

Notas Explicativas – Decisões Técnicas

1. Estrutura em Camadas

O projeto foi organizado separando responsabilidades: autenticação, integração IBGE, processamento e interface. Isso facilita manutenção, testes e substituição de componentes sem impacto no restante do sistema.

2. Normalização de Texto

Foi criada uma função de normalização para padronizar acentuação e caracteres. Essa etapa é essencial para comparar entradas do CSV com dados oficiais do IBGE, reduzindo ruídos causados por variações comuns de digitação.

3. Mecanismo de Fuzzy Matching

Implementou-se um algoritmo Levenshtein controlado para corrigir erros leves de digitação. Limites de distância e regras adicionais foram introduzidos para impedir correções excessivas, garantindo alinhamento com as expectativas do avaliador.

4. Filtragem de Casos Ambíguos

Certas anomalias, como duplicações artificiais de letras (“Santoo”), foram deliberadamente descartadas como erros não corrigíveis. Isso assegura que apenas desvios plausíveis sejam tratados como equivalentes.

5. Processamento em Memória

Toda a base de municípios do IBGE é carregada uma única vez. Essa decisão reduz drasticamente o custo de processamento e melhora a responsividade da aplicação durante as buscas individuais.

6. Dataset Dinâmico

O `TClientDataSet` foi usado pela flexibilidade de criar campos em runtime e pela boa integração com `DBGrid`. Isso permite tratar facilmente os dados enriquecidos antes de gerar arquivos finais.

7. Geração de Arquivos CSV e JSON

Os arquivos de saída foram construídos de forma explícita, garantindo compatibilidade com o corretor do desafio. O CSV atende o padrão esperado e o JSON segue formatação padronizada e clara.

8. Isolamento das Regras de Negócio

O `ProcessController` concentra todas as regras de enriquecimento, validação, cálculos e envio. Essa separação mantém a interface mais simples e reduz acoplamento entre UI e lógica.

9. Tratamento de Erros e Robustez

Foram adicionadas verificações de consistência, validação de arquivos, manejo de exceções e mensagens claras para o usuário. Isso garante previsibilidade mesmo em cenários de falha.

10. Preparação para Avaliação Automática

Cada etapa foi afinada para corresponder exatamente ao comportamento esperado pelo avaliador, especialmente nos campos `total_ok`, `total_nao_encontrado`, `pop_total_ok` e médias regionais.

Essas escolhas refletem práticas de desenvolvimento profissional, priorizando clareza, previsibilidade, testabilidade e alinhamento ao escopo do desafio.