

# **Tópicos Especiais em Sistemas para Internet III**

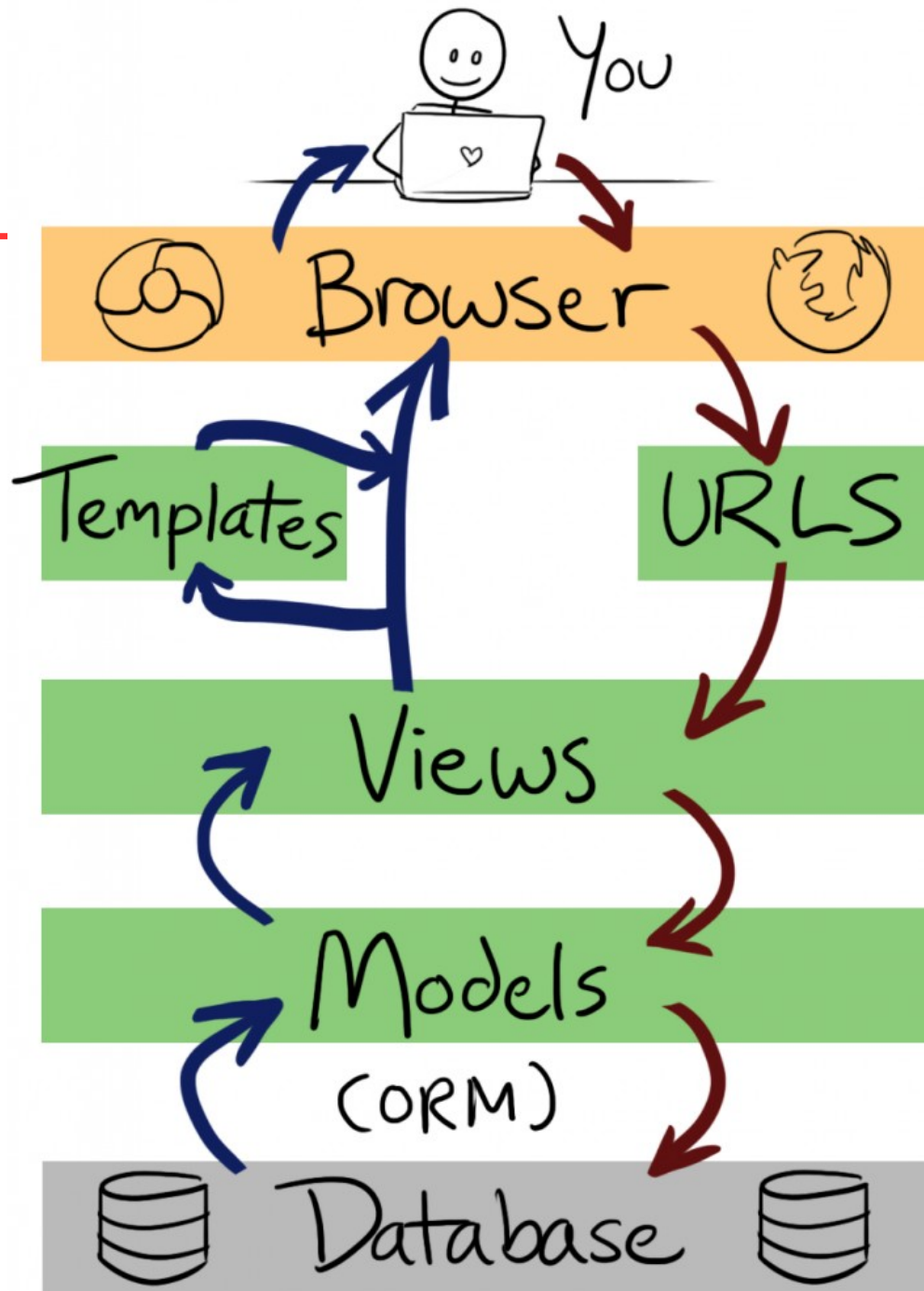
**Prof. Orlando Saraiva Júnior**  
**[orlando.nascimento@fatec.sp.gov.br](mailto:orlando.nascimento@fatec.sp.gov.br)**

# Agenda

---

- Django
  - Forms
  - ModelForms
- O Projeto

# Fluxo MTV



# **Ativando o Ambiente Virtual**

# Vamos começar ... No Windows

---

No prompt de comando (cmd), digite:

- pip install virtualenv
- virtualenv venv
- cd venv
- cd Scripts
- Activate.bat
- (venv)

# Vamos começar ...

## No Linux

---

No terminal, digite:

- `virtualenv -p python3 venv`
- `source venv/bin/activate`
- `(venv)`

# Projeto Feriado v. 0.5

Os formulários no Django possui duas funções

- Renderizar: Renderizar o Form e seus dados como inputs no HTML
- Sanitizar: Transformar os dados submetidos em objetos válidos



```
from django import forms

class FeriadoForm(forms.Form):
    nome = forms.CharField(max_length=50)
    dia = forms.IntegerField()
    mes = forms.IntegerField()

    def clean_nome(self):
        nome = self.cleaned_data['nome']
        Return nome.upper()
```

# Forms

## Bounded vs Unbounded

---

- Bounded: com dados
  - `FeriadoForm(request.POST)`
- Unbounded: sem dados
  - `FeriadoForm()`

```
dados1 = {'nome':'Tiradentes', 'dia':14,'mes':4}
dados2 = {'nome':'Tiradentes', 'dia':14}

form1 = FeriadoForm(dados1)
form2 = FeriadoForm(dados2)
form3 = FeriadoForm()

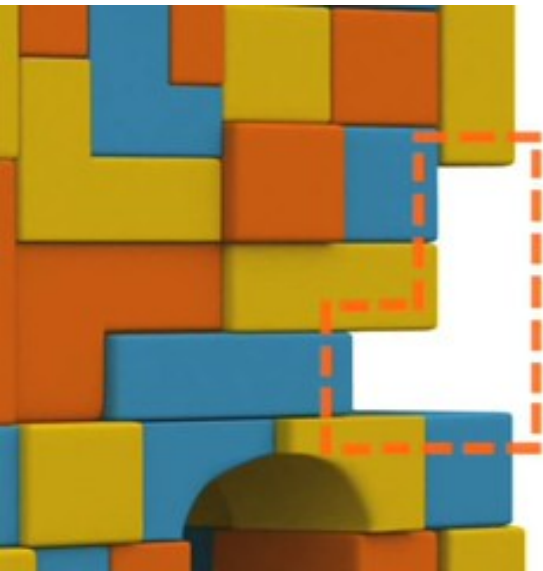
form1.is_bound
form2.is_bound

form1.is_valid()
form2.is_valid()
print(form3.as_ul())

for field in form3:
    print(str(field))
```

# Test it !

---



# Teste unitário

## /feriado/tests.py

```
from feriado.forms import FeriadoForm

class FeriadoFormTest(TestCase):
    def test_form_has_fields(self):
        form = FeriadoForm()
        expected = ['nome', 'dia', 'mes']
        self.assertEqual(expected, list(form.fields))

    def test_must_be_capitalized(self):
        form = self.make_validated_form(nome='dia de são nunca')
        self.assertEqual('DIA DE SÃO NUNCA', form.cleaned_data['nome'])
```

# Teste unitário

**/feriado/tests.py ( continuação)**

---

```
def test_must_be_capitalized(self):
    form = self.make_validated_form()
    self.assertEqual('TIRADENTES', form.cleaned_data['nome'])

def make_validated_form(self, **kwargs):
    valid = dict(
        nome='Tiradentes',
        dia=14,
        mes=4
    )
    data = dict(valid, **kwargs)
    form = FeriadoForm(data)
    form.is_valid()
    return form
```

# ModelForm

---

Modelforms são formulários criados a partir de uma inspeção do modelo.



# forms.py ( do app )

```
from feriado.models import FeriadoModel
from django.forms import ModelForm

class FeriadoFormModel(ModelForm):
    class Meta:
        model = FeriadoModel
        fields = ['nome','dia','mes']

dados = {'nome':'Natal','dia':25,'mes':12}
form = FeriadoFormModel(dados)
Form.is_valid()
form.save()
```

# Dúvidas

**Prof. Orlando Saraiva Júnior**  
**[orlando.nascimento@fatec.sp.gov.br](mailto:orlando.nascimento@fatec.sp.gov.br)**

---

Análise de um projeto “real”

[https://github.com/orlandosaraivajr/  
cadastro\\_eleitores](https://github.com/orlandosaraivajr/cadastro_eleitores)

Incluir um form no projeto “Feriado”