

## Progresso TCC

Adriner Maranhão de Andrade <adrinera@furb.br>

Seg, 28/09/2020 22:51

**Para:** Dalton Solano dos Reis <dalton@furb.br>

Opa professor, boa noite!

Falando então sobre o **progresso do TCC nesta semana:**

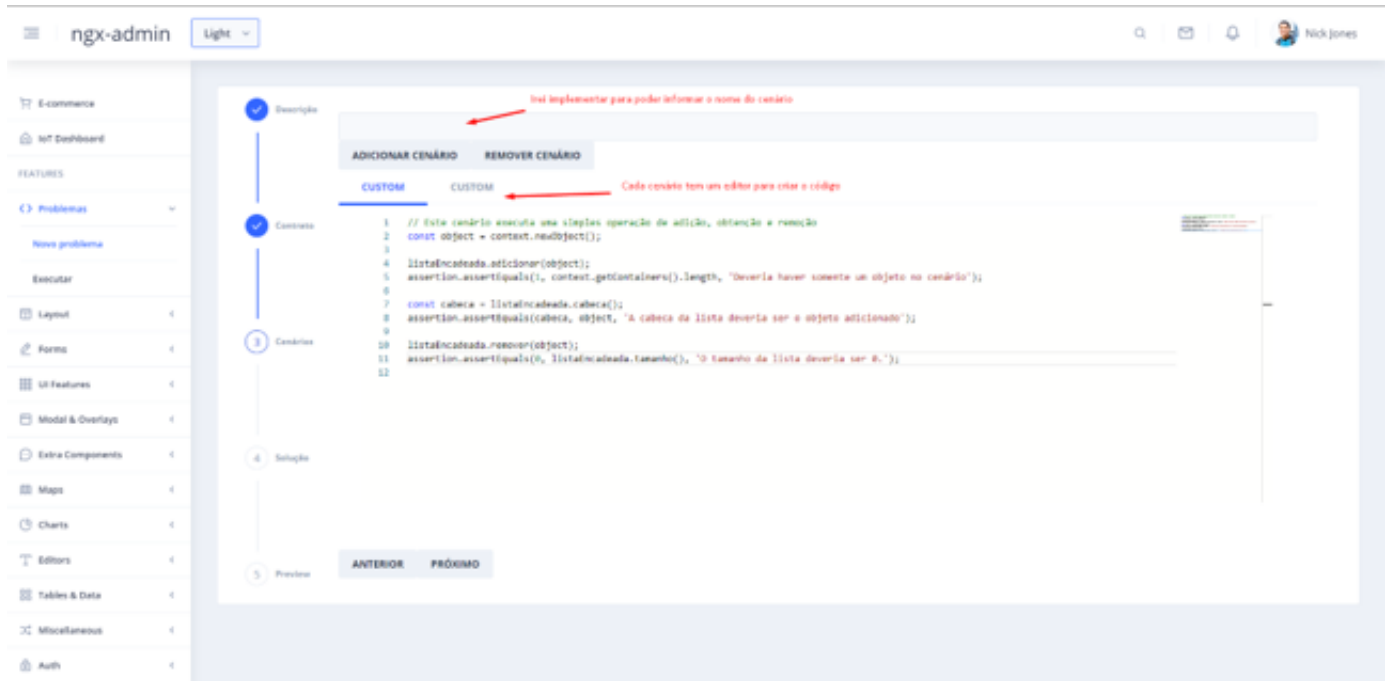
Eu consegui desenvolver algumas telas, acredito que já dê para imaginar o encaminhamento conforme o descrito no projeto que entregamos em TCC I.

Primeiramente eu desenvolvi a tela de cadastro de um contrato. Abaixo está um exemplo:

The screenshot shows the 'ngx-admin' interface with a sidebar on the left containing navigation links like 'E-commerce', 'Dashboard', 'Features', 'Problemas', 'Novo problema', 'Executar', 'Layout', 'Formas', 'UI Features', 'Modal & Overlays', 'Extra Components', 'Maps', 'Charts', 'Editors', 'Tables & Data', 'Miscellaneous', and 'Auth'. The main content area is titled 'Contrato' and shows a configuration form for a contract named 'ListaEncadeada'. The form is divided into four steps: 1. Descrição (Define o nome do contrato), 2. Contrato (Define os campos), 3. Cenários (Define os métodos), and 4. Solução. The 'Contrato' step is currently active, showing a table with columns 'Nome' and 'Descrição (t)'. The table contains one row with 'cabeca' and 'Nó inicial da lista'. There are buttons for adding, editing, and deleting rows. The 'Cenários' step shows a table with columns 'Nome', 'Parâmetros (t)', and 'Descrição (t)'. It contains four rows: 'adicionar' (no), 'contém' (no), 'remove' (no), and 'tamanho' (Obtém o tamanho da lista). The 'Solução' step is currently empty.

Estou permitindo o cadastro em tabela do contrato. No exemplo acima, estou definindo o contrato de uma lista encadeada. São configuráveis 3 valores: o nome do contrato, os campos que deve ter e os métodos.

A segunda que estou terminando de desenvolver agora, é a tela onde será possível definir os cenários de teste em cima desse contrato. Foi a parte que mais tomou tempo, porque tive que me virar um pouco pra conseguir gerar a interface cadastrada pro editor visual e gerenciar alterações ao atualizar ela. Mas no final acredito que consegui um resultado interessante. Abaixo alguns exemplos:

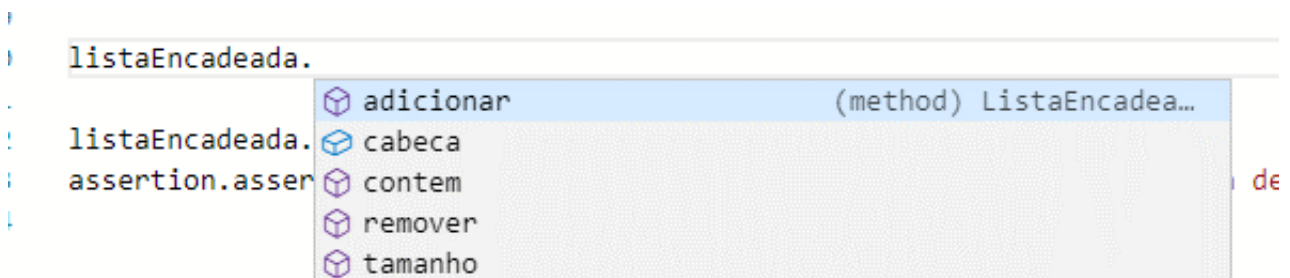


Abaixo o autocomplete para auxiliar na definição do cenário:

```

7  const cabeca = listaEncadeada.cabeca();
8  assertion.assertEquals(cabeca, object, 'A cabeca da lista deveria
9
10 list
11  listaEncadeada const listaEncadeada:...
12  listaEncadeada.remove(object);
13  assertion.assertEquals(0, listaEncadeada.tamanho(), 'O tamanho da

```



```

const cabeca = (method) ListaEncadeada.adicionar(no: any): any
assertion.asser Adiciona um nó na lista
listaEncadeada.adicionar()

```

A interface que gerei para o editor:

```

interface SmalgObjectReadOnly {
    get(name: string);
}

interface SmalgContainerReadOnly {
    get(index: number);
}

interface Context {
    newObject(): SmalgObjectReadOnly;
    newContainer(): SmalgContainerReadOnly;
    getObjects(): SmalgObjectReadOnly[];
    getContainers(): SmalgContainerReadOnly[];
}

declare const context: Context;

interface Assertion {
    assertEquals(expected: any, actual: any, message: string): void;
    assertTrue(actual: any, message: string): void;
    assertFalse(actual: any, message: string): void;
}

declare const assertion: Assertion;

interface ListaEncadeada {
    /** Nó inicial da lista */cabeca: any;
    /** Verifica se um nó está presente na lista */contem(no): any;
    /** Obtém o tamanho da lista */tamanho(): any;
    /** Remove um nó da lista */remover(no): any;
    /** Adiciona um nó na lista */adicionar(no): any;
}

declare const listaEncadeada: ListaEncadeada;

```

Esses cenários configurados pelo professor, eu uso pra executar o código do aluno. Agora tenho que pegar esse contrato e gerar uma classe para ele, onde o aluno irá desenvolver. Quando ele clicar em executar, eu utilizo os cenários definidos na criação do problema. Provavelmente vou adicionar um campo de descrição no cenário, para que quando o aluno selecionar o cenário que quer executar, ele ler uma descrição do professor, dizendo o que aquele cenário faz especificamente.

**O que pretendo fazer:**

Até quinta-feira, pretendo terminar o cadastro de um problema. E até semana que vem integrar a parte da execução com a definição do problema. Não consegui pegar muito a questão da linguagem de programação devido a algumas complicações que tive no desenvolvimento :/

Se o professor tiver alguma contribuição ou conselho eu acho que poderíamos marcar um outro dia para conversarmos. Se o professor acha que o planejamento está correto, por mim tudo bem, posso ir tocando.

Atenciosamente,  
Adriener Maranhão de Andrade

