

## Documento de Projeto e Testes

**Projeto:** Sistema de Fila Virtual - Testes de Unidade

**Disciplina:** Oficina de Integração 1

**Grupo:**

- Tiago Navarro Bettanin
- Estevan Garrido Hernandez Sastre

---

### Descrição

Este pacote contém a implementação do módulo de lógica de gerenciamento da fila (`FilaManager.js`) para o projeto de um Sistema de Fila Virtual e seus respectivos testes de unidade (`FilaManager.test.js`) utilizando Jest.

O objetivo é validar o "coração" do sistema de forma isolada, garantindo seu correto funcionamento antes da integração com o servidor e a interface do cliente.

### Arquivos Incluídos

- `FilaManager.js`: O módulo com a lógica pura da fila (adicionar, remover, consultar).
- `FilaManager.test.js`: O conjunto de testes para o `FilaManager.js`.
- `package.json`: Arquivo de configuração do projeto Node.js, contendo as dependências e scripts.
- `README.pdf`: Este arquivo de instruções.

### Cenários de Teste Verificados

O arquivo `FilaManager.test.js` foi projetado para cobrir os principais cenários de uso e casos extremos da lógica da fila. Cada teste valida um comportamento específico:

- **Adição de usuário:** Verifica se um usuário é adicionado corretamente e se o tamanho da fila aumenta.
  - **Adição múltipla e ordem FIFO:** Garante que múltiplos usuários são adicionados na ordem correta (o primeiro a entrar deve ser o primeiro da lista).
  - **Chamada do próximo:** Testa a remoção do primeiro usuário da fila, validando se o usuário correto é retornado e se a fila é atualizada como esperado.
  - **Robustez da chamada:** Valida o comportamento do sistema ao tentar remover um item de uma fila que já está vazia, esperando um retorno `null`.
  - **Consulta de posição:** Confere se a busca de posição retorna o número correto para um usuário específico.
  - **Consulta de usuário inexistente:** Verifica se a busca por um usuário que não está na fila falha de maneira controlada (retornando `-1`).
  - **Validação de entrada:** Garante que entradas inválidas (nomes vazios, nulos ou com apenas espaços) não sejam adicionadas à fila, mantendo a integridade dos dados.
-

## Como Executar os Testes

Para executar os testes, siga os passos abaixo no terminal:

### 1. Pré-requisitos:

- Ter o Node.js instalado em seu computador.

### 2. Instalar as Dependências:

Navegue até a pasta do projeto pelo terminal e execute o comando para instalar o Jest (a única dependência de desenvolvimento):

None

```
npm install
```

### 3. Rodar os Testes:

Após a instalação, execute o seguinte comando para rodar todos os testes definidos:

None

```
npm test
```

## Resultado Esperado

O terminal deverá exibir uma saída indicando que todos os 7 testes passaram com sucesso, similar a esta:

None

```
PASS ./FilaManager.test.js
  ✓ deve adicionar um usuário à fila corretamente (2ms)
  ✓ deve adicionar múltiplos usuários na ordem correta
  ✓ deve chamar o próximo usuário (o primeiro da fila) (1ms)
  ✓ deve retornar null ao tentar chamar o próximo de uma fila vazia
  ✓ deve retornar a posição correta de um usuário (base 1)
```

- ✓ deve retornar -1 para um usuário que não está na fila
- ✓ não deve adicionar um usuário se o nome for vazio ou inválido (1ms)

Test Suites: 1 passed, 1 total  
Tests: 7 passed, 7 total  
Snapshots: 0 total  
Time: ...s