Testes Unitários

Comando utilizado: npx jest --verbose

Retorno:

```
PASS  tests/Cliente.test.ts
  Cliente
    constructor
      √ deve criar uma instância de Cliente com nome e número corretos (24 ms)
      √ deve permitir criação com valores vazios
    getters
      √ deve retornar o nome corretamente
      √ deve retornar o número corretamente
    casos especiais
      √ deve aceitar caracteres especiais no nome (1 ms)
      √ deve aceitar números no campo nome

 PASS  tests/Mesa.test.ts
  Mesa
    constructor
      √ deve criar uma instância de Mesa com os valores corretos (25 ms)
    métodos
      √ deve atualizar a disponibilidade da mesa (11 ms)
      √ deve realizar um pedido e atualizar a disponibilidade para false (6 ms)
      √ deve calcular a conta corretamente quando houver um pedido (1 ms)
      √ deve retornar zero ao calcular a conta se nenhum pedido foi realizado (1 ms)
    getters
      √ deve retornar o nome, número e disponibilidade corretamente
      √ deve retornar o pedido corretamente após realizar um pedido (1 ms)

 PASS  tests/Prato.test.ts
  Prato
    constructor
      √ deve criar uma instância de Prato com os valores corretos e quantidade padrão
(12 ms)
      √ deve criar uma instância de Prato com quantidade informada
    adicionar_quantidade
      √ deve atualizar a quantidade do prato corretamente (1 ms)
    getters
      √ deve retornar os valores corretos através dos getters
    casos especiais
      √ deve atualizar a quantidade para zero quando informado zero

  console.log
    Garçom Carlos registrou o pedido para a mesa 1.
```

```
      at Garcom.registrarPedido (src/Garcom.ts:14:17)

  console.log
      Conta total para a mesa 1: R$ 105.00

      at Garcom.calcularConta (src/Garcom.ts:20:17)

  console.log
      Mesa 1 está indisponível.

      at Garcom.atualizarMesa (src/Garcom.ts:27:17)

 PASS  tests/Garcom.test.ts
  Garcom
      √ deve registrar um pedido (75 ms)
      √ deve calcular a conta com taxa (3 ms)
      √ deve atualizar a disponibilidade da mesa (3 ms)
      √ deve retornar o nome do garçom (1 ms)
      √ deve retornar a taxa do garçom
      √ deve retornar a disponibilidade do garçom (1 ms)
      √ deve definir corretamente o status da mesa como disponível ou indisponível (1 ms)

 PASS  tests/Pedido.test.ts
  Pedido
      constructor
      √ deve criar uma instância de Pedido com o cliente informado e sem pratos (9 ms)
      adicionarPrato
      √ deve adicionar um prato ao pedido (1 ms)
      calcularTotal
      √ deve retornar 0 quando nenhum prato for adicionado
      √ deve calcular corretamente o total com um prato adicionado
      √ deve calcular corretamente o total com múltiplos pratos adicionados (1 ms)
      getters
      √ deve retornar o cliente corretamente através do getter
      √ deve retornar os pratos corretamente através do getter

Test Suites: 5 passed, 5 total
Tests:       32 passed, 32 total
Snapshots:   0 total
Time:        3.989 s, estimated 4 s
Ran all test suites.
```

ao usar npx jest --coverage, obtemos também a tabela de cobertura do código:

```
------------|---------|----------|---------|---------|-------------------
File        | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
------------|---------|----------|---------|---------|-------------------
```

```
Cliente.ts |     100 |   100 |   100 |   100 |
Garcom.ts  |   100 |   100 |   100 |   100 |
Mesa.ts|        100 |   100 |   100 |   100 |
Pedido.ts  |   100 |   100 |   100 |   100 |
Prato.ts   |     100 |   100 |   100 |   100 |
-----------|---------|----------|---------|---------|-------------------
```

Modelagem

Relação	Tipo	Multiplicidade	Explicação
Cliente — Pedido	**Associação**	**1:1**	Um pedido pertence a um cliente
Pedido — Prato	**Composição**	**1:N**	Um pedido contém vários pratos
Mesa — Pedido	**Agregação**	**0:1**	Uma mesa pode ter um pedido ou estar vazia
Garcom — Mesa	**Dependência Leve**	**N:N**	`Garcom` depende de `Mesa`, mas não mantém uma relação permanente.
Garcom — Pedido	**Dependência Leve**	**1:N**	O garçom depende de `Pedido` para registrar pedidos em uma mesa, mas não mantém referência direta a eles

Diagrama de classes projeto

Bernardo Lopes Braga | February 18, 2025