

FGA 0238 - Testes de Software – Turma: 1

Semestre: <2024.1>

Nome: <Marcos Castilhos>

Matrícula: 221008300

Equipe: Equipe 6 - TLE

Atividade 3 – Desenvolver Testes de Unidade

1 Identificação do Projeto

Monkeytype

2 Cobertura de testes

Backend

```
✓ __tests__/dal/blocklist.spec.ts (13) 384ms
✓ __tests__/utils/pb.spec.ts (3)
> __tests__/utils/misc.spec.ts (14)
  > Misc Utils (14)
    ✓ getCurrentDayTimestamp
    ✓ matchesAPattern
    ✓ kogascore
    ✓ identity
    ✓ flattenObjectDeep
    ✓ sanitizeString
    ✓ getOrdinalNumberString
    ✗ getStartOfWeekTimestamp
    ✗ getCurrentWeekTimestamp
    ✓ getStartOfDayTimestamp
    ✓ isToday
    ✓ isYesterday
    ✓ mapRange
    ✓ formatSeconds
✓ __tests__/utils/validation.spec.ts (5)
✓ __tests__/dal/preset.spec.ts (12) 420ms
✓ __tests__/dal/result.spec.ts (4) 464ms
✓ __tests__/middlewares/auth.spec.ts (9) 1222ms
✓ __tests__/dal/user.spec.ts (87) 1802ms
✓ __tests__/dal/leaderboards.spec.ts (10) 877ms
✓ __tests__/dal/public.spec.ts (3) 441ms
✓ __tests__/migration/testActivity.spec.ts (2) 449ms
✓ __tests__/utils/daily-leaderboards.spec.ts (1) 317ms
✓ __tests__/services/weekly-xp-leaderboard.spec.ts (1)
✓ __tests__/dal/api-keys.spec.ts (1)
✓ __tests__/utils/monkey-mail.spec.ts (1)
✓ __tests__/api/controllers/user.spec.ts (19) 408ms
✓ __tests__/api/controllers/result.spec.ts (11) 303ms
✓ __tests__/api/controllers/leaderboard.spec.ts (1)
```

```
Failed Tests 2

FAIL: __tests__/utils/misc.spec.ts > Misc Utils > getStartOfWeekTimestamp
AssertionError: expected 1661817600000 to deeply equal 1662330000000
- Expected
+ Received
- 1662330000000
+ 1661817600000

> __tests__/utils/misc.spec.ts:316:51
314|
315|     testCases.forEach(({ input, expected }) => {
316|       expect(misc.getStartOfWeekTimestamp(input)).toEqual(expected);
317|     });
318|   });
> __tests__/utils/misc.spec.ts:315:15

FAIL: __tests__/utils/misc.spec.ts > Misc Utils > getCurrentWeekTimestamp
AssertionError: expected 824774400000 to be 824688000000 // Object.is equality
- Expected
+ Received
- 824688000000
+ 824774400000

> __tests__/utils/misc.spec.ts:324:25
322|     const currentWeek = misc.getCurrentWeekTimestamp();
323|     expect(currentWeek).toBe(824688000000); // Mon Feb 19 1996 00:00:00 GMT+0000
324|   });
325|
326|

Test Files 1 failed | 17 passed (18)
Tests 2 failed | 195 passed (197)
Start at 23:42:19
Duration 4.94s (transform 2.41s, setup 9.28s, collect 14.39s, tests 7.39s, environment 8ms, prepare 3.38s)
```

Frontend

```
✓ __tests__/utils/tag-builder.spec.ts (5)
✓ __tests__/utils/numbers.spec.ts (4)
✓ __tests__/test/misc.spec.ts (2)
✓ __tests__/test/lazy-node.spec.ts (7)
✓ __tests__/test/config.spec.ts (3)
> __tests__/elements/test-activity-calendar.spec.ts (22)
  > test-activity-calendar.ts (22)
    > TestActivityCalendar (15)
      > getMonths (7)
        ✓ for lastDay in april
        ✓ for lastDay in april, not test for the current week
        ✓ for lastDay in january
        ✓ for lastDay and full year starting with sunday
        ✓ for lastDay and full year starting with monday
        ✗ for first day in june
        ✗ no double month for for 16th june
      > getDays (8)
        ✗ for lastDay in april
        ✗ for full leap year
        ✗ for full year
        ✓ ignores data before calendar range
        ✗ handles missing data in calendar range
        ✓ for lastDay in february
        ✗ current day mid of month
        ✓ for lastDay in february full year
      > ModifiableTestActivityCalendar (5)
        > increment (5)
          ✓ increments on lastDay
          ✗ increments after lastDay
          ✗ increments after two months
          ✗ increments in new year
          ✓ fails increment in the past
        ✓ getFullYearCalendar (1)
        ✓ getTotalTests (1)
✓ __tests__/utils/format.spec.ts (19)
```

Failed Tests 10

3 Método a ser testado

<Apresentar o nome do método, descrever seu propósito e a classe/pacote a que pertence>

Nome: updateAltGrState()

Propósito: A função updateAltGrState é responsável por monitorar o estado da tecla AltGr

A função detecta quando a tecla AltGr (ou Alt Right) é pressionada (keydown) e quando é liberada (keyup). Dependendo do tipo de evento (keydown ou keyup), ela atualiza a variável global isAltGrPressed.

A função também trata de um caso especial para macOS onde a tecla AltLeft pode ser usada como AltGr em algumas configurações. Isso é feito verificando se o userAgent do navegador contém "Mac".

A função pertence ao módulo layout-emulator

<Apresentar o código do método a ser testado. Pode ser uma captura de tela>

```
function updateAltGrState(event: JQuery.KeyboardEventBase): void {  
  const shouldHandleLeftAlt =  
    event.code === "AltLeft" && navigator.userAgent.includes("Mac");  
  if (event.code !== "AltRight" && !shouldHandleLeftAlt) return;  
  if (event.type === "keydown") isAltGrPressed = true;  
  if (event.type === "keyup") isAltGrPressed = false;  
}
```

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

[módulo Layout-emulator](#)

4 Testes existentes

Não existem testes para o método descrito.

5 Tabela de decisões/condições

<Apresentar a tabela de decisões e condições>

Decisão 1: Linha `const shouldHandleLeftAlt = event.code === "AltLeft" && navigator.userAgent.includes("Mac");`

- **Condição 1:** `event.code === "AltLeft"`
- **Condição 2:** `navigator.userAgent.includes("Mac")`

Decisão 2: Linha `if (event.code !== "AltRight" && !shouldHandleLeftAlt) return;`

- **Condição 3:** `event.code !== "AltRight"`
- **Condição 4:** `!shouldHandleLeftAlt`

Decisão 3: Linha `if (event.type === "keydown") isAltGrPressed = true;`

- **Condição 5:** `event.type === "keydown"`

Decisão 4: Linha `if (event.type === "keyup") isAltGrPressed = false;`

- **Condição 6:** `event.type === "keyup"`

Tabela 1: Decisões e Condições

ID	Decisão (linha)	Condição	Cenário de entrada para Verdadeiro	Cenário de entrada para Falso
CD1	<code>const shouldHandleLeftAlt = event.code === "AltLeft" && navigator.userAgent.includes("Mac");</code>	<code>event.code === "AltLeft"</code>	<code>event.code = "AltLeft"</code>	<code>event.code ≠ "AltLeft"</code>

CD2	const shouldHandleLeftAlt = event.code === "AltLeft" &&navigator.userAgent.includes("Mac");	navigator.userAgent.includes("Mac")	navigator.userAgent contém "Mac"	navigator.userAgent não contém "Mac"
CD3	if (event.code !== "AltRight" && !shouldHandleLeftAlt) return;	event.code !== "AltRight"	event.code ≠ "AltRight"	event.code = "AltRight"
CD4	if (event.code !== "AltRight" && !shouldHandleLeftAlt) return;	!shouldHandleLeftAlt	shouldHandleLeftAlt = False	shouldHandleLeftAlt = V
CD5	if (event.type === "keydown") isAltGrPressed = true;	event.type === "keydown"	event.type = "keydown"	event.type ≠ "keydown"
CD6	if (event.type === "keyup") isAltGrPressed = false;	event.type === "keyup"	event.type = "keyup"	event.type ≠ "keyup"

6 Tabelas verdade, pares de independência e combinações de condições MC/DC

<Apresentar a tabela verdade para cada decisão com mais de uma condição>

Tabela 2: Tabela Verdade para Decisão da linha N

Pares de Independência para Decisão 1

- Decisão: const shouldHandleLeftAlt = event.code === "AltLeft" && navigator.userAgent.includes("Mac");
 - Condição 1: event.code === "AltLeft"
 - Condição 2: navigator.userAgent.includes("Mac")

ID	event.code === "AltLeft"	navigator.userAgent.includes("Mac")	Resultado
1	V	V	V
2	F	V	F

3	V	F	F
4	F	F	F

Pares de Independência para Decisão 2

- Decisão: `if (event.code !== "AltRight" && !shouldHandleLeftAlt) return;`
 - Condição 3: `event.code !== "AltRight"`
 - Condição 4: `!shouldHandleLeftAlt`

ID	<code>event.code !== "AltRight"</code>	<code>!shouldHandleLeftAlt</code>	Resultado
5	V	V	return
6	F	V	continue
7	V	F	continue
8	F	F	continue

Pares de Independência para Decisão 3

- Decisão: `if (event.type === "keydown") isAltGrPressed = true;`
 - Condição 5: `event.type === "keydown"`

ID	<code>event.type === "keydown"</code>	Resultado
9	V	<code>isAltGrPressed = true</code>
10	F	sem alteração

Pares de Independência para Decisão 4

- Decisão: `if (event.type === "keyup") isAltGrPressed = false;`
 - Condição 6: `event.type === "keyup"`

ID	event.type === "keyup"	Resultado
11	V	isAltGrPressed = false
12	F	sem alteração

7 Especificação dos Casos de Testes

<Apresentar os casos de teste derivados das múltiplas-condições>

Tabela 3: Casos de Testes

ID	Nome do CT	Entrada	Saída Esperada	Cobertura de Condições (Condição + Situação V ou F)	Cobertura MC/DC (linhas das tabelas verdade)
CT1	AltLeft e Mac	event.code = "AltLeft", navigator.userAgent.includes("Mac") = true, event.type = "keydown"	isAltGrPressed = true	CD1V, CD2V, CD3V, CD4F, CD5V, CD6F	1, 2, 5, 9
CT2	AltLeft não é Mac	event.code = "AltLeft", navigator.userAgent.includes("Mac") = false, event.type = "keydown"	isAltGrPressed = true	CD1V, CD2F, CD3V, CD4F, CD5V, CD6F	1, 3, 5, 9
CT3	AltRight e Mac	event.code = "AltRight", navigator.userAgent.includes("Mac") = true, event.type = "keydown"	isAltGrPressed = true	CD1F, CD2V, CD3V, CD4F, CD5V, CD6F	2, 4, 5, 9
CT4	AltRight não é Mac	event.code = "AltRight", navigator.userAgent.includes("Mac") = false, event.type = "keydown"	isAltGrPressed = true	CD1F, CD2F, CD3V, CD4F, CD5V, CD6F	3, 4, 5, 9

CT5	AltLeft e Mac (keyup)	event.code = "AltLeft", navigator.userAgent.includes("Mac") = true, event.type = "keyup"	isAltGrPressed = false	CD1V, CD2V, CD3F, CD4F, CD5F, CD6V	1, 2, 10, 11
CT6	AltLeft não é Mac (keyup)	event.code = "AltLeft", navigator.userAgent.includes("Mac") = false, event.type = "keyup"	isAltGrPressed = false	CD1V, CD2F, CD3F, CD4F, CD5F, CD6V	1, 3, 10, 11
CT7	AltRight e Mac (keyup)	event.code = "AltRight", navigator.userAgent.includes("Mac") = true, event.type = "keyup"	isAltGrPressed = false	CD1F, CD2V, CD3F, CD4F, CD5F, CD6V	2, 4, 10, 11
CT8	AltRight não é Mac (keyup)	event.code = "AltRight", navigator.userAgent.includes("Mac") = false, event.type = "keyup"	isAltGrPressed = false	CD1F, CD2F, CD3F, CD4F, CD5F, CD6V	3, 4, 10, 11
CT9	AltLeft e Mac (não Alt)	event.code != "AltRight", event.code != "AltLeft", navigator.userAgent.includes("Mac") = true	Sem mudança	CD1F, CD2V, CD3F, CD4V	4,6
CT10	AltLeft não é Mac (não Alt)	event.code != "AltRight", event.code != "AltLeft", navigator.userAgent.includes("Mac") = false	Sem mudança	CD1F, CD2F, CD3F, CD4V	3,4,6
CT11	AltRight e Mac (não Alt)	event.code != "AltRight", event.code != "AltLeft", navigator.userAgent	Sem mudança	CD1F, CD2V, CD3F, CD4V	2,6

		nt.includes("Mac") = true			
CT12	AltRight não é Mac (não Alt)	event.code !== "AltRight", event.code !== "AltLeft", navigator.userAgent nt.includes("Mac") = false	Sem mudança	CD1F, CD2F, CD3F, CD4V	3,4,6

8 Implementação dos Casos de Teste

<Apresentar o código dos casos de teste implementados. Pode ser uma captura de tela>

```

TS layout-emulator.spec.ts X TS layout-emulator.ts
frontend > _tests_ > test > TS layout-emulator.spec.ts > describe("LayoutEmulator") callback > describe("updateAltGrState") callback
1 import { describe, it, expect, afterEach } from "vitest";
2 import {
3   updateAltGrState,
4   getIsAltGrPressed,
5 } from "../../src/ts/test/layout-emulator";
6
7 describe("LayoutEmulator", () => {
8   describe("updateAltGrState", () => {
9     afterEach(() => {
10       // Reset isAltGrPressed state after each test
11       // Simulate keyup event to reset state
12       const event = createEvent("AltRight", "keyup");
13       updateAltGrState(event);
14     });
15
16     const createEvent = (
17       code: string,
18       type: string
19     ): JQuery.KeyboardEventBase => {
20       ({
21         code,
22         type,
23       } as JQuery.KeyboardEventBase);
24     };
25
26     it("should set isAltGrPressed to true on AltRight keydown", () => {
27       const event = createEvent("AltRight", "keydown");
28       updateAltGrState(event);
29       expect(getIsAltGrPressed()).toBe(true);
30     });
31
32     it("should set isAltGrPressed to false on AltRight keyup", () => {
33       const event = createEvent("AltRight", "keyup");
34       updateAltGrState(event);
35       expect(getIsAltGrPressed()).toBe(false);
36     });
37
38     it("should set isAltGrPressed to true on AltLeft keydown on Mac", () => {
39       Object.defineProperty(window.navigator, "userAgent", {
40         value: "Mac",
41         configurable: true,
42       });
43       const event = createEvent("AltLeft", "keydown");
44       updateAltGrState(event);
45       expect(getIsAltGrPressed()).toBe(true);
46     });
47
48     it("should set isAltGrPressed to false on AltLeft keyup on Mac", () => {
49       Object.defineProperty(window.navigator, "userAgent", {

```

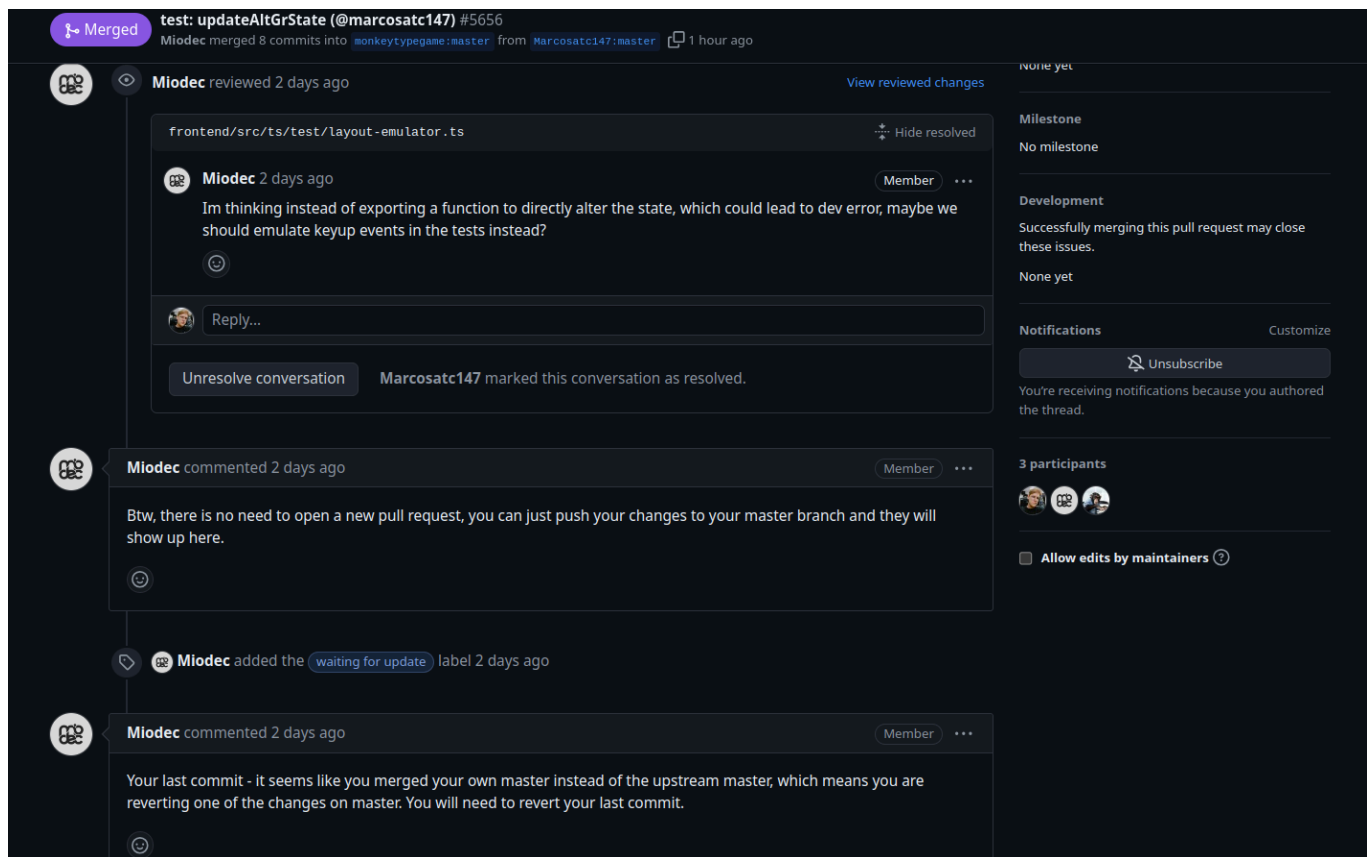
```
TS layout-emulator.spec.ts x TS layout-emulator.ts
frontend > __tests__ > test > TS layout-emulator.spec.ts > describe("LayoutEmulator") callback > describe("updateAltGrState") callback
7 describe("LayoutEmulator", () => {
8   describe("updateAltGrState", () => {
47     it("should set isAltGrPressed to false on AltLeft keyup on Mac", () => {
48       Object.defineProperty(window.navigator, "userAgent", {
49         value: "Mac",
50         configurable: true,
51       });
52       const event = createEvent("AltLeft", "keyup");
53       updateAltGrState(event);
54       expect(getIsAltGrPressed()).toBe(false);
55     });
56
57     it("should not change isAltGrPressed on AltLeft keydown on non-Mac", () => {
58       Object.defineProperty(window.navigator, "userAgent", {
59         value: "Windows",
60         configurable: true,
61       });
62       const event = createEvent("AltLeft", "keydown");
63       updateAltGrState(event);
64       expect(getIsAltGrPressed()).toBe(false);
65     });
66
67     it("should not change isAltGrPressed on AltLeft keyup on non-Mac", () => {
68       Object.defineProperty(window.navigator, "userAgent", {
69         value: "Windows",
70         configurable: true,
71       });
72       const event = createEvent("AltLeft", "keyup");
73       updateAltGrState(event);
74       expect(getIsAltGrPressed()).toBe(false);
75     });
76
77     it("should not change isAltGrPressed on keydown of other keys", () => {
78       const event = createEvent("KeyA", "keydown");
79       updateAltGrState(event);
80       expect(getIsAltGrPressed()).toBe(false);
81     });
82
83     it("should not change isAltGrPressed on keyup of other keys", () => {
84       const event = createEvent("KeyA", "keyup");
85       updateAltGrState(event);
86       expect(getIsAltGrPressed()).toBe(false);
87     });
88   });
89 });
```

[classe github](#)

9 Análises e Resultados

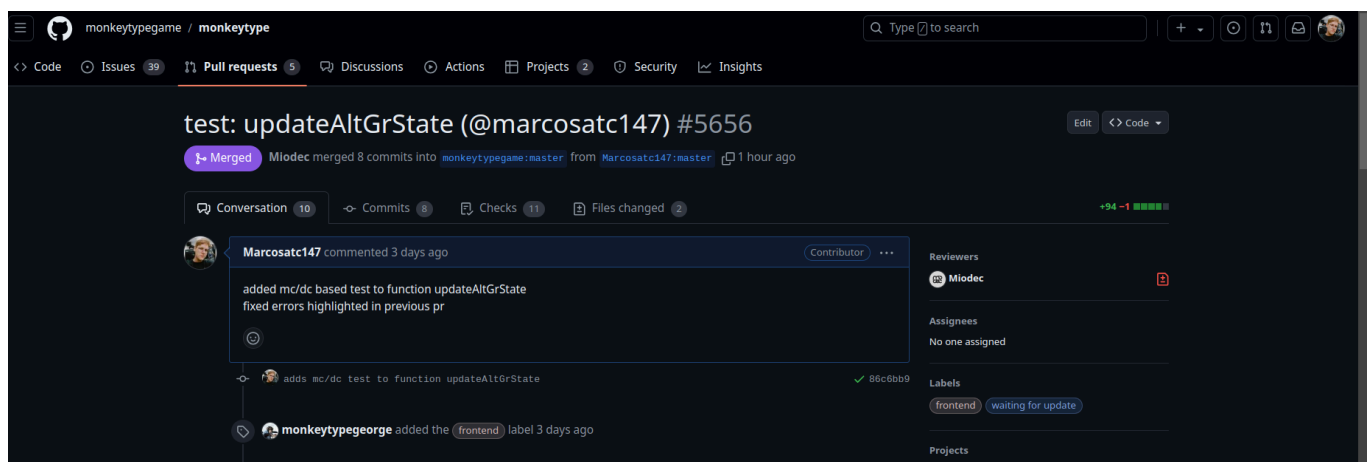
```
✓ __tests__/test/layout-emulator.spec.ts (8)
```

Foram pedidas algumas mudanças em relação a primeira implementação



Optei por trocar o pull request para fazer um outro fork do projeto porque as coisas acabaram ficando um pouco confusas na minha máquina - percebi depois que não era necessário.

10 Pull Request



11 Links

Fork do projeto:

[Monkeytype Fork](#)

Implementação no projeto:

[Implementação](#)

Commit da implementação:

[commit implementação](#)

Pull Request:

[pull request #5656](#)