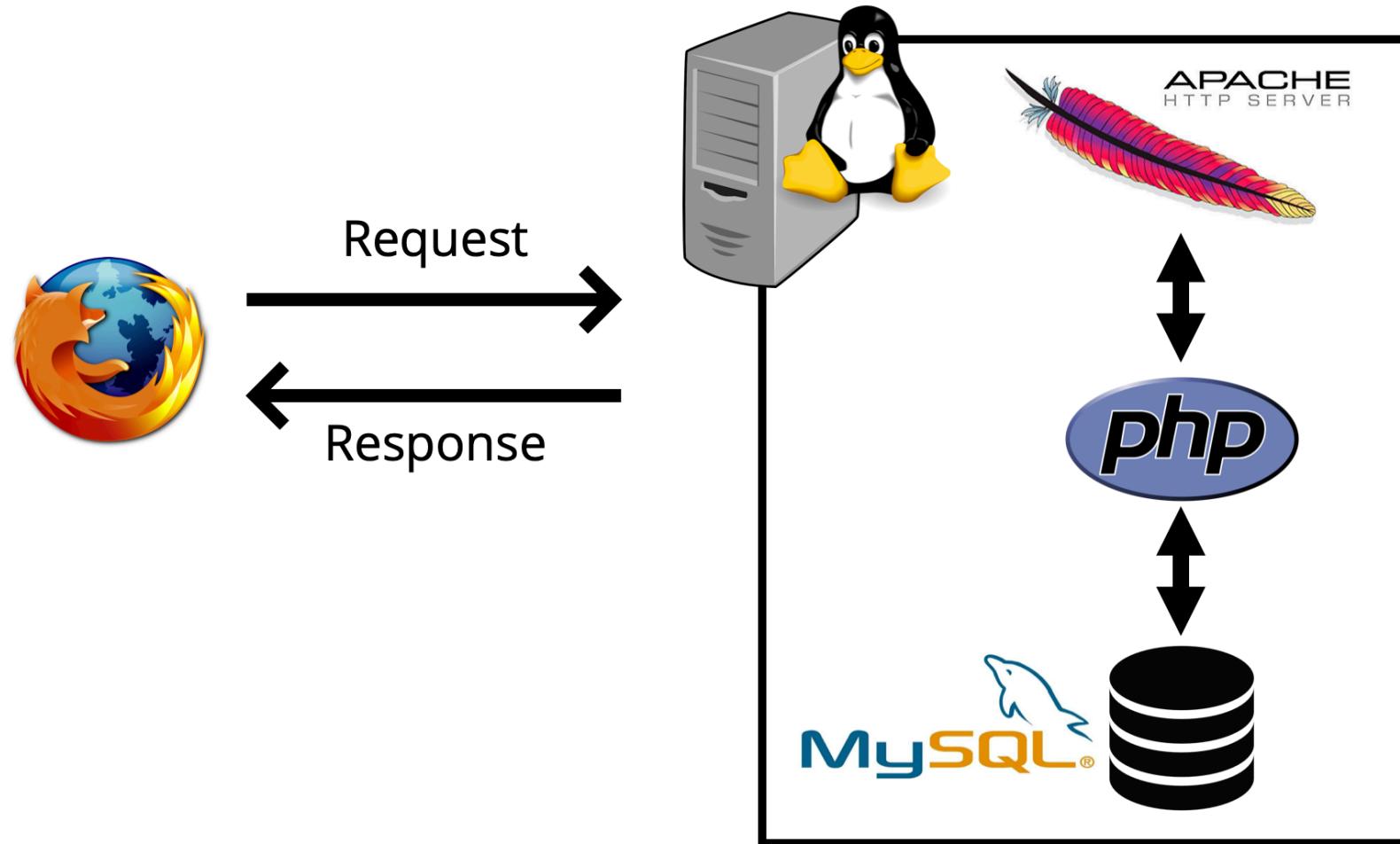


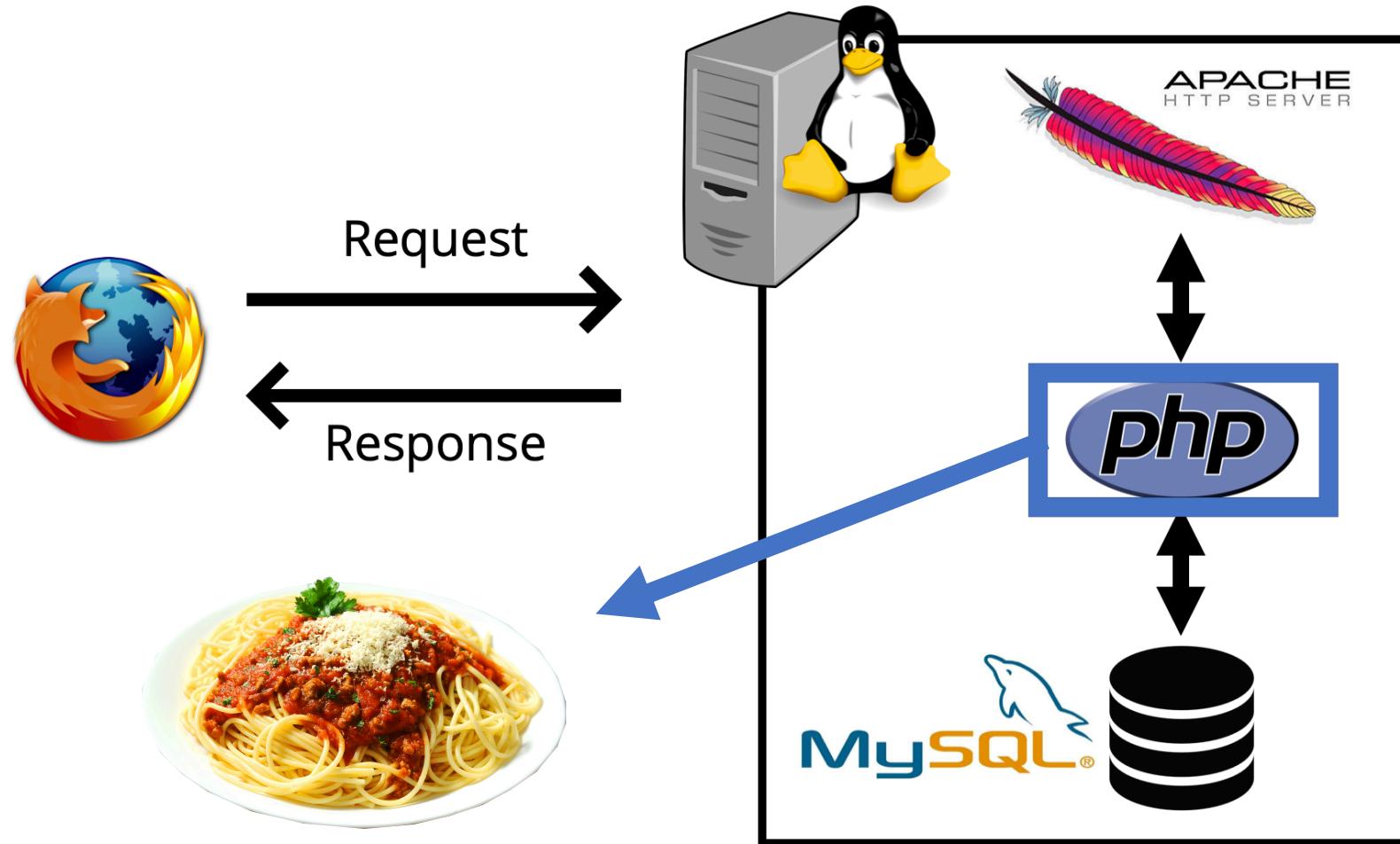
Developing a web application with **Flask**

Pau Andrio [pau.andrio@bsc.es]

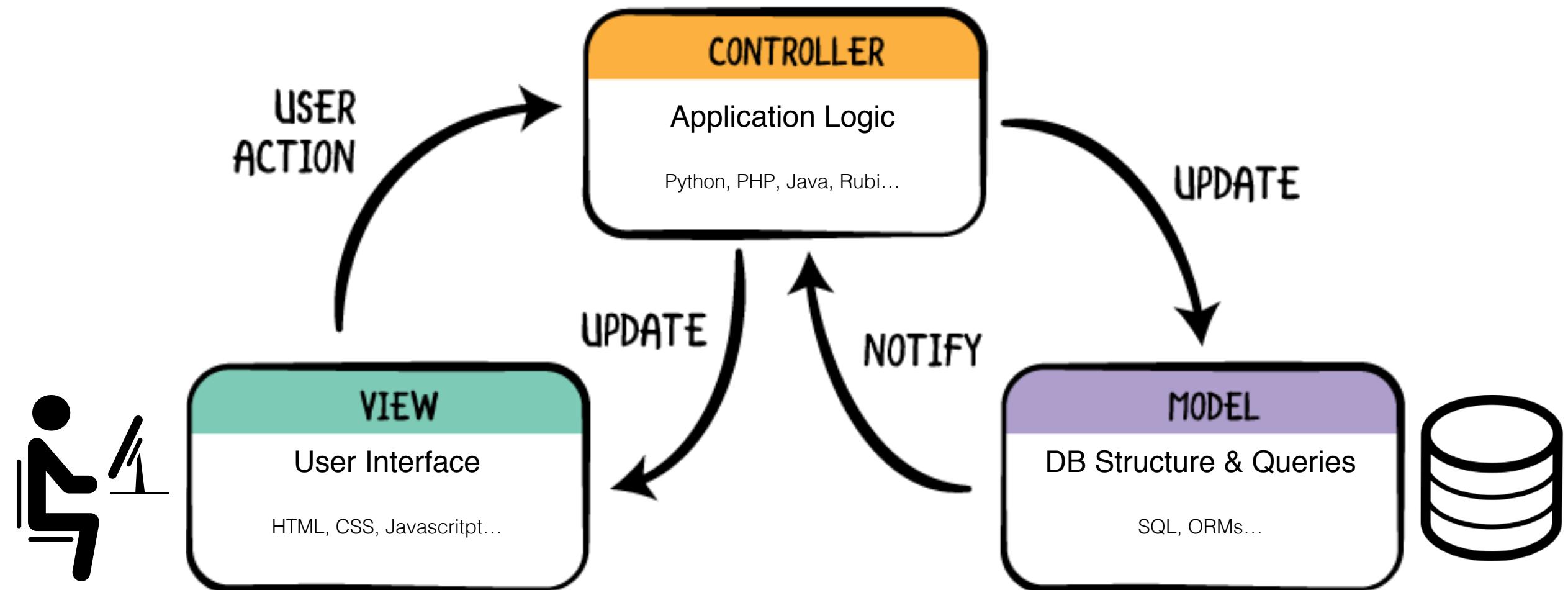
LAMP: Linux, Apache, MySQL, PHP



LAMP: Linux, Apache, MySQL, PHP



MVC: Model View Controller



Questions

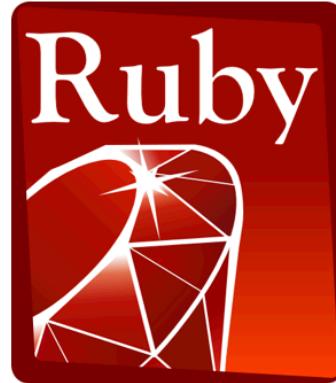
Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Are you familiar with the MVC software design pattern?
2. Do you understand the benefits of dividing the code in components?
3. Have you ever heard the term Spaghetti code before?

Example of an answer in the chat:

MyUserName: YYN

(Big) Frameworks



(Micro μ) Frameworks



a micro framework for PHP

Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Have you ever used a developing framework of any kind?
2. Are you going to use PHP/Slim for your project?
3. Are you going to use Python/Flask for your project?

Example of an answer in the chat:

MyUserName: YYN



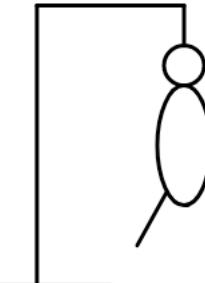
python



Flask

web development,
one drop at a time

Hangman game



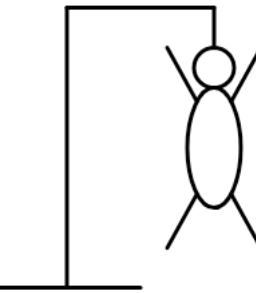
CONTROLLER

You won!

[Home](#) [Play again](#)

Errors (3/6): V G A

Hangman game



CONTROLLER

You lost :-(

[Home](#) [Play again](#)

Errors (6/6): I Y U P A W

The hangman game

Hangman game

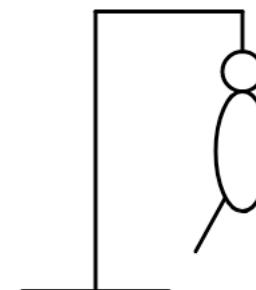
Your name Play!

Top 10

#	Player	Word	Errors	Points
1	Phoebe	CONFIRMATION		130
2	Chandler	REPRESENTS	U	112
3	Ross	CONFIDENTIAL	W Q	112
4	Rachel	EXPLICITLY	W R Q	96
5	Monica	ACQUISITIONS	B G W X Y	80
6	Joey	MECHANICAL	Z G Y T O	76



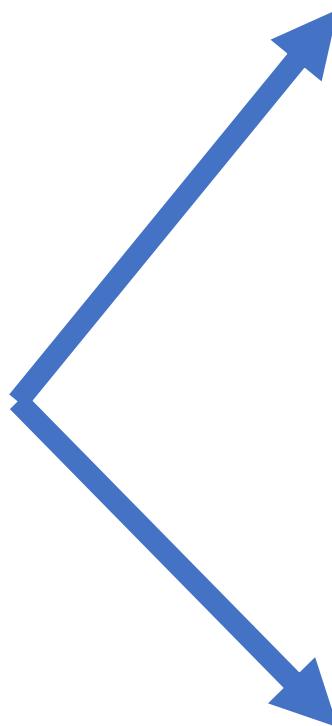
Hangman game



C _ N _ R _ _ _ R

Errors (3/6): V G A

Try letter



Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Are you familiar with the Hangman rules?

Example of an answer in the chat:

MyUserName: Y

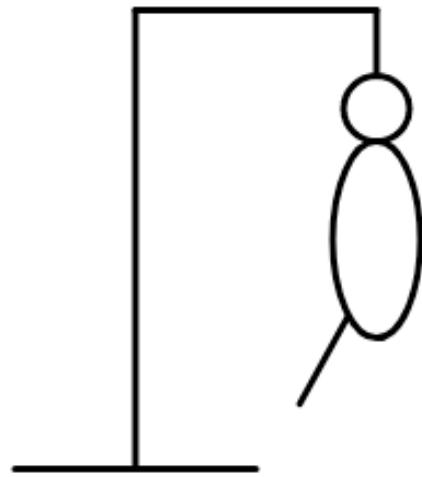
Home view

Hangman game

Top 10				
#	Player	Word	Errors	Points
1	Phoebe	CONFIRMATION		130
2	Chandler	REPRESENTS	U	112
3	Ross	CONFIDENTIAL	W Q	112
4	Rachel	EXPLICITLY	W R Q	96
5	Monica	ACQUISITIONS	B G W X Y	80
6	Joey	MECHANICAL	Z G Y T O	76

Play view

Hangman game



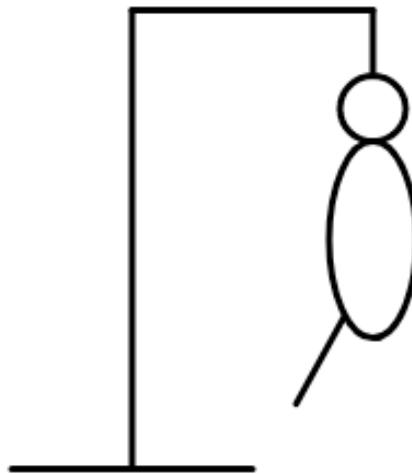
C _ N _ R _ _ _ _ R

Errors (3/6): V G A

Try letter

Play view

Hangman game



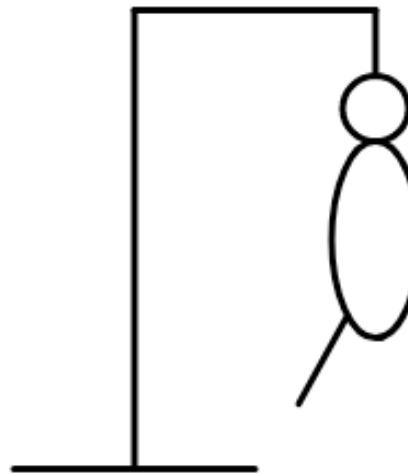
C _ N _ R _ _ _ R

Errors (3/6): V G A

Try letter



Hangman game



CONTROLLER

You won!

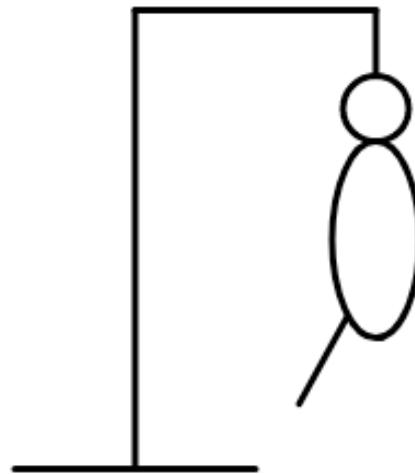
Home

Play again

Errors (3/6): V G A

Play view

Hangman game

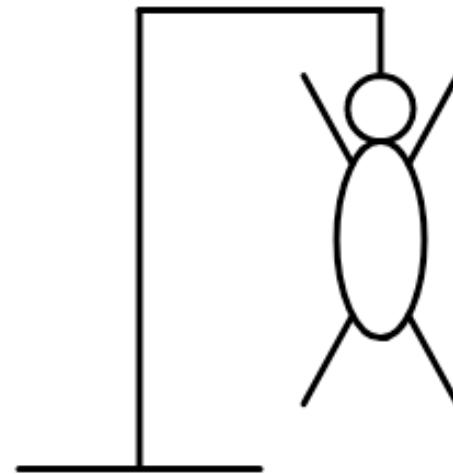


C _ N _ R _ _ _ R

Errors (3/6): V G A

Try letter

Hangman game



C O N T R O L L E R

You lost :-(

Home

Play again

Errors (6/6): I Y U P A W

Download & Execute hangman

Terminal:

```
pip install Flask Flask-SQLAlchemy Werkzeug==0.16.1
```

```
git clone https://github.com/PauAndrio/hangman.git
```

```
cd hangman
```

```
python -c 'from hangman import db; db.create_all()'
```

```
python hangman.py
```

Firefox:

```
http://localhost:5000/
```

Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Have you been able to run the game in Firefox?

2. Have you been able to open the code in your code editor?

Example of an answer in the chat:

MyUserName: YY

Project Structure

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

Project Structure

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

words.txt

motherboard runtime inbox focal bibliographic eden distant incl champagne ala decimal hq deviation superintendent propecia dip nbc samba hostel housewives employ mongolia penguin magical influences inspections irrigation miracle manually reprint reid wt hydraulic centered robertson flex yearly penetration wound belle rosa conviction hash omissions writings hamburg lazy mv mpg retrieval qualities cindy lolita fathers carb charging cas marvel lined cio dow prototype importantly rb petite apparatus upc terrain dui pens explaining yen strips gossip rangers nomination empirical mh rotary worm dependence discrete beginner boxed lid sexuality polyester cubic deaf commitments suggesting sapphire kinase skirts mats remainder crawford labeled privileges televisions specializing marking commodities pvc serbia sheriff griffin declined guyana spies blah mime neighbor motorcycles elect highways thinkpad concentrate intimate reproductive preston deadly cunt feof bunny chevy molecules rounds longest refrigerator tions intervals sentences dentists usda exclusion workstation holocaust keen flyer peas dosage receivers urls customize disposition variance navigator investigators cameroon baking adaptive computed needle baths enb gg cathedral brakes og nirvana ko fairfield owns til invision sticky destiny generous madness emacs climb blowing fascinating landscapes

Pick random word: (hangman.py)

```
def random_word():
    words = [line.strip() for line in open('words.txt') if len(line) > 10]
    return random.choice(words).upper()
```

Project Structure: The View

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

Project Structure: The View + Ajax

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

View Files:

Content: HTML

```
<div id="mega-green">tree</div>
```

tree

Style: CSS

```
#mega-green { color: green; font-size: xxx-large; }
```

tree

main.css:

```
1 th {  
2   text-align: center  
3 }  
4  
5 .vspace {  
6   margin-top: 25px;  
7 }  
8  
9 .spaced {  
10  letter-spacing: 15px;  
11 }  
12  
13 /* Drawing */  
14  
15 #drawing {  
16   fill: none;  
17   stroke: black;  
18   stroke-width: 3;  
19   stroke-linecap: round;  
20 }  
21  
22 #hangman-drawing > * {  
23   display: none  
24 }
```

Hangman game

 Play!

Top 10				
#	Player	Word	Errors	Points
1	Phoebe	CONFIRMATION		130
2	Chandler	REPRESENTS	U	112
3	Ross	CONFIDENTIAL	W Q	112
4	Rachel	EXPLICITLY	W R Q	96
5	Monica	ACQUISITIONS	B G W X Y	80
6	Joey	MECHANICAL	Z G Y T O	76

Hangman game

Your name

Play!

Top 10

#	Player	Word	Errors	Points
1	Phoebe	CONFIRMATION		130
2	Chandler	REPRESENTS	U	112
3	Ross	CONFIDENTIAL	Q W	112
4	Rachel	EXPLICITLY	R Q W	96
5	Monica	ACQUISITIONS	W G X B Y	80
6	Joey	MECHANICAL	G T Z O Y	76

Inspector

Console

Debugger

Style Editor

Performance

»

...

X

+

✖

bootstrap.min.css
1186 rules.

* main.css
5 rules.

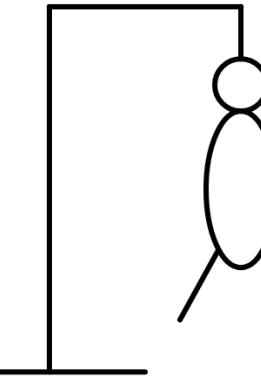
Save

```
1
2 ▼ th {
3   text-align: right
4 }
5
6 ▼ .vspace {
7   margin-top: 25px;
8 }
9
10 ▼ .spaced {
11   letter-spacing: 15px;
12 }
13
14 ▼ /* Drawing */
15
16 #drawing {
```

Hangman game

main.css:

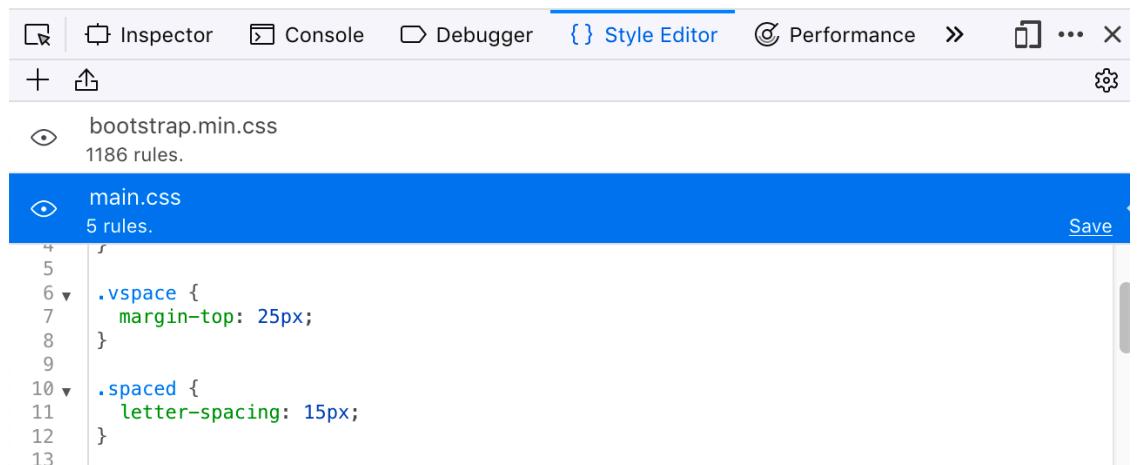
```
1 th {  
2   text-align: center  
3 }  
4  
5 .vspace {  
6   margin-top: 25px;  
7 }  
8  
9 .spaced {  
10  letter-spacing: 15px;  
11 }  
12  
13 /* Drawing */  
14  
15 #drawing {  
16   fill: none;  
17   stroke: black;  
18   stroke-width: 3;  
19   stroke-linecap: round;  
20 }  
21  
22 #hangman-drawing > * {  
23   display: none  
24 }
```



F O _ _ D _ T _ O _ _

Errors (3/6): C X Z

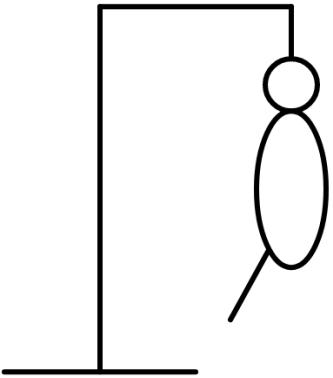
Try letter



The screenshot shows a browser's developer tools interface with the "Style Editor" tab selected. At the top, there are tabs for Inspector, Console, Debugger, and Performance. Below the tabs, there are buttons for adding and removing styles. The sidebar lists "bootstrap.min.css" with 1186 rules and "main.css" with 5 rules, which is currently selected. The main area displays the content of main.css:

```
1 .vspace {  
2   margin-top: 25px;  
3 }  
4  
5 .spaced {  
6   letter-spacing: 15px;  
7 }
```

Hangman game



FO_D_T_O_

Errors (3/6): CXZ



Try letter

A screenshot of the Chrome DevTools interface, specifically the Style Editor tab. The tab bar at the top includes icons for Inspector, Console, Debugger, Style Editor (which is highlighted in blue), and Performance. Below the tab bar, there are buttons for creating new styles (+) and deleting existing ones (trash can). The main area shows two style definitions:

```
* main.css
  5 rules.
```

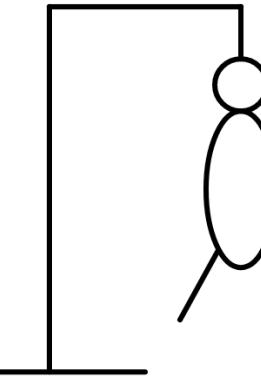
```
4
5
6 .vspace {
7
8 }
9
10 .spaced {
11
12 }
```

A blue bar highlights the first rule under the heading "main.css". On the far right of this bar, there is a "Save" button. The code editor shows numbered lines 4 through 13, with the cursor positioned at the start of line 4.

Hangman game

main.css:

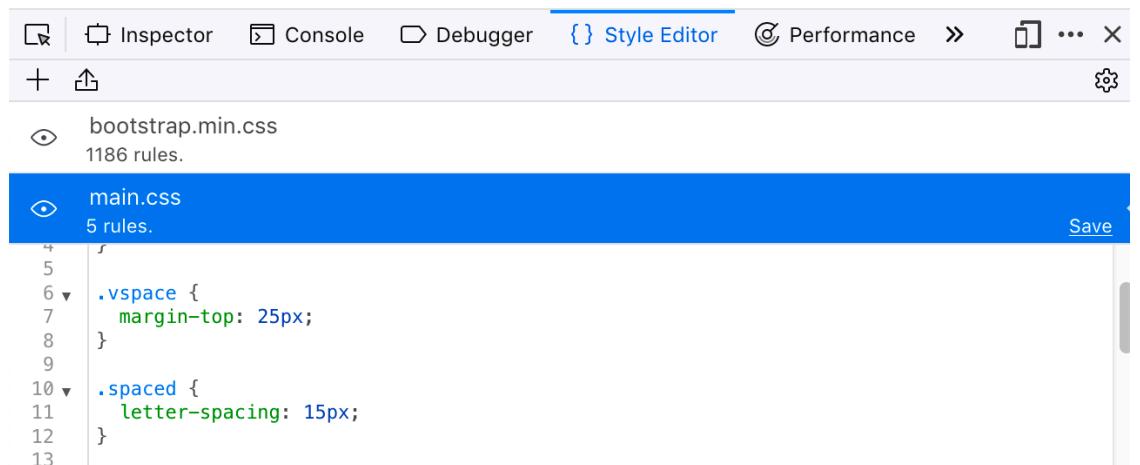
```
1 th {  
2   text-align: center  
3 }  
4  
5 .vspace {  
6   margin-top: 25px;  
7 }  
8  
9 .spaced {  
10  letter-spacing: 15px;  
11 }  
12  
13 /* Drawing */  
14  
15 #drawing {  
16   fill: none;  
17   stroke: black;  
18   stroke-width: 3;  
19   stroke-linecap: round;  
20 }  
21  
22 #hangman-drawing > * {  
23   display: none  
24 }
```



F O _ _ D _ T _ O _ _

Errors (3/6): C X Z

Try letter



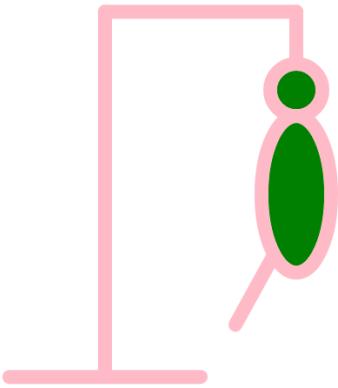
Style Editor

bootstrap.min.css 1186 rules.

main.css 5 rules.

```
1 .vspace {  
2   margin-top: 25px;  
3 }  
4  
5 .spaced {  
6   letter-spacing: 15px;  
7 }
```

Hangman game



F O _ _ D _ T _ O _ _

Errors (3/6): C X Z



Try letter

File Inspector Console Debugger Style Editor Performance ... ×

+ ↗

bootstrap.min.css 1186 rules.

* main.css 5 rules. Save

```
16 ▼ #drawing {  
17   fill: green;  
18   stroke: pink;  
19   stroke-width: 8;  
20   stroke-linecap: round;  
21 }  
22  
23 ▼ #hangman-drawing > * {  
24   display: none  
25 }
```

Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Are you familiar with CSS?

2. Have you been able to play with the styles in Firefox?

Example of an answer in the chat:

MyUserName: YY

Project Structure: The View + Ajax

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

index.html: html + Jinja2

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8     <title>Hangman game</title>
9
10    <!-- Bootstrap -->
11    <link rel="stylesheet"
12      href="{{ url_for('static', filename='bootstrap.min.css') }}">
13    <link rel="stylesheet"
14      href="{{ url_for('static', filename='main.css') }}">
15  </head>
16  <body>
17    <div class="container text-center">
18      <h1>Hangman game</h1>
19      <div class="row vspace">
20        <div class="col-md-6 col-md-offset-3">
21          {% block content %}{% endblock %}
22        </div>
23      </div>
24    </div>
25
26    <script src="{{ url_for('static', filename='jquery.min.js') }}"/></script>
27    <script src="{{ url_for('static', filename='bootstrap.min.js') }}"/></script>
28
29    {% block bottom %}{% endblock %}
30  </body>
31 </html>
```

home.html: html + Jinja2

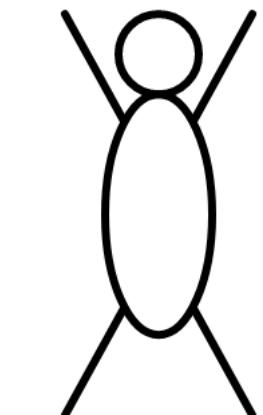
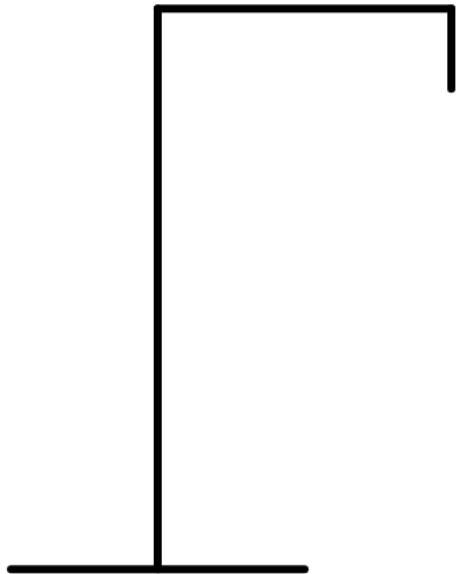
```
1  {% extends 'index.html' %}  
2  
3  {% block content %}  
4      <form action="{{ url_for('new_game') }}" class="form-inline">  
5          <div class="form-group vspace">  
6              <input name="player" class="input-lg" required="required"  
7                  placeholder="Your name">  
8              <button class="btn btn-primary btn-lg" type="submit">Play!</button>  
9          </div>  
10     </form>  
11  
12     <div class="panel panel-default vspace">  
13         <div class="panel-heading">Top 10</div>  
14         <table class="table">  
15             <thead>  
16                 <tr>  
17                     <th>#</th>  
18                     <th>Player</th>  
19                     <th>Word</th>  
20                     <th>Errors</th>  
21                     <th>Points</th>  
22             <tbody>  
23                 {% for game in games %}  
24                     <tr>  
25                         <td>{{ loop.index }}</td>  
26                         <td>{{ game.player }}</td>  
27                         <td class="text-success">{{ game.current }}</td>  
28                         <td class="text-danger spaced">{{ game.errors }}</td>  
29                         <td>{{ game.points }}</td>  
30                 {% endfor %}  
31             </tbody>  
32         </table>  
33     </div>  
34 {% endblock %}
```

home.html:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8   <title>Hangman game</title>
9
10  <!-- Bootstrap -->
11  <link rel="stylesheet" href="/static/bootstrap.min.css">
12  <link rel="stylesheet" href="/static/main.css">
13
14 </head>
15 <body>
16   <div class="container text-center">
17     <h1>Hangman game</h1>
18     <div class="row vspace">
19       <div class="col-md-6 col-md-offset-3">
20
21       <form action="/play" class="form-inline">
22         <div class="form-group vspace">
23           <input name="player" class="input-lg" required="required"
24             placeholder="Your name">
25           <button class="btn btn-primary btn-lg" type="submit">Play!</button>
26         </div>
27       </form>
28
29       <div class="panel panel-default vspace">
30         <div class="panel-heading">Top 10</div>
31         <table class="table">
32           <thead>
33             <tr>
34               <th>#</th>
35               <th>Player</th>
36               <th>Word</th>
37               <th>Errors</th>
38               <th>Points</th>
39           <tbody>
40
41             <tr>
42               <td>1</td>
43               <td>Phoebe</td>
44               <td class="text-success">CONFIRMATION</td>
45               <td class="text-danger spaced"></td>
46               <td>130</td>
47
48             <tr>
49               <td>2</td>
50               <td>Chandler</td>
51               <td class="text-success">REPRESENTS</td>
52               <td class="text-danger spaced">U</td>
53               <td>112</td>
54
55             <tr>
56               <td>3</td>
57               <td>Ross</td>
58               <td class="text-success">CONFIDENTIAL</td>
59               <td class="text-danger spaced">QW</td>
60               <td>112</td>
61
```

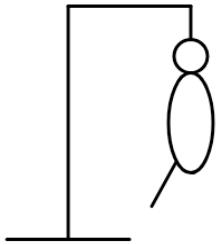
play.html: html + Jinja2 1/3

```
1  {% extends 'index.html' %}  
2  
3  {% block content %}  
4      <svg width="250" height="230" id="drawing">  
5          <!-- Structure -->  
6          <line y1="220" y2="220" x1="10" x2="120" />  
7          <line y1="10" y2="220" x1="65" x2="65" />  
8          <line y1="10" y2="10" x1="65" x2="175" />  
9          <line y1="10" y2="40" x1="175" x2="175" />  
10  
11         <!-- Hangman -->  
12         <g id="hangman-drawing">  
13             <circle cx="175" cy="55" r="15" />  
14             <ellipse cx="175" cy="115" rx="20" ry="45" />  
15             <line y1="150" y2="190" x1="162" x2="140" />  
16             <line y1="150" y2="190" x1="188" x2="210" />  
17             <line y1="80" y2="40" x1="162" x2="140" />  
18             <line y1="80" y2="40" x1="188" x2="210" />  
19         </g>  
20     </svg>
```



play.html: 2/3

Hangman game

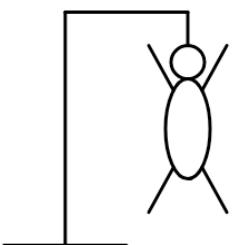


C _ N _ R _ _ _ _ R

Errors (3/6): V G A

 Try letter

Hangman game



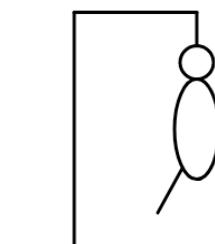
CONTROLLER

You lost :-(

Home Play again

Errors (6/6): I Y U P A W

Hangman game



CONTROLLER

You won!

Home Play again

Errors (3/6): V G A

```
22  {% if game.finished %}  
23      <h1>  
24          {% for letter in game.word %}  
25              <span class="  
26                  text-{% if letter in game.tried %}success{% else %}danger{% endif %}">  
27                  {{ letter }}  
28              </span>  
29          {% endfor %}  
30      </h1>  
31      <h3>  
32          {% if game.won %} You won! {% else %} You lost :-( {% endif %}  
33      </h3>  
34      <p>  
35          <a class="btn btn-default" href="{{ url_for('home') }}>Home</a>  
36          <a class="btn btn-primary"  
37              href="{{ url_for('new_game') }}?player={{ game.player }}>  
38              Play again  
39          </a>  
40      </p>  
41  {% else %}  
42      <h1 id="current" class="spaced">{{ game.current }}</h1>  
43  {% endif %}  
44  
45  <h2 class="vspace" id="errors">  
46      Errors ({{ game.errors|length }}/6):  
47      <span class="text-danger spaced">{{ game.errors }}</span>  
48  </h2>  
49  
50  {% if not game.finished %}  
51      <form method="post" id="letter-form" class="form-inline vspace">  
52          <div class="form-group">  
53              <input class="input-lg text-uppercase" maxlength="1" size="1"  
54                  name="letter" autocomplete="off" autofocus>  
55          </div>  
56          <button type="submit" class="btn btn-default btn-lg">Try letter</button>  
57      </form>  
58  {% endif %}  
59  {% endblock %}
```

Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Do you understand what the view in the MVC pattern is?

2. Do you understand how html + jinja2 templates are rendered?

Example of an answer in the chat:

MyUserName: YY

play.html: html + Jinja2 3/3

```
61  {% block bottom %}  
62      <script src="{{ url_for('static', filename='main.js') }}"></script>  
63      <script>  
64          updateDrawing('{{ game.errors }}');  
65      </script>  
66  {% endblock %}
```

How do we update the values of play.html without reloading every time we send a letter?

AJAX

AJAX

Asynchronous JavaScript and XML

AJAX

Asynchronous JavaScript and XML

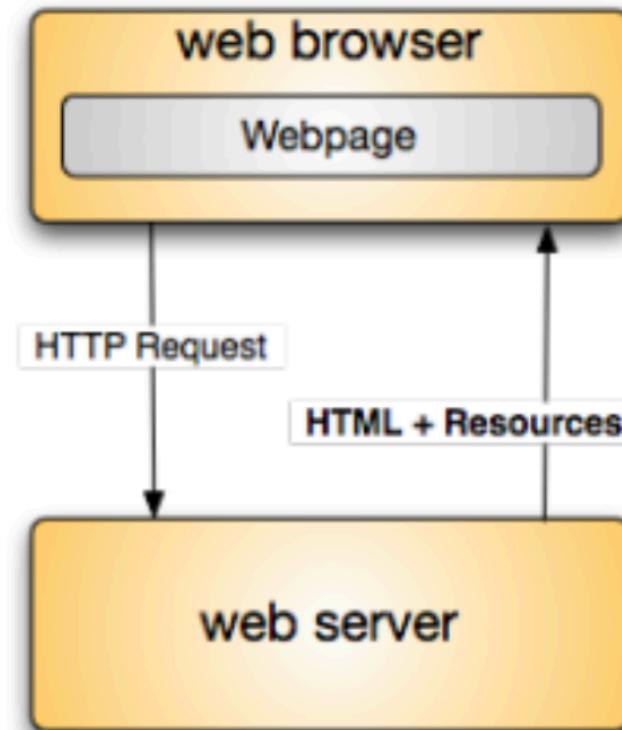
Change web content without reloading

AJAX

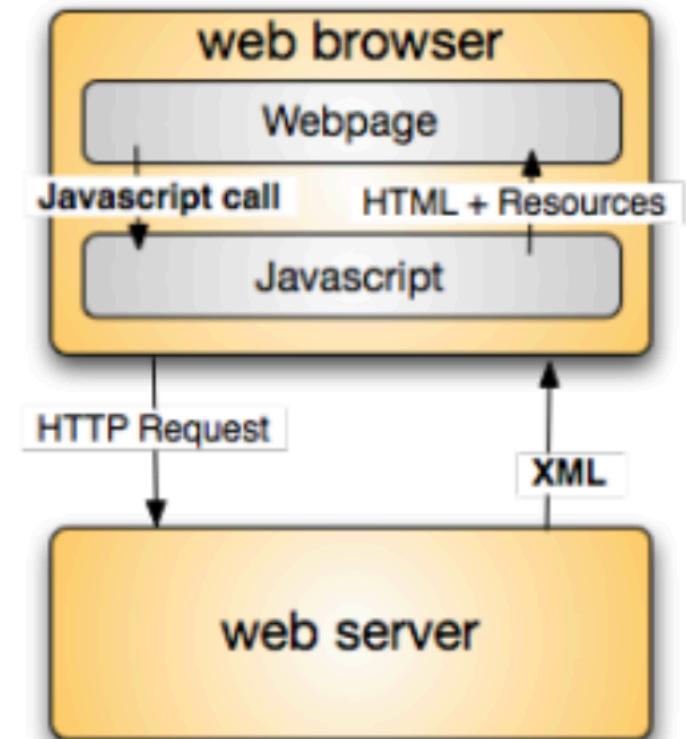
Change web content without refreshing



Traditional web model



AJAX web model



main.js: Javascript Jquery

```
1 /* Submit letter */
2
3 $('#letter-form').submit(function(e) {
4     var data = $("#letter-form").serialize();
5
6     /* Empty input */
7     $('#letter-form input').val('');
8
9     $.ajax({
10         type: "POST",
11         url: '',
12         data: data,
13         success: function(data) {
14             /* Refresh if finished */
15             if (data.finished) {
16                 location.reload();
17             }
18             else {
19                 /* Update current */
20                 $('#current').text(data.current);
21
22                 /* Update errors */
23                 $('#errors').html(
24                     'Errors (' + data.errors.length + '/6): ' +
25                     '<span class="text-danger spaced">' + data.errors + '</span>');
26
27                 /* Update drawing */
28                 updateDrawing(data.errors);
29             }
30         }
31     });
32     e.preventDefault();
33 });
34
35 function updateDrawing(errors) {
36     $('#hangman-drawing').children().slice(0, errors.length).show();
37 }
```

main.js: Javascript + Jquery

Status	Method	File	Domain	Cause	Type	Transferred	Size
200	GET	bootstrap.min.css	localhost:5000	stylesheet	css	cached	118.42 KB
200	POST	1748091743	localhost:5000	xhr	json	216 B	70 B

Headers Cookies Params Response Timings Stack Trace

Request URL: <http://localhost:5000/play/1748091743>

Request method: POST

Remote address: 127.0.0.1:5000

Status code: 200 OK [Edit and Resend](#) Raw headers

Version: HTTP/1.0

[Filter headers](#)

Response headers (146 B)

- Content-Length: 70
- Content-Type: application/json
- Date: Tue, 29 Jan 2019 22:30:32 GMT
- Server: Werkzeug/0.14.1 Python/3.7.1

Request headers (419 B)

- Accept: */*
- Accept-Encoding: gzip, deflate
- Accept-Language: en-US,en;q=0.5
- Connection: keep-alive
- Content-Length: 8
- Content-Type: application/x-www-form-urlencoded; charset=UTF-8
- Host: localhost:5000
- Referer: <http://localhost:5000/play/1748091743>
- User-Agent: Mozilla/5.0 (Macintosh; Intel ...) Gecko/20100101 Firefox/63.0
- X-Requested-With: XMLHttpRequest

Headers Cookies Params

[Filter request parameters](#)

Form data

letter: B

Headers Cookies Params Response

[Filter properties](#)

JSON

current: B
errors: A
finished: false

Response payload

```
1 {  
2   "current": "_____B____",  
3   "errors": "A",  
4   "finished": false  
5 }  
6
```

Questions

Please answer the following questions in the chat using a "Y" for yes and an "N" for no.

1. Any doubt about AJAX?
2. Where is the javascript code executed the (S)erver, the (B)rowser, or the (D)atabase?

Example of an answer in the chat:

MyUserName: ND

Project Structure: The Model

```
hangman
├── hangman.db
├── hangman.py
└── static
    ├── bootstrap.min.css
    ├── bootstrap.min.js
    ├── jquery.min.js
    ├── main.css
    └── main.js
└── templates
    ├── home.html
    ├── index.html
    └── play.html
└── words.txt
```

hangman.py (Model): SQLAlchemy + SQLite

```
8 # Database
9
10 app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///hangman.db'
11 db = SQLAlchemy(app)
12
13 # Model
14
15 def random_pk():
16     return random.randint(1e9, 1e10)
17
18 def random_word():
19     words = [line.strip() for line in open('words.txt') if len(line) > 10]
20     return random.choice(words).upper()
21
22 class Game(db.Model):
23     pk = db.Column(db.Integer, primary_key=True, default=random_pk)
24     word = db.Column(db.String(50), default=random_word)
25     tried = db.Column(db.String(50), default='')
26     player = db.Column(db.String(50))
27
28     def __init__(self, player):
29         self.player = player
30
31     @property
32     def errors(self):
33         return ''.join(set(self.tried) - set(self.word))
34
35     @property
36     def current(self):
37         return ''.join([c if c in self.tried else '_' for c in self.word])
38
39     @property
40     def points(self):
41         return 100 + 2*len(set(self.word)) + len(self.word) - 10*len(self.errors)
42
43     --#
44
45     # Game status
46
47     @property
48     def won(self):
49         return self.current == self.word
50
51
52     @property
53     def lost(self):
54         return len(self.errors) == 6
55
56
57     @property
58     def finished(self):
59         return self.won or self.lost
60
61
62
```

Project Structure: The Controller

```
hangman
|__ hangman.db
|__ hangman.py
|__ static
    |__ bootstrap.min.css
    |__ bootstrap.min.js
    |__ jquery.min.js
    |__ main.css
    |__ main.js
|__ templates
    |__ home.html
    |__ index.html
    |__ play.html
|__ words.txt
```

hangman.py (Controller): Flask

```
65 # Controller
66
67 @app.route('/')
68 def home():
69     games = sorted(
70         [game for game in Game.query.all() if game.won],
71         key=lambda game: -game.points)[:10]
72     return flask.render_template('home.html', games=games)
73
74 @app.route('/play')
75 def new_game():
76     player = flask.request.args.get('player')
77     game = Game(player)
78     db.session.add(game)
79     db.session.commit()
80     return flask.redirect(flask.url_for('play', game_id=game.pk))
81
82 @app.route('/play/<game_id>', methods=['GET', 'POST'])
83 def play(game_id):
84     game = Game.query.get_or_404(game_id)
85
86     if flask.request.method == 'POST':
87         letter = flask.request.form['letter'].upper()
88         if len(letter) == 1 and letter.isalpha():
89             game.try_letter(letter)
90
91     if flask.request.is_xhr:
92         return flask.jsonify(current=game.current,
93                             errors=game.errors,
94                             finished=game.finished)
95     else:
96         return flask.render_template('play.html', game=game)
```

```
44
45     def try_letter(self, letter):
46         if not self.finished and letter not in self.tried:
47             self.tried += letter
48             db.session.commit()
```

Deployment

Terminal:

```
ssh uXXXXXX@mmb.pcb.ub.es -p 22122
```

```
git clone your_git_project_url.git
```

```
cd yourproject
```

```
python3 -m virtualenv yourprojectenv
```

```
source yourprojectenv/bin/activate
```

```
pip install uwsgi Flask Werkzeug==0.16.1 # Add here any other dependency of your project
```

~/yourproject/wsgi.py

```
from yourproject import app

if __name__ == "__main__":
    app.run()
```

~/yourproject/yourproject.ini

```
[uwsgi]
module = wsgi:app

master = true
processes = 1

socket = yourproject.sock
chmod-socket = 777
vacuum = true

die-on-term = true

logto = /home/youruser/yourproject/yourproject.log
```

yourproject.service (email this file to sysadmin)

```
[Unit]
Description=uWSGI instance to serve yourproject
After=network.target

[Service]
User=youruser
Group=youruser
WorkingDirectory=/home/youruser/yourproject
Environment="PATH=/home/youruser/yourproject/yourprojectenv/bin"
ExecStart=/home/youruser/yourproject/yourprojectenv/bin/uwsgi --ini yourproject.ini

[Install]
WantedBy=multi-user.target
```

yourproject (email this file to sysadmin)

```
server {
    listen 8080;

    location /yourproject/ {
        include uwsgi_params;
        uwsgi_pass unix:///home/youruser/yourproject/yourproject.sock;
    }
}
```