

Apresentação de plano de curso e avaliação Final

Projeto Final Integrador: "SmartNet Dashboard"

Monitor de Infraestrutura e Performance Inteligente

1. O Conceito

Vocês não vão apenas criar um site ou configurar uma rede isolada. Vocês serão responsáveis por desenvolver o **SmartNet Dashboard**, uma interface web que monitora o status de uma infraestrutura de rede que vocês mesmos configurarão.

O sistema deve ser capaz de informar, em tempo real, se os dispositivos da rede (roteadores, servidores ou PCs simulados) estão "Online" ou "Offline".

2. A Integração (O "Pulo do Gato")

Para este projeto, utilizaremos as três camadas de conhecimento:

- **Camada de Infraestrutura (Redes):** Vocês montarão e configurarão a topologia da rede no laboratório.
- **Camada de Inteligência (Python/Flask):** Vocês escreverão scripts que "perguntam" para a rede se ela está viva (usando comandos de rede via Python).
- **Camada de Interface (HTML/CSS - Reuso):** Vocês pegarão as páginas web que criaram no módulo anterior e as transformarão em um painel de controle (Dashboard) dinâmico, onde os dados da rede serão exibidos.

3. Fases do Desenvolvimento

O projeto será desenvolvido de forma modular, acompanhando as aulas:

1. **Fase de Design (Semana 1-4):**
 - Adaptação do seu layout HTML/CSS anterior para receber dados dinâmicos.
 - Criação da topologia de rede no Cisco Packet Tracer.
2. **Fase de Implementação Backend (Semana 5-12):**
 - Configuração do servidor Flask.
 - Criação das rotas e lógica em Python para realizar o monitoramento dos dispositivos.
3. **Fase de Integração Final (Semana 13-20):**
 - Conexão do Python com o HTML (usando Jinja2).
 - Testes de estresse (o que acontece se um cabo for "desconectado" na simulação?).
 - Apresentação do Dashboard funcionando.

4. Critérios de Avaliação (Rubrica)

Para garantir transparência, a nota final (de 0 a 10) será composta pelos seguintes pilares:

Critério	Peso	Descrição do que será avaliado
Infraestrutura de Rede	3.0	Topologia funcional, endereçamento IP correto e conectividade testada.
Lógica de Programação	3.0	Código Python limpo, uso correto de bibliotecas e tratamento de erros.
Integração Web	2.0	Uso eficiente do Flask para renderizar dados dinâmicos no seu HTML/CSS.
Funcionalidade/UX	1.0	O Dashboard é intuitivo e responde corretamente ao status da rede.
Documentação/Defesa	1.0	Apresentação clara do projeto e capacidade de explicar o fluxo de dados.

5. Por que este projeto?

Como profissionais de TI, vocês raramente trabalharão apenas com redes ou apenas com programação. No mercado, o profissional de destaque é o **DevOps** ou o **Engenheiro de Redes com automação** — alguém que sabe configurar o hardware (Redes) e automatizar o controle desse hardware via software (Python).

Este projeto é o primeiro passo para vocês se tornarem profissionais de alto nível, capazes de enxergar o sistema completo, do cabo de rede até a interface que o usuário final vê.