



**LaDICA**

**Laboratório de Desenvolvimento  
Interdisciplinar de Computação Aplicada**

## **Documento de Casos de Uso**

**Carcará - Simulador de Circuitos Elétricos e Eletrônicos**

LaDICA

**Build 1.0a**



# Histórico de Revisões

<b>Data</b>	<b>Descrição</b>	<b>Autor(es)</b>
10/08/2019	Concepção do documento	Henderson Chalegre
26/05/2020	Casos de Usos da Arquitetura	Lara Esquivel
01/06/2020	Casos de Usos da Arquitetura	Lara Esquivel
09/06/2020	Casos de Uso da Arquitetura	Lara Esquivel

## SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>3</b>
1	Propósito do Documento . . . . .	3
2	Público Alvo do Documento . . . . .	3
3	Escopo . . . . .	3
<b>2</b>	<b>Visão Geral do Documento</b>	<b>4</b>
0.1	Plataforma. . . . .	4
0.2	Conteúdo. . . . .	4
0.3	Descrição da ferramenta . . . . .	4
0.4	Público-Alvo . . . . .	4
0.5	Premissas e restrições . . . . .	4
<b>3</b>	<b>Acrônimos e Abreviações</b>	<b>5</b>
<b>4</b>	<b>Casos de Uso</b>	<b>6</b>
0.1	[UC 001] Salvar Módulo de Cálculo . . . . .	6
0.2	[UC 002] Multiplataforma . . . . .	6
0.3	[UC 003] Armazenar Resultados da Simulação . . . . .	7
0.4	[UC 004] Realizar Simulação . . . . .	7

# 1 | Introdução

## 1. Propósito do Documento

Este documento especifica os casos de uso do projeto do Simulador de Circuitos Elétricos Carcará. a ser desenvolvido pelo Laboratório de Desenvolvimento Interdisciplinar de Computação Aplicada fornecendo aos desenvolvedores as informações iniciais necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema.

## 2. Público Alvo do Documento

Este documento se destina a toda a equipe de desenvolvimento.

## 3. Escopo

Este documento realiza a elicitação dos casos de uso do Simulador de Circuitos Elétricos Carcará

## 2 | Visão Geral do Documento

O presente documento é apresentado como mostrado abaixo:

### **0.1. Plataforma.**

O simulador será desenvolvido para funcionamento em multiplataformas.

### **0.2. Conteúdo.**

Requisitos de software.

### **0.3. Descrição da ferramenta**

### **0.4. Público-Alvo**

Esta ferramenta tem como publico alvo professores, estudantes e profissionais de engenharia

### **0.5. Premissas e restrições**

- A
- a

### 3 | Acrônimos e Abreviações

Sigla	Descrição
UC	Casos de Uso
FR	Requisito Funcional
NFR	Requisito Não Funcional

## 4 | Casos de Uso

### 0.1. [UC 001] Salvar Módulo de Cálculo

Um módulo de cálculo é cadastrado no servidor para posteriormente ser escolhido para realizar a simulação.

*Atores*

Usuário

Módulo de Cálculo.

*Pré-condições*

- O sistema deve ser capaz de enviar dados aos módulos de cálculo por meio de uma API;
- O sistema deve ser capaz de receber dados do módulo de cálculo por meio de uma API;
- Os módulos de cálculo devem ser do tipo arquivo ou tcp;

*Pós-condições*

- O módulo poderá ser escolhido para realizar simulação do circuito;

*Fluxo Principal de Eventos*

- P1. O usuário decide cadastrar um módulo de cálculo;
- P2. O módulo sendo do tipo tcp deve ser fornecido ip, nome e porta;
- P3. O módulo é salvo no servidor;

### 0.2. [UC 002] Multiplataforma

O simulador é multiplataforma. A versão mobile deve ter as mesmas funcionalidades que a desktop.

*Atores*

Usuário

*Pré-condições*

- Deve atender a todos os Requisitos Funcionais.



*Pós-condições*

- Ao ser utilizado em um dispositivo móvel as funcionalidades devem ser mantidas.

*Fluxo Principal de Eventos*

- P1. O usuário acessa o sistema por um computador ou notebook;
- P2. O usuário faz uso do sistema;
- P3. O usuário acessa o sistema por meio de um dispositivo mobile;
- P4. O usuário faz uso do sistema com as mesmas funcionalidades que o sistema desktop;

**0.3. [UC 003] Armazenar Resultados da Simulação**

O resultado da simulação chegará a interface por meio de um JSON enviado da API. O resultado pode ser armazenado no dispositivo.

*Atores*

Usuário

Dispositivo

Módulo de Cálculo

*Pré-condições*

- O módulo de cálculo cadastrado deve receber dados do sistema para efetuar os cálculos;
- O módulo de cálculo cadastrado deve retornar o resultado para a API;

*Pós-condições*

- A interface deve receber um JSON da API;
- O usuário pode armazenar o resultado da simulação;

*Fluxo Principal de Eventos*

- P1. Módulo de cálculo é chamado pela API para efetuar os cálculos;
- P2. O módulo, devolve os resultados a API;
- P3. A API devolve um JSON para a interface;
- P4. Os resultados podem ser salvos num dispositivo;

**0.4. [UC 004] Realizar Simulação**

Dado o circuito desejado pelo usuário, deve ser possível realizar uma simulação do funcionamento desse.

*Atores*

Usuário

Módulo de Cálculo

*Pré-condições*

- O sistema deve enviar dados do circuito para o módulo de cálculo;
- O sistema deve receber dados do módulo de cálculo;
- O módulo de cálculo deve estar cadastrado no servidor;

*Pós-condições*

- A interface deve receber um JSON que corresponde ao resultado da simulação realizada nos módulos de cálculo;

*Fluxo Principal de Eventos*

- P1. O usuário solicita a simulação do circuito desejado;
- P2. O sistema envia os dados para os módulos de cálculo;
- P3. Os módulos de cálculo retornam o resultado da simulação para a API;
- P4. A API deve devolver um JSON para a interface;
- P5. Se o usuário desejar, é possível guardar o resultado da simulação;

*Fluxo Secundário: Alternativo*

**[SF1] Gerando Gráficos ou Tabelas**

---

1. O usuário solicita a simulação do circuito desejado;
2. O sistema envia os dados para os módulos de cálculo;
3. Os módulos de cálculo retornam o resultado da simulação para a API;
4. A API deve devolver um JSON para a interface;
5. É possível visualizar os resultados da simulação em gráficos ou em tabelas.
6. Se o usuário desejar, é possível guardar o resultado da simulação.