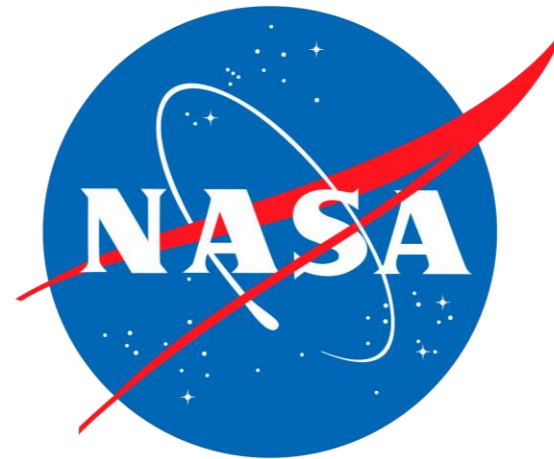# wagtail
CMS open source

Pourquoi Wagtail ?

Wagtail est utilisé par de grandes entreprises et institutions grâce à :

- Flexibilité

- Extensibilité

- Communauté

- Performances
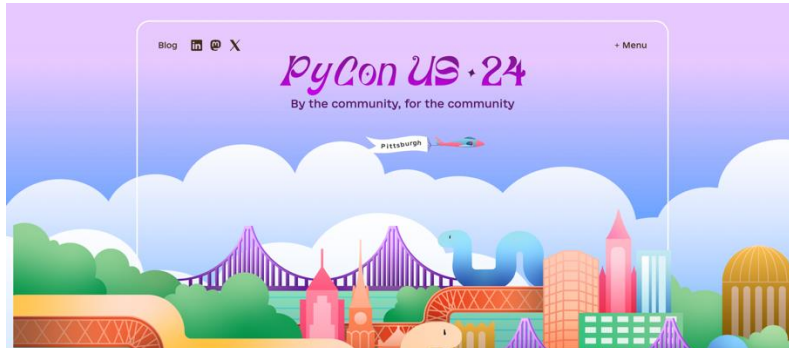
- Python/Django

- Open Source

Utilisation ?

- Sites d'entreprises

- Blogs

- E-commerce

- Sites éducatifs
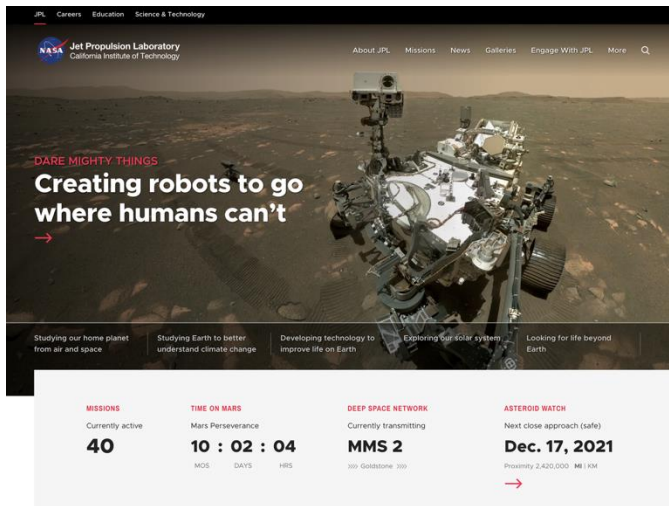
- Sites gouvernementaux

- etc...

Made with Wagtail

The PyCon 2021 conference is a production of the **Python Software Foundation**.
This site is built using **Django** and **Wagtail**. Hosting is provided by **Heroku**.

blog.google

grow.google

flutter.dev

blog.youtube.com

twilio.com

give.thunderbird.net

climate.esa.int/en/

transform.england.nhs.uk

about.google

docs.sendgrid.com

quintagroup.com
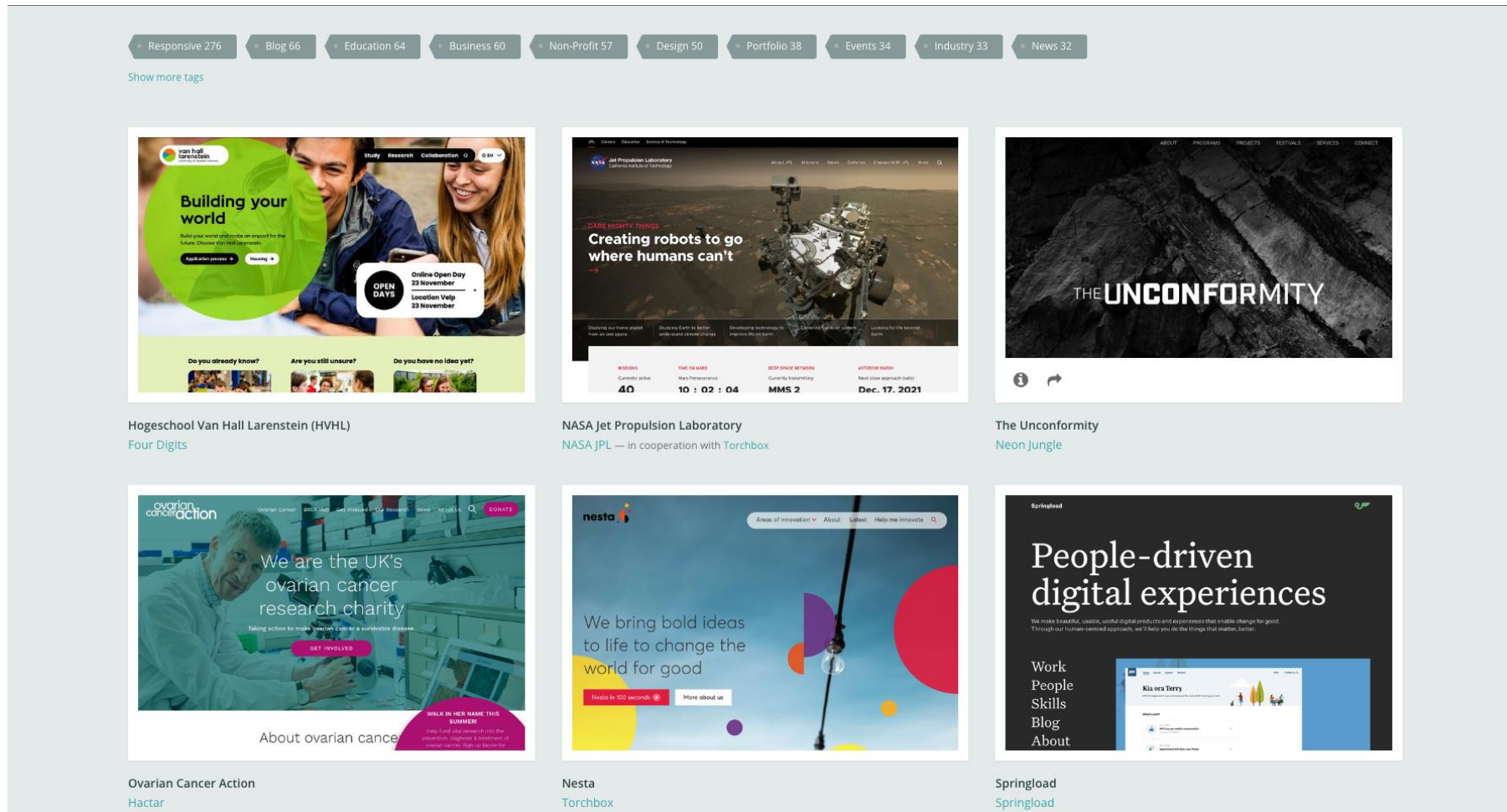
Made with Wagtail

madewithwagtail.org

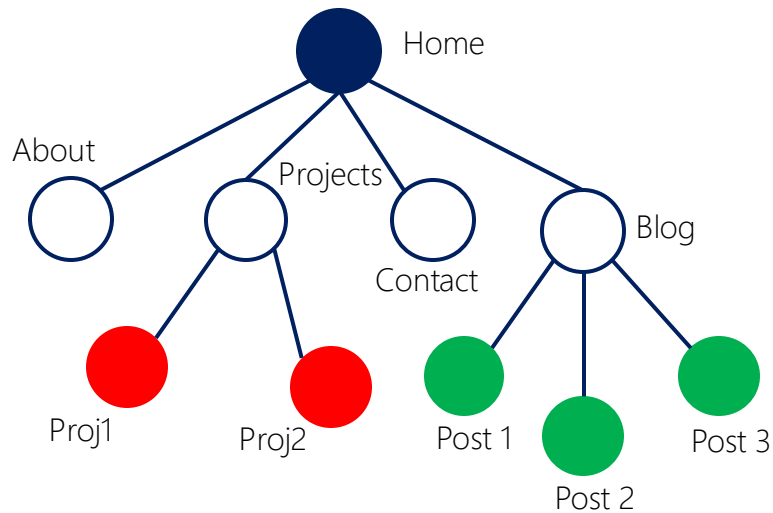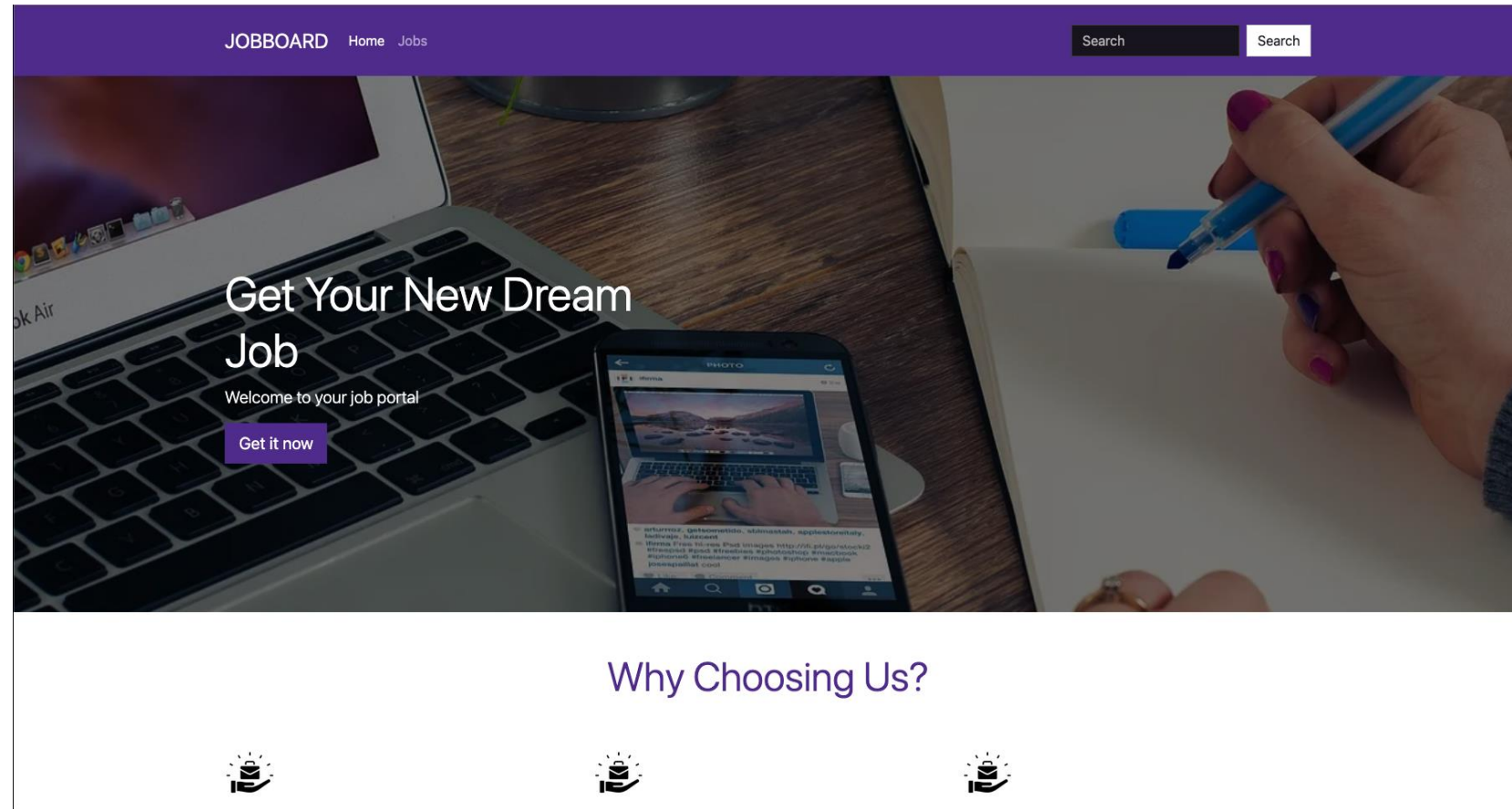# Structure technique et caractéristiques

- Page

- StreamField

- Blocks

- Snippets

- Interface d'utilisateur

- Images et documents

- Permissions et rôles

Démo

- Création de pages

- StreamField

- Style (bootstrap)

- Menu

- Pied de page

- Recherche

Création de page

```python
class JobPage(Page):
    date = models.DateField("Published Date", default=timezone.now)
    location = models.CharField(max_length=100)
    short_description = models.CharField(max_length=200, help_text="Max 200
    characters")
    thumbnail = models.ForeignKey(
        "wagtailimages.Image",
        null=True,
        blank=True,
        related_name="+",
        on_delete=models.SET_NULL,
    help_text="Choose a image for your cover")
    salary = models.IntegerField(verbose_name="A salary approximation")
    company = models.CharField(max_length=200)
```
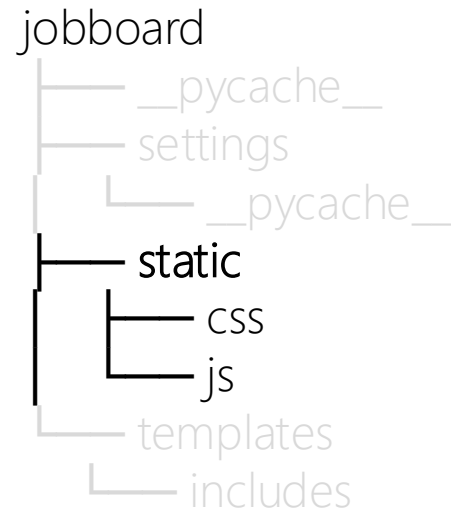
```python
content_panels =
Page.content_panels + [
FieldPanel("date"),
FieldPanel("location"),
FieldPanel("short_description"),
FieldPanel("thumbnail"),
FieldPanel("salary"),
FieldPanel("company"),
]
```

```
{{ page.title }}
```

StreamField

```python
features = StreamField(
        [
                ('feature_icon', ImageChooserBlock(required=False)),
                ('feature_title', blocks.CharBlock(required=True)),
                ('feature_desc', blocks.CharBlock(required=True))
        ]


content = StreamField(
    [
    ('heading', blocks.CharBlock(template='blocks/heading.html')),
    ('paragraph', blocks.RichTextBlock(template='blocks/paragraph.html')),
    ('blockquote', blocks.BlockQuoteBlock(template='blocks/blockquote.html')),
    ('list', blocks.ListBlock(blocks.CharBlock(), template='blocks/list.html')),
    ('html', blocks.RawHTMLBlock(template='blocks/html.html')),
    ('image', ImageChooserBlock())
    ]
```
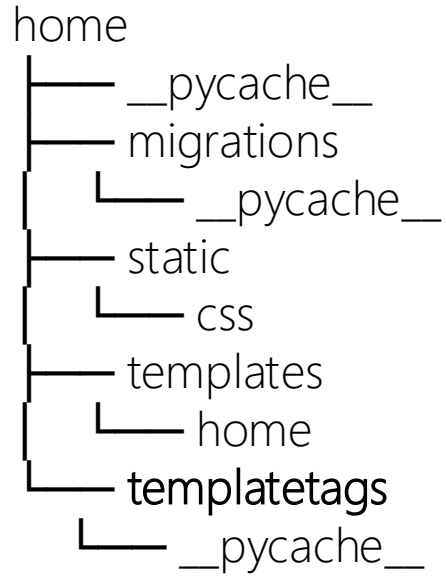
Style

```
jobboard
├── __pycache__
├── settings
│   └── __pycache__
│
├── static
│   ├── css
│   └── js
└── templates
    └── includes
```

<link rel="stylesheet" type="text/css" href="{% static 'css/jobboard.css' %}">
<link rel="stylesheet" type="text/css" href="{% static 'css/bootstrap.min.css' %}">

Menu

```
home
├──── __pycache__
├──── migrations
│     └──── __pycache__
├──── static
│     └──── css
├──── templates
│     └──── home
└──── templatetags
      └──── __pycache__
```

 # navigation_tags.py

```python
register = template.Library()
@register.simple_tag(takes_context=True)
def get_site_root(context):
    return Site.find_for_request(context["request"]).root_page
```

 # template.html

```html
{% load wagtailcore_tags navigation_tags %}
{% get_site_root as site_root %}
<ul>
  <li>
    <a href="{% pageurl site_root %}">Home</a>
  </li>

{% for menuitem in site_root.get_children.live.in_menu %}
<li>
  <a href="{% pageurl menuitem %}">{{ menuitem.title }}</a>
</li>
{% endfor %}
```

Pied de page

```
@register_setting
class NavigationSettings(BaseGenericSetting):
linkedin_url = models.URLField(verbose_name="Linkedin URL", blank=True)
github_url = models.URLField(verbose_name="GitHub URL", blank=True)
mastodon_url = models.URLField(verbose_name="Mastodon URL", blank=True)


panels = [
    MultiFieldPanel(
        [
        FieldPanel("linkedin_url"),
        FieldPanel("github_url"),
        FieldPanel("mastodon_url")
    ],
    "Social settings"
    )
]
```

```
INSTALLED_APPS = [
"home",
"job",
"search",
…
"wagtail.contrib.settings"
]
```

```
"wagtail.contrib.settings.context_processors.settings",
```

Recherche

```python
search_fields = Page.search_fields + [
index.SearchField('location'),
index.SearchField('company'),
]

python manage.py update_index
```

```html
<form action="{% url 'search' %}" method="get">
<input type="search"
        name="query"{% if search_query %}
        value="{{ search_query }}"{% endif %}>
<buttontype="submit">Search</button>
</form>
```

Appréhender Wagtail ?

- La Documentation officielle – docs.wagtail.org

- Site Bakery - https://github.com/wagtail/bakerydemo

- Starter template - https://github.com/torchbox/wagtail-news-template/

Pour infos et contact

https://github.com/Thevenel/wagtail-pyconfr-2024

Thevenel Joazard

thevenel25@gmail.com

thevenel.thev-academy.com

Questions ?