

PROC001 - Núcleo de validação e formatação de CNPJ

	Projeto/Sistema: GERENCIAQUI - MIGRAÇÃO CNPJ ALFANUMERICO (NT 2026.004)	Versão do Template: 1.2
	Processo: PROC001 - Núcleo de validação e formatação de CNPJ	Versão do Documento: 1.0
	Responsável(eis): Luís Leite	Data: 05/01/2026

1. Introdução

Este processo descreve a adaptação dos utilitários centrais de CNPJ do projeto GerencieAqui (CnpjUtil moderno, CnpjUtil legado, CpfOrCnpjUtil, ValidaCNPJ e ModValidador) para suportar o CNPJ Alfanumérico previsto pela NT 2026.004 (Instrução Normativa RFB nº 2.229/2024), com entrada em produção em 01/07/2026.

O CNPJ Alfanumérico mantém 14 caracteres, sendo as 12 primeiras posições alfanuméricas (letras maiúsculas A-Z e dígitos 0-9) e os 2 últimos dígitos verificadores numéricos. A máscara de exibição XX.XXX.XXX/XXXX-XX permanece inalterada. O cálculo dos DV usa o valor de cada caractere por ASCII menos 48 (módulo 11), o que já é implementado pelo ModValidador atual.

Esta especificação é a FUNDAÇÃO de toda a migração — todas as demais frentes (NF-e, NFC-e, CT-e, MDF-e, NFS-e, DCE, integrações, banco e front-end) reutilizam estes utilitários. A correção do núcleo deve ser a primeira entrega da iniciativa.

2. Processo

O núcleo de CNPJ é exercido em três fluxos: (1) validação, quando o sistema verifica se uma string informada é um CNPJ válido (componentes JSF, APIs REST, parsers); (2) limpeza/normalização, quando o sistema recebe o CNPJ com máscara e precisa armazená-lo apenas com os 14 caracteres úteis; (3) formatação, quando o sistema precisa exibir o CNPJ ao usuário ou em relatórios/XML/arquivos bancários.

Todas estas três operações precisam ser revistas: as regex `\d{14}` rejeitam letras, as limpezas via `replaceAll("\D", "")` apagam letras silenciosamente, e a formatação por `replaceFirst` com grupos `\d` falha quando há letras. A correção concentrada no núcleo resolve, em cascata, dezenas de pontos espalhados pelo sistema.

Processo	Envolvidos	Dados de Entrada	Dados de Saída
Processo	N/A	N/A	N/A

Processos Relacionados

- PROC002 — Adaptação de Cadastros

Especificação Funcional

PROC001 - Núcleo de validação e formatação de CNPJ:

Descrição dos Campos:

Campo	Tipo	Tamanho	Obrig. ?	Máscara	Observações
CNPJ (entrada)	Texto	14 ou 18	-	99.999.999/9999-99 (legado) ou *9.***.***/*-99 (alfanumérico)	As 12 primeiras posições alfanuméricas A-Z e 0-9; as 2 últimas posições numéricas (DV).
CNPJ (armazenado)	Texto	14	Sim	-	Sempre em maiúsculas, sem pontuação. Coluna varchar(14) a varchar(20) já existente.
CNPJ (exibido)	Texto	18	N/A	XX.XXX.XXX/XXXX-XX	-

Dígitos Verificadores	Numérico	2	Sim	99	Calculado por módulo 11 com pesos 5,4,3,2,9,8,7,6,5,4,3,2 sobre (char-48).
-----------------------	----------	---	-----	----	--

Regras de Negócio:

- 1. Algoritmo de validação dos DV (mantido):** O cálculo permanece módulo 11 sobre os 12 primeiros caracteres, utilizando o valor (char-48) de cada posição: '0'→0 ... '9'→9, 'A'→17 ... 'Z'→42. ModValidador.java já realiza esta operação corretamente (linhas 179, 192).
- 2. Limpeza preservando letras:** CnpjUtil.clean (linha 35) deve substituir replaceAll("\D", "") por replaceAll("[^A-Za-z0-9]", "") e em seguida toUpperCase(). Isto preserva as letras do CNPJ alfanumérico e elimina a corrupção silenciosa do dado.
- 3. Padrão de validação simples:** CnpjUtil.isValidSimple (linha 31) deve trocar matches("\d{14}") por matches("[A-Z0-9]{12}[0-9]{2}") sobre o valor limpo, e contemplar variação com máscara: matches("[A-Z0-9]{2}\.[A-Z0-9]{3}\.[A-Z0-9]{3}/[A-Z0-9]{4}-[0-9]{2}").
- 4. Formatação agnóstica:** CnpjUtil.format (linhas 38-47) deve usar replaceFirst("(.{2})(.{3})(.{3})(.{4})(.{2})", "\$1.\$2.\$3/\$4-\$5") em vez de grupos \d, suportando qualquer caractere.
- 5. Consolidação do util legado @Deprecated:** O CnpjUtil legado em util/CnpjUtil.java (com isCNPJValido / isCNPJ / imprimeCNPJ via charAt-48 e substring) é tecnicamente compatível com alfanumérico mas mantém duplicidade. Consolidar no util moderno e marcar para remoção em release subsequente.
- 6. Caracteres repetidos:** A verificação matches("^(\\d)\1*\$") em CnpjUtil.java (linha 16) já não casa com alfanumérico — sem quebra. Avaliar estender para detectar também sequências repetidas de letras (ex.: AAAAAAAAAAAAA00).
- 7. Desambiguação CPF × CNPJ em CpfOrCnpjUtil:** A distinção continua por comprimento (11 vs 14) sobre o valor limpo. Toda lógica que dependa de "ser numérico" para classificar precisa ser revista — um valor de 14 caracteres com letras é necessariamente CNPJ.

©SOFTEN SISTEMAS 2026

Revision #1

Created 25 May 2026 17:21:01 by Luís Leite

Updated 25 May 2026 17:52:25 by Luís Leite