UNIVERSIDADE
FEDERAL DO CARIRI

Arthur Ricardo Macêdo Pereira e Everton Lohan Pereira Ferreira

Programação Orientada à Objetos

Documentação do Projeto - To-Do List

Juazeiro do Norte - CE
2024

SUMÁRIO

Link do repositório : https://github.com/ev3rton0/toDoListJava

1. Tema: Gerenciador de Tarefas (to-do-list)

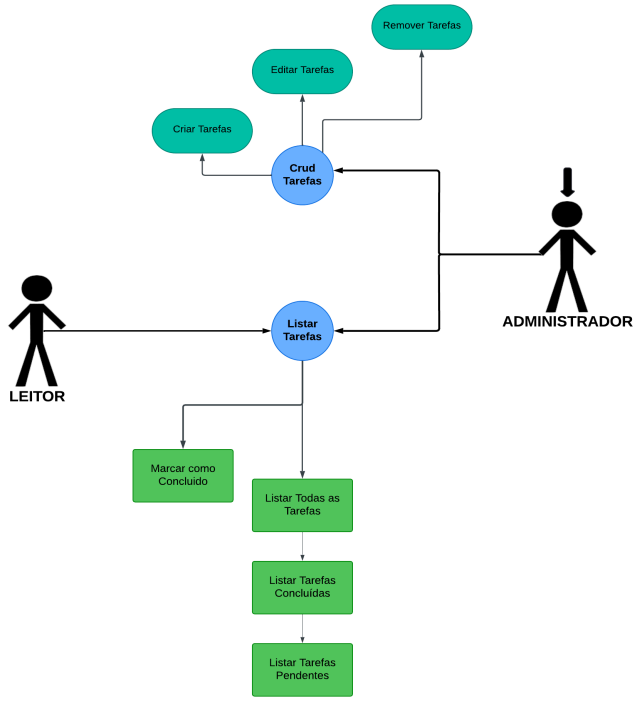
Um sistema no qual o usuário poderá criar tarefas do seu dia a dia, tendo como opções criar, editar, remover, marcar como concluído e listar as tarefas.

2. Funcionalidades e Stakeholders:

FUNCIONALIDADES:

- Criar Tarefas;
- Editar Tarefas;
- Remover Tarefas;
- Ver Tarefas;
- Marcar Tarefas como Concluída;
- Tarefas Concluídas.

STAKEHOLDERS:



- Usuário Administrador: Acesso a todas as funcionalidades;
- Usuário Leitor: Não terá acesso a criar, remover e editar listas.

3. Classes Básicas:

Classe Tarefa:

-Na classe tarefa temos os atributos:

- Título;
- Descrição;
- Data de Entrega;
- Concluída;

- Além dos atributos temos o construtor, os setters e os getters e por fim temos os override dos métodos toString e equals.
- A classe Tarefa é o tipo oficial dos objetos usados nas próximas classes.

```java
public class Tarefa {
    private String titulo;
    private String descricao;
    private LocalDate dataEntrega;
    private boolean concluida;

    public Tarefa(String titulo, String descricao, LocalDate dataEntrega) {
        this.titulo = titulo;
        this.descricao = descricao;
        this.dataEntrega = dataEntrega;
        this.concluida = false;
    }

    public String getTitulo() {
        return titulo;
    }

    public String getDescricao() {
        return descricao;
    }

    public LocalDate getDataEntrega() {
        return dataEntrega;
    }
```

Classe listarTarefas:

- Na classe listar Tarefas, temos 3 métodos:
    - listarTarefas: Esse método lista todas as tarefas existentes no programa;
    - listarTarefasConcluidas: Esse método lista todas as tarefas que foram marcadas como concluídas;
    - listarTarefasPendentes: Esse método lista todas as tarefas que ainda não foram concluídas.

```java
public void listarTarefas(List<Tarefa> tarefas) {
    if (tarefas.isEmpty()) {
        System.out.println("Não há tarefas a serem listadas.");
    } else {
        System.out.println("\n|Lista de Tarefas:");
        for (Tarefa tarefa : tarefas) {
            System.out.println(tarefa);
        } System.out.println("==========================");
    }
}
```

```java
public void listarTarefasConcluidas(List<Tarefa> tarefas) {
    System.out.println("\n|Tarefas Concluídas:");
    if (tarefas.isEmpty()){
        System.out.println("Não há tarefas concluídas");
    } else {
        for (Tarefa tarefa : tarefas) {
            if (tarefa.isConcluida()) {
            System.out.println(tarefa);
            }
    } System.out.println("==========================");

    }
}
```

```java
public void listarTarefasPendentes(List<Tarefa> tarefas) {
    System.out.println("\n|Tarefas Pendentes:");
    if(tarefas.isEmpty()){
        System.out.println("Não há tarefas pendentes");
    } else {
        for (Tarefa tarefa : tarefas) {
            if (!tarefa.isConcluida()) {
                System.out.println(tarefa);}
    }System.out.println("==========================");
    }
}
```

4. Classes de Repositório(CRUD):

Na classe de repositório CRUD, temos os principais métodos:

- **Criar tarefa**: Permite ao usuário adicionar uma nova tarefa à lista de tarefas.
    - Implementação: Utiliza o método que recebe os detalhes da tarefa (título, descrição, data de entrega) e adiciona à lista de tarefas.

- **Editar tarefa**: Permite ao usuário modificar os detalhes de uma tarefa existente.
    - Implementação: Utiliza um método que recebe o índice da tarefa a ser editada e os novos detalhes, e então atualiza a tarefa na lista.

- **Remover tarefa**: Permite ao usuário deletar uma tarefa da lista.
    - Utiliza um método que recebe o índice da tarefa a ser removida e a exclui da lista.

- **Marcar tarefa como concluída:** Permite ao usuário marcar uma tarefa como concluída.
    - Utiliza um método T que altera o estado da tarefa para: concluída.

- **Listar tarefas:** Exibe todas as tarefas, independentemente do seu estado (concluídas ou pendentes).
    - **Tarefas concluídas:** Exibe apenas as tarefas que foram marcadas como concluídas.
    - **Tarefas pendentes:** Exibe apenas as tarefas que ainda não foram concluídas.

(CÓDIGO DISPONÍVEL NO REPOSITÓRIO)

5. Interface via Console:

- Na interface temos duas páginas:
- A primeira para selecionar o tipo de usuário

- A segunda para usar todas as funcionalidades de acordo com o tipo de usuário, abaixo está as funcionalidades do usuário Administrador: