

## ***Planejamento - Projeto - Studex***

- Python/MySQL/Flask
- HTML/CSS
- API Google Maps

### **Recursos do Site (Ordem de Desenvolvimento - Front):**

1. Tela de Login
2. Forms
3. Perfil - Interface
4. Homepage (Nav+Botões)
5. Pesquisa de Nomes (Pokédex)
6. Mapa

### **Ordem de Desenvolvimento (Back):**

1. Funcionamento do Registro + Login
2. Funcionamento do Forms + Usuários no BD
3. Perfil único pegando os dados do banco (pode editar)
4. Pesquisa para procurar os perfis já cadastrados
5. Mapa que mostra os perfis logados na aplicação

### **Trilha (Studex):**

- Registro
- Login
- Formulário
- Perfil
- Pesquisa (Pokédex)
- Mapa

Relacionamento entre duas tabelas...

**tabela de usuário (login):**

login: e-mail (primary key) → identificador

senha: senha

**tabela de formulário (perfil)**

id →

nome →

email → mesmo email do login para associar as duas tabelas

ra →

2\_ling\_pref →

tempo\_de\_prog →

s\_o →

js →

py →

c →

htmlcss →

java →

resumo\_pessoal →

—

INSERT E UPDATE na tabela do forms..

—

**ordem-de-paginas:**

**primeiro acesso**

formulário → login → index(perfil, mapa, pesquisa)

**já tem um conta**

index → pode editar, pesquisar, mapa, ver o index...

## O que precisa ser feito:

### -Perfil

- Conseguir puxar os dados do banco e inserir na página HTML com o Python; ✓
- Fazer a função de editar todos os campos;
- Permitir que a pessoa consiga ver/editar o perfil só quando logada;
- Deixar o link de cada perfil com perfil/ra ou perfil/nome;
- (Eu) só consigo editar o (meu) perfil autenticado;
- Colocar uma imagem aleatória de um Pokémon na foto do perfil (Usar API ou fazer uma lista).

### -Pesquisa

- Desejável: Fazer uma pesquisa que atualize a medida que o usuário digite, mostrando os perfis adequados ao input;
- Aceitável: Quando ele clicar em buscar (ENTER) mostra todos os resultados adequados à pesquisa;
- Filtros:
  - Filtros para indicar os melhores usuários em cada linguagem;
  - Filtro de Semestre;
  - Filtro de Sistema Operacional;
  - Filtro de Tempo de Programação.
- Quando pesquisar+clicar em algum usuário, o /perfil/ deve ser mostrado.

## **\*\*Mapa**

### **- Opções:**

1. Quando o usuário logar, solicitar a localização atual e mostrar no mapa somente usuários logados (como se fosse o snap);
2. Assim que a pessoa se registrar no forms, inserir no banco de dados a posição dela. Assim, mostrar no mapa todos os usuários cadastrados;
3. ...

<https://github.com/flask-extensions/Flask-GoogleMaps>

<https://developers.google.com/maps/documentation/places/web-service/get-api-key>

<https://morioh.com/p/f458678a9ffa>

## **Sobre**

- Explicar o Time e as funções de todos no projeto

## **Como funciona**

- Explicar o Funcionamento e Cada seção e requisitos do site