



FGA 0238 - Testes de Software – Turma:		1	Semestre:	2024.1
Nome:	Marcos Castilhos		Matrícula:	221008300
Equipe:	Equipe 6 - TLE			

## Atividade 4 – TDD

### 1 Funcionalidade

#### 1.1 Identificação da Issue

Não foi aberta issue no github

#### 1.2 Descrição da Funcionalidade

Suítes de testes do frontend

#### 1.3 Ciclos

Alguns testes do Frontend estavam falhando. Eu escolhi resolver os testes de uma função que estava com alguns problemas, a função getMonths.

No entanto, o meu pull request chamou atenção para um problema geral nos testes - eles estavam rodando no tempo da máquina, não em UTC.

Então um outro desenvolvedor assíduo do projeto resolveu o problema e citou o meu pr.

### 2 Execução

#### 2.1 Primeiro Ciclo

<identifique o primeiro teste e apresente uma breve descrição>

Teste: Identifiquei que o método getMonths estava falhando para o primeiro dia de junho.

```
getMonths(): MonkeyTypes.TestActivityMonth[] {
  const months: Date[] = eachMonthOfInterval({
    start: this.startDay,
    end: this.endDay,
  });
  const results: MonkeyTypes.TestActivityMonth[] = [];

  for (let i = 0; i < months.length; i++) {
    const month: Date = months[i] as Date;
    let start = i === 0 ? this.startDay : startOfMonth(month);
    let end = i === months.length - 1 ? this.endDay : endOfMonth(start);

    if (!isSunday(start)) {
      start = (i === 0 ? previousSunday : nextSunday)(start);
    }
    if (!isSaturday(end)) {
      end = nextSaturday(end);
    }

    const weeks = differenceInWeeks(end, start, { roundingMethod: "ceil" });
    if (weeks > 2) {
      results.push({
        text: format(month, "MMM").toLowerCase(),
        weeks: weeks,
      });
    } else if (i == 0) {
      results.push({ text: "", weeks: weeks });
    }
  }
  return results;
}
```

<apresente o código de teste>

```
describe("TestActivityCalendar", () => {
  describe("getMonths", () => {
    it("for lastDay in april", () => { ...
    });

    it("for lastDay in april, not test for the current week", () => { ...
    });
    it("for lastDay in january", () => { ...
    });
    it("for lastDay and full year starting with sunday", () => { ...
    });
    it("for lastDay and full year starting with monday", () => { ...
    });
    it("for first day in june", () => { ...
    });
    it("no double month for for 16th june", () => { ...
    });
  });
});
```

```
307 it("for first day in june", () => {
308   //set today
309   vi.setSystemTime(getDate("2024-06-01"));
310   const calendar = new TestActivityCalendar([], getDate("2024-06-01"));
311
312   expect(calendar.getMonths()).toEqual([
313     {
314       text: "",
315       weeks: 1,
316     },
317     {
318       text: "jun",
319       weeks: 4,
320     },
321   ])
```

<apresente o resultado da execução do teste (falha)>

```
Failed Tests 10
FAIL: __tests__/elements/test-activity-calendar.spec.ts > test-activity-calendar.ts > TestActivityCalendar > getMonths > for first day in june
AssertionError: expected [ { text: '', weeks: 1 }, ... ] to deeply equal [ { text: 'jun', weeks: 4 }, ... ]
- Expected
+ Received

Array [
  Object {
    + "text": "",
    + "weeks": 1,
  },
  Object {
    "text": "jun",
    "weeks": 4,
  },
]
```



<apresente o código da funcionalidade (já refatorado)>

```
.gitignore m Extension: Prettier - Code Formatter Extension: Vitest Extension: Auto Markdown TOC TS test-activity-calendar.spec.ts X TS test-activity-calendar.ts
frontend > __tests__ > elements > TS test-activity-calendar.spec.ts > describe("test-activity-calendar.ts") callback > describe("TestActivityCalendar") callback > describe("getMonths") callback > it("for first day in June") callback
  9 describe("test-activity-calendar.ts", () => {
 16 describe("TestActivityCalendar", () => {
 17   describe("getMonths", () => {
 18     //
 19     it("for first day in June", () => {
 20       //set today
 21       vi.setSystemTime(getDate("2024-06-01"));
 22       const calendar = new TestActivityCalendar([], getDate("2024-06-01"));
 23
 312      expect(calendar.getMonths()).toEqual([
 24        {
 25          text: "jun",
 26          weeks: 4,
 27        },
 28        {
 29          text: "jul",
 30          weeks: 5,
 31        },
 32        {
 33          text: "aug",
 34          weeks: 4,
 35        },
 36        {
 37          text: "sep",
 38          weeks: 4,
 39        },
 40        {
 41          text: "oct",
 42          weeks: 5,
 43        },
 44        {
 45          text: "nov",
 46          weeks: 4,
 47        },
 48        {
 49          text: "dec",
 50          weeks: 5,
 51        },
 52        {
 53          text: "jan",
 54          weeks: 4,
 55        },
 56        {
 57          text: "feb",
 58          weeks: 4,
 59        },
 60        {
 61          text: "mar",
 62          weeks: 5,
 63        },
 64        {
 65          text: "apr",
 66          weeks: 4,
 67        },
 68        {
 69          text: "may",
 70          weeks: 4,
 71        },
 72      ]);
 73    });
 74  });
 75 }
```

Expected	Actual
1 Array []	1 Array [
	2+ Object {
	3+   "text": "",
	4+   "weeks": 1,
	5+ },
2 Object {	6 Object {
3   "text": "jun",	7   "text": "jun",
4   "weeks": 4,	8   "weeks": 4,
5 },	9 },
6 Object {	10 Object {
7   "text": "jul",	11   "text": "jul",
8   "weeks": 5,	12   "weeks": 5,
9 },	13 },
10 Object {	14 Object {
11   "text": "aug",	15   "text": "aug",
12   "weeks": 4,	16   "weeks": 4,
13 },	17 },
14 Object {	18 Object {
15   "text": "sep",	19   "text": "sep",
16   "weeks": 4,	20   "weeks": 4,
17 },	21 },
18 Object {	22 Object {
19   "text": "oct",	23   "text": "oct",
20   "weeks": 5,	24   "weeks": 5,
21 },	25 },
22 Object {	26 Object {
23   "text": "nov",	27   "text": "nov",
24   "weeks": 4,	28   "weeks": 4,
25 },	29 },
26 Object {	30 Object {
27   "text": "dec",	31   "text": "dec",

Removi o caso de teste que estava diferente do esperado.

Ainda não há captura de tela do teste funcionando pois ainda é preciso arrumar outro caso.

## 2.2 Segundo Ciclo

<identifique o segundo teste e apresente uma breve descrição>

**Teste:** Identifiquei que havia um erro no teste de duplicidade para o mês de junho no dia 16.

<apresente o código de teste>

```
it("no double month for for 16th june", () => {
  expect(calendar.getMonths()).toHaveLength(
    [
      {
        text: "",
        weeks: 2,
      },
      {
        text: "jul",
        weeks: 5,
      },
      {
        text: "aug",
        weeks: 4,
      },
      {
        text: "sep",
        weeks: 4,
      },
      {
        text: "oct",
        weeks: 5,
      },
      {
        text: "nov",
        weeks: 4,
      },
      {
        text: "dec",
        weeks: 5,
      },
      {
        text: "jan",
        weeks: 4,
      },
      {
        text: "feb",
        weeks: 4,
      },
      {
        text: "mar",
        weeks: 5,
      },
      {
        text: "apr",
        weeks: 4,
      },
      {
        text: "may",
        weeks: 4,
      },
      {
        text: "jun",
        weeks: 3,
      },
    ],
  );
});
```



<apresente o resultado da execução do teste (falha)>

Esperado na esquerda, atual na direita.

```
36 },
37 Object {
38   "text": "mar",
39   "weeks": 5,
40 },
41 },
42 Object {
43   "text": "apr",
44   "weeks": 4,
45 },
46 Object {
47   "text": "may",
48   "weeks": 4,
49 },
50 Object {
51   "text": "jun",
52   "weeks": 3,
53 },
54 ],
55 ]
```



```
FAIL: __tests__/elements/test-activity-calendar.spec.ts > test-activity-calendar.ts > TestActivityCalendar > getMonths > no double month for for 16th june
AssertionError: expected [ { text: ' ', weeks: 2 }, ... ] to deeply equal [ { text: ' ', weeks: 2 }, ... ]

- Expected
+ Received

Array [
  Object {
    "text": " ",
    "weeks": 2,
  },
  Object {
    "text": "jul",
    "weeks": 5,
  },
  Object {
    "text": "aug",
    "weeks": 4,
  },
  Object {
    "text": "sep",
    "weeks": 4,
  },
  Object {
    "text": "oct",
    "weeks": 5,
  },
  Object {
    "text": "nov",
    "weeks": 4,
  },
  Object {
    "text": "dec",
    "weeks": 5,
  },
  Object {
    "text": "jan",
    "weeks": 4,
  },
  Object {
    "text": "feb",
    "weeks": 4,
  },
  Object {
    "text": "mar",
    "weeks": 5,
  },
  Object {
    "text": "apr",
    "weeks": 4,
  },
  Object {
    "text": "may",
    "weeks": 4,
  },
  Object {
    "text": "jun",
    "weeks": 3,
  },
]

> __tests__/elements/test-activity-calendar.spec.ts:368:38
366|         const calendar = new TestActivityCalendar([], getDate('2024-06-0_
367|
368|         expect(calendar.getMonths()).toEqual([
    |                                     ^
369|           {
370|             text: " ",
```

<apresente o código da funcionalidade>

```
367 it("no double month for for 16th june", () => {
368   //set today
369   vi.setSystemTime(getDate("2024-06-16"));
370   const calendar = new TestActivityCalendar([], getDate("2024-06-01"));
371
372   expect(calendar.getMonths()).toEqual([
373     {
374       text: "",
375       weeks: 2,
376     },
377     {
378       text: "jul",
379       weeks: 5,
380     },
381     {
382       text: "aug",
383       weeks: 4,
384     },
385     {
386       text: "sep",
387       weeks: 4,
388     },
389     {
390       text: "oct",
391       weeks: 5,
392     },
393     {
394       text: "nov",
395       weeks: 4,
396     },
397     {
398       text: "dec",
399       weeks: 5,
400     },
401     {
402       text: "jan",
403       weeks: 4,
404     },
405     {
406       text: "feb",
407       weeks: 4,
408     },
409     {
410       text: "mar",
411       weeks: 5,
412     },
413     {
414       text: "apr",
415       weeks: 4,
416     },
417     {
418       text: "may",
419       weeks: 4,
420     },
421   ]);
422 });
```

Removido o caso que estava causando erro.

```
> test-activity-calendar.ts (22) 491ms
> TestActivityCalendar (15) 410ms
  ✓ getMonths (7)
  \ getDays (8) 371ms
```

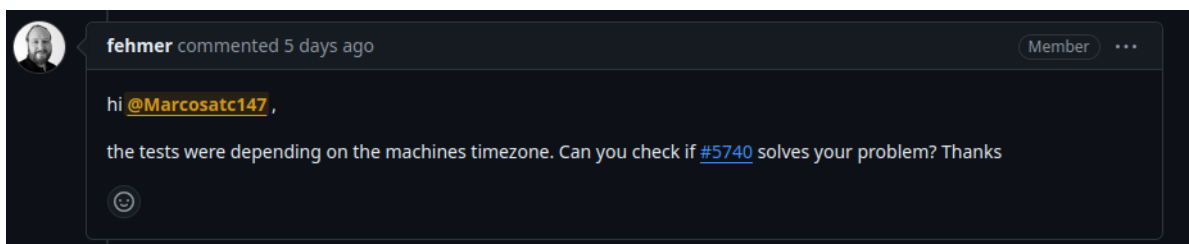
Após resolver os problemas citados anteriormente, os testes da função `getMonths` passaram todos.

## 2.3 Terceiro Ciclo



<identifique o n-ésimo teste e apresente uma breve descrição>

No entanto, a minha solução se provou ser como uma falsificação, existia um problema maior que eu não conhecia e, ao ver o meu PR, um outro desenvolvedor propôs uma solução geral para a suíte de testes do frontend.



<apresente o código de teste>

```
3 backend/__tests__/global-setup.ts
... -1,5 +1,6 @@
1 1 import * as MongoDBMock from "vitest-mongodb";
2 - export async function setup({ provide }): Promise<void> {
2 + export async function setup(): Promise<void> {
3 +   process.env.TZ = "UTC";
3 4   await MongoDBMock.setup({
4 5     serverOptions: {
5 6       binary: {
6 7         version: "6.0.12",
7 8       },
8 9     },
9 10   });
10 11 }
11 12
12 13 export async function teardown(): Promise<void> {
13 14   await MongoDBMock.teardown();
14 15 }
```

O que está em vermelho foi removido e o que está em verde, adicionado.



<apresente o resultado da execução do teste (falha)>

```

Failed Tests 10
FAIL __tests__/elements/test-activity-calendar.spec.ts > test-activity-calendar.ts > TestActivityCalendar > getMonths > for first day in june
AssertionError: expected [ { text: '', weeks: 1 }, _(12) ] to deeply equal [ { text: 'jun', weeks: 4 }, _(11) ]

- Expected
+ Received

Array [
  Object {
+   "text": "",
+   "weeks": 1,
+ },
+ Object {
    "text": "jun",
    "weeks": 4,
  }
]

```

Por causa do fato dos testes estarem rodando no tempo da máquina, os testes estavam falhando em 10 casos.

<apresente o código da funcionalidade>

```
3 frontend/__tests__/global-setup.ts
...
1 + export const setup = () => {
2 +   process.env.TZ = "UTC";
3 + };

1 frontend/vitest.config.js
↑
-4,6 +4,7 export default defineConfig({
4   test: {
5     globals: true,
6     environment: "happy-dom",
7 +   globalSetup: "__tests__/global-setup.ts",
8     setupFiles: ["__tests__/setup-tests.ts"],
9   },
10  coverage: {
.....
↓
```

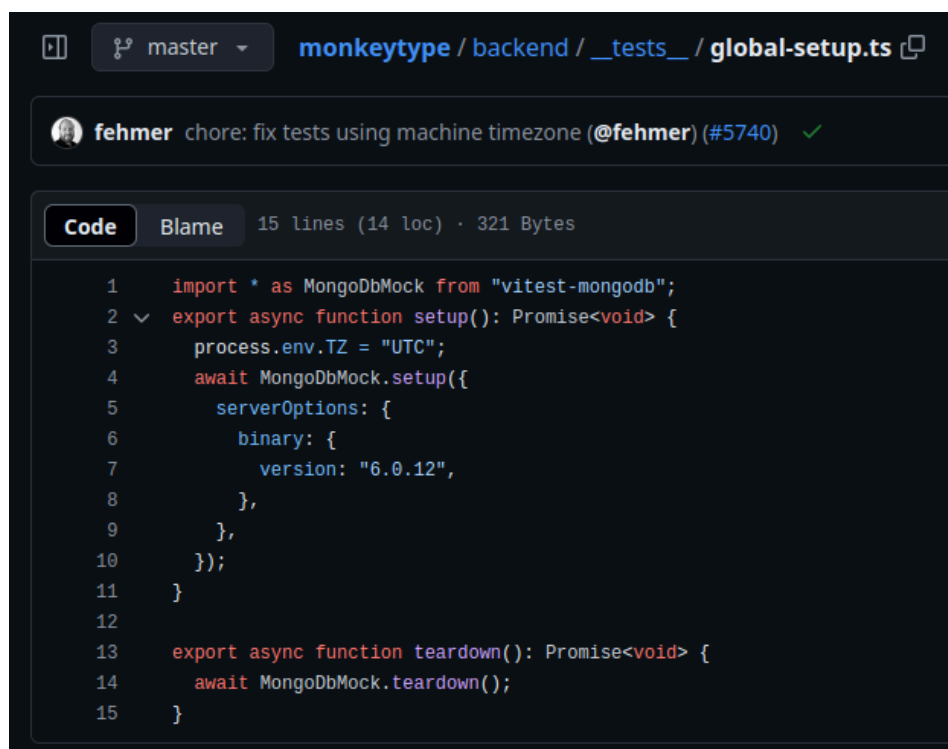
<apresente o resultado da execução do teste (sucesso)>

```
✓ __tests__/utils/tag-builder.spec.ts (5)
✓ __tests__/test/lazy-mode.spec.ts (7)
✓ __tests__/utils/numbers.spec.ts (4)
✓ __tests__/test/misc.spec.ts (2)
✓ __tests__/utils/ip-addresses.spec.ts (5) 1087ms
✓ __tests__/utils/colors.spec.ts (2)
✓ __tests__/utils/config.spec.ts (3)
✓ __tests__/elements/test-activity-calendar.spec.ts (22) 434ms
✓ __tests__/utils/generate.spec.ts (1)
✓ __tests__/utils/format.spec.ts (19)
✓ __tests__/root/config.spec.ts (89)
✓ __tests__/test/layout-emulator.spec.ts (8)

Test Files 12 passed (12)
Tests      167 passed (167)
```

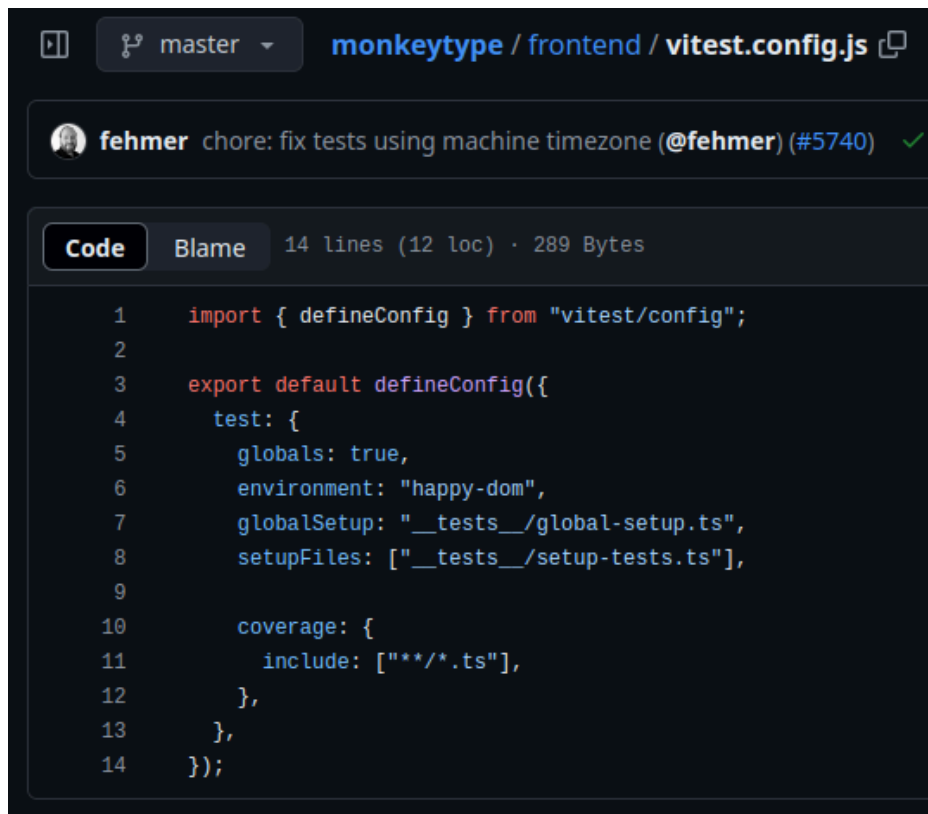
### 3 Código Fonte Testes

<Apresentar a versão final do código fonte dos testes>



The screenshot shows a GitHub commit interface. At the top, it says "master" and "monkeytype / backend / \_\_tests\_\_ / global-setup.ts". Below that, a commit by "fehmer" is shown with the message "chore: fix tests using machine timezone (@fehmer) (#5740)". The commit is marked as "Code" and "Blame" with "15 lines (14 loc) · 321 Bytes". The code itself is as follows:

```
1 import * as MongoDBMock from "vitest-mongodb";
2 export async function setup(): Promise<void> {
3   process.env.TZ = "UTC";
4   await MongoDBMock.setup({
5     serverOptions: {
6       binary: {
7         version: "6.0.12",
8       },
9     },
10  });
11 }
12
13 export async function teardown(): Promise<void> {
14   await MongoDBMock.teardown();
15 }
```



The screenshot shows a GitHub commit interface for the file `monkeytype / frontend / vitest.config.js`. The commit is by user `fehmer` with the message `chore: fix tests using machine timezone (@fehmer) (#5740)`. The commit is marked as merged with a green checkmark. Below the commit message, there are tabs for `Code` and `Blame`. The `Code` tab is selected, showing the content of `vitest.config.js`. The code is as follows:

```
1  import { defineConfig } from "vitest/config";
2
3  export default defineConfig({
4    test: {
5      globals: true,
6      environment: "happy-dom",
7      globalSetup: "__tests__/global-setup.ts",
8      setupFiles: ["__tests__/setup-tests.ts"],
9
10     coverage: {
11       include: ["**/*.ts"],
12     },
13   },
14 });
```

<Indicar o link para a classe no seu repositório Github (fork do projeto)>

[Global setup backend](#)

[Suíte de testes - Frontend](#)

## 4 Resultado Final Execução Testes

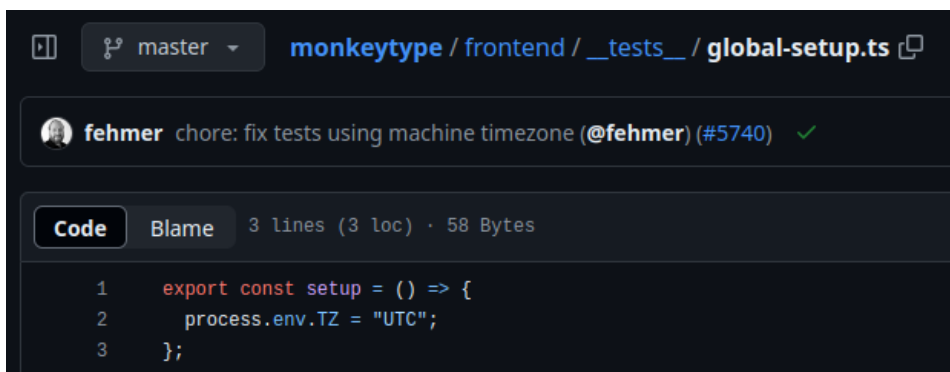
<Apresentar o resultado final da execução de todos os testes>

```
✓ __tests__/utils/tag-builder.spec.ts (5)
✓ __tests__/test/lazy-mode.spec.ts (7)
✓ __tests__/utils/numbers.spec.ts (4)
✓ __tests__/test/misc.spec.ts (2)
✓ __tests__/utils/ip-addresses.spec.ts (5) 1087ms
✓ __tests__/utils/colors.spec.ts (2)
✓ __tests__/utils/config.spec.ts (3)
✓ __tests__/elements/test-activity-calendar.spec.ts (22) 434ms
✓ __tests__/utils/generate.spec.ts (1)
✓ __tests__/utils/format.spec.ts (19)
✓ __tests__/root/config.spec.ts (89)
✓ __tests__/test/layout-emulator.spec.ts (8)

Test Files 12 passed (12)
Tests      167 passed (167)
```

## 5 Código Fonte da Funcionalidade Implementada

<Apresentar a versão final do código fonte do funcionalidade implementada>



```
1 export const setup = () => {
2   process.env.TZ = "UTC";
3 };
```

Foi criado o arquivo global-setup.ts para garantir que os testes do frontend rodassem em UTC.

<Indicar o link para a classe no seu repositório Github (fork do projeto)>

[Global Setup - Frontend](#)

## 6 Pull Request

[Pull Request - 5739](#)



## 7 Conclusão

Desenvolver usando TDD foi uma experiência que me permitiu não apenas corrigir um problema existente, mas também identificar uma questão mais ampla sobre a dependência dos testes no fuso horário da máquina. Pude aprender bastante conversando com outro desenvolvedor, que inclusive foi solícito para ajudar a resolver o problema na minha máquina e também pude fornecer feedback sobre a funcionalidade da suíte de testes. A implementação em TDD garantiu que cada mudança fosse validada incrementalmente (mesmo que sem querer), o que resultou em um código mais robusto e confiável.