



**BearStudio**

# JADE

**LE MOTEUR DE TEMPLATE MODERNE**

présenté par **Thibault Barske** et **David Endico**



# BearStudio

 [\\_BearStudio](#)



## Thibault Barske

ANALYSTE DÉVELOPPEUR

 [tibs245](#)



## David Endico

UI/UX DESIGNER

 [DavidEndico](#)



# JADE, QU'EST-CE QUE C'EST ?

**Jade** a été créé en 2009, inspiré de Haml (qui exploitait Ruby), c'est un **moteur de template** basé sur JavaScript pour NodeJS. Simple à étudier et à prendre en main, il est à la fois **modulable** et **performant**. Jade s'adapte à différents langages comme PHP, Ruby et Python.



PARTIE 1

# LES BASES

Don't worry, be happy



**VOUS NE  
FERMEREZ  
PAS**

© The Lord of the Ring

5

## PARTIE 1 : LES BASES

# ADIEU BALISES FERMANTES...

### JADE

```
h1
div
script(src="script.js")
img
// Commentaire
// if lt IE 8
```



### HTML

```
<h1></h1>
<div></div>
<script src="script.js"></script>
<img/>
<!-- Commentaire -->
<!--[if lt IE 8]> -->
```

## PARTIE 1 : LES BASES

# ...BONJOUR INDENTATION !

JADE

```
div
  p
  i
    strong
```



HTML

```
<div>
<p></p>
<i><strong></strong></i>
</div>
```

```
header: nav: ul
  li: a
  li: a
```



```
<header><nav><ul>
<li><a></a></li>
<li><a><a></li>
</ul></nav></header>
```



## PARTIE 1 : LES BASES

# 3 MANIÈRES D'AJOUTER DU TEXTE

### JADE

```
h1 Hello world !  
p.  
  Ceci est une phrase  
  de texte.  
  
p  
  | Ceci est une seconde  
  | phrase de texte.
```



### HTML

```
<h1>Hello World !</h1>  
<p>Ceci est une phrase de texte.</p>  
<p>Ceci est une seconde phrase de  
texte.</p>
```



## PARTIE 1 : LES BASES

# BALISES IMBRIQUÉES

### JADE

```
p
  | Ceci est une
  b  première
  | phrase de texte.

p.
  Ceci est une <b>seconde</b>
  phrase de texte.
```



### HTML

```
<p>Ceci est une <b>première</b> phrase
de texte.</p>

<p>Ceci est une <b>seconde</b> phrase
de texte.</p>
```

## PARTIE 1 : LES BASES

# AJOUTER DES ATTRIBUTS, UN JEU D'ENFANT !

### JADE

```
p#id  
p.class  
p.class#id.other-class
```



### HTML

```
<p id="id"></p>  
<p class="class"></p>  
<p class="class other-class" id="id"></p>
```

```
h1#title Hello world !  
button.btn(type="submit") Réservez !
```



```
<h1 id="title">Hello world !</h1>  
<button class="btn", type="submit">  
  Réservez !</button>
```

## PARTIE 1 : LES BASES

# ARCHITECTURE ET SIMPLIFICATION DU CODE AVEC LES INCLUDES

JADE

```
doctype html
html
  head
    style
      include style.css
  body
    include content.jade
```

```
//style.css
h1 { color : red; }
```

```
//content.jade
h1 Hello world !
p C'est trop de la balle !
```



HTML

```
<!doctype html>
<html>
  <head>
    <style>
      h1 { color : red; }
    </style>
  </head>
  <body>
    <h1>Hello world !</h1>
    <p>C'est trop de la balle !</p>
  </body>
</html>
```

## PARTIE 1 : LES BASES

# ÉCRITURE D'AUTRES LANGAGES

### JADE

```
//PHP
ul
  - foreach ($this->list as $list):
    li!= $list
  - endforeach

//Javascript
script(type='text/javascript')
  if (foo) {
    bar(1 + 5)
  }
```



### HTML

```
<!-- PHP -->
<ul>
  <?php foreach ($this->list as $list): ?>
    <li><?php echo $list; ?></li>
  <?php endforeach; ?>
</ul>

<!-- Javascript -->
<script type="text/javascript">
  if (foo) {bar(1 + 5)}
</script>
```



**SUCCESS!**

PARTIE 2

# MANŒUVRES AVANCÉES

Raise your weapon

## PARTIE 2 : MANŒVRES AVANCÉES

# CONFIGURATION DE VALEURS

### JADE

```
-var movie="Cohérence, 05.50€"
```

```
-var film={name:"Victoria", price:"07.40€"}
```

```
ul
  li The Signal, 06.50€
  li= movie
  li #{film.name}, #{film.price}
```



### HTML

```
<ul>
<li>The Signal, 06.50€</li>
<li>Cohérence, 05.50€</li>
<li>Victoria, 07.40€</li>
</ul>
```



## PARTIE 2 : MANŒUVRES AVANCÉES

# ITERATIONS EACH OU FOR

JADE

```
ul
  each item in ['Lun', 'Mar', 'Mer']
    li= item
```



HTML

```
<ul>
<li>Lun</li>
<li>Mar</li>
<li>Mer</li>
</ul>
```

```
ul
  for item, index in ['Lun', 'Mar', 'Mer']
    li= index + ': ' + item
```



```
<ul>
<li>0: Lun</li>
<li>1: Mar</li>
<li>2: Mer</li>
</ul>
```

## PARTIE 2 : MANŒUVRES AVANCÉES

# ITERATIONS WHILE

JADE

```
- var n = 0
ul
  while n < 4
    li= n++
```

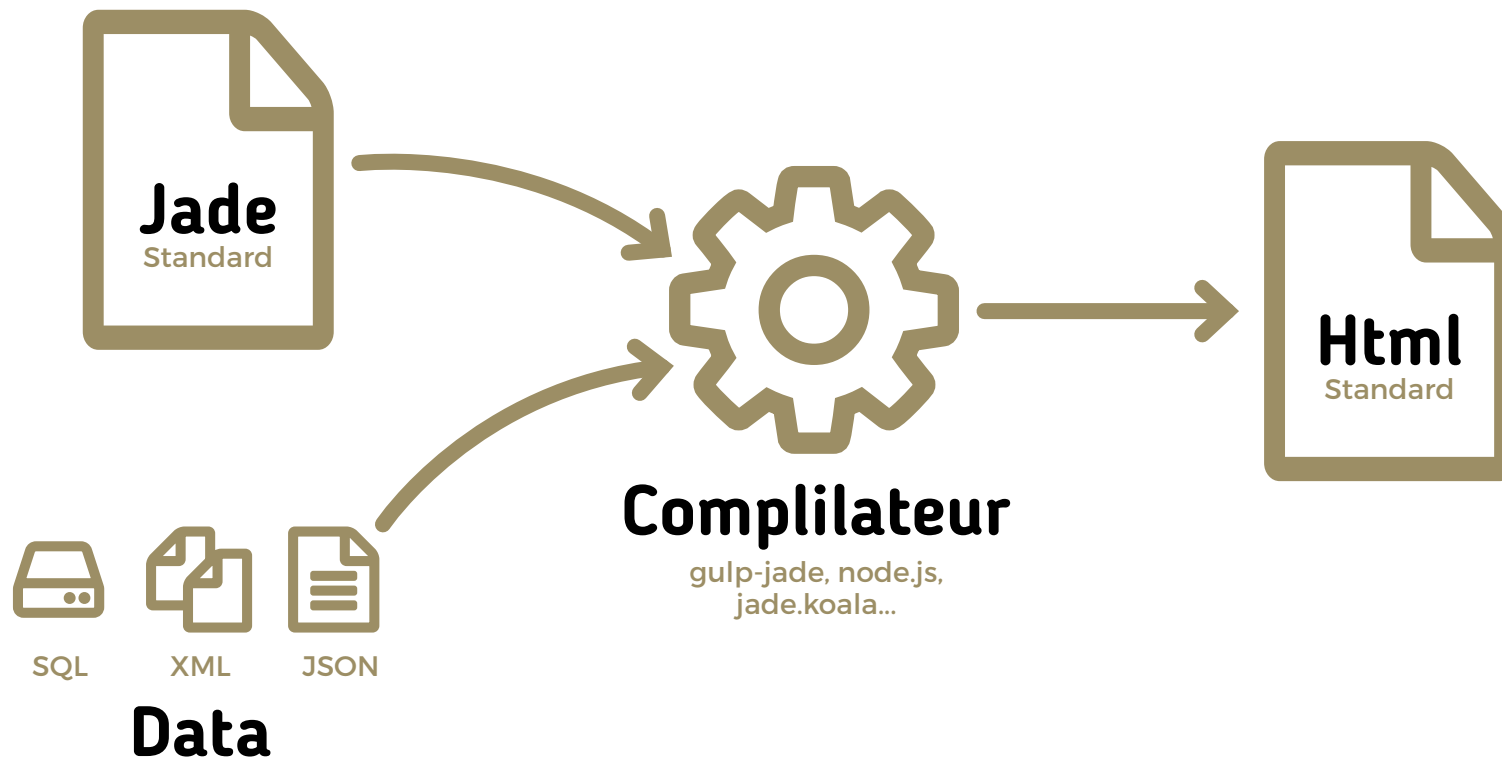


HTML

```
<ul>
<li>0</li>
<li>1</li>
<li>2</li>
<li>3</li>
</ul>
```

## PARTIE 2 : MANŒUVRES AVANCÉES

# RÉCUPÉRATION DE VARIABLES



## PARTIE 2 : MANŒVRES AVANCÉES

# RÉCUPÉRATION DE VARIABLES

### JADE

```
- each name in names
  label
    input(
      type="radio",
      name="name",
      value=name
    )
    span Prénom : #{name}
```



### HTML

```
<label>
<input type="radio", name="David",
value="David"></input>
<span> Prénom : David</span>
</label>

<label>
<input type="radio", name="Francis",
value="Francis"></input>
<span> Prénom : Francis</span>
</label>

...
```

```

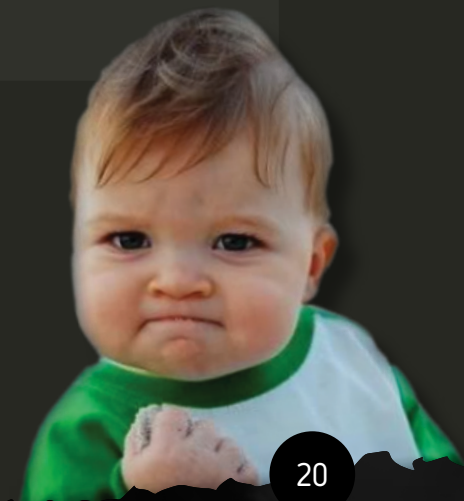
16 input(type='text',name='tuesdayEndTime',id='tuesdayEndTime',placeholder='5:00')
17 .one.ninth.mobile
18 span#tuesdayEndSuffix.suffix pm
19 hr
20 #wednesdayHours
21 .hoursLabel
22 label(for="wednesdayStartTime") Wednesday:
23 .hoursClosed
24 input(type='checkbox',id='wednesdayClose',value='Closed')
25 |Closed
26 .clear
27 #wednesdayInput.row
28 .three.ninth.mobile
29 input(type='text',name='wednesdayStartTime',id='wednesdayStartTime',placeholder='9:00')
30 .one.ninth.mobile
31 span#wednesdayStartSuffix.suffix am
32 .one.ninth.mobile.hoursSeparator
33 |to
34 .three.ninth.mobile
35 input(type='text',name='wednesdayEndTime',id='wednesdayEndTime',placeholder='5:00')
36 .one.ninth.mobile
37 span#wednesdayEndSuffix.suffix pm
38 hr
39 #thursdayHours
40 .hoursLabel
41 label(for="mondayStartTime") Thursday:
42 .hoursClosed
43 input(type='checkbox',id='thursdayClose',value='Closed')
44 |Closed
45 .clear
46 #thursdayInput.row
47 .three.ninth.mobile
48 input(type='text',name='thursdayStartTime',id='thursday
49 .one.ninth.mobile
50 span#thursdayStartSuffix.suffix am
51 .one.ninth.mobile.hoursSeparator
52 |to
53 .three.ninth.mobile
54 input(type='text',name='thursdayEndTime',id='thursdayEn
55 .one.ninth.mobile
56 span#thursdayEndSuffix.suffix pm
57 hr
58 #fridayHours
59 .hoursLabel
60 label(for="fridayStartTime") Friday:
61 .hoursClosed
62 input(type='checkbox',id='fridayClose',value='Closed')
63 |Closed
64 .clear
65 #fridayInput.row
66 .three.ninth.mobile
67 input(type='text',name='fridayStartTime',id='fridayStar
68 .one.ninth.mobile
69 span#fridayStartSuffix.suffix am
70 .one.ninth.mobile.hoursSeparator
71 |to
72 .three.ninth.mobile
73 input(type='text',name='fridayEndTime',id='fridayEndTime',placeholder='5:00')
74 .one.ninth.mobile
75 span#fridayEndSuffix.suffix pm
76 hr
77 #saturdayHours
78 .hoursLabel
79 label(for="saturdayStartTime") Saturdays:
80 .hoursClosed
81 input(type='checkbox',id='saturdayClose',value='Closed')
82 |Closed
83 .clear
84 #saturdayInput.row
85 .three.ninth.mobile
86 input(type='text',name='saturdayStartTime',id='saturdayStartTime',placeholder='9:00')
87 .one.ninth.mobile
88 span#saturdayStartSuffix.suffix am
89 .one.ninth.mobile.hoursSeparator
90 |to
91 .three.ninth.mobile
92 input(type='text',name='saturdayEndTime',id='saturdayEndTime',placeholder='5:00')
93 .one.ninth.mobile
94 span#saturdayEndSuffix.suffix pm

```

```

1 each day in ['Monday','Tuesday','Wednesday','Thursday','Friday','Saturday','Sunday']
2 - var dayLower = day.toLowerCase();
3 div(id=dayLower+'Hours')
4 .hoursLabel
5 label(for=dayLower+"StartTime")= day+": "
6 .hoursClosed
7 input(type='checkbox',id=dayLower+'Close',value='Closed')
8 |Closed
9 .clear
10 div(id=dayLower+'Input',class='row')
11 .three.ninth.mobile
12 input(type='text',name=dayLower+'StartTime',id=dayLower+'StartTime',placeholder='9:00')
13 .one.ninth.mobile
14 span(id=dayLower+'StartSuffix',class='suffix') am
15 .one.ninth.mobile.hoursSeparator
16 |to
17 .three.ninth.mobile
18 input(type='text',name=dayLower+'EndTime',id=dayLower+'EndTime',placeholder='5:00')
19 .one.ninth.mobile
20 span(id=dayLower+'EndSuffix',class='suffix') pm
21 hr

```





ALLO CHÉRIE ?  
**ÇA VA  
MIXER**

# PRINCIPE DE BASE DES MIXINS

## JADE

```
// Déclaration

mixin list
  ul
    li Fruits
    li Légumes

// Utilisation

+list
+list
```



## HTML

```
<ul>
<li>Fruits</li>
<li>Légumes</li>
</ul>
<ul>
<li>Fruits</li>
<li>Légumes</li>
</ul>
```



## PARTIE 2 : MANŒVRES AVANCÉES

# MIXINS À VALEUR ÉDITABLE

### JADE

```
// Déclaration

mixin menu(name)
  li.nav-li
    a= name

// Utilisation

ul
  +list('Accueil')
  +list('Infos')
  +list('Contact')
```



### HTML

```
<ul>
  <li class="nav-li"><a>Accueil</a></li>
  <li class="nav-li"><a>Infos</a></li>
  <li class="nav-li"><a>Contact</a></li>
</ul>
```

## PARTIE 2 : MANŒVRES AVANCÉES

# MIXINS À VALEURS ÉDITABLES

```
// Déclaration

mixin article(title, content, author)
  div.article-wrapper
    h1= title
    p= content
    span= author

// Utilisation

+list(
  'Pirates',
  'Je suis un pirate.',
  'Barbe Noire'
)
+list(
  'Banquier',
  'Je suis un banquier.',
  'Bernard Madoff'
)
```



```
<div class="article-wrapper">
  <h1>Pirate</li>
  <p>Je suis un pirate.</li>
  <span>Barbe Noire</li>
</div>
<div class="article-wrapper">
  <h1>Banquier</li>
  <p>Je suis un banquier.</li>
  <span>Bernard Madoff</li>
</div>
```

## PARTIE 2 : MANŒVRES AVANCÉES

# MIXIN BLOCK

### JADE

```
// Déclaration

mixin article(title)
  .article
    h1= title
    if block
      block
    else
      p Article is empty

// Utilisation

+list('Presentation')

+list('Presentation')
  p.
    Bonjour, permettez-moi
    de me présenter.
```



### HTML

```
<div class="article">
  <h1>Presentation</h1>
  <p>Article is empty</p>
</div>
<div class="article">
  <h1>Presentation</h1>
  <p>Bonjour, permettez-moi
  de me présenter.</p>
</div>
```



**I HAVE ZE  
POWER!**

© Les Maîtres de l'Univers

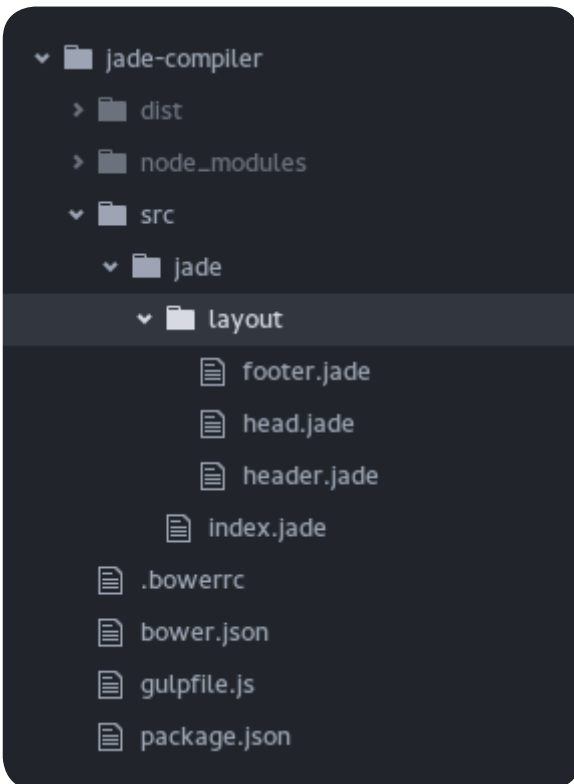
PARTIE 3

# CONFIGURATION DE GULP

Houston, we haven't problem

## PARTIE 3 : CONFIGURATION DE GULP

# MINIMUM VALUABLE PRODUCT



```
var gulp = require('gulp'),
    jade = require('gulp-jade');

gulp.task('jade', function() {
  return gulp.src('src/jade/*.jade')
    .pipe(jade())
    .pipe(gulp.dest('dist/'));
});
```



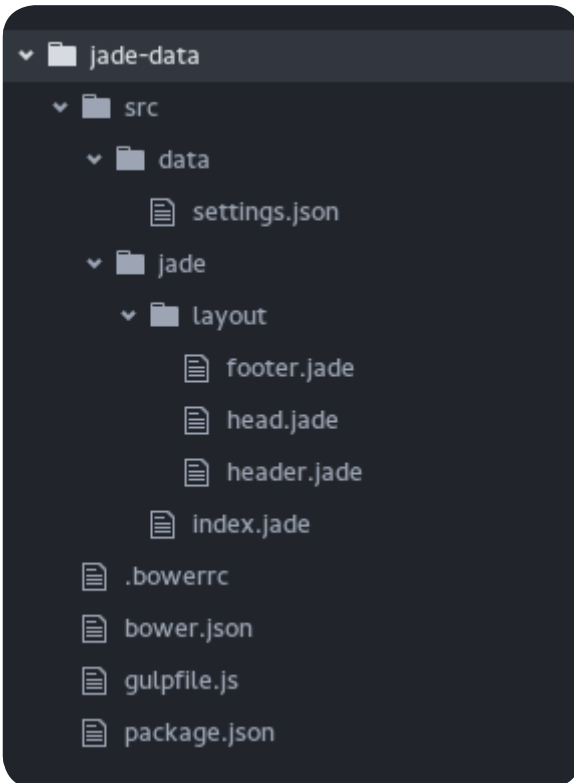


**BAH !?!  
ELLES SONT OÙ LES  
DATAS ?**



## PARTIE 3 : CONFIGURATION DE GULP

# LES DATAS



```
var gulp = require('gulp'),
    jade = require('gulp-jade'),
    data = require('gulp-data');

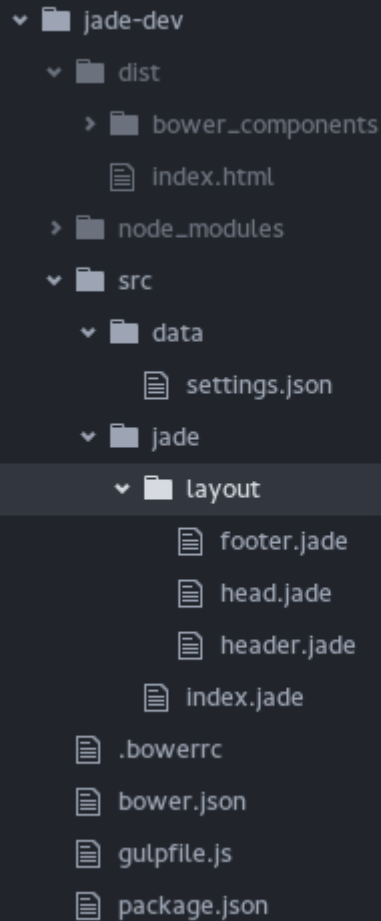
gulp.task('jade', function() {
  return gulp.src('src/jade/*.jade')

    .pipe(data(function() {
      return require('./src/data/settings.json');
    }))

    .pipe(jade())
    .pipe(gulp.dest('dist/'));
});
```

## PARTIE 3 : CONFIGURATION DE GULP

# LE HTML "(DÉ)CONTRACTÉ"



```
▼ jade-dev
  ▼ dist
    > bower_components
      index.html
    > node_modules
  ▼ src
    ▼ data
      settings.json
    ▼ jade
      ▼ layout
        footer.jade
        head.jade
        header.jade
        index.jade
      .bowerrc
      bower.json
      gulpfile.js
      package.json
```

```
gulp.task('jade', function() {
  return gulp.src('src/jade/*.jade')
    .pipe(data(function() {
      return require('./src/data/settings.json');
    }))

    .pipe(jade({
      pretty: true
    }))

    .pipe(gulp.dest('dist/'));
});
```

## PARTIE 3 : CONFIGURATION DE GULP

# AUTO-COMPILATION

```
gulp.task('watch', function () {  
  gulp.watch('src/jade/**/*.jade',  
    ['jade']);  
});
```

# EN CONCLUSION


Jade, c'est bien !

- **SYNTAXE PROPRE ET CONDENSÉE**
- **CODE SIMPLIFIÉ ET LISIBLE**
- **MEILLEURES PERFORMANCES**  
(RAPIDITÉ, LISIBILITÉ, RÉPÉTITIONS...)
- **MANIABLE ET PERMISSIF**  
(ITERATIONS, MIXINS...)
- **FACILITÉ DE MAINTENANCE**  
(LIBILITÉ, INCLUDES...)

RETROUVEZ CES DOCUMENTS SUR



Mais d'abord, suivez-nous sur twitter ;)



**BOUJOU  
BIEN !**