apresentado, avalie a documentação gerada de acordo com os itens abaixo. Use a escala Likert de 5 pontos quando aplicável (1 = Discordo totalmente / Muito ruim, 5 = Concordo totalmente / Excelente). Responda honestamente e consulte o trecho de código durante o julgamento.

**A — INTEGRALIDADE (verificações estruturais)**

Use Sim/Não onde indicado ou Likert onde especificado.

B1. A documentação inclui um resumo claro de uma linha do propósito do método? (Likert 1–5)

B2. Todos os parâmetros do método estão documentados em entradas @param (se houver)? (Opções: Sim / Parcialmente / Não / Não aplicável)

B3. O @return é documentado corretamente quando o método retorna um valor? (Sim / Parcialmente / Não / Não aplicável)

B4. As exceções lançadas são documentadas em @throws quando o código declara ou pode lançar exceções? (Sim / Parcialmente / Não / Não aplicável)

B5. A estrutura geral do Javadoc e as seções necessárias estão presentes (resumo, parâmetros, retornos, lançamentos, exemplos, se relevantes)? (Likert 1–5)

B6. Pontuação rápida de completude (para seu uso interno): Indique o número de campos obrigatórios presentes (inteiro) — opcional, o formulário pode calcular automaticamente.

Comentário curto opcional (texto): Observações sobre completude.

**B — Utilidade (usabilidade prática)**

C1. A documentação é clara e fácil de ler. (Likert 1–5)

C2. A documentação me ajuda a entender o propósito do método rapidamente. (Likert 1–5)

C3. A documentação fornece contexto suficiente para usar ou manter o método. (Likert 1–5)

C4. A linguagem e as convenções correspondem aos padrões do projeto (tom, nomes, exemplos de código). (Likert 1–5)

C5. Pontuação geral de utilidade (subjetiva 1–5). (Likert 1–5)

Comentário curto opcional (texto): Observações de utilidade.

**C — VERANIZIDADE (consistência factual)**

D1. Todas as entidades nomeadas (variáveis, classes, exceções, métodos referenciados) existem no código apresentado. (Sim / Parcialmente / Não)

D2. A documentação não declara comportamentos que não aparecem no código (sem alucinações). (Likert 1–5)

D3. As referências a APIs/bibliotecas externas são precisas (se presentes). (Sim / Não / Não aplicável)

D4. Consistência factual geral (estimativa da Razão de Existência entre 0 e 1): (numérico 0,00–1,00) — opcional se você