



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

TRABALHO FINAL - LINGUAGENS DE PROGRAMAÇÃO

LÍGIA CALINA BUENO BONIFÁCIO DRE: 122046065

Rio de Janeiro-RJ
2024

SUMÁRIO

1	INTRODUÇÃO	2
2	INTERFACE	3
2.1	DESCRIÇÃO TÉCNICA	3
2.1.1	Entradas e Saídas da Interface	3
2.1.2	Desenho Ilustrativo da Arquitetura	4
2.1.3	Formato das Chamadas e Interação	4
2.1.3.1	Opção 1 - Adicionar tarefa	4
2.1.3.2	Opção 2 - Listar Tarefas	4
2.1.3.3	Opção 3 - Analisar tarefa	4
2.1.3.4	Opção 4 - Remover tarefa	5
2.1.3.5	Opção 5 - Marcar tarefa como concluída	5
2.1.3.6	Opção 6 - Listar tarefas concluídas	5
2.1.3.7	Opção 0 - Sair	5

1 INTRODUÇÃO

O objetivo deste projeto é criar um sistema de gerenciamento de tarefas que permita aos usuários adicionar, listar, analisar, remover e marcar tarefas como concluídas. O sistema é dividido em duas partes principais: um gerenciador de tarefas em C++ e um módulo de análise de texto em Python. O gerenciador em C++ cuida da manipulação e armazenamento das tarefas, enquanto o script em Python realiza a análise detalhada das descrições das tarefas, identificando palavras-chave e contando palavras. Este projeto demonstra a integração entre C++ e Python, destacando a eficiência da combinação dessas duas linguagens para diferentes propósitos.

As opções do menu incluem:

1. Adicionar tarefa;
2. Listar tarefas;
3. Analisar tarefa;
4. Remover tarefa;
5. Marcar tarefa como concluída;
6. Listar tarefas concluídas;
7. Sair.

2 INTERFACE

2.1 DESCRIÇÃO TÉCNICA

O programa gerenciador em C++ é responsável por toda a interação com o usuário e o gerenciamento das tarefas. Suas responsabilidades incluem:

1. Adicionar novas tarefas à lista;
2. Listar todas as tarefas atuais;
3. Remover tarefas específicas;
4. Marcar tarefas como concluídas;
5. Listar tarefas que foram concluídas;
6. Exportar a descrição de uma tarefa para um arquivo e chamar um script Python para análise.

As funções externas, implementadas em Python, são responsáveis pela análise detalhada do texto das tarefas. Especificamente, o script Python:

1. Lê a descrição da tarefa a partir de um arquivo de texto;
2. Realiza a contagem de palavras na descrição;
3. Identifica palavras-chave na descrição, baseando-se em critérios como comprimento mínimo;
4. Retorna os resultados da análise, que são exibidos no terminal.

2.1.1 Entradas e Saídas da Interface

A comunicação entre o programa C++ e o script Python é realizada através de arquivos de texto e comandos do sistema.

Aqui estão os detalhes:

- Entrada para o Script Python: Um arquivo de texto (**'tarefa.txt'**) contendo a descrição da tarefa.
- Saída do Script Python: Resultados da análise impressos no terminal, incluindo a contagem e a lista de palavras-chave.

2.1.2 Desenho Ilustrativo da Arquitetura

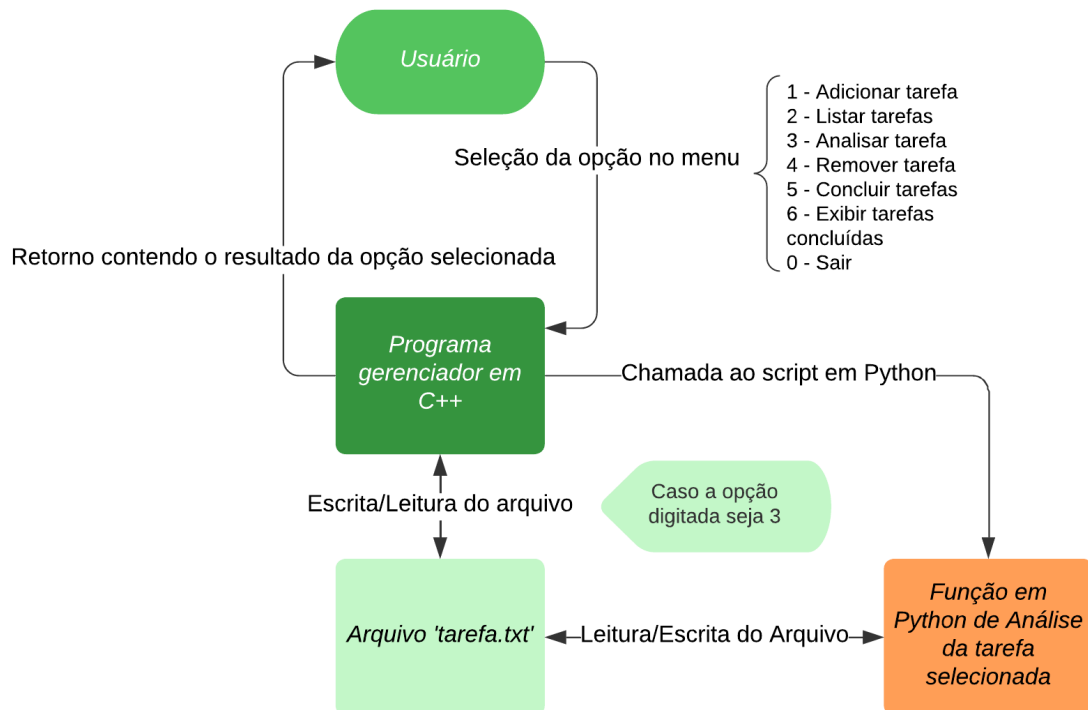


Figura 1 – Desenho da arquitetura do programa

2.1.3 Formato das Chamadas e Interação

2.1.3.1 Opção 1 - Adicionar tarefa

O usuário escreve uma string referente a tarefa. Cada nova tarefa é adicionada a um vetor de tarefas em memória *vector<string>*. O vetor armazena todas as tarefas correntes do usuário.

2.1.3.2 Opção 2 - Listar Tarefas

As tarefas escritas no vetor serão printadas no terminal ao lado de seu índice para que o usuário tenha controle sobre as tarefas em sua lista.

2.1.3.3 Opção 3 - Analisar tarefa

Quando o usuário seleciona a opção de "Analisar Tarefa", o programa em C++ busca a descrição da tarefa a partir do vetor de tarefas armazenadas. Em seguida, o programa escreve a tarefa selecionada no arquivo (*tarefa.txt*) e chama o script Python que realizará a análise.

O usuário escolhe uma tarefa específica da lista de tarefas armazenadas no vetor. O programa em C++ então:

- Lê a tarefa selecionada do vetor.
- Escreve a descrição da tarefa no arquivo de texto (*tarefa.txt*).
- Utiliza um comando do sistema para chamar o script Python (*analisaTarefa.py*), passando o caminho do arquivo como argumento.

2.1.3.4 Opção 4 - Remover tarefa

As tarefas são mostradas ao usuário para que ele selecione o índice da tarefa que ele deseja remover e, então, ela é retirada da lista.

2.1.3.5 Opção 5 - Marcar tarefa como concluída

As tarefas são mostradas ao usuário para que ele selecione o índice da tarefa que ele deseja marcar como concluída. Então, ela é retirada do vetor de tarefas é adicionada a um vetor de tarefas concluídas em memória *vector<string>*. O vetor armazena todas as tarefas concluídas do usuário.

2.1.3.6 Opção 6 - Listar tarefas concluídas

As tarefas escritas no vetor serão printadas no terminal ao lado de seu índice para que o usuário tenha controle sobre as tarefas concluídas.

2.1.3.7 Opção 0 - Sair

Encerra a execução do programa.