Tela para Cálcular de imposto

O objetivo da tela é realizar o cálculo de imposto de um pedido e emitir uma nota fiscal em arquivo XML e gravado os dados no banco de dados.

Foram solicitados alguma alterações na regra de negócio.

1. Gerar Arquivo .XML com os dados da nota fiscal

No app.config chave de configuração para o destino dos arquivos.

```
<appSettings>
      <add key="CaminhoArquivoXMLNotaFiscal" value="ArquivosXML"/>

</appSettings>
```

Na camada de dados foi criado um método para serializar e salvar em arquivo os dados da Nota Fiscal.

2. Gravar os dados da nota fiscal no banco de dados

No app.config chave de configuração com a string de conexão.

```
<connectionStrings>

   <add name="SQL"
connectionString="Server=localhost;Database=Teste;Trusted_Connection=True;"/>

</connectionStrings>
```

Foi gerados dois reposítorios na camada de dados, um para gravar os dados da Nota Fiscal e o outro para gravar os itens da Nota Fiscal.

3. Adicionar novos campos na classe NotaFiscalItem

Foram adicionados os campos novos na classe na camada de dominio:

```
BaseCalculoIPI
AliquotaIPI
ValorIPI
```

Os mesmos foram replicados para o XML e Banco de dados.

Script para a criação dos novos campos no banco de dados encontra-se na pasta SQL.

CAMPOS_NOVOS.sql

4. Desenvolver uma procedure para atender sistemas externos

Na pasta SQL foi criado um novo arquivo com o script da Procedure

P_NOTA_FISCAL_IMPOSTOS.sql

5. Correção do bug quando estado de origem SP e destino RO o sistema deveria definir a CFOP como 6.006

Foi desenvolvido um teste de unidade para reproduzir a falha, e com isso foi feito o ajuste os valores de estado de origem e estado destino estavam invertidos.

6. Melhorias na tela

Foi implementado validação em todos os campos, ou seja, todos os campos são obrigatórios. Caso algum campo não seja preenchido exibirá uma mensagem de alerta.

Após a geração da nota fiscal os campos são limpos.

Validação dos estados de origem e destinos se são válidos.

Campos de estado de origem e destino com limitação de 2 caracteres e caixa alta.

7. Criar campo de desconto no item da Nota Fiscal

Foi adicionado o campo Desconto na classe NotaFiscalItem, e a regra de negócio foi implementada na mesma classe isolando sua responsabilidade.

8. Melhoria técnica

Foram feitos algumas melhorias tecnicas, seguindo os principios do SOLID. Como por exemplo.

Isolando as responsabilidades das classes NotaFiscal e NotaFiscalItem

Criar interfaces de repositório para NotaFiscal e NotaFiscalItem

Regras de negócios foram isoladas nas camadas de dominio

Testes de unidades foram implementados cobrindo 100% as classes de dominio.

9. Criar testes de unidade

Foram criados os testes de unidade e junto com a ferramenta do Visual Studio Enterprise foi executado o Living Test e a cobertura das classes de dominio foi 100%, foi encontrado um bug de regra de négocio, está com TODO: na camada de testes;