



THE  
DEVELOPER'S  
CONFERENCE



TWITTER 50MIN



# THE DEVELOPER'S CONFERENCE

## **Integração: TV Digital e Twitter em 50 min**

**Felipe Iasi e Shyrles Monteiro**

# Quem somos?



THE  
DEVELOPER'S  
CONFERENCE



# Proposta



THE  
DEVELOPER'S  
CONFERENCE

- Prover uma infra-estrutura para ate 500 mil twits / hora.
- Moderar essas mensagens e disponibilizar um ponto de acesso para mensagens moderadas.
- Exibir as mensagens moderadas em uma TV conectada.

# Infraestrutura



THE  
DEVELOPER'S  
CONFERENCE



DigitalOcean

140DEV

140DEV



PHP



WEB2PY



MySQL



Ginga NCL





THE  
DEVELOPER'S  
CONFERENCE

# Vamos subir a infra?

# Twitter API



THE  
DEVELOPER'S  
CONFERENCE

- Criar chaves de acesso em [dev.twitter.com](https://dev.twitter.com)
- `consumer_key = TWITTER_CONSUMER_KEY`
- `consumer_secret = TWITTER_CONSUMER_SECRET`
- `access_token = OAUTH_TOKEN`
- `access_token_secret = OAUTH_SECRET`

# Digital Ocean



THE  
DEVELOPER'S  
CONFERENCE

- Criar uma conta com em [digitalocean.com](https://digitalocean.com)
- Utilize uns dos vouchers:
  - SHIPITFAST10
  - ALLSSD10



# Scripts



THE  
DEVELOPER'S  
CONFERENCE

➤ <https://gist.github.com/ficosta/06e2a799c1745cc9df39Utilitarios>

➤ web2py

➤ phpmyadmin

➤ 140dev

➤ criar a base de dados

➤ configurar chaves de database

➤ subir a aplicação web2py

➤ testar o ponto de acesso JSON



THE  
DEVELOPER'S  
CONFERENCE

# Moderar?



THE  
DEVELOPER'S  
CONFERENCE

# JSON final?

# Aplicacao



THE  
DEVELOPER'S  
CONFERENCE

## > main.ncl

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Generated by NCL Eclipse -->
<ncl id="main" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="rgTela" left="0" top="0" width="1280" height="720" zIndex="1"/>
      <region id="rgFundo" left="0" top="0" width="1280" height="720" zIndex="0"/>
    </regionBase>

    <descriptorBase>
      <descriptor id="dApp" region="rgTela" focusIndex="0" focusBorderWidth="0" />
      <descriptor id="dFundo" region="rgFundo" focusIndex="0" focusBorderWidth="0" />
    </descriptorBase>
  </head>

  <body>
    <port id="pInicioFundo" component="fundo"/>
    <port id="pInicio" component="0"/>
    <media id="nSettings" type="application/x-ginga-settings">
      <property name="service.currentKeyMaster" value="0" />
    </media>

    <media id="fundo" descriptor="dFundo" src="http://magicpaname.com/wp-content/uploads/2013/03/Rooney-avec-Manchester.jpg"/>
    <media id="0" descriptor="dApp" src="main.lua"/>
  </body>
</ncl>
```

# Aplicacao



THE  
DEVELOPER'S  
CONFERENCE

## >main.lua

```
json = require("json")

function callbackFunctionPrevisao(pResult, pData)
    if (pResult) then --true
        previsaoConteudo = json.decode(pData)
        apresentacaoTwiter();
    else -- false
        print("ERRO");
    end
end

dofile('cConn.lua');
```

```

function endApp()
    event.unregister(defaultHandler);
    event.post( "out", { class = "ncl", type = "presentation", action = "stop" } );
end

function Control(keys)
    log(__LOG_DEBUG, "Control");
    event.post("in", { class = "key", type = "press", key = keys })
end

function defaultHandler(evt)
    if ( evt.class == 'key' and evt.type == 'press' ) then
        if ( evt.key == 'CURSOR_LEFT' ) then
            if (indice > 1) then
                indice = indice - 1;
            else
                indice = qtdeTwitter;
            end
            return apresentacaoTwiter();
        elseif ( evt.key == 'CURSOR_RIGHT' ) then
            if (indice < qtdeTwitter) then
                indice = indice + 1;
            else
                indice = 1;
            end
            return apresentacaoTwiter();
        end
    end
end

event.register(defaultHandler);
startBaseConn();

```

# Aplicacao



THE  
DEVELOPER'S  
CONFERENCE

## >cCon.lua

```
if (pEvento.type == 'connect') then
    value = debugTextValue .. "Class: {" .. tostring(pEvento.class);
    value = value .. "} - Type: {" .. tostring(pEvento.type);
    value = value .. "} - Host: {" .. tostring(pEvento.host);
    value = value .. "} - Port: {" .. tostring(pEvento.port);
    value = value .. "} - Timeout: {" .. tostring(pEvento.timeout);
    value = value .. "} - Connection: {" .. tostring(pEvento.connection);
    value = value .. "} - Error: {" .. tostring(pEvento.error);
    value = value .. "} - Value: BEGIN {";
    value = value .. tostring(pEvento.value);
    value = value .. "} - Value: END";
```

```
event.post({
    class='tcp',
    type='data',
    connection=pEvento.connection,
    value="GET http://162.243.129.79/cockpit/social/tv_digital.json HTTP/1.0\nHost: 162.243.129.79\n\n",
    timeout=5
});
```

# Aplicacao



THE  
DEVELOPER'S  
CONFERENCE

## ➤ JSON Parser

- JSON4Lua: JSON encoding / decoding support for the Lua language.
- json Module.
- Author: Craig Mason-Jones
- Homepage: <http://json.luaforge.net/>
- Version: 0.9.50
- This module is released under the MIT License (MIT).
- Please see LICENCE.txt for details.



# Aplicacao

➤ Aplicativo



THE  
DEVELOPER'S  
CONFERENCE

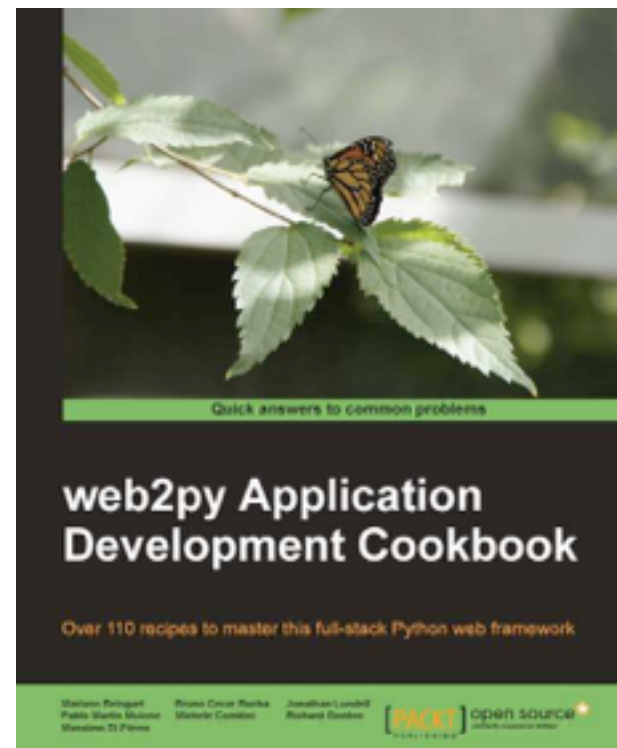
# O que ler?



➤ Twitter API engagement



➤ web2py cookbook



# Duvidas???



THE  
DEVELOPER'S  
CONFERENCE

> [felipeiasi@gmail.com](mailto:felipeiasi@gmail.com)

> [shyrlesmonteiro@gmail.com](mailto:shyrlesmonteiro@gmail.com)