
Laboratório 28: Construindo a Experiência do Usuário (UX)

Objetivo do Laboratório

O objetivo desta atividade é construir o Front-end do projeto e realizar a comunicação (*fetching*) com os *Endpoints* desenvolvidos no laboratório passado. O produto da *Startup* da equipe deve sair das telas de código e terminal e passar a operar visualmente em um Navegador Web.

Atividade Prática

1. Configuração do Servidor (CORS)

A primeira barreira ao separar o Front-end do Back-end no navegador é o *Cross-Origin Resource Sharing* (CORS). Certifique-se de que o seu servidor Python ou Java autorize conexões locais.

No FastAPI:

```
from fastapi.middleware.cors import CORSMiddleware
app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"], # Atenção: Em produção limite isso para a URL do Front
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)
```

2. Escrevendo a Interface

Divida o trabalho na equipe. Um membro deve criar os componentes gráficos visuais (seja num arquivo `index.html` isolado, num projeto `React` ou numa aba `Streamlit`).

Concentre-se em construir:

- Um contêiner que simule uma tela de conversa.

-
- Um campo de entrada (*input text*) e botão de Enviar.
 - Um indicador de "A IA está pensando...".

3. O Fetch (Comunicação)

Faça a interligação lógica. O Front-end não raciocina, apenas espelha dados.

```
// Exemplo em JavaScript nativo atrelado a um botão "Enviar":
async function enviarMensagem() {
    const texto = document.getElementById("inputUsuario").value;
    mostrarNoChat(texto, "usuario");
    mostrarLoading();

    try {
        const resposta = await fetch("http://localhost:8000/api/chat", {
            method: "POST",
            headers: { "Content-Type": "application/json" },
            body: JSON.stringify({ mensagem: texto })
        });
        const dados = await resposta.json();
        removerLoading();
        mostrarNoChat(dados.resposta, "ia");
    } catch (erro) {
        alert("O servidor de IA está offline ou com problemas.");
    }
}
```

4. Testes de Usabilidade

Passe 15 minutos do final da aula tentando "quebrar" a própria interface. O que acontece se o usuário clicar no botão de enviar 10 vezes seguidas? A UI trata isso ou a IA responde 10 vezes? Corrijam esses pequenos "bugs" de usabilidade.

Critério de Avaliação

O laboratório 28 é validado com sucesso pela submissão do código visual e comprovação presencial ou via print da aplicação consumindo com sucesso o servidor Backend construído na aula 27, diretamente de uma página web.