



Trường ĐH Khoa Học Tự Nhiên Tp. Hồ Chí Minh  
**TRUNG TÂM TIN HỌC**

# Lập trình viên Python

## Bài 1: *Flask WTF*

Phòng LT & Mạng

<http://csc.edu.vn/lap-trinh-va-csdl>

# Nội dung

---

1. Tổng quát về Form
2. Flask WTF

# Tổng quát về Form

---

## □ Giới thiệu

- Trong web có một thành phần không thể thiếu là nơi cho người dùng nhập thông tin và gửi yêu cầu đi. Thành phần này gọi là Form.
- Flask cung cấp các thư viện giúp ta xây dựng các Forms để nhận thông tin của người dùng sau đó xử lý và phản hồi kết quả.

# Tổng quát về Form

---

## □ HTML form

- Trong HTML, một form là một bộ các element nằm trong tag `<form>...</form>`, nó cho phép người dùng nhập liệu, chọn lựa trên các đối tượng hoặc điều khiển, sau đó gửi thông tin về server
- Form element rất nhiều loại như text input hay checkbox, radio button, submit button,... Trong đó, `<input>` element là form element quan trọng nhất. Nó có thể hiển thị dưới nhiều dạng khác nhau tùy vào thuộc tính ***type***

# Tổng quát về Form

---

## □ GET và POST

- Là những phương thức HTTP được sử dụng để làm việc với Form
- GET và POST thường được sử dụng cho các mục đích khác nhau.
- Bất kỳ yêu cầu nào có thể được sử dụng để thay đổi trạng thái của hệ thống - ví dụ như một yêu cầu thay đổi cơ sở dữ liệu - nên sử dụng POST.
- GET chỉ nên được sử dụng cho các yêu cầu không ảnh hưởng đến trạng thái của hệ thống.

# Tổng quát về Form

---

- Với GET, khi gửi yêu cầu thông tin sẽ xuất hiện trên URL , do đó không nên dùng khi cần bảo mật cho thông tin. Và GET không phù hợp khi gửi đi một lượng thông tin lớn. GET thích hợp cho những tìm kiếm trên web vì URL thể hiện yêu cầu GET có thể dễ dàng được đánh dấu, chia sẻ hoặc gửi lại.
- POST, cùng với các bảo vệ khác như CSRF của Flask cung cấp khả năng kiểm soát truy cập nhiều hơn.

# Tổng quát về Form

---

## ❑ Vai trò của Flask trong Form

- Quản lý form là một công việc phức tạp. Vì vậy Flask cung cấp chức năng cho Form để đơn giản và tự động hóa các công việc liên quan đến Form và làm cho công việc này trở nên an toàn hơn.

# Tổng quát về Form

---

- Flask xử lý 3 thành phần riêng biệt của công việc liên quan đến Form.
  - Chuẩn bị và tổ chức dữ liệu giúp cho nó sẵn sàng để render
  - Tạo HTML form cho dữ liệu
  - Tiếp nhận và xử lý các form đã submit và dữ liệu từ khách hàng.



# Nội dung

---

1. Tổng quát về Form
2. Flask WTF

# Flask WTF

---

- ❑ Một đặc điểm bất lợi của HTML form là rất khó khăn khi muốn tạo ra các form element tự động. Bản thân HTML không cung cấp cách để xác thực dữ liệu mà người dùng nhập.
- ❑ Vì vậy mà **WTForms**, một dạng form linh hoạt, có thư viện để tạo và kiểm tra dữ liệu tiện dụng, được sử dụng. Flask-WTF cung cấp một giao diện đơn giản với thư viện **WTForms** này.
- ❑ Sử dụng **Flask-WTF**, ta có thể định nghĩa form element trong Python script và render chúng bằng cách sử dụng một HTML template. Và có thể thực hiện việc kiểm tra dữ liệu (validation) cho **WTF** field.

# Flask WTF

---

## ❑ Để có thể làm việc với wtf form

- Cần cài đặt thư viện wtf: `pip install flask_wtf`
- Trong thư viện này có class Form được sử dụng làm lớp cha cho các form mà ta sẽ khai báo sau này.

# Flask WTF

- ❑ Trong WTforms package chứa định nghĩa của nhiều form element khác nhau. Một số form element hay dùng:

1	TextField ~ <code>&lt;input type = 'text'&gt;</code> HTML form element
2	BooleanField ~ <code>&lt;input type = 'checkbox'&gt;</code> HTML form element
3	DecimalField ~ TextField hiển thị số kiểu decimals
4	IntegerField ~ TextField hiển thị số kiểu integer
5	RadioField ~ <code>&lt;input type = 'radio'&gt;</code> HTML form element
6	SelectField ~ select form element
7	TextAreaField ~ <code>&lt;textarea&gt;</code> html form element
8	PasswordField ~ <code>&lt;input type = 'password'&gt;</code> HTML form element
9	SubmitField ~ <code>&lt;input type = 'submit'&gt;</code> form element

# Flask WTF

---

## ❑ Các bước thực hiện

- Bước 1: Trong thư mục Xu\_ly tạo tập tin Xu\_ly\_Form.py để chứa các class form cần thiết cho ứng dụng
- Bước 2: Trong Xu\_ly\_Form.py, chèn các thư viện cần thiết ở đầu file

```
from flask_wtf import Form
```

```
from wtforms import TextField, IntegerField,  
TextAreaField, SubmitField, RadioField, SelectField  
from wtforms import validators, ValidationError
```

# Flask WTF

- Bước 3: Trong Xu\_ly\_Form.py, tạo các class form cần thiết cho ứng dụng
  - Trong class có các thuộc tính kèm theo các validators tương ứng cho các thuộc tính đó bằng cách sử dụng validators.xxx. Ví dụ:
    - `validators.Required("Thông báo bắt buộc nhập liệu")`
    - `validators.Email("Thông báo nội dung nhập phải là email")`
    - `validators.DataRequired("Thông báo data bắt buộc phải có")`
    - `validators.EqualTo(fieldname, message="Thông báo dữ liệu hai field không giống nhau")`
    - `validators.Length(min=?, max=?, message="Thông báo dữ liệu nhập ngắn hơn min hoặc dài hơn max")`
    - `validators.URL(require_tld=True, message="Thông báo URL chưa đúng chuẩn")`

# Flask WTF

- Ví dụ: tạo class Form\_Lien\_he cho người dùng gửi ý kiến

```
class Form_Lien_he(Form):
    Th_Ho_ten = TextField("Tên khách hàng",[validators.Required("Vui lòng nhập tên.")])
    Th_Gioi_tinh = RadioField('Giới tính', choices =
    [('M','Nam'),('F','Nữ'), ('D', "Khác")])
    Th_Dia_chi = TextAreaField("Địa chỉ")
    Th_Email = TextField("Email",[validators.Required("Vui lòng nhập email"), validators.Email("Vui lòng nhập email")])
    Th_Tuoi = IntegerField("Tuổi")
    Th_Ly_do = SelectField('Góp ý cho', choices = [('TGGH', 'Thời gian giao hàng'), ('CSKH', 'Chăm sóc khách hàng'), ('BHSP', 'Bảo hành sản phẩm') ])
    Th_Noi_dung = TextAreaField("Nội dung")
    Th_submit = SubmitField("Gửi ý kiến")
```

# Flask WTF

---

- Bước 4: Trong app\_xxx.py, import Xu\_ly\_Form vào đầu trang và viết phương thức gửi form vừa tạo ra trang html để hiển thị.
  - Ví dụ: gửi form Form\_Lien\_he thông qua render\_template đến trang html

```
@app.route("/Gui_y_kien", methods=['GET', 'POST'])
def Gui_y_kien():
    form = Form_Lien_he(request.form)
    Khung = render_template('Khach_mua_hang_Online/MH_Lien_he.html',
    form = form)
    return Khung
```




# Flask WTF

- Bước 5: Tạo trang html, sử dụng form vừa gửi ở bước 4 vào trang

- Ví dụ:

```
{{ form.csrf_token }}
<tbody>
  <tr>
    <td>{{ form.Th_Ho_ten.label }} </td>
    <td>{{ form.Th_Ho_ten }} </td>
  </tr>
  <tr>
    <td>{{ form.Th_Gioi_tinh.label }}</td>
    <td>{{ form.Th_Gioi_tinh}}</td>
  </tr>
  <tr>
    <td>{{ form.Th_Dia_chi.label }}</td>
    <td>{{ form.Th_Dia_chi }}</td>
  </tr>
  <tr>
    <td>{{ form.Th_Email.label }}</td>
    <td>{{ form.Th_Email }}</td>
  </tr>
  <tr>
    <td colspan="2">{{
      form.Th_submit}}</td>
  </tr>
</tbody>
```

## ■ Kết quả sau khi tạo form

 Mô hình - Công nghệ - Kỹ thuật  
Chào mừng quý khách đến cửa hàng

**GỬI Ý KIẾN - YÊU CẦU HỖ TRỢ**

Tên khách hàng

Nguyễn Thanh Xuân

Giới tính

- ☐ Nam
- ☒ Nữ
- ☐ Khác

Địa chỉ

123 Nguyễn Chí Thanh  
p9 q5 tp hcm

Email

ntxuan@gmail.com

Góp ý cho

Bảo hành sản phẩm ▾

Nội dung

Bảo hành không chu  
đáo, liên hệ khó khăn

Gửi ý kiến

# Flask WTF

---

- Bước 6: Trong `app_xxx.py`: viết phương thức xử lý khi người dùng nhập liệu xong
  - Lấy dữ liệu trên form
  - Kiểm tra dữ liệu
  - Hiển thị/ghi/cập nhập dữ liệu
  - ...


# Flask WTF

- Ví dụ: Xử lý kiểm tra dữ liệu người dùng nhập ở form ý kiến, lấy dữ liệu và ghi vào tập tin .json

```
@app.route("/Gui_y_kien",  
methods=['GET', 'POST'])  
def Gui_y_kien():  
    Chuoi_ket_qua = ""  
    form = Form_Lien_he(request.form)  
    if form.validate_on_submit():  
        Ho_ten =  
            request.form['Th_Ho_ten']  
        Gioi_tinh =  
            request.form['Th_Gioi_tinh']  
        Dia_chi =  
            request.form['Th_Dia_chi']  
        Email =  
            request.form['Th_Email']  
        Ly_do =  
            request.form['Th_Ly_do']  
        Noi_dung =  
            request.form['Th_Noi_dung']
```

```
        if len(Ho_ten)>3 and len(Email)>6:  
            Ngay = datetime.now().strftime('%d-%m-%Y')  
            Y_kien = {"Ho_ten": Ho_ten,  
                    "Gioi_tinh": Gioi_tinh, "Dia_chi": Dia_chi,  
                    "Email": Email, "Ly_do": Ly_do, "Noi_dung":  
                    Noi_dung, "Ngay_gui": Ngay}  
            Ghi_Y_kien(Y_kien)  
            print(Y_kien)  
            Chuoi_ket_qua += "Cảm ơn " + Ho_ten + " đã  
            gửi ý kiến"  
            flash("Cảm ơn " + Ho_ten + " đã gửi ý  
            kiến")  
        else:  
            flash('Error: Cần nhập đủ các nội dung yêu cầu')  
    Khung=  
    render_template('Khach_mua_hang_Online/MH_Lien_he.htm  
l',  
    form = form, Chuoi_ket_qua = Chuoi_ket_qua)  
    return Khung
```

## ■ Kết quả

 Mô hình - Công nghệ - Kỹ thuật  
Chào mừng quý khách đến cửa hàng

**GỬI Ý KIẾN - YÊU CẦU HỖ TRỢ**

Tên khách hàng

Giới tính

☐ Nam  
☒ Nữ  
☐ Khác

Địa chỉ

123 Nguyễn Chí Thanh  
p9 q5 tp hcm

Email

Góp ý cho

Bảo hành sản phẩm ▼

Nội dung


Bảo hành không chu  
đáo, liên hệ khó khăn

Gửi ý kiến

Vui lòng nhập tên.

Vui lòng nhập email

Cảm ơn quý khách. Chúng tôi sẽ phản hồi trong thời gian sớm nhất

 Mô hình - Công nghệ - Kỹ thuật  
Chào mừng quý khách đến cửa hàng

**GỬI Ý KIẾN - YÊU CẦU HỖ TRỢ**

Tên khách hàng

Nguyễn Thanh Xuân

Giới tính

☐ Nam  
☒ Nữ  
☐ Khác

Địa chỉ

123 Nguyễn Chí Thanh  
p9 q5 tp hcm

Email

ntxuan@gmail.com

Góp ý cho

Bảo hành sản phẩm ▼

Nội dung

Bảo hành không chu  
đáo, liên hệ khó khăn

Gửi ý kiến

Cảm ơn quý khách. Chúng tôi sẽ phản hồi trong thời gian sớm nhất

Cảm ơn Nguyễn Thanh Xuân đã gửi ý kiến

