

André Luiz Re Filho

PORTFÓLIO ACADÊMICO

André Luiz Re Filho

PORTFÓLIO ACADÊMICO

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Araras - Antônio Brambilla, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientadores:

Prof. Me Daniel Dos Santos Robledo

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

André Luiz Re Filho

PORTFÓLIO ACADÊMICO

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Araras - Antônio Brambilla, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientadores:

Prof. Me Daniel Dos Santos Robledo

Banca Examinadora:

Prof. Me Daniel Dos Santos Robledo- Orientador
Faculdade de Tecnologia de Araras – Antônio Brambilla

Prof. Me Júlio César de Lemos - Orientador
Faculdade de Tecnologia de Araras – Antônio Brambilla

Prof. Dr Leonardo de Souza Lima – Coordenador do Curso de Sistemas para Internet
Faculdade de Tecnologia de Araras – Antônio Brambilla

Prof. Euclides Armando Santos
Convidado Externo

Araras, ____ de _____ de 2021

Figuras

<i>Figura 1 - Estrutura do Framework Django</i>	6
<i>Figura 2 - Página de login da Plataforma Fatec</i>	8
<i>Figura 3 - Página de entrada da Plataforma Fatec</i>	9
<i>Figura 4 - Página de entrada do Módulo Vitrine dentro da Plataforma Fatec</i>	9
<i>Figura 5 - Página de cadastro para estágio e/ou emprego</i>	12
<i>Figura 6 - Página de listagem de alunos para vaga de estágio e/ou emprego</i>	13
<i>Figura 7 - Página para remover cadastro dos alunos</i>	14

Sumário

PROJETO 2 – Plataforma FATEC(Módulo Vitrine)	7
INTRODUÇÃO	9
REVISÃO BIBLIOGRÁFICA (Arrumar)	10
MÉTODOS E FERRAMENTAS	11
RELATO DO PROCESSO	13
RESULTADOS	20
CONSIDERAÇÕES	21
REFERÊNCIAS	22

PROJETO 2 – *Plataforma FATEC (Módulo Vitrine)*

RESUMO

No meio da pandemia de 2020, onde as aulas em modelo presencial foram substituídas pelas remotas, as de vagas de emprego e estágio que ficavam nos quadros de anúncios da FATEC deixaram de ser vistas e uma solução para isso foi utilizar o Sistema Plataforma FATEC e adicionar o módulo vitrine para os alunos poderem se cadastrar e terem visibilidade para as empresas contratantes.

Palavras-chave: *Python, Django, CRUD, Sistemas para Internet*

ABSTRACT

During the 2020 pandemic, where in-person classes were replaced by remote ones, the job and internship vacancies that were on FATEC's bulletin boards were no longer seen, and one solution to this was to use the FATEC Platform System and add the showcase module so that students could register and have visibility to hiring companies.

Keywords: *Python, Django, CRUD, Internet Systems.*

INTRODUÇÃO

No período em que um estudante está se graduando, é necessário que o aluno realize a matéria de estágio em algum momento de sua formação, ou esperando encontrar uma vaga de emprego para entrar no mercado de trabalho.

Levando em consideração essa necessidade do aluno, foi criado através do Projeto Plataforma da FATEC Araras, realizado por Prof. Me. Orlando Saraiva do Nascimento Júnior, Prof. Dr. Leonardo de Souza Lima e Profa. Me. Dheborá Souza Umbelino Silva, um módulo denominado Vitrine que tem como objetivo criar registros de alunos fatecano para vagas de emprego, estágio ou ambos, e estes sendo expostos no mesmo módulo para empresas interessadas. O projeto foi desenvolvido pelos alunos do 6º Semestre do curso de Sistemas para Internet, esses André Luiz Re Filho, Caio Teixeira, Marcelo Eliseu, Matheus Delmunde e Vinicius Vieira, utilizando a linguagem *Python* juntamente com o *framework Django*.

REVISÃO

Em relação a anos anteriores, o ano de 2021 demonstrou crescimento no mercado de TI (Tecnologia da informação). Através de pesquisa da empresa brasileira de consultoria *advance*(2021), no primeiro e segundo trimestre de 2021 registrou-se um crescimento de 15,7% e 26,5% em comparação ao mesmo período de 2020, respectivamente, esperando que cresça no mínimo 21% em relação a todo o ano.

Conforme Silva (2021), em um levantamento feito pela Associação Brasileira das Empresas de Tecnologia da Informação e Comunicação (Brasscom), em 2024 o mercado nacional necessitará de 420 mil profissionais na área de TI, porém anualmente se formam 46 mil profissionais. Além disso, em notícia do próprio portal a *International Data Corporation (IDC)* (2021), informa que o crescimento para o mercado de TI no Brasil foi de 11% em 2021.

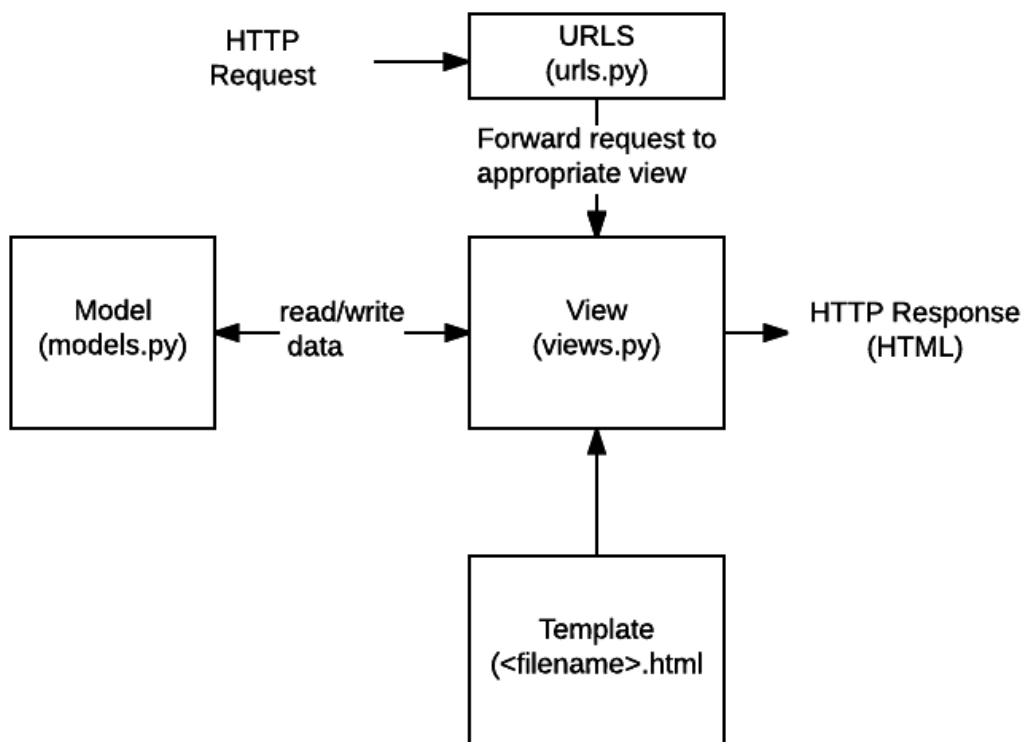
MÉTODOS E FERRAMENTAS

Este projeto foi desenvolvido com base na linguagem *python* que é uma linguagem de programação de alto nível, orientada a objetos e de tipagem dinâmica e forte, juntamente com o seu *framework Django*.

Ser de alto nível, significa resumidamente que é uma linguagem bem próxima a humana, orientada a objetos, significa não ser escrita em somente um fluxo estrutural, e finalmente tipagem dinâmica e forte, significa que um atributo não está preso a um tipo, e não é possível fazer operações misturando tipos, respectivamente.

O *framework Django* consiste em uma estrutura com ferramentas já prontas, baseada em *Python*, que permite o desenvolvimento de sites de forma rápida, segura e de fácil manutenção. Segundo explicação da comunidade *Mozilla* (2021), o *framework Django* segue a seguinte estruturação:

Figura 1 - Estrutura do Framework Django



Fonte: <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Django/Introduction> (2021)

URLs são utilizadas para redirecionar solicitações HTTP para a *view* apropriada com base na *URL* da solicitação. O mapeador de *URLs* também pode corresponder a padrões específicos de *strings* ou dígitos que aparecem em uma *URL* e transmiti-los a uma função *view* como dados.

View acessam os dados necessários para satisfazer solicitações por meio dos *models* e encarregam a formatação da resposta aos *templates*.

Models são objetos em *Python* que definem a estrutura dos dados de um aplicativo, e fornece mecanismos para gerenciar (adicionar, modificar e excluir) e consultar registros no banco de dados.

Templates é um arquivo de texto que define a estrutura ou o *layout* de um arquivo, com espaços reservados usados para representar o conteúdo real.

RELATO DO PROCESSO

Na página de login da Plataforma FATEC (Figura 2), foi adicionado a opção de “Acesso à Vitrine”, onde se pode ter acesso a lista dos alunos cadastrados (Figura 6).

Código fonte html: acesso a página do módulo vitrine(listagem)

```
<div class="text-center">
  <a class="small" href="{% url 'vitrine:showcase' %}">Acesso à Vitrine</a>
</div>
```

Figura 2 - Página de login da Plataforma Fatec



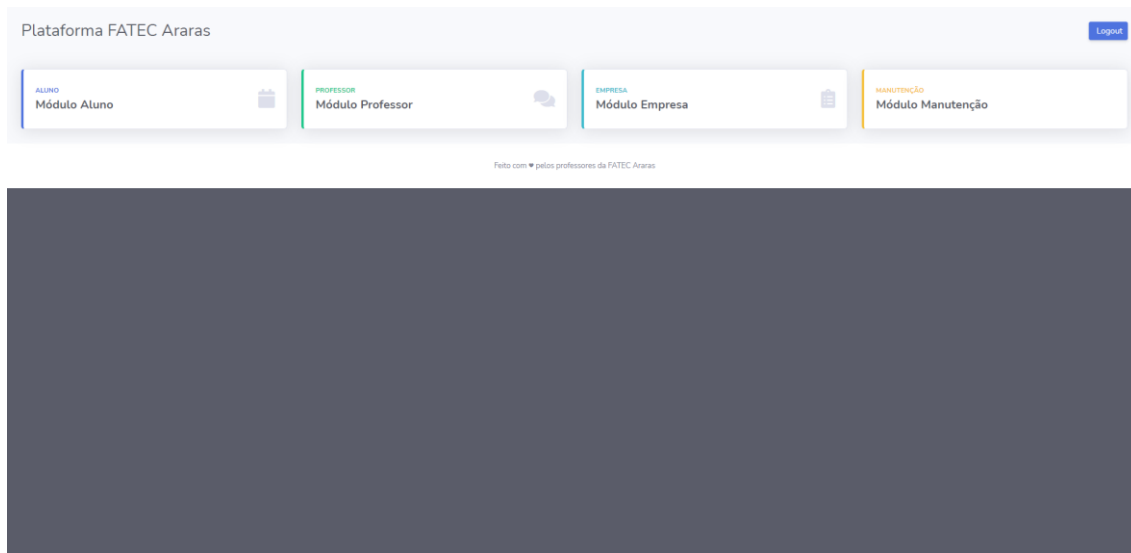
Fonte: de autor (2020)

Código fonte html: acesso a página do módulo vitrine(cadastro/deletar)

```
<!-- Módulo Aluno -->
<div class="col-xl-3 col-md-6 mb-4">
  <a href="{% url 'core:core_index_aluno' %}">
    <div class="card border-left-primary shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-primary text-uppercase mb-1">Aluno</div>
            <div class="h5 mb-0 font-weight-bold text-gray-800">Módulo Aluno</div>
          </div>
          <div class="col-auto">
            <i class="fas fa-calendar fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>
```

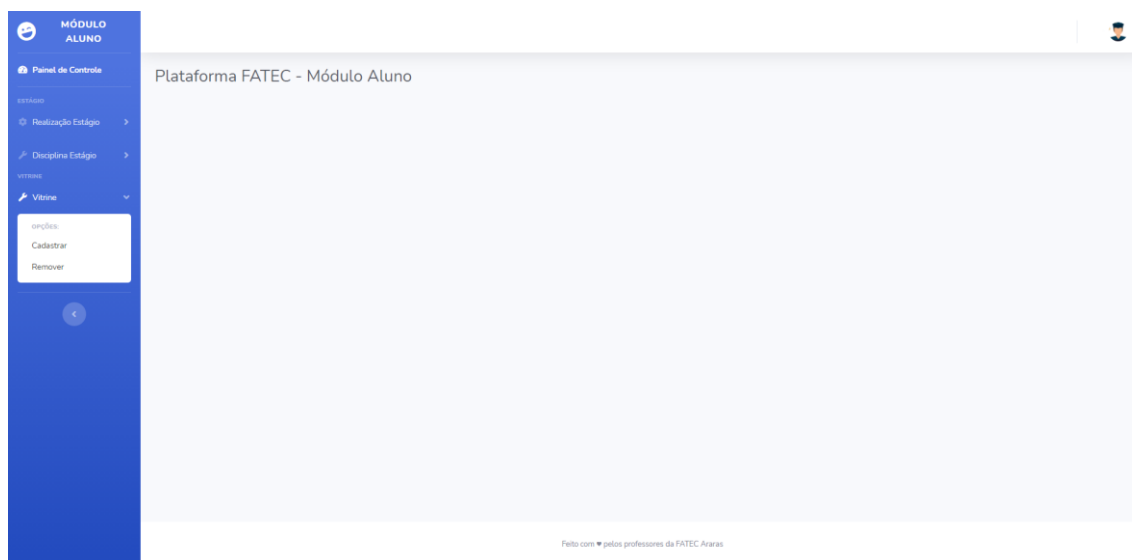
```
</div>  
</a>  
</div>
```

Figura 3 - Página de entrada da Plataforma Fatec



Fonte: de autor (2020)

Figura 4 - Página de entrada do Módulo Vitrine dentro da Plataforma Fatec



Fonte: de autor (2020)

Foi utilizado CRUD para manuseio da informação. CRUD nada mais é que um acrônimo para *Create, Read, Update and Delete*.

Create (Criar): Através deste método se consegue criar um novo registro no banco de dados, como por exemplo o cadastro de um estudante.

Código fonte python: Código referente ao CREATE

```
@area_student
def cadastro(request):
    if request.method == 'GET':
        context = {'cabecalho': 'Plataforma FATEC - Módulo Vitrine',
                    'form': VitrineForm(), 'user': request.user}
        return render(request, 'cadastro.html', context)
    else:
        form = VitrineForm(request.POST)
        if form.is_valid():
            VitrineModel.objects.filter(aluno_id=request.user.pk).delete()
            VitrineModel.objects.create(**form.cleaned_data)
            context = {'cabecalho': 'Plataforma FATEC - Módulo Vitrine',
                        'mensagem': 'Cadastro feito com sucesso'}
            return render(request, 'cadastro_feito.html', context)
        else:
            context = {'form': form, 'user': request.user}
            return render(request, 'cadastro.html', context)
```

Para coletar os dados dos alunos fatecanos, foi criado um formulário e um modelo de entidade para o banco de dados, segue ambos os trechos de código.

Código fonte python: Form.py

```
class VitrineForm(ModelForm):
    class Meta:
        model = VitrineModel
        fields = ('aluno', 'curso', 'tipo_vaga', 'descricao',
                  'linkedin', 'github')
        labels = {
            'aluno': 'Nome',
            'descricao': 'Descricao',
            'linkedin': 'Link Linkedin',
            'github': 'Link Github',
            'curso': 'Curso',
            'tipo_vaga': 'Tipo da Vaga'
        }
        widgets = {
            'aluno': forms.HiddenInput(attrs={'class': 'form-control'}),
            'curso': forms.Select(attrs={'class': 'form-control'}),
            'tipo_vaga': forms.Select(attrs={'class': 'form-control'}),
            'descricao': forms.Textarea(attrs={'class': 'form-control'}),
            'linkedin': forms.URLInput(attrs={'class': 'form-control'}),
            'github': forms.URLInput(attrs={'class': 'form-control'}),
        }
        error_messages = {
            'descricao': {
                'required': ("Você precisa nos contar algo sobre você."),
            },
            'linkedin': {
                'required': ("Você precisa colocar seu endereço do Linkedin"),
            }
        }
```

```

    }
}

def clean(self):
    self.cleaned_data = super().clean()
    sem_linkedin = not self.cleaned_data.get('linkedin')
    sem_github = not self.cleaned_data.get('github')
    if sem_linkedin and sem_github:
        raise ValidationError('Informe ao menos uma rede social.')
    return self.cleaned_data

```

Código fonte python: Model.py

```

TIPO_VAGA = (
    ('0', 'Emprego'),
    ('1', 'Estágio'),
    ('2', 'Emprego ou Estágio'),
)

class VitrineManager(models.Manager):
    def get_alunos(self):
        one_month_ago = timezone.now() - timedelta(days=30)
        return VitrineModel.objects.filter(
            updated_at__gte=one_month_ago
        ).order_by('updated_at')

    def get_alunos_emprego(self):
        alunos = self.get_alunos()
        return alunos.filter(Q(tipo_vaga=0) | Q(tipo_vaga=2))

    def get_alunos_estagio(self):
        alunos = self.get_alunos()
        return alunos.filter(Q(tipo_vaga=1) | Q(tipo_vaga=2))

    def get_tipo_vaga(self, tipo_vaga):
        alunos = self.get_alunos()
        return alunos.filter(Q(tipo_vaga=tipo_vaga) | Q(tipo_vaga=2))

    def get_tipo_curso(self, curso):
        alunos = self.get_alunos()
        return alunos.filter(curso=curso)

class VitrineModel(models.Model):
    aluno = models.ForeignKey(User, on_delete=models.CASCADE)
    descricao = models.TextField()
    linkedin = models.URLField(default="", blank=True)
    github = models.URLField(default="", blank=True)
    curso = models.CharField(
        verbose_name='Curso Matriculado',
        max_length=2,
        choices=CURSO_FATEC,
        default='0'
    )
    tipo_vaga = models.CharField(

```



```

        verbose_name='Tipo da Vaga',
        max_length=2,
        choices=TIPO_VAGA,
        default='2'
    )
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
    objects = VitrineManager()

    class Meta:
        verbose_name_plural = 'Alunos cadastrados'
        verbose_name = 'Aluno cadastrado'

    def __str__(self):
        return f'{self.aluno.first_name} - {self.aluno.email}'

```

Figura 5 - Página de cadastro para estágio e/ou emprego

Fonte: de autor (2020)

Read (Ler): Com esse método podemos acessar os dados(alunos) cadastrados no banco de dados e exibi-los em uma lista.

Código fonte python: Código referente ao READ

```

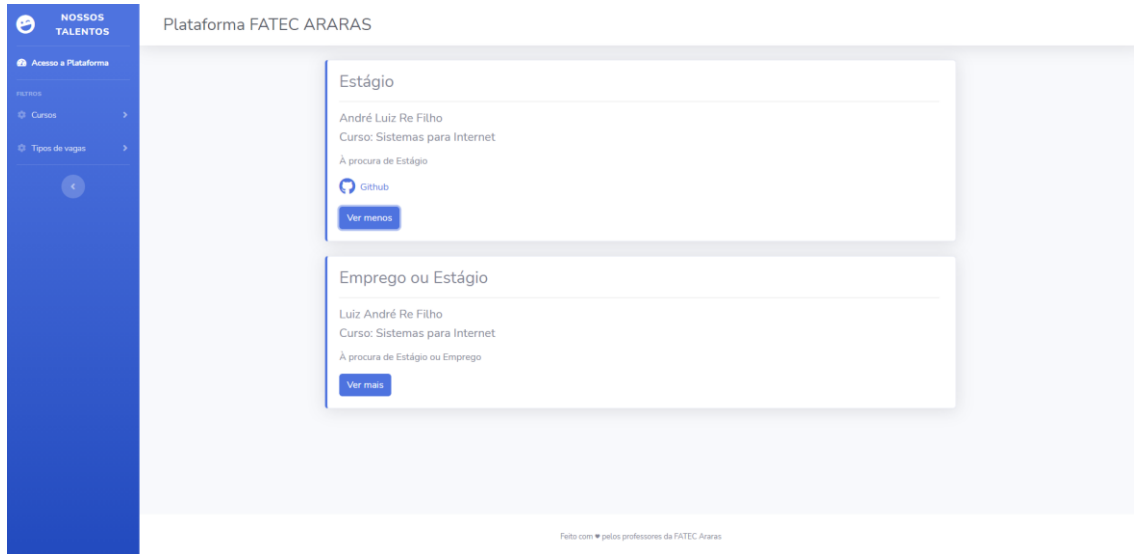
#Listagem Geral
def showcase(request):
    alunos = VitrineModel.objects.get_alunos()
    return render(request, 'showcase.html', {'alunos': alunos})

#Listagem por tipo de vaga
def showcase_tipo_vaga(request, tipo_vaga):
    alunos = VitrineModel.objects.get_tipo_vaga(tipo_vaga)
    return render(request, 'showcase.html', {'alunos': alunos})

```

```
#Listagem por Curso
def showcase_tipo_curso(request, curso):
    alunos = VitrineModel.objects.get_tipo_curso(curso)
    return render(request, 'showcase.html', {'alunos': alunos})
```

Figura 6 - Página de listagem de alunos para vaga de estágio e/ou emprego



Fonte: de autor (2020)

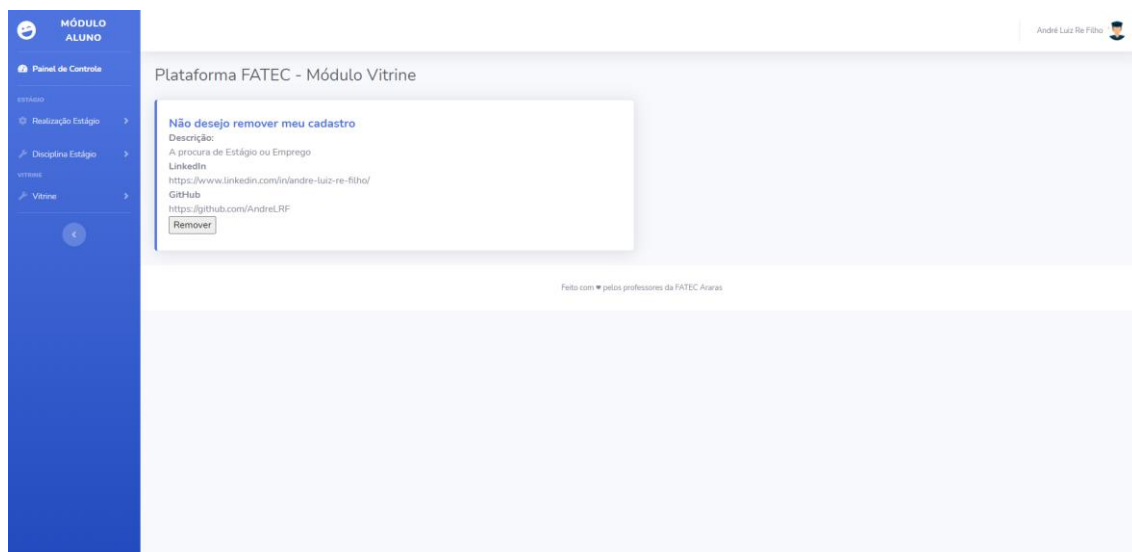
Update (Atualizar): Esse método é utilizado para editar o dado de algum registro no banco de dados. Mas não foi implementado no projeto

Delete (Deletar): Apaga o registro do aluno no banco de dados.

Código fonte python: Código referente ao DELETE

```
@area_student
def remover(request):
    if request.method == 'GET':
        cadastro = VitrineModel.objects.filter(aluno_id=request.user.pk)
        if cadastro:
            context = {'cadastro': cadastro[0],
                       'user': request.user}
            return render(request, 'remover.html', context)
        else:
            context = {'cabecalho': 'Você não possui registro ainda',
                       'form': VitrineForm(), 'user': request.user}
            return render(request, 'cadastro.html', context)
    else:
        VitrineModel.objects.filter(aluno_id=request.user.pk).delete()
        context = {'cabecalho': 'Plataforma FATEC - Módulo Vitrine',
                   'mensagem': 'Seu anúncio foi removido.'}
        return render(request, 'cadastro_feito.html', context)
```

Figura 7 - Página para remover cadastro dos alunos



Fonte: de autor (2020)

RESULTADOS

O projeto foi desenvolvido por alunos e pelo docente, na forma de *Coding Dojo*, ou seja, a cada dia um aluno era o piloto (Pessoa que codifica), enquanto os demais faziam o papel de copiloto (Pessoas que auxiliam o codificador). Desta forma todos compartilham conhecimentos e experiências de mercado para os demais.

O Módulo Vitrine foi concluído no término do semestre, onde poderá ser acessado através do repositório do docente, e também com a colaboração dos alunos acessando o link https://github.com/orlandosaraivajr/FATEC_plataforma.

CONSIDERAÇÕES

O referido projeto foi desenvolvido na matéria de Tópicos Especiais em Sistemas para Internet III, ministrada pelo Prof. Me. Orlando Saraiva do Nascimento Júnior, com o objetivo de aprendizado da linguagem *python/django* em executar um projeto real próximo aos utilizados no mercado de trabalho.

Este projeto foi de grande valia para os discentes da referida matéria, pois, tratando-se de um projeto solicitado pela instituição, proporcionou oportunidades próximas de um projeto real, havendo uma inter-relação e troca de experiência entre os participantes.

REFERÊNCIAS

ADVANCE, **iMonitor IT (*Brazil IT market survey*) Estudo Trimestral do Mercado Brasileiro de TI Agosto de 2021**, 2021. Disponível em: <https://www.advanceconsulting.com.br/pesquisa>. Acesso em: setembro de 2021.

IDC, **Previsões da IDC Brasil para 2021 apontam que mercado de TIC crescerá 7%**, 05/02/2021, Disponível em: <https://www.idc.com/getdoc.jsp?containerId=prLA47452221>
Acesso em: outubro de 2021

MOZILLA, ***Django introduction***, 2021. Disponível em: <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction>.
Acesso em: outubro de 2021

SILVA, I. M., **Até abril, empresas de tecnologia contrataram mais que em todo o ano de 2020**, 23/06/2021. Disponível em: <https://brasscom.org.br/ate-abril-empresas-de-tecnologia-contrataram-mais-que-em-todo-o-ano-de-2020/>. Acesso em: outubro de 2021