

**FACULDADE DE ENGENHARIA**

**DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA**

**ENGENHARIA INFORMÁTICA**

**Compiladores**

**Construção da linguagem de programação Dondza usando a biblioteca JFlex do Java**

**Discentes:**

Júnior, Delfim Luís Uqueio

Macamo, Mónica Olga Ricardo

Machanguele, José Francisco

**Docente:**

Eng. Ruben Manhice

Eng. Edson Michaque

Maputo, Abril de 2020

**Índice**

[1 Introdução 3](#_Toc39235251)

[1.1 Objectivos 4](#_Toc39235252)

[1.1.1 Geral 4](#_Toc39235253)

[1.1.2 Específicos 4](#_Toc39235254)

[2 Metodologias 5](#_Toc39235255)

[3 API Java JFlex 6](#_Toc39235256)

[4 Linguagem Dondza 6](#_Toc39235257)

[4.1 Caracteristicas 6](#_Toc39235258)

[4.2 Implementação do Analizador Léxico 6](#_Toc39235259)

[4.2.1 Ferramentas e/ou Tecnologias Usadas 6](#_Toc39235260)

[4.2.2 Passos de Implementação 7](#_Toc39235261)

[5 Referências Bibliográficas 12](#_Toc39235262)

# Introdução

## **Objectivos**

### Geral

### Específicos

# Metodologias

**Quanto aos procedimentos**

* Pesquisas bibliográficas;

**Quanto à abordagem**

**Quanto ao objectivo**

# API Java JFlex

# Linguagem Dondza

O Termo Dondza vem do Xichangana, língua predominante no sul de Moçambique, que significa estudar como verbo em português. Tecnicamente o Dondza é uma linguagem de programação em desenvolvimento, que permitirá o aprendizado da programação usando a língua local Changana, com maior impacto para as crianças em regime de língua bilingue Changana no currículo Moçambicano de educação. Entre tanto, é de referir que todos podem aprender a programar em Dondza, assim como o fazem para outras linguagens com base inglesa. Esta de facto será uma forma de incentivar o gosto pela programação de quem é fluente em Changana e também funcionar como uma base de dados de aprendizado de changana para crianças não fluentes na língua em simultâneo com o aprendizado de programação.

## Caracteristicas

* Códigos baseados em Changana;
* Possui variáveis reservadas(Identificadores);
* Conta com os três paradigmas de programação: estrutural, procedural e orientada a objectos;
* Possui delimitadores de inicio e fim do programa na língua padrão;
* O escopo de identificadores é delimitado por ponto e virgula.

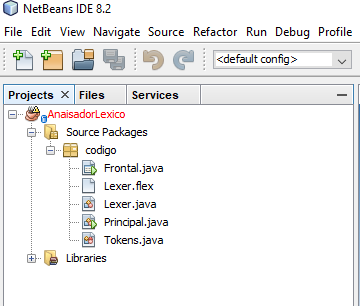
## Implementação do Analizador Léxico

O primeiro passo de projeccao do Dondza foi a construção do analisador Léxico, onde foi definido a estrutura e nomenclaturas de artefactos que fazem parte da linguagem e o uso de um simples programa em java var confirmar as regras estabelecidas no analisador.

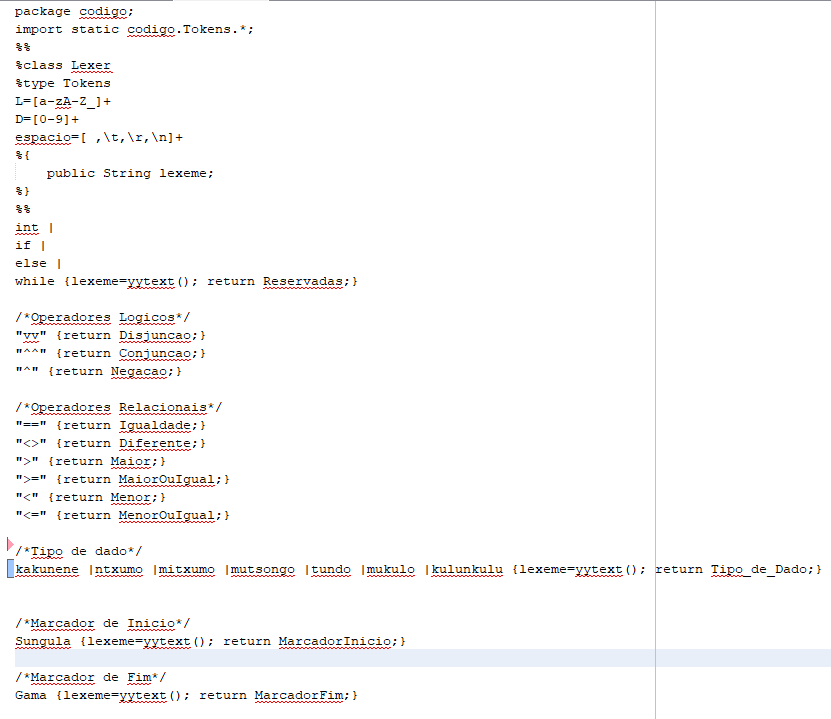
### Ferramentas e/ou Tecnologias Usadas

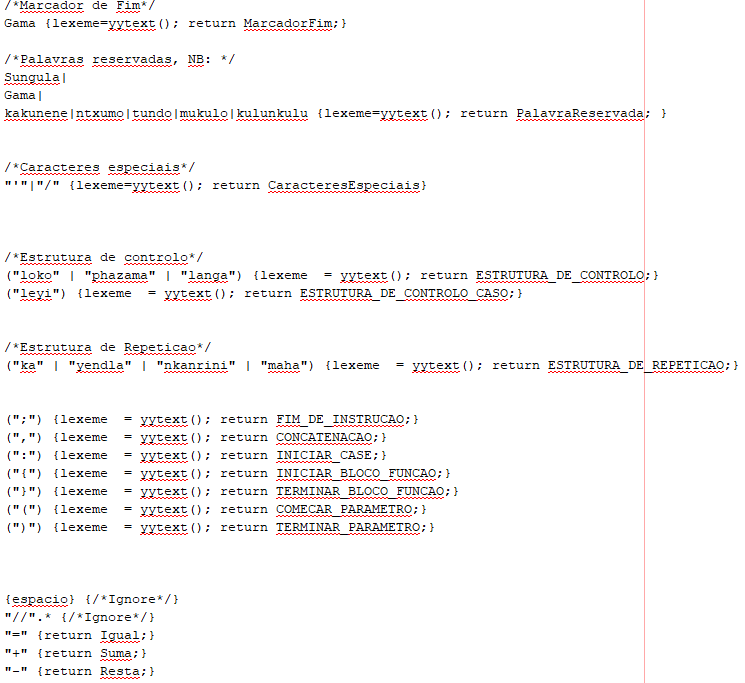
* IDE NetBeans;
* Linguagem Java e seus recursos, com maior ênfase a API JFlex;
* Ambiente de desenvolvimento colaborativo Github como repositório do código;
* O Git para versionar as etapas de desenvolvimento da linguagem.

### Passos de Implementação

1. **Criação de um projecto Java no NetBeans.**
2. **Agregar o JFlex a biblioteca do projecto Criado.**
3. **Criação de uma estrutura padrão para o trabalho.**

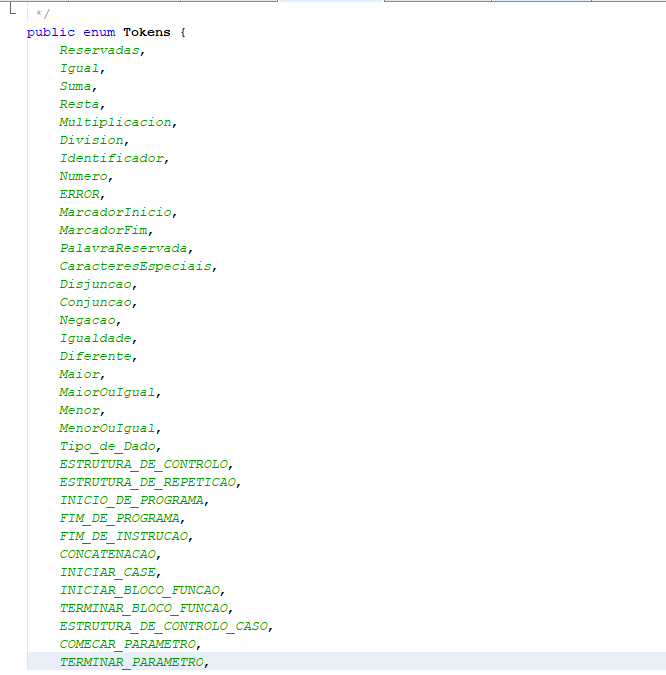
1. **Ficheiro Lexer.flex**





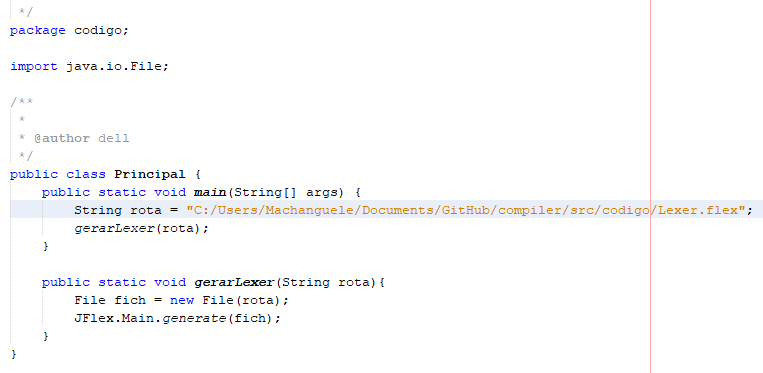
**5. Definicao dos Tokens**

Aqui estao listadas as designações de lexemas usados do ficheiro lexer.flex.

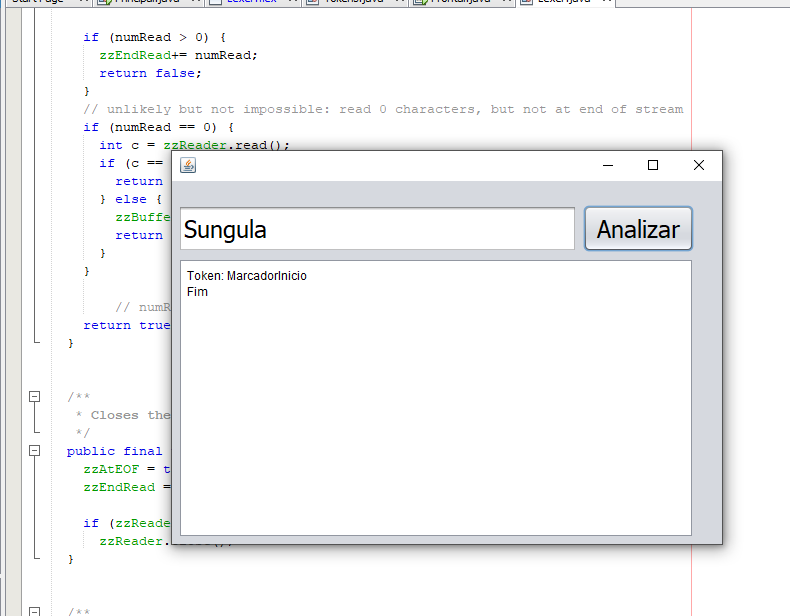


1. **Classe Principal.Java**

Aqui é definida a URL que índica o ficheiro lexer.flex, e ao compilar este, Cria automaticamente elementos da Classe Lexer.Java, configurando cada token aos seus lexemas.



1. **Classe Lexer.Java (Gerada Automaticamente)**
2. **Classe de testes do analisador Léxico, Frontal. Java**



# Referências Bibliográficas