

Quadros

Trabalho IoT Camargo

CAMARGO TEAM Free | Visível para o Time | PH SF Convidar

Backlog

- Reunião (18 de set)
- Definição das tasks de cada integrante (18 de set) PH
- Definição da estrutura back-end do projeto (18 de set) PH
- Manter o Github atualizado com os códigos
- Como extrair imagem do Aluno?
- Como enviar a imagem para o servidor?
- Como acessar o banco de dados do servidor?
- Como integrar Django com scripts externos?

A Fazer

- Integrar API de Reconhecimento do Docker/Heroku com o Django (PH)
- Integrar Front com Back (PH SF)

Em andamento

- Script para acessar a câmera e extrair a imagem (30 de out) PH

Fase de teste

- Estudar a possibilidade de deploy no heroku. (21 de set) 2 0/2

Concluído

- Preencher Seções 1 e 2 do Projeto (26 de out) SF
- Criar Artefatos e Planejamento do projeto (26 de out) PH SF
- Criar um banco de dados local com as imagens (SF)
- SEMINÁRIO CAMARGO (5 de out)
- Validar uso de script externo no Django

Diagrama de Fluxo:

```

graph LR
    subgraph CLIENTE [CLIENTE]
        direction TB
        C[HTML  
start  
se ok, OK  
SE DESEJAR]
    end
    subgraph SERVIDOR [SERVIDOR]
        direction TB
        S[Django  
[Java, OK]]
        I[Imagem]
        S -- "jpg" --> I
        I -- "ok" --> S
        I -- "jpg -> string" --> S
    end
    C -- "jpg" --> S
    
```

Script Python (Visualizado na coluna Concluído):

```

class CustomerListView(ListView):
    template_name = "customer/customer-list.html"
    model = Customer
    paginate_by = 2

    def get_queryset(self):
        name = self.request.GET.get("name")

        if name:
            rec = facerec6
            result = rec.procurar("imagem.jpg")
            print(result)
        else:
            result = None
    
```