



# CALCULADORA

por Gustavo Costa de Andrade & Luiz Eduardo Tojal

Grupo 7

**SAMSUNG**



- Link do repositório: <https://github.com/Grupo7-CTEDS/calculadora>

## 1. Objetivo

*“Queremos desenvolver nossos conhecimentos obtidos no curso CTEDS 2º oferecimento na Poli-USP e também expandir nossos horizontes, pesquisando e aprendendo novas tecnologias e formas de aprimorar nossa produtividade.” – Grupo 7*

## 2. DESCRIÇÃO DO PROJETO

Com base nos temas indicados pelos responsáveis da disciplina D10 - Projeto Final optamos pelo desenvolvimento de uma aplicação de calculadora para Windows. As tecnologias utilizadas são C# no framework WPF.

## 3. REQUISITOS FUNCIONAIS

O programa deve cumprir os seguintes requisitos: realização dos cálculos corretamente, possibilidade de salvar operações, possibilidade de "cópia e cola de operações" e possibilidade de limpar operações. A aplicação deve ser capaz de realizar adição, subtração, multiplicação, divisão, raiz quadrada, porcentagem, resto da divisão, exponenciais, logaritmo e fatorial.

## 4. REQUISITOS NÃO FUNCIONAIS

A aplicação deve ter processamento rápido, código leve, e ambiente agradável (interface e sistema) para o usuário.

## 5. CRONOGRAMA COMPLETO

### Semana 1 - 16/11 - 22/11

#### Cronograma semanal

- Setup do ambiente de desenvolvimento
- Setup da organização e repositório no GitHub

- Implementar algoritmo de cálculo de uma string contendo números e operações básicas
- Aprofundar nosso conhecimento sobre XAML.

## FEEDBACK DO DESENVOLVIMENTO

O desenvolvimento essa semana foi relevante, muito por conta da dedicação a aprender o XAML

### APRENDIZADO DA EQUIPE

Pudemos aprofundar nosso conhecimento em XAML e na utilização de árvores binárias para representar uma sequência de operações.

### AUTO AVALIAÇÃO DOS MEMBROS

**Gustavo C:** Implementar todas as metodologias e funções lógicas que aprendemos até então é um desafio, claro, mas além disso a facilidade de um bom planejamento pode nos levar a harmonia no desenvolvimento da nossa calculadora. Sinto que preciso estudar além do tempo de aula presencial no curso para acompanhar o desenvolvimento, mas tudo bem.

**Luiz Eduardo:** Na primeira semana, tive que sair mais cedo da aula. Contudo realizei progresso em casa, desenvolvendo a função de cálculo da string de operações.

---

## Semana 2 - 23/11 - 29/11

### Cronograma semanal

- Desenvolver uma interface inicial da calculadora
- Implementar a possibilidade da string de operações receber operações mais complexas (raiz quadrada, seno, cosseno, exponencial, fatorial)

## FEEDBACK DO DESENVOLVIMENTO

As tarefas previstas para essa semana foram realizadas, exceto pela implementação da operação de raiz quadrada.

## APRENDIZADO DA EQUIPE

O desenvolvimento feito essa semana permitiu que aprendêssemos mais sobre a programação em XAML

## AUTO AVALIAÇÃO DOS MEMBROS

**Gustavo C:** Tive uma produção um pouco abaixo do planejado pois tive outros compromissos para somar à carga horária semanal. Acredito que na próxima semana meu rendimento será melhor e irei finalizar a interface da calculadora.

**Luiz Eduardo:** Acredito que minha produção essa semana foi razoável uma vez que cumpri o planejamento.

---

## Semana 3 - 30/11 - 6/12

### Cronograma semanal

- Terminar a interface de usuário
- Implementar a persistência

## FEEDBACK DO DESENVOLVIMENTO

## APRENDIZADO DA EQUIPE

Conseguimos implementar a persistência no back-end e quase finalizamos a interface, só falta alguns detalhes de estética e funcionalidades.

## AUTO AVALIAÇÃO DOS MEMBROS

**Gustavo C:** Consegui desenvolver todos os objetivos propostos na semana anterior. Finalmente criei a interface utilizando XAML, depois de muita pesquisa em documentos e fóruns especializados.

**Luiz Eduardo:** Ao longo da última semana tive provas na graduação. Como consequência, não avancei no projeto fora de aula. Contudo, durante a aula passada consegui implementar a persistência de operações em um banco SQLite.

---

## Semana 4 - 7/11 - 14/11

### Cronograma semanal

- Integrar a interface com back-end
- Preparar apresentação

### FEEDBACK DO DESENVOLVIMENTO

#### APRENDIZADO DA EQUIPE

Percebemos que a união da equipe pode ser muito útil quando é necessário resolver algum problema. Teve momentos, por exemplo, que uma chamada online ajudou na parte da integração do front-end e back-end. Facilitando o entendimento da situação e a resolução.

#### AUTO AVALIAÇÃO DOS MEMBROS

**Gustavo C:** Pude resolver, de minha parte, todos os itens que eram importantes para a apresentação. Acredito que foi a semana de desenvolvimento mais gratificante, pois nossos esforços começaram a mostrar resultados.

**Luiz Eduardo:** Nessa semana, contribui com a integração do backend com o front-end e corrige problemas encontrados nas em algumas operações.

---

## Semana 5 - 15/11 - 18/11

### Cronograma semanal

- Organizar a entrega
- Arrumar bugs e pendências no código

## FEEDBACK DO DESENVOLVIMENTO

### APRENDIZADO DA EQUIPE

Essa semana final do curso foi de suma importância para a conclusão dos trabalhos. Conseguimos finalizar itens importantes da entrega e retiramos o que foi necessário do escopo.

### AUTO AVALIAÇÃO DOS MEMBROS

**Gustavo C:** Durante essa semana consegui trabalhar na organização de alguns itens para a entrega final. Como a edição do vídeo e o refatoramento do código para o padrão MVC. Porém precisei tirar do escopo o “Opcional: Testes de unidade” pois não pude implementá-lo suficientemente a fim de garantir uma cobertura satisfatória no projeto.

**Luiz Eduardo:** Implementei operações com números negativos e a operação de exponenciação com base e.

## 6. DOCUMENTAÇÃO BREVE

O Back-end tem cinco classes principais:

- As classes Context e Operation fornecem as bases para realizar a persistência de dados. A Context realiza a criação do banco de dados e a inserção de duas operações de teste. A Operation determina os atributos que uma linha do banco de dados tem.
- A classe Nó representa um nó da árvore binária de operações. Cada operador tem um nó associado a ele e os operandos também terão seus nós, sejam eles números ou outros operadores.
- A classe InfixTree não é instanciada, ela guarda o método de construção da árvore a partir do string de operações e o método de cálculo de um valor a partir da árvore.
- A classe InfixTreeRepositories também não é instanciada e contém uma série de funções auxiliares a construção e cálculo de árvore. Por exemplo, o método format e método de cálculo do fatorial de um número.

O Front-end tem 2 classes XAML/cs principais.

- MainWindowController: Essa classe é literalmente a “main”, isso porque todas as outras se estruturam a partir dela. A principal funcionalidade da classe é controlar a interação do usuário e enviar os dados recebidos para a model(classes do Back-end).
- PersistenciaContasController: Essa classe/window, assim como a main, recebe o resultado da interação, mas não dados e sim informações, considerando que os dados foram processados pela model e repassados via controller para essa view.

Foi utilizado nessas duas windows o XAML do WPF para desenvolver a estrutura visual e "mecânica"(xaml.cs) da janela.

Lembrando que utilizamos o SQLite para a persistência de dados nessa window.

Sobre os testes unitários chegamos a implementar parte deles(como havíamos comentado na apresentação). Mas durante essa semana final de desenvolvimento do projeto decidimos retirá-los do código, já que não alcançamos uma cobertura satisfatória de testes aprovados.

Segue abaixo os links do repositório github e vídeo de apresentação do projeto no youtube.

- <https://github.com/Grupo7-CTEDS/calculadora>
- [https://www.youtube.com/watch?v=8c4lb\\_OmA6s](https://www.youtube.com/watch?v=8c4lb_OmA6s)