

# Cliente API REST con LungoJS

Adolfo Sanz De Diego

Octubre 2013



# Acerca de

- El GUL es el **Grupo de Usuarios de Linux de la UC3M**.
- Grupo de personas con inquietudes en torno a la informática.
- Con la idea común de la utilización y promoción del **Software Libre**.
- Quedamos de vez en cuando y organizamos actividades sobre todo esto.
- El punto de unión es la **lista de correo** que está abierta a todo el mundo.

- Twitter: <http://twitter.com/guluc3m>
- Lista: [gul@gul.uc3m.es](mailto:gul@gul.uc3m.es)
- Ftp: <ftp://ftp.gul.uc3m.es/>
- Web: <http://www.gul.uc3m.es>
- Podcast: <http://holamundo.gul.es/>
- Blog: <http://planeta.gul.uc3m.es/>
- Linkedin: <http://www.linkedin.com/groups?gid=3451836>

# Adolfo Sanz De Diego



- **Antiguo programador web JEE**
- Hoy en día:
  - **Profesor de FP de informática:**
    - Hardware, Sistemas Operativos, Redes, Programación
  - **Formador Freelance:**
    - Java, Android
    - JavaScript, jQuery
    - JSF, Spring, Hibernate
    - Groovy & Grails
  - **Me gusta programar**

# Hackathon Lovers

- Para los amantes de los hackathones
- Meetup: <http://www.meetup.com/Hackathon-Lovers/>
- Twitter: <http://twitter.com/HackathonLovers>



- Es un **analizador de tweets** que extrae información semántica para conocer si el sentimiento general de los tweets de un determinado tema es positivo o negativo.
- **Web:** <http://tweetssentiment.com/>
- **Twitter:** <http://twitter.com/TweetsSentiment>

- Mi nick: **asanzdiego**
- AboutMe: <http://about.me/asanzdiego>
  - GitHub: <http://github.com/asanzdiego/>
  - Twitter: <http://twitter.com/asanzdiego>
  - Blog: <http://asanzdiego.blogspot.com.es>
  - LinkedIn: <http://www.linkedin.com/in/asanzdiego>

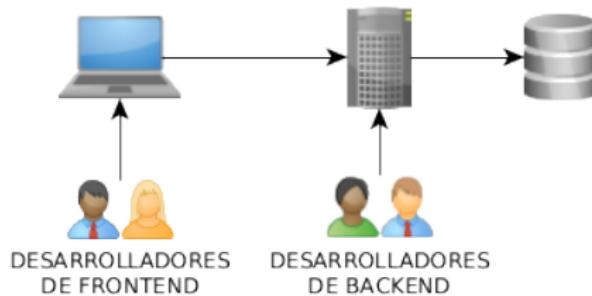
- Estas **transparencias** están hechas con:
  - <https://github.com/asanzdiego/deck.js-pandoc-slides>

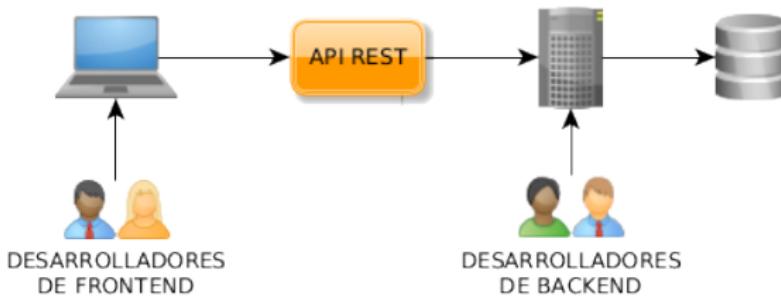
- Estas **transparencias** están bajo una licencia:
  - Creative Commons Reconocimiento-CompartirIgual 3.0
- El **código** de los programas están bajo una licencia:
  - GPL 3.0

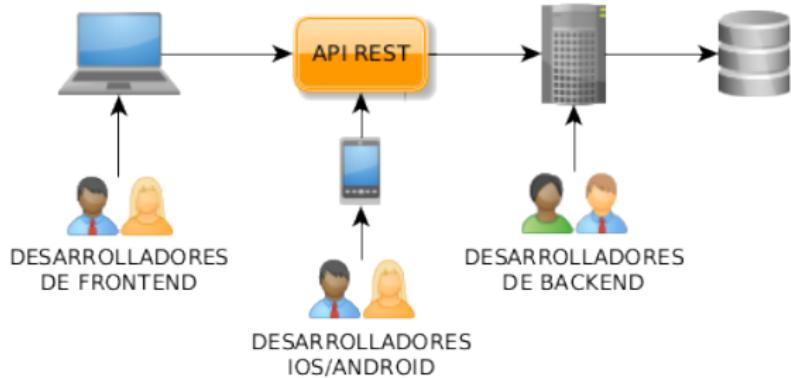
- Transparencias:
  - SlideShare:
    - <http://www.slideshare.net/asanzdiego/api-restfullungojsclientbeamer>
  - GitHub Pages:
    - <http://asanzdiego.github.io/curso-api-restful-lungojs-client-2013/slides/export/api-restful-lungojs-client.slides.html>
- Código:
  - GitHub:
    - <http://github.com/asanzdiego/curso-api-restful-lungojs-client-2013/tree/master/src>

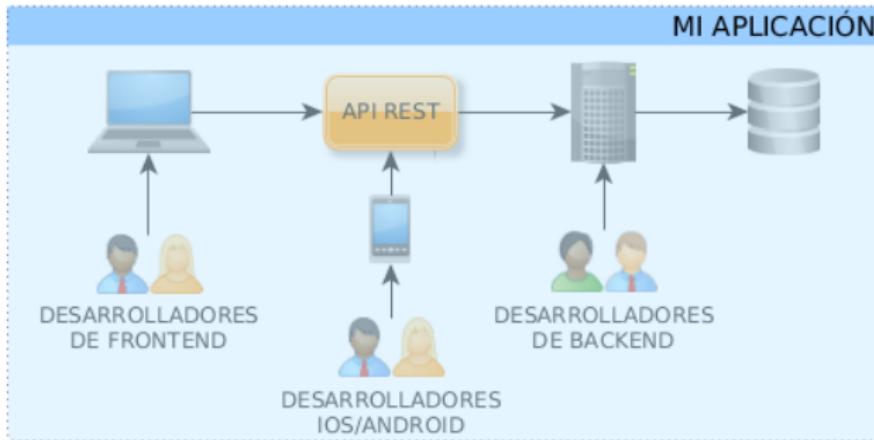
||

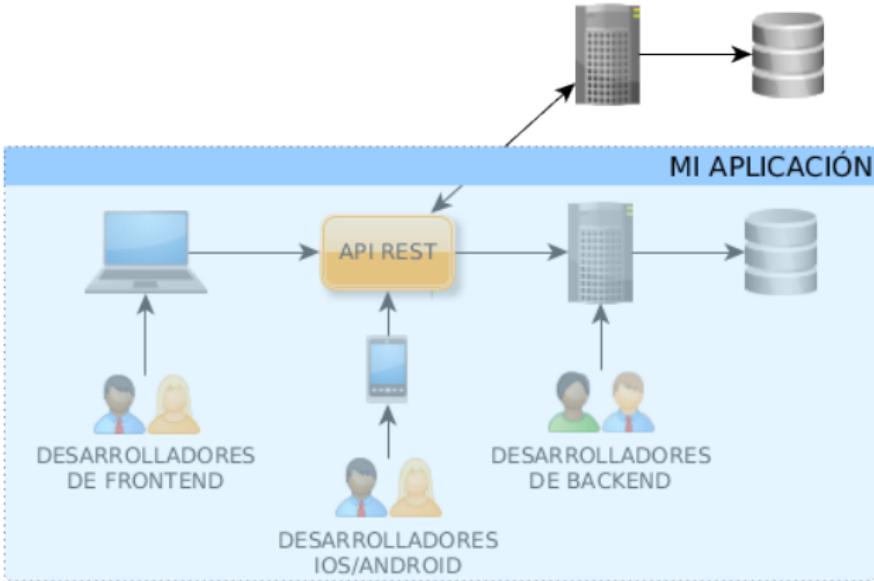
APIs ¿Para qué?

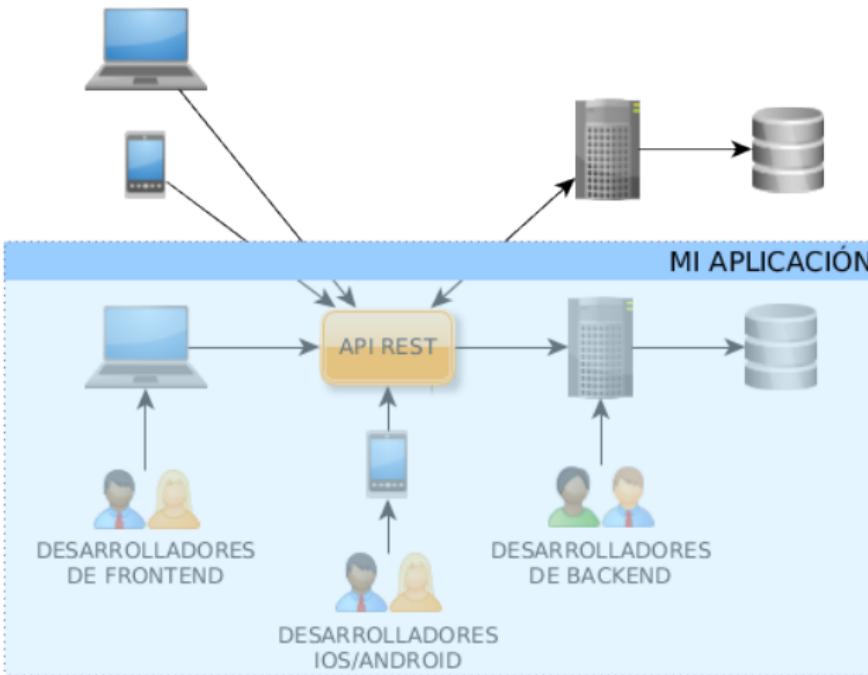


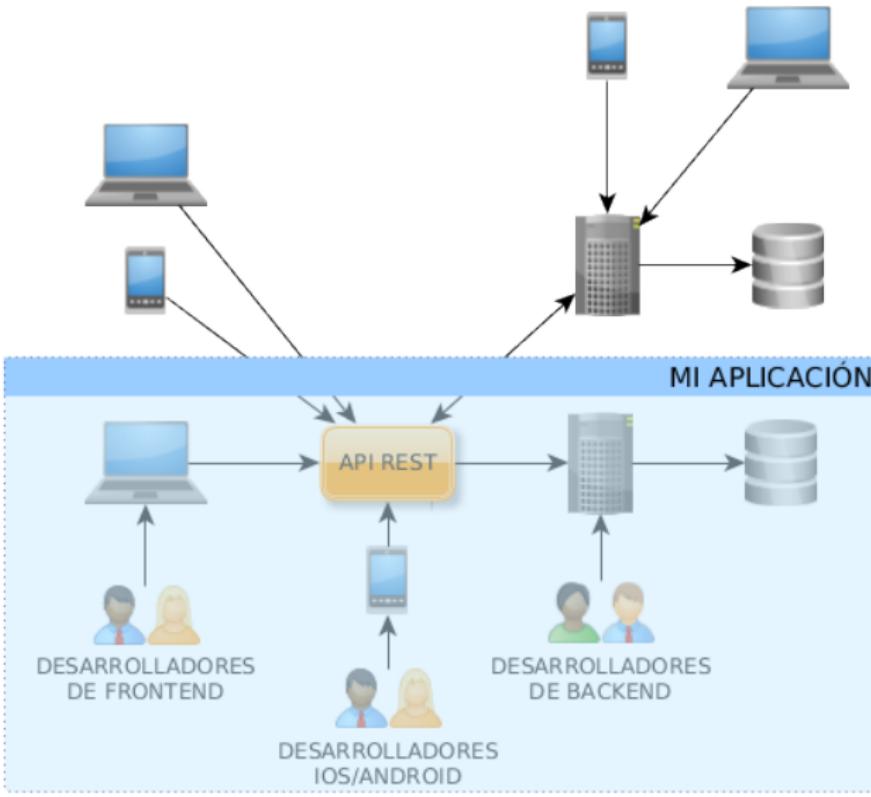




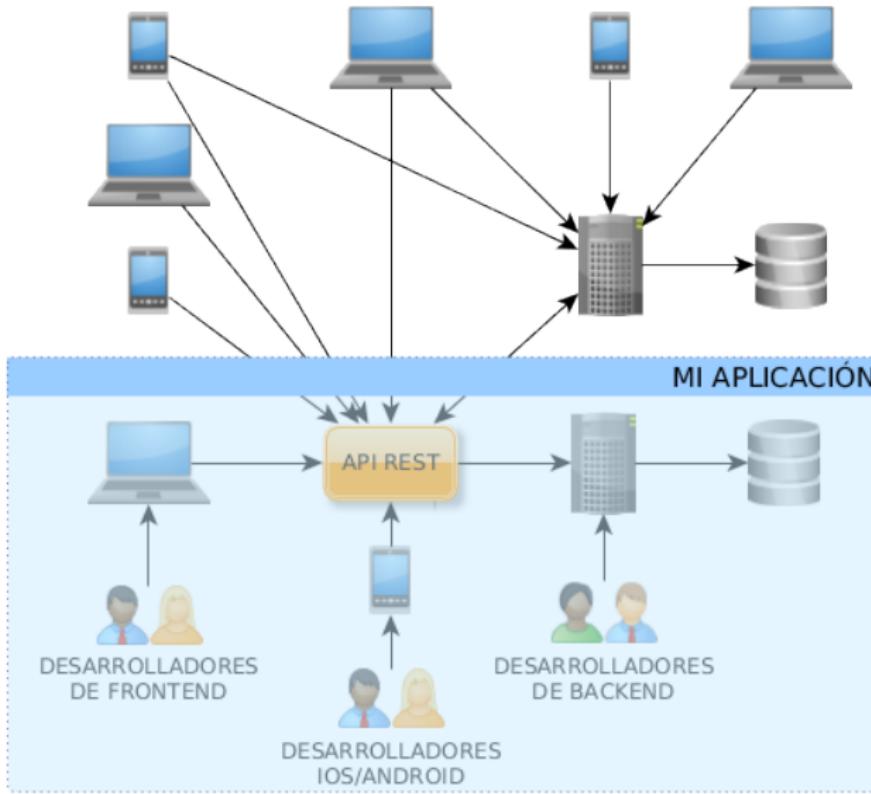


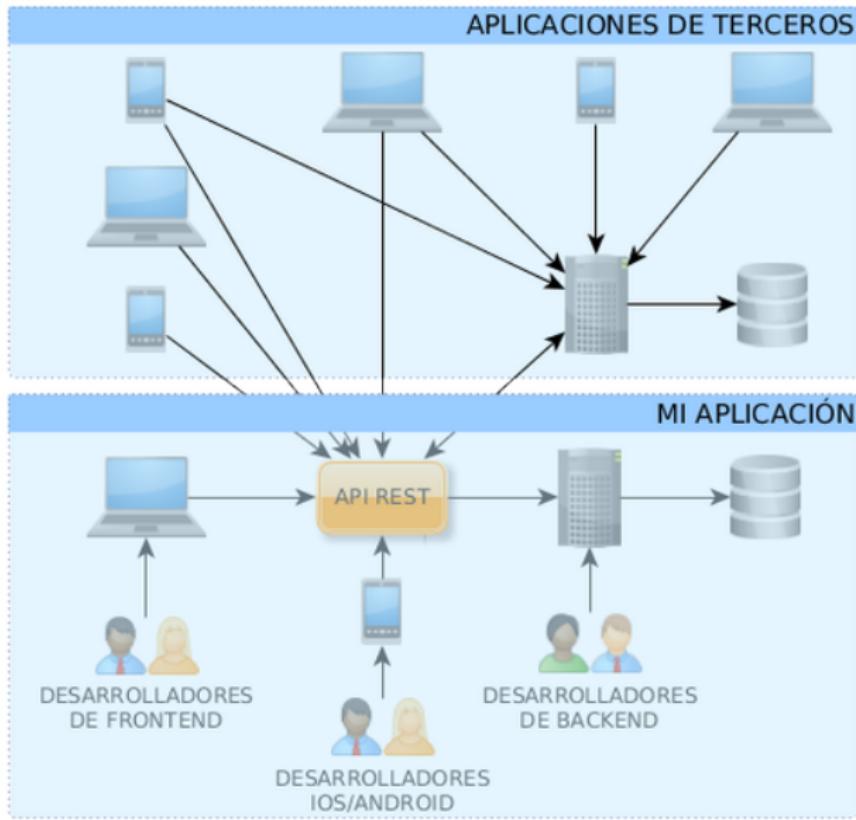






# Apps mixtas











Exponlas tú





Exponlas tú





## APIs RESTful

- REST (**R**epresentational **S**tate **T**ransfer) es una técnica de arquitectura de software para sistemas hipermedia distribuidos como la World Wide Web.
- En REST una **URL** (Uniform Resource Locator) representa un **recurso**.
- Se puede acceder al recurso o modificarlo mediante los **métodos del protocolo HTTP**:

GET, POST, PUT, DELETE

- **http://myhost.com/talk**
  - GET > Devuelve todas las charlas.
  - POST > Crear una nueva charla.
- **http://myhost.com/talk/123**
  - GET > Devuelve la charla con id=123
  - PUT > Actualiza la charla con id=123
  - DELETE > Borra la charla con id=123

- Se pueden utilizar los errores del protocolo HTTP:

- 200 Successful
- 201 Created
- 202 Accepted
- 301 Moved Permanently
- 400 Bad Request
- 401 Unauthorised
- 402 Payment Required
- 403 Forbidden
- 404 Not Found
- 405 Method Not Allowed
- 500 Internal Server Error
- 501 Not Implemented

- Es **más sencillo** (tanto la API como la implementación).
- Es **más rápido** (peticiones más ligeras que se pueden cachear).
- Es **multiformato** (HTML, XML, JSON, etc.).
- Se complementa muy bien con **AJAX**.

- REST se refiere a un tipo de arquitectura de software
  - Se utiliza como **nombre**
  - Se utiliza como por ejemplo: success = éxito.
- Si un servicio web es **RESTful** indica que implementa dicha arquitectura.
  - Se utiliza como **adjetivo**
  - Se utiliza como por ejemplo: successful = exitoso).

- A veces el **ful** se confunde con **full** = completo.
  - Y se refiere a los servicios web REST**full**  
Aquellos que implementan una API con todos los métodos del protocolo HTTP.
  - Y se refiere a los servicios web REST (**sin el full**)  
Aquellos que NO implementan una API con todos los métodos del protocolo HTTP.

# IV

## Clientes multidispositivo

# Distintos dispositivos







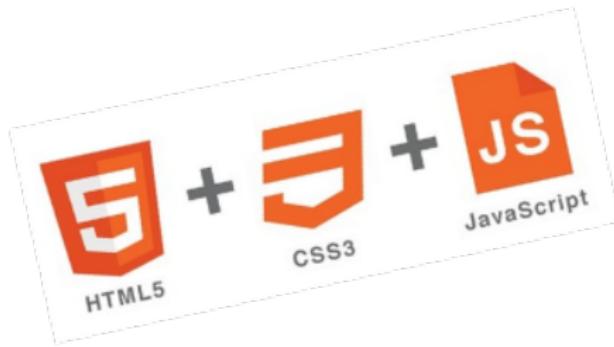
# Distintos navegadores



¡¡¡Socorro!!!

# ¡¡¡SOCORRO!!!



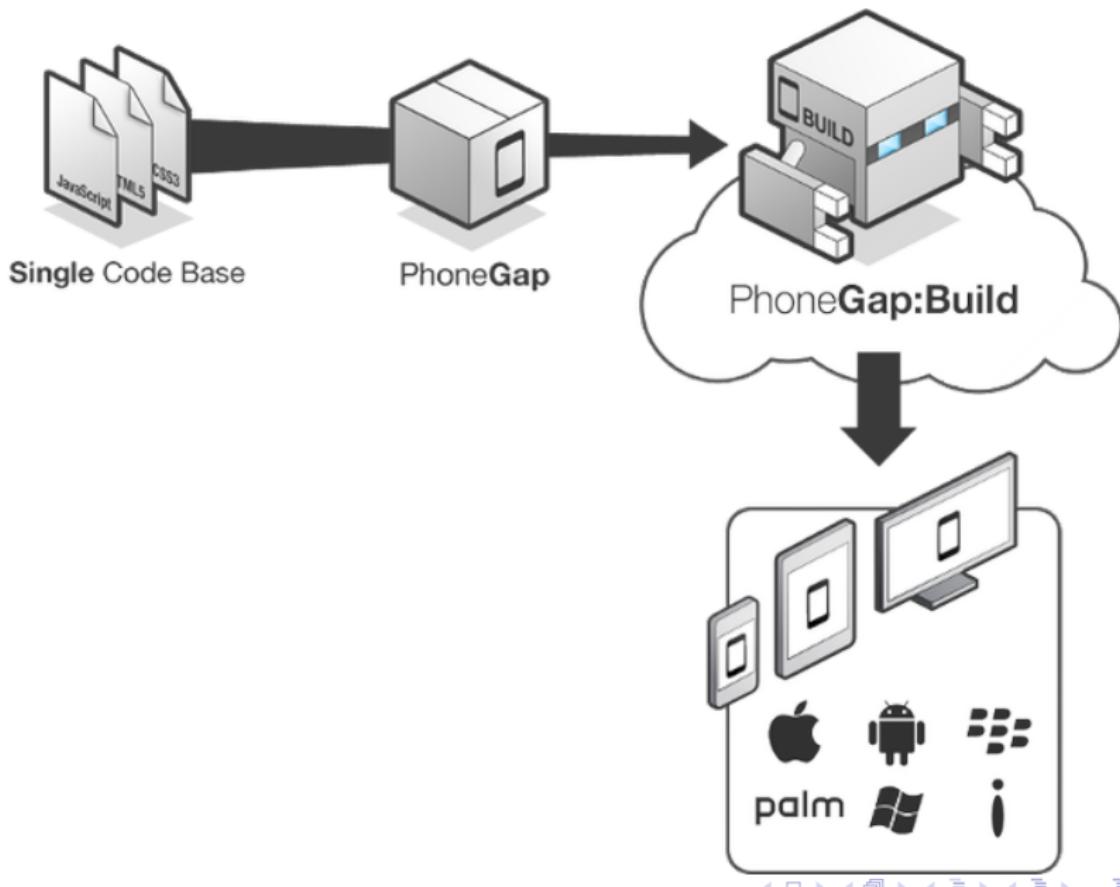






# Progressive Enhancement





V

Lungo.js

- jQTouch: <http://jqtjs.com/>
- Sencha Touch: <http://www.sencha.com/products/touch/>
- jQueryMobile: <http://jquerymobile.com/>
- hammer.js: <http://eightmedia.github.io/hammer.js/>

# ¿Por qué usar lungo?



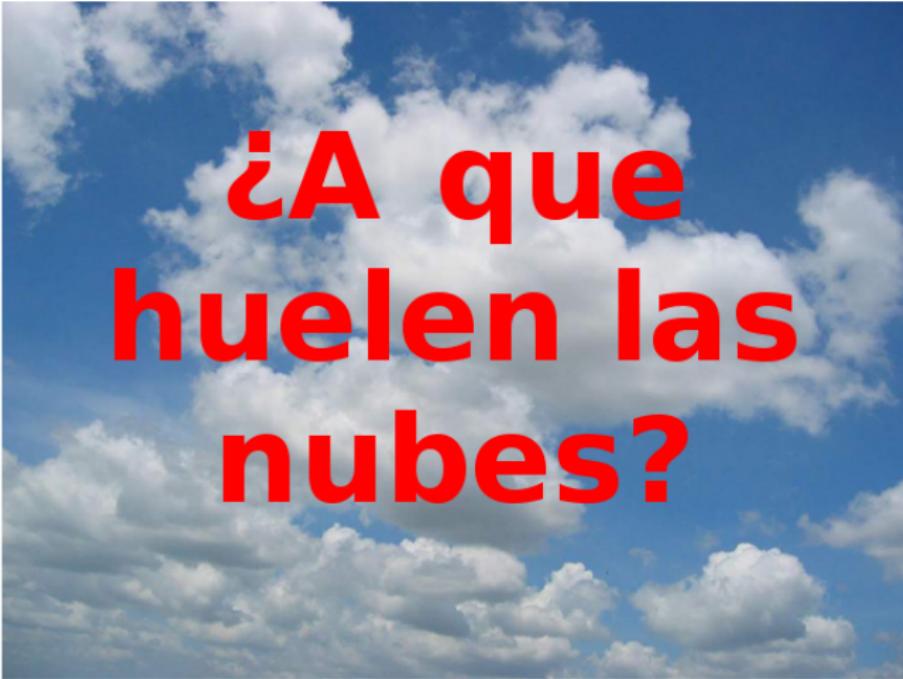


Pues no



**¡PUES NO!**

**¡ESTÁBAIS  
EQUIVOCADOS!**



¿A que  
huelen las  
nubes?



VI

## Código



- Puedes usar la estructura de directorios que quieras.
- Yo he usado esta:
  - **html**: los HTML con una carpeta para cada entidad del domino
  - **js**: los JS con una carpeta para cada entidad del domino
  - **lib**: carpeta con las dependencias a otros proyectos
  - Lungo depende de Quo que es una librería de los mismos autores de 'tipo' jQuery enfocada a móviles

```
// Search
$$('document').ready(function(){
    Lungo.dom('input[type=search]').on('keyup', ...);
});

// Server URL
var util_server_url = "http://localhost:3000";

// Error Notification
var util_errorNotification = function(message, error) {
    Lungo.Notification.error(message, "", "warning-sign", 2);
};

// Métodos REST
var util_ajaxGet = function(url, data, callback) {
    $$_.get(util_server_url+url, data, ..., 'json');
};
var util_ajaxPost = function(url, data, callback) {...};
var util_ajaxPut = function(url, data, callback) {...};
var util_ajaxDelete = function(url, data, callback) {...};
```

```
<body class="app">
<section id="main" data-transition="">
  <header data-title="Add Talk" class="extended"></header>
  <footer>
    <nav>
      <a href="talk-list.html" data-icon="list"></a>
      <a href="talk-add.html" data-icon="plus" class="active"></a>
    </nav>
  </footer>
  <article id="main-article" class="active list">
    <div class="form">
      <fieldset>
        <label>NAME:</label><input type="text" id="talkName" />
      </fieldset>...
    </div>
    <div>
      <a href="#addTalk" id="addTalk" class="button">Add</a>
      <a href="talk-list.html" class="button cancel">Cancel</a>
    </div>
  </article>
</section>
</body>
```

### Add Talk

TALK NAME: Enter the talk name

TALK DATE: dd/mm/yyyy hh:mm:ss

SPEAKER NAME: Enter the speaker name

SPEAKER MAIL: speaker@mail.es

TALK POINTS: 0

**Add** **Cancel**

The image shows a mobile application interface. At the top is a blue header bar with the title "Add Talk". Below it is a white form with five input fields. The first field is labeled "TALK NAME" with the placeholder "Enter the talk name". The second field is labeled "TALK DATE" with the placeholder "dd/mm/yyyy hh:mm:ss". The third field is labeled "SPEAKER NAME" with the placeholder "Enter the speaker name". The fourth field is labeled "SPEAKER MAIL" with the placeholder "speaker@mail.es". The fifth field is labeled "TALK POINTS" with the value "0". To the right of the "TALK POINTS" field is a small vertical scroll bar. At the bottom of the form are two buttons: a blue "Add" button on the left and a red "Cancel" button on the right. At the very bottom of the screen is a black navigation bar with several icons: a grid icon, a plus sign icon, and icons for back, forward, search, and other functions.

```
$$('#addTalk').tap(function() {  
  
    var data = {  
        talkName:      $$("#talkName").val(),  
        talkDate:      util_stringToDate($$("#talkDate").val()),  
        talkSpeaker:   $$("#talkSpeaker").val(),  
        talkSpeakerMail: $$("#talkSpeakerMail").val(),  
        talkPoints:    $$("#talkPoints").val()  
    };  
  
    // send data to server  
    util_ajaxPost('/talk', data, function(json) {  
        if(!json || json.error) {  
            util_errorNotification('ERROR adding talk', json.error);  
        } else {  
            util_successNotification('Talk saved', function() {  
                window.location.replace('talk-list.html');  
            });  
        }  
    });  
});  
});
```

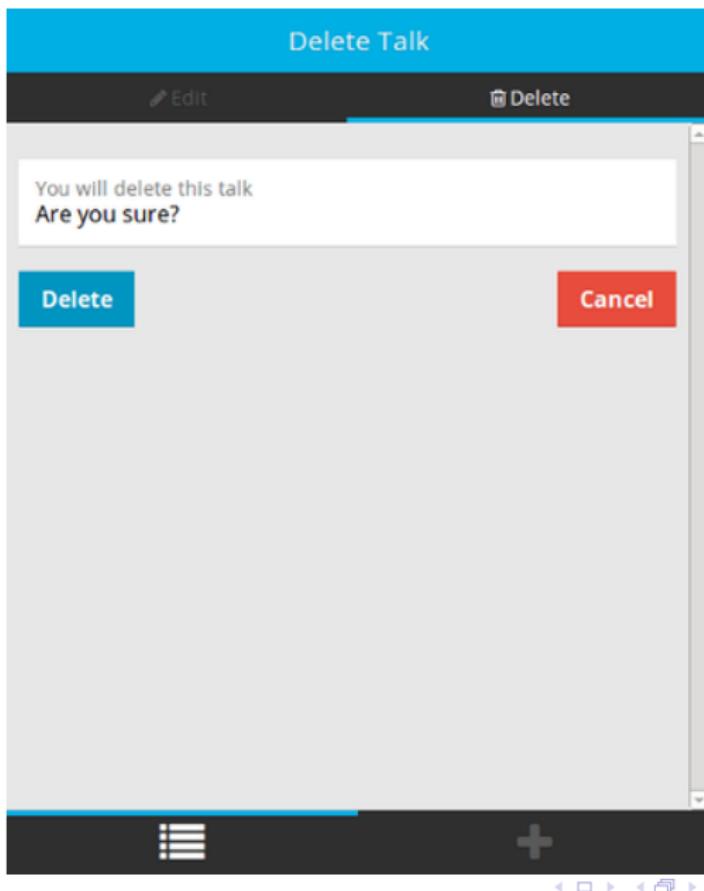
```
<body class="app">
  <section id="main" data-transition="">
    <header data-title="Edit Talk" class="extended"></header>
    <nav data-control="groupbar">
      <a href="#editTalkArticle" ...>Edit</a>
      <a href="#deleteTalkArticle" ...>Delete</a>
    </nav>
    <footer>
      ...
    </footer>
    <article id="editTalkArticle" class="list indented scroll">
      ...
    </article>
    <article id="deleteTalkArticle" class="list indented scroll">
      ...
    </article>
  </section>
</body>
```

## Edit Talk

[Edit](#)  [Delete](#)

TALK NAME:	Clientes APIs RESTful con Lungo
TALK DATE:	22/10/2012 19:00:00
SPEAKER NAME:	Adolfo Sanz De Diego
SPEAKER MAIL:	asanzdiego@gmail.com
TALK POINTS:	10

[Save](#) [Cancel](#)



```
$(document).ready(function(){

  var talkId = util_urlParams["talkId"];

  // get data from server
  util_ajaxGet('/talk/' + talkId, {}, function(json) {
    if(!json || json.error) {
      util_errorNotification('ERROR retrieving talk', json.error);
    } else {
      var talk = json;
      console.log('Talk retrieved');
      drawTalk(talk);
    }
  });

  // draw data
  var drawTalk = function(talk) {
    $('#talkName').val(talk.talkName);

    ...
  };
});
```

```
$$('#saveTalk').tap(function(){
    var talkId = util_urlParams['talkId'];

    var data = {
        talkName:      $$("#talkName").val(),
        ...
    };

    // send data to server
    util_ajaxPut('/talk/' + talkId, data, function(json) {
        if(!json || json.error) {
            util_errorNotification('ERROR saving talk', json.error);
        } else {
            util_successNotification('Talk saved', function() {
                window.location.replace('talk-list.html');
            });
        }
    });
});
```

```
$$('#deleteTalk').tap(function(){
  var talkId = util_urlParams['talkId'];
  // send data to server
  util_ajaxDelete('/talk/'+talkId, {}, function(json) {
    if(!json || json.error) {
      util_errorNotification('ERROR deleting talk', json.error);
    } else {
      util_successNotification('Talk deleted', function() {
        window.location.replace('talk-list.html');
      });
    }
  });
});
```

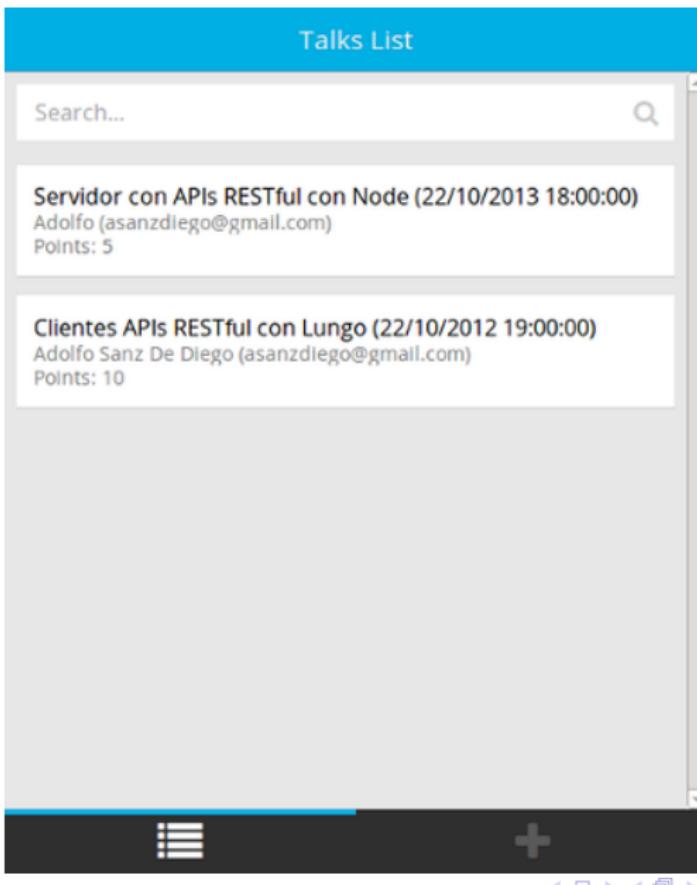
```
<body class="app">
  <section id="main" data-transition="">
    <header data-title="Talks List" class="extended"></header>
    <footer>
      <nav>
        <a href="talk-list.html" data-icon="list" class="active"></a>
        <a href="talk-add.html" data-icon="plus"></a>
      </nav>
    </footer>
    <article id="main-article" class="active list indented scroll">
      <div class="form">
        <fieldset data-icon="search">
          <input type="search" placeholder="Search..."/>
        </fieldset>
      </div>
      <ul id="talks"></ul>
    </article>
  </section>
</body>
```

## Talks List

Search... 

**Servidor con APIs RESTful con Node (22/10/2013 18:00:00)**  
Adolfo (asanzdiego@gmail.com)  
Points: 5

**Clientes APIs RESTful con Lungo (22/10/2012 19:00:00)**  
Adolfo Sanz De Diego (asanzdiego@gmail.com)  
Points: 10



Adolfo Sanz De Diego

Cliente API REST con LungoJS

The screenshot shows a mobile application interface titled "Talks List". At the top, there is a search bar containing the letter "Lu" and a magnifying glass icon. Below the search bar, a list item is displayed: "Clientes APIs RESTful con Lungo (22/10/2012 19:00:00)" followed by "Adolfo Sanz De Diego (asanzdiego@gmail.com)" and "Points: 10". The bottom of the screen features a navigation bar with icons for back, forward, and other controls.

Lu

Talks List

Clientes APIs RESTful con Lungo (22/10/2012 19:00:00)  
Adolfo Sanz De Diego (asanzdiego@gmail.com)  
Points: 10

+

Adolfo Sanz De Diego

Cliente API REST con LungoJS

```
$$('document').ready(function(){

    // get data from server
    util_ajaxGet('/talk', {}, function(json) {
        if(!json || json.error) {
            util_errorNotification('ERROR retrieving talks', json.error);
        } else {
            drawTalks(json);
        }
    });
}

// draw data
var drawTalks = function(talks) {
    for (var i = 0; i < talks.length; i++) {
        var talk = talks[i];
        $$("#talks").append(
            '<li data-action="search" class="selectable">' +
            '<a href="talk-edit.html?talkId=' + talk._id + '">' +
            '<strong>' + talk.talkName + '</strong></a></li>');
    }
};
});
```

VII

Demo

## VIII

¿Alguna pregunta?