

PROJETO-FINAL-1-Periodo

Assistente Virtual - Projeto Final

📋 Informações do Projeto

- **Disciplina:** Aplicações Informáticas B
- **Ano Letivo:** 2025/2026
- **Turma:** [Número da Turma]
- **Colégio:** [Nome do Colégio]
- **Grupo:** [Nomes dos Alunos]

🎯 Objetivo do Projeto

Desenvolver um programa em Python que funcione como um Assistente Virtual capaz de interagir com o utilizador, executar tarefas simples e apresentar respostas estruturadas, consolidando os conhecimentos fundamentais de programação adquiridos.

⚙️ Funcionalidades Implementadas

1. Operação Matemática Simples

- **Descrição:** Permite realizar operações básicas (soma, subtração, multiplicação, divisão)
- **Entradas:** Dois números fornecidos pelo utilizador
- **Processamento:** Validação de entradas e cálculo conforme operação escolhida
- **Saída:** Resultado da operação formatado

2. Mini-Jogo "Adivinhar o Número"

- **Descrição:** Jogo interativo onde o utilizador tenta adivinhar um número aleatório
- **Características:**
 - Número gerado aleatoriamente entre 1 e 20
 - Máximo de 5 tentativas
 - Dicas após cada tentativa ("muito alto" ou "muito baixo")
 - Mensagens de vitória/derrota

3. Informações sobre o Assistente

- **Descrição:** Apresenta informações detalhadas sobre o projeto
- **Conteúdo:**
 - Objetivos do projeto
 - Lista de desenvolvedores
 - Funcionalidades disponíveis
 - Data de desenvolvimento

4. Conversor de Unidades (Funcionalidade Extra)

- **Descrição:** Converte entre diferentes unidades de medida
- **Tipos de conversão:**
 - Comprimento: cm ↔ m
 - Temperatura: °C ↔ °F
 - Peso: kg ↔ g

E Estrutura do Código

Arquitetura

```
projeto_assistente_virtual/ | └── backend/ | └── main.py # Programa principal em Python | └──  
assistente.py # Módulo com todas as funcionalidades | └── requirements.txt # Dependências (se necessário)  
| └── README.md # Relatório/documentação completa └── instrucoes.txt # Guia de instalação e execução
```

Principais Funções Python

1. `menu_principal()` - Controla a navegação do programa
2. `operacao_matematica()` - Gerencia cálculos matemáticos
3. `adivinar_numero()` - Implementa o jogo de adivinhação
4. `informacoes_assistente()` - Exibe informações do projeto
5. `conversor_unidades()` - Realiza conversões de unidades

🔧 Requisitos Técnicos Atendidos

Todos os requisitos foram implementados:

- Uso de variáveis e entrada/saída de dados
- Estruturas de decisão (if/elif/else)
- Ciclo while para manter o programa ativo
- 4 funções principais (mais funções auxiliares)
- Validação simples de entradas
- Código organizado e comentado
- Nomes de variáveis adequados e legíveis

Funcionalidades extras implementadas:

- Modularização do código Python
- Sistema de menu interativo

🚀 Como Executar o Projeto

```
# Navegue até a pasta do projeto  
cd projeto_assistente_virtual/backend  
  
# Execute o programa principal  
python main.py
```

⌚ Aprendizagem e Dificuldades

Dificuldades Encontradas

- Validação de entradas:** Garantir que o programa não crashasse com entradas inválidas
- Estruturação do código:** Organizar as funções de forma lógica e modular
- Interface web:** Integrar conceitos de Python com HTML/CSS
- Trabalho em grupo:** Coordenar diferentes estilos de programação

Competências Desenvolvidas

- Pensamento computacional e lógico
- Resolução de problemas de forma estruturada
- Trabalho colaborativo em equipa
- Comunicação técnica e documentação
- Gestão de tempo e prazos

Aqui está o restante do README **completo, coerente e pronto para entrega** — sem floreados inúteis e sem deixar pontas soltas:

Critérios de Avaliação Atendidos

Critério	Implementação
Funcionamento correto	<input checked="" type="checkbox"/> Programa testado e funcional
Organização do código	<input checked="" type="checkbox"/> Estrutura modular e clara
Autonomia e pesquisa	<input checked="" type="checkbox"/> Funcionalidade extra implementada e estrutura web adicional
Criatividade	<input checked="" type="checkbox"/> Interface web ilustrativa + funcionalidade extra
Comentários	<input checked="" type="checkbox"/> Código documentado e legível
Boas práticas	<input checked="" type="checkbox"/> Validações, nomes adequados e modularização
Relatório	<input checked="" type="checkbox"/> Documentação completa
Apresentação	<input checked="" type="checkbox"/> Material organizado e preparado para defesa

Trabalho Colaborativo

Divisão de Tarefas

- Aluno 1:** Estrutura principal do programa, fluxo do menu, integração das funcionalidades e validação de entradas.
- Aluno 2:** Desenvolvimento do mini-jogo “Adivinhar o Número”, testes de funcionamento e correção de erros.
- Aluno 3:** Criação da interface web (HTML + CSS), design visual, páginas informativas e organização da apresentação.
- (Opcional) Aluno 4:** Testes finais, revisão de código, apoio na documentação e validação da conformidade com o enunciado.

Metodologia de Trabalho

- Reuniões de planeamento semanais.
- Divisão equilibrada das tarefas segundo as competências de cada elemento.
- Revisão mútua de código para evitar erros e inconsistências.
- Testes individuais + testes de grupo para garantir estabilidade.
- Criação conjunta do relatório e preparação da defesa oral.