

## **Estrutura de Dados no Desenvolvimento do MVP – Disciplina de Extensão**

A presente Atividade Integradora tem por objetivo apresentar a estrutura lógica de parte do aplicativo de Lista de Compras, sendo desenvolvido por meio da disciplina de Extensão II. Este módulo tratará do cadastro dos produtos a serem ordenados no aplicativo. Este cadastro, inicialmente, contará apenas com os nomes dos produtos, sendo aprimorado posteriormente.

A estrutura de dados escolhida para o desenvolvimento do código foi a Fila. Isso porque a Fila é uma estrutura de dados simples e eficiente que permite armazenar elementos em uma ordem específica e processá-los de acordo com essa ordem. Além disso, a Fila é uma estrutura de dados relativamente simples de ser implementada, podendo ser criada utilizando um array ou uma lista encadeada. No cenário construído, a Fila será implementada por meio de uma Lista Encadeada (*LinkedList*), isto pois, ao contrário de um array, a *LinkedList* não tem um tamanho fixo, permitindo que a Fila cresça dinamicamente à medida que novos elementos são adicionados. Assim, a *LinkedList* oferece uma combinação de eficiência, flexibilidade e facilidade de uso.

A ordenação dos elementos será dada por meio do método *Collections.sort*, da linguagem de programação Java, pois traz rapidez e eficiência ao código, tendo em vista que este método verifica o tamanho da lista e, através disso, escolhe o melhor script de ordenação para os elementos.

A linguagem escolhida foi Java pois se trata de uma linguagem portátil, ou seja, um programa desenvolvido em Java pode ser executado em diferentes sistemas operacionais sem a necessidade de modificações significativas e, tendo em vista que a aplicação primária do aplicativo será em dispositivos móveis, o Java tem suporte nativo para este uso. Ademais, o Java possui uma grande biblioteca de classes e APIs que facilitam o desenvolvimento de programas.

A implementação da estrutura de dados foi feita utilizando a biblioteca *Java.util*, por meio das classes: *Collections*, *LinkedList*, *List* e *Queue*, de forma que promovem rapidez de codificação e eficiência na execução dos códigos.

Desta forma, segue abaixo a codificação da estrutura de dados para o desenvolvimento do cadastro de produtos no aplicativo de Lista de Compras:

```
import java.util.Collections;
import java.util.LinkedList;
import java.util.List;
import java.util.Queue;

public class FilaProdutos {
    private Queue<String> produtos;

    public FilaProdutos() {
        this.produtos = new LinkedList<String>();
    }

    public void adicionarProduto(String produto) {
        this.produtos.add(produto);
    }

    public String removerProduto() {
        return this.produtos.remove();
    }

    public boolean filaVazia() {
        return this.produtos.isEmpty();
    }

    public void ordenarProdutos() {
        List<String> listaProdutos = new LinkedList<>(this.produtos);
        Collections.sort(listaProdutos);
        this.produtos = new LinkedList<>(listaProdutos);
    }

    public static void main(String[] args) {

        FilaProdutos fila = new FilaProdutos();

        fila.adicionarProduto("Macarrão");
        fila.adicionarProduto("Oleo de Soja");
        fila.adicionarProduto("Arroz");
        fila.adicionarProduto("Sal");

        System.out.println("Fila antes da ordenação: " + fila.produtos);

        fila.ordenarProdutos();

        System.out.println("Fila depois da ordenação: " + fila.produtos);
    }
}
```