

# PROC001 - Núcleo de validação e formatação de CNPJ

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

## 1. Introdução

Este processo descreve a adaptação dos utilitários centrais de CNPJ do projeto GerencieAqui 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

Processo	Envolvidos	Dados de Entrada	Dados de Saída
----------	------------	------------------	----------------

PROC001 - Núcleo de validação e formatação de CNPJ	-	-	-
----------------------------------------------------	---	---	---

## Processos Relacionados

## 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).

**1.1** O CNPJ alfanumérico é composto por doze caracteres alfanuméricos e dois dígitos verificadores numéricos. Os dígitos verificadores (DV) são calculados a partir dos doze primeiros caracteres em duas etapas, utilizando o módulo de divisão 11 e pesos distribuídos de 2 a 9.

**1.2** Para cada um dos caracteres do CNPJ, atribuir o valor da coluna conforme a tabela abaixo. O restante do cálculo do primeiro e segundo DV devem permanecer conforme

**MANUAL DE CALCULO DO DV DO CNPJ:**

Caractere	ASCII	Valor no cálculo
'0' a '9'	48-57	0 a 9
'A'	65	17
'B'	66	18
'C'	67	19
'D'	68	20
'E'	69	21
'F'	70	22
'G'	71	23
'H'	72	24
'I'	73	25
'J'	74	26
'K'	75	27
'L'	76	28
'M'	77	29
'N'	78	30
'O'	79	31
'P'	80	32
'Q'	81	33
'R'	82	34
'S'	83	35
'T'	84	36
'U'	85	37
'V'	86	38
'W'	87	39

' X'	88	40
' Y'	89	41
' Z'	90	42

**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:** 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.

**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 x 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.

## Questões Técnicas:

### 1. Alterações:

Arquivo : linha	Risco	Descrição / Ação
util/datatypes/CnpjUtil.java : ~31	Crítico	isValidSimple — trocar matches("\d{14}") por padrão alfanumérico.
util/datatypes/CnpjUtil.java : ~35	Crítico	clean — substituir replaceAll("\D", "") por replaceAll("[^A-Za-z0-9]", "") + toUpperCase().
util/datatypes/CnpjUtil.java : ~38-47	Alto	format — usar regex genérica (.{2})(.{3})(.{3})(.{4})(.{2}).
validadores/ModValidador.java : ~179, 192	Sem ação	charAt(i) - '0' já é o algoritmo oficial alfanumérico.

Arquivo : linha	Risco	Descrição / Ação
util/datatypes/CpfOrCnpjUtil.java : ~8-42	Alto	Delega a CnpjUtil — herda correções.
util/CnpjUtil.java (legado @Deprecated)	Baixo	Tolerante a alfanumérico. Consolidar no util moderno.
util/ValidaCNPJ.java	Alto	Reescrever detecção CPF×CNPJ; aceitar alfanumérico 14 sem máscara e 18 com máscara.
test/.../CnpjUtilTest.java	Alto	Adicionar casos alfanuméricos válidos e inválidos.

**2. Mudança na base de dados:** Nenhuma alteração de schema. Todas as colunas de CNPJ já comportam 14 caracteres alfanuméricos (varchar(14)-varchar(20)).

©SOFTEN SISTEMAS 2026

Revision #5

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

Updated 5 June 2026 11:44:24 by Luís Leite