

Documento de requisitos

Linguagem utilizada:



Equipe 18:

2314031 – Andrew Ribeiro Pires

2326338 – Igor Carvalho Figueira

2226022 – Maurício Conde

2323748 – Vinícius Linhares Alves de Oliveira

Repositório:

[GitHub](https://github.com/igorcodigo/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADE)

<https://github.com/igorcodigo/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADE>

1. Introdução

1.1 Objetivo

O objetivo deste sistema é o de testar as habilidades do grupo 18 na implementação de código python tendo como norte a aplicação dos conceitos de programação funcional, documentação do código e testes.

2. Definição de Papéis

Nome	Papel	Responsabilidade
Andrew Ribeiro Pires	Documentação	Desenvolver o documento de requisitos.
Igor Carvalho Figueira	Implementação	Desenvolver o código fonte.
Maurício conde	Testes	Aplicação de testes do código.
Vinícius Linhares Alves de Oliveira	Implementação	Desenvolver o código fonte.

3. Requisitos

3.1. Requisitos Funcionais

ID	Descrição	Implementação
01	O sistema deverá ter ao menos uma função Lambda .	Este requisito está implementado dentro do arquivo "lambda.py", sendo declarado na função anônima atribuída a variável 'quadrado'. A função retorna o quadrado de um número inteiro.
02	O sistema deverá ter ao menos uma função contendo uma Closure .	Este requisito está implementado dentro do arquivo "closure.py", sendo evidenciado na função 'hi_message' que cria e retorna a função 'message'.
03	O sistema deverá ter ao menos uma List Comprehension .	Este requisito está implementado dentro do arquivo "list_comprehension.py". O primeiro exemplo está presente na lista 'quadrados_com' e realiza o

ID	Descrição	Implementação
		quadrado de uma lista de números e os armazena. O segundo exemplo está presente na lista ' <i>pares_com</i> ' e filtra os números pares de uma lista e os armazena.
04	O sistema deverá ter ao menos uma função de Alta Ordem .	Este requisito está implementado dentro do arquivo " <i>higher_order.py</i> ", sendo descrita na função ' <i>define_operation</i> ' que recebe uma função ' <i>Fn</i> ' como argumento para realizar operações matemáticas com os parâmetros ' <i>a</i> ' e ' <i>b</i> '.

3.2. Requisitos Não Funcionais

ID	Descrição	Implementação
01	O código deve ser escrito de forma clara e legível, facilitando o entendimento dos conceitos abordados.	As partes mais relevantes contém comentários explicativos.
02	O código deve ser modular	O código está dividido em contextos

4. Construções de Programação Funcional

Construção	Código Relacionado
Função Lambda	<code>quadrado = lambda x: x ** 2</code>
Closure	<pre>def hi_message(nome, inicial): def message(): nonlocal inicial inicial += 1 return f"Olá, {nome}! Esta é minha {inicial}ª saudação." return message</pre>
List Comprehension 1	<code>quadrados_com = [num ** 2 for num in range(1, 11)]</code>
List Comprehension 2	<code>pares_com = [num for num in numeros if num % 2 == 0]</code>

High Order	<pre>def define_operation(fn, a, b): return fn(a, b)</pre>
------------	--

5. Casos de Teste

A verificação da funcionalidade dos códigos foi realizada por meio de testes manuais, utilizando a função `print()` do Python para exibir os resultados no console. Os prints foram configurados para apresentar os valores calculados ou processados pelas funções e algoritmos.

O código foi validado ao comparar os resultados exibidos nos prints com os resultados esperados. Quando o código produziu os resultados corretos, os outputs exibidos coincidiram com os valores esperados, confirmando o bom funcionamento do programa.

5.1. Função Lambda

```
(base) conde@creed:~/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES$ cd /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES ; /usr/bin/env /bin/python3 /home/conde/.vscode/extensions/ms-python.debugpy-2025.4.1-linux-x64/bundled/libs/debugpy/adapter/../../debugpy/launcher 42017 -- /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES/lambda.py  
O quadrado de 1 é 1  
O quadrado de 2 é 4  
O quadrado de 3 é 9  
O quadrado de 4 é 16  
O quadrado de 5 é 25  
O quadrado de 6 é 36  
O quadrado de 7 é 49  
O quadrado de 8 é 64  
O quadrado de 9 é 81  
O quadrado de 10 é 100
```

5.2. Closure

```
(base) conde@creed:~/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES$ /usr/bin/env /bin/python3 /home/conde/.vscode/extensions/ms-python.debugpy-2025.4.1-linux-x64/bundled/libs/debugpy/adapter/../../debugpy/launcher 47139 -- /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES/closure.py  
<function hi_message.<locals>.message at 0x74fc65b4a160>  
Olá, Vini! Esta é minha 1ª saudação.  
Olá, Vini! Esta é minha 2ª saudação.  
Olá, Vini! Esta é minha 3ª saudação.  
Olá, Ana! Esta é minha 10ª saudação.  
Olá, Ana! Esta é minha 11ª saudação.  
Olá, Ana! Esta é minha 12ª saudação.
```

5.3. List Comprehension 1 e 2

```
(base) conde@creed:~/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES$ cd /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES ; /usr/bin/env /bin/python3 /home/conde/.vscode/extensions/ms-python.debugpy-2025.4.1-linux-x64/bundled/libs/debugpy/adapter/../../debugpy/launcher 39947 -- /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES/list_comprehension.py  
Quadrados sem List Comprehension: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]  
Quadrados com List Comprehension: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]  
Pares sem List Comprehension: [2, 4, 6, 8, 10]  
Pares com List Comprehension: [2, 4, 6, 8, 10]
```

5.4. High Order

```
(base) conde@creed:~/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADES$ cd /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADE ; /usr/bin/env /bin/python3 /home/conde/.vscode/extensions/ms-python.debugpy-2025.4.1-linux-x64/bundled/libs/debugpy/adapter/../../debugpy/launcher 38597 -- /home/conde/Documents/Estudos/Unifor/PROGRAMACAO-FUNCIONAL-TRABALHO-FACULDADE/higher_order.py
12
8
26
```

6. Uso de Chatbots

Durante o desenvolvimento, utilizamos para gerar exemplos, tirar dúvidas e explicar alguns contextos relacionados aos conceitos de programação Funcional.

Além disso, foi utilizado ChatGPT para o desenvolvimento de um template básico de um documento de requisitos, o qual modificamos e adaptamos para nossa proposta.