

PhoneGap Build

Configuración para Generación de Aplicaciones iOS

Gabriel Huecas

27 de enero de 2016

Objetivo

- Configurar PhoneGap Build para que genere una aplicación nativa iOS
 - Son muchos pasos, pero es necesario disponer de un dispositivo Apple que habrá que registrar

Índice

- Ficheros necesarios en PhoneGap Build
- Certificado de desarrollador Apple
- Perfil de Suministro
- Configuración de PhoneGap Build

Compilación en PhoneGap Build

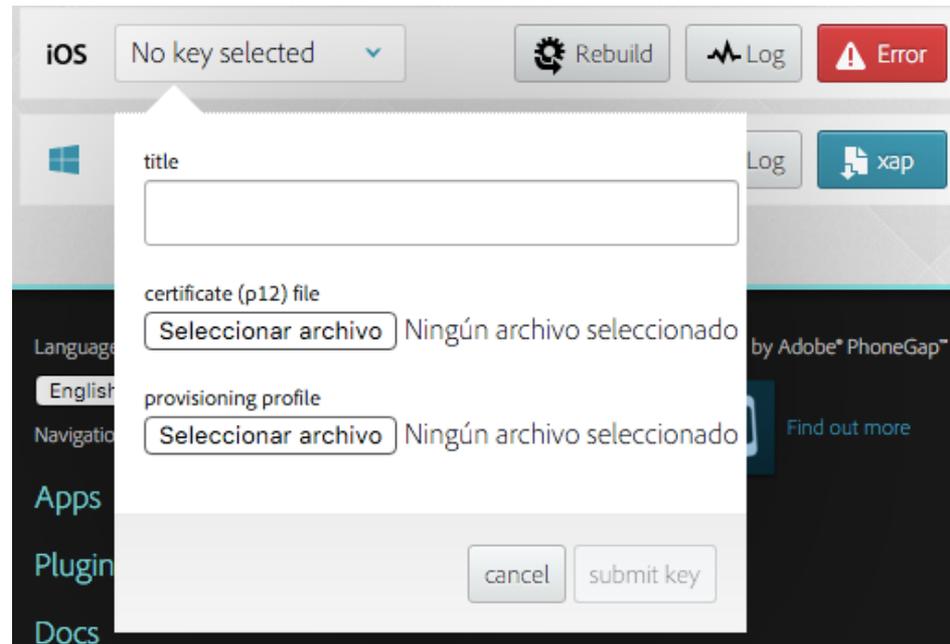
The screenshot shows the PhoneGap Build interface for an application named 'TemporizadorV' (Kitchen Timer). The app is owned by 'gabriel@ditupm.es' and is currently in the 'iOS' build configuration. The interface includes a 'Builds' tab, a 'Plugins' section, and a 'Collaborators' section. The 'Builds' section shows a table with columns for App ID, Version, Owned by, PhoneGap (iOS / Android / Windows), and Source. The 'iOS' build configuration is selected, and the status is 'No key selected'. There is a red 'Error' button next to the 'Rebuild' button.

App ID	Version	Owned by	PhoneGap (iOS / Android / Windows)	Source
1882234	1.0.0	gabriel@ditupm.es	cli-5.2.0 (3.9.1 / 4.1.1 / 3.8.1)	zip package

- Para iOS no tenemos certificado
 - ‘No key selected’
 - se muestra un mensaje de error

Ficheros requeridos

- Al pulsar en “no key selected” debemos proporcionar:
 - un titulo o identificador
 - podemos tener varias claves
 - Un certificado de desarrollador de Apple
 - en formato p12
 - Un perfil de suministro
 - provisioning profile



Pasos

1. ID de Apple
2. Licencia de Desarrollo
3. Certificado de Desarrollador
4. Registrar el Dispositivo Apple
5. Perfil de Suministro

1.- ID de Apple

- Se necesita ID de Apple
 - para los que ya son usuarios de Apple, puede usarse el mismo ID de iCloud o de AppStore
 - se puede obtener en <https://appleid.apple.com/account>

2.- Licencia de Desarrollo

- Para obtener una Licencia de Desarrollo
 - unirse al Apple Developer Program
<https://developer.apple.com/programs/>
- Elegir entre desarrollador independiente o empresa de desarrollo
- Cuesta 99\$ US / año

Certificado de Desarrollador Apple

- Apple proporciona dos tipos de certificado
 - desarrollador y distribuidor
 - necesitamos el primero
- Las aplicaciones han de estar firmadas por un certificado válido para poder:
 - ejecutar en dispositivo Apple
 - publicar en AppStore

3.- Obtención del Certificado

- Accedemos al IOS Provisioning Portal
 - <https://developer.apple.com>
 - Pulsamos en ‘Member Center’, introducimos el ID de Apple
 - Pulsamos en ‘Certificates, Identifiers & Profiles’



SDKs

Download the SDKs and the latest beta software.



Forums

Find answers and discuss with other developers and Apple engineers.



Certificates, Identifiers & Profiles

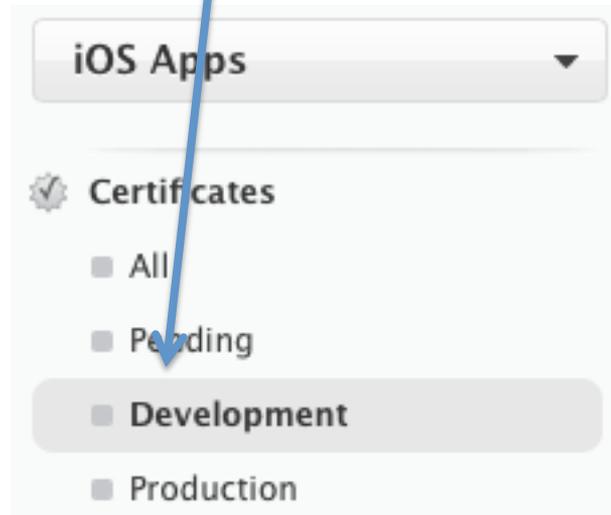
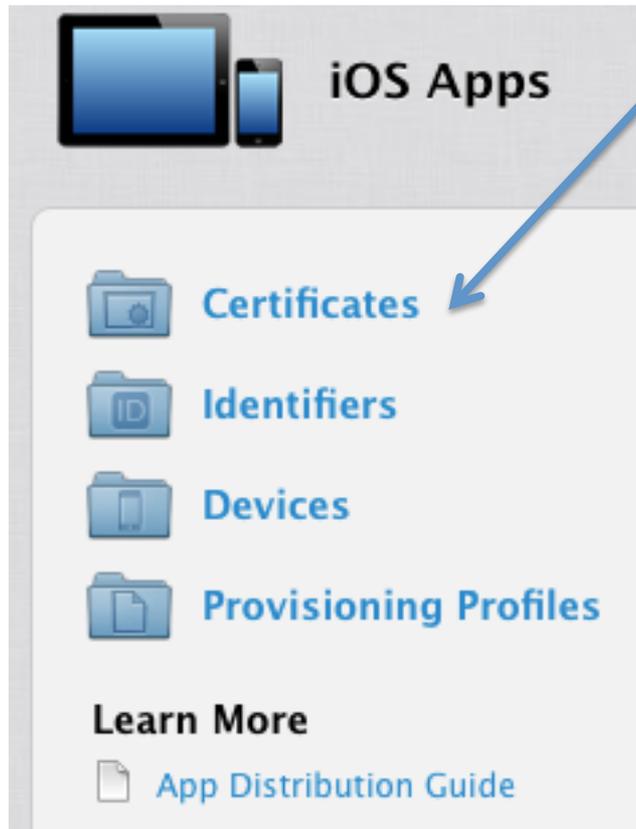
Manage your certificates, identifiers, devices, and profiles for your apps.



Bug Reporting

Submit bugs or request enhancements to APIs and developer tools.

- Después en 'Certificates'
- Y desplegamos 'Certificates' 'Development'
- Y en la esq sup. derecha, pinchamos el botón Añadir (+)



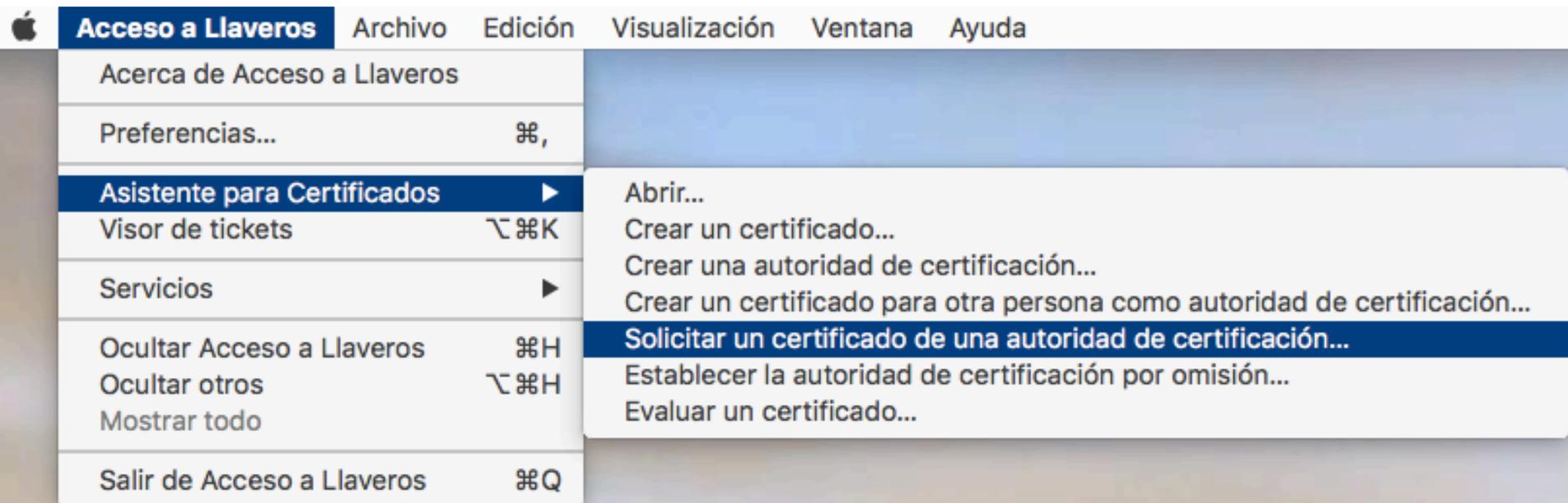
- Seleccionamos iOS App Development y 'Continue'
- Se nos solicita un archivo CSR (Certificate Signing Request) o Solicitud de Firma de Certificado
 - vienen las instrucciones de cómo hacerlo

Archivo CSR

- Hay muchas herramientas de gestión de certificados,
 - algunas incluidas en el S.O., otras se pueden instalar
- Mac OS X, desde el “Acceso a Llaverio”
- Windows, Ubuntu: herramientas externas
 - openssl, gestores de Certificados de IE, Firefox,...

Generar CSR en Mac OS X

- Abrimos “Acceso a Llaveros”
 - En ‘Aplicaciones’ -> ‘Utilidades’
- En el menú principal “Acceso a Llaveros” seleccionamos ‘Asistente para Certificados’
- Seleccionamos ‘Solicitar un certificado de una autoridad de certificación...’



Rellene el formulario

- Introduzca su dirección de correo
- Su nombre
- Seleccione 'guardar en disco'
 - y pulse 'Continuar'
- Obtenemos el fichero 'CSR'

Certificate Information

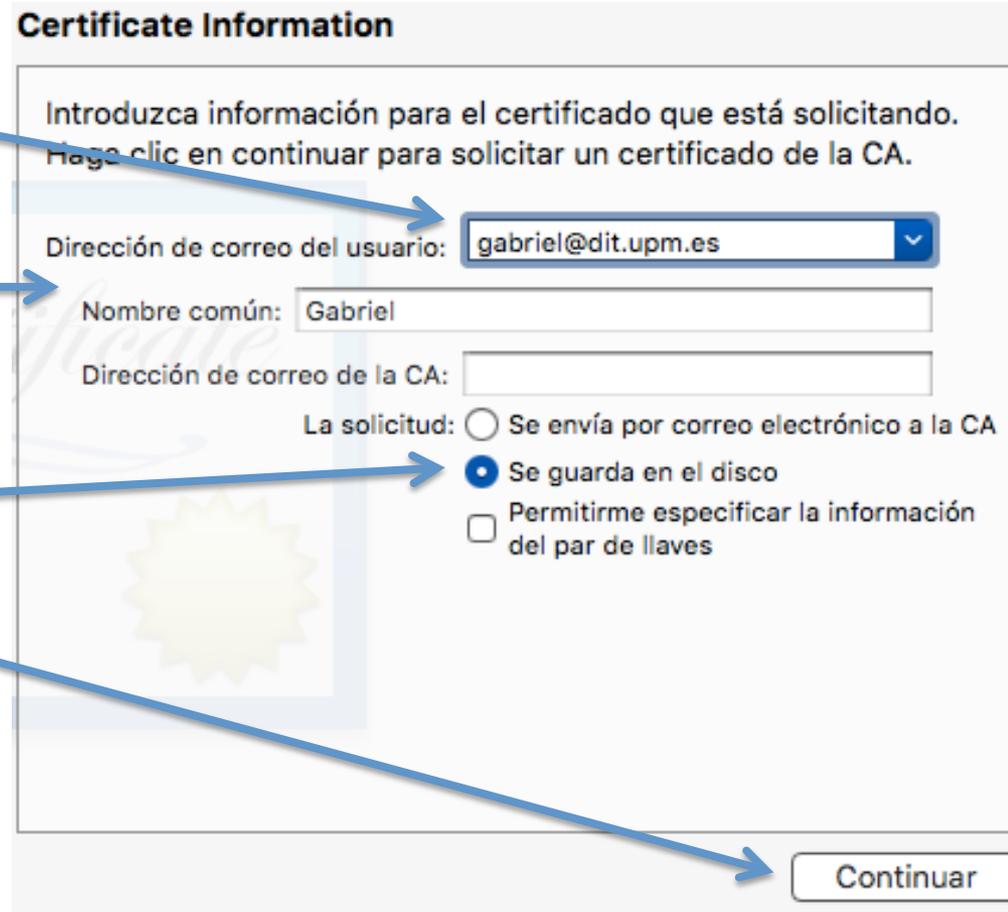
Introduzca información para el certificado que está solicitando.
Haga clic en continuar para solicitar un certificado de la CA.

Dirección de correo del usuario:

Nombre común:

Dirección de correo de la CA:

La solicitud: Se envía por correo electrónico a la CA
 Se guarda en el disco
 Permitirme especificar la información del par de llaves



Solicitar el certificado

- Volvemos a la web de desarrolladores de Apple y subimos el fichero 'CSR'
- Pulsamos el botón 'Generate'
- Nuestro certificado está listo para instalar
 - pulsamos en 'Download', salvamos el fichero .cer y hacemos doble click para instalarlo en 'Acceso a Llaveros'
 - conviene hacer una copia de seguridad y mantenerlo a buen recaudo
 - pulsamos 'done' para proseguir

Certificado listo

- En 'Certificates' -> 'Development' podemos ver nuestro certificado y la fecha de expiración
 - si pinchamos, aparece información adicional



The screenshot shows a certificate card with a blue header and a light gray body. The header contains the name 'Gabriel Huecas', the type 'iOS Development', and the expiration date 'Jan 27, 2017'. The body features a 'Certificate' icon, the name 'Gabriel Huecas', the type 'iOS Development', and the expiration date 'Jan 27, 2017'. At the bottom, there are two buttons: 'Revoke' and 'Download'.

Name	Type	Expires
Gabriel Huecas	iOS Development	Jan 27, 2017

Buttons: Revoke, Download

Formato p12

- El certificado está listo, pero PhoneGap Build exige formato p12
- En 'Acceso a Llaveros' -> 'Certificados'
 - Menú contextual sobre 'iPhone Developer...'
 - Seccionamos la opción 'Exportar ...'
 - Nos pedirá una clave, hay que recordarla para poder subirla a PhoneGap Build



4.- Registrar el dispositivo

- En Member Center:
 1. En 'Certificates, Identifiers & Profiles', seleccionamos 'Devices'
 2. Seleccionamos 'All'
 3. Pinchamos el botón 'Añadir' (+) en la esquina superior derecha
 4. Seleccionamos 'Register Device'
 5. Introducimos el nombre e identificador (UDID) de nuestro dispositivo
 6. Pinchamos 'Continue' y luego 'Register'

Identificador de Dispositivo

- Hay que introducir el UDID
 - no confundir con Número de Serie
- Conectamos el dispositivo al ordenador
- Desde iTunes:
 - Suele mostrar el número de serie, pinchamos hasta que ponga UDID
- Desde Xcode (si disponemos de un Mac):
 - Menu “Window” -> “Devices”

5.- Perfil de Suministro

- Ya podemos obtener el 'Provisioning Profile'
- En la sección 'Provisioning Profile' pinchamos en Añadir (+), rellenamos los datos y bajamos el fichero generado
 - extensión: .mobileprovisioning

Descargar el Perfil de Suministro

- Desplegamos el perfil y pulsamos en 'Download'
- Ya tenemos el segundo fichero



Configuración PhoneGap Build

- Rellenamos los campos
 - titulo
 - “GH1” por ejemplo
 - certificado p12
 - “Certificados.p12”
 - Debemos recordar la clave asignada al fichero
 - perfil de suministros
 - “.mobileprovision”

ios No key selected

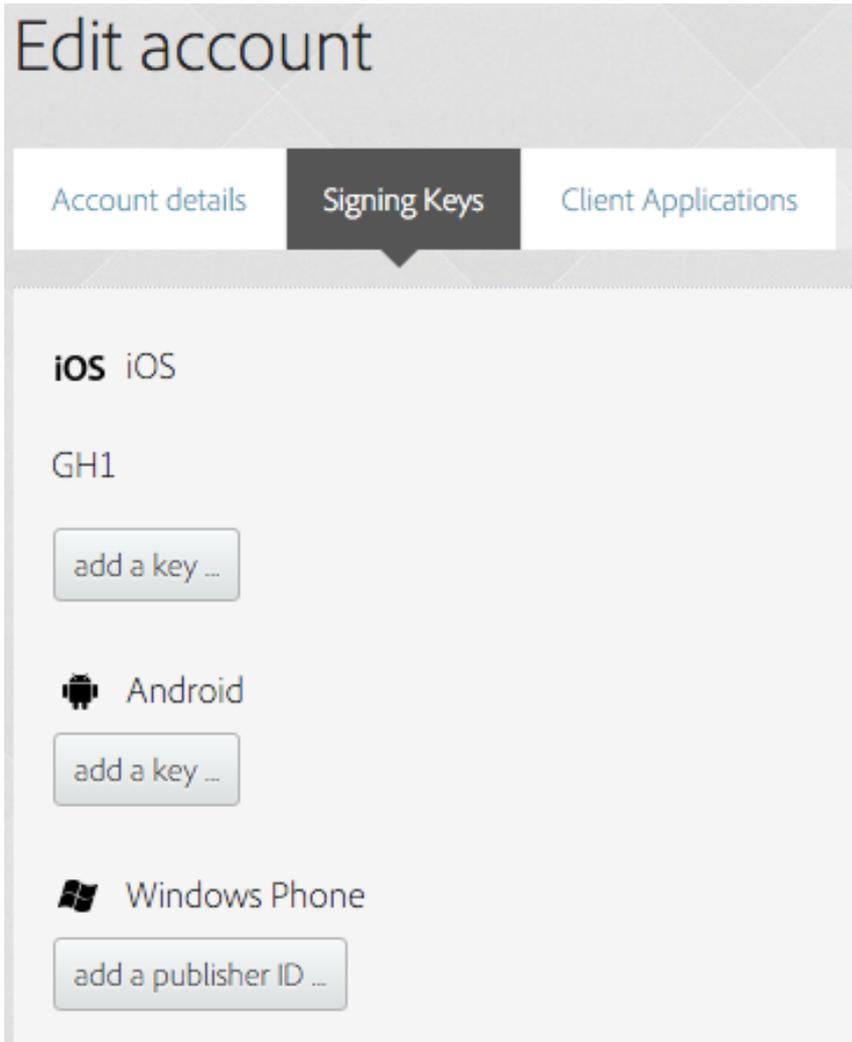
title
GH1

certificate (p12) file
Seleccionar archivo Certificados.p12

provisioning profile
Seleccionar archivo Docencia_UPM...leprovision

cancel submit key

Otras claves



- En PhoneGap Build -> 'Edit Account' podemos acceder a nuestras claves firmadas

Recompilamos todo

The screenshot shows the PhoneGap Build dashboard for an application named "TemporizadorV" (Kitchen Timer). The app icon is a white kitchen timer. The subtitle is "Segundero de Cocina". A QR code is visible in the top right. The interface includes buttons for "Install", "Update code", and "Rebuild all". A navigation bar contains "Builds", "Plugins", "Collaborators", and "Settings". Below this is a table with app details:

App ID	Version	Owned by	PhoneGap (iOS / Android / Windows)	Source	Last built (7)
1882234	1.0.0	gabriel@dit.upm.es	cli-5.2.0 (3.9.1 / 4.1.1 / 3.8.1)	zip package	1 minute

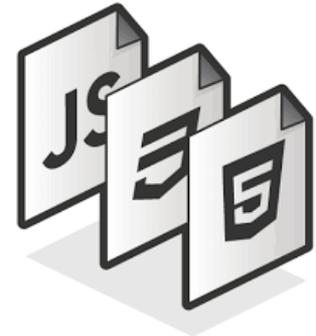
Below the table, there are three rows representing different platforms:

- iOS:** Platform selected as "GH1". Actions: Rebuild, Log, Download ipa.
- Android:** Platform selected as "No key selected". Actions: Rebuild, Log, Download apk.
- Windows:** Platform selected as "No publisher ID selec...". Actions: Rebuild, Log, Download xap.

A vertical "Feedback" button is located on the left side of the interface.

¡Gracias por su atención!

- G. Huecas



Twitter: @ghuecas

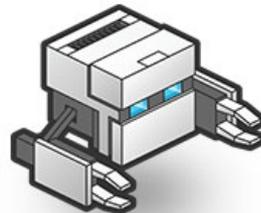
- www.dit.upm.es/~gabriel



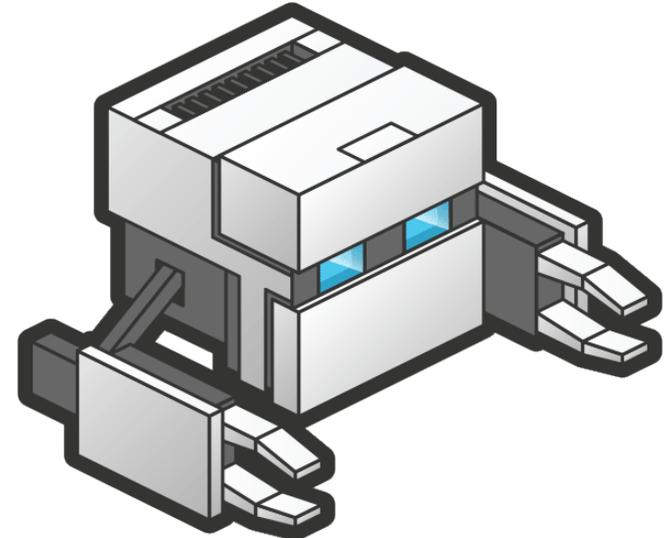
APACHE
CORDOVA™



Adobe®
PhoneGap



Adobe®
PhoneGap Build





Aplicaciones FirefoxOS



FirefoxOS

◆ FirefoxOS es el nuevo SO

- para móviles y tabletas
 - ◆ Desarrollado por la Fundación Mozilla

◆ Sus aplicaciones se programan en

- HTML5, CSS y Javascript

◆ Recursos y tutoriales

- https://marketplace.firefox.com/developers/docs/firefox_os
- https://marketplace.firefox.com/developers/docs/quick_start
- https://developer.mozilla.org/es/docs/Aplicaciones/Comenzando_aplicaciones
- <https://developer.mozilla.org/en-US/Apps/Reference>



Apps FirefoxOS



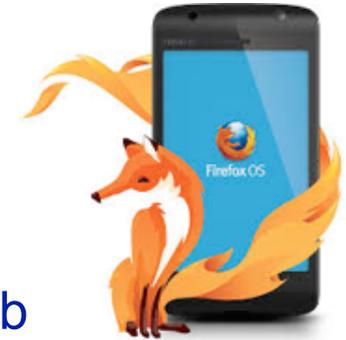
◆ Aplicaciones para instalar en móviles y tabletas

- Funcionando con o sin conexión a Internet
 - ◆ Construidas con tecnologías Web: URLs, HTTP, HTML, CSS, JavaScript, ...
- También pueden instalarse en PCs o navegadores

◆ Tipos de WebApps Firefox OS

- **hosted:** alojadas en un servidor
- **packaged:** se descargan empaquetadas
 - ◆ Hay tres tipos
 - **Plain:** aplicaciones Web convencionales
 - **Certified:** con acceso a algunos elementos restringidos
 - **Privileged:** con acceso a todos los recursos del móvil

Hosted Apps



◆ Webapps publicadas en modo ejecutable en un servidor Web

- Pueden ejecutarse desde un navegador
- Además de ser publicadas en una tienda a través de su URL
 - ◆ para su instalación en un dispositivo FirefoxOS
- Mas info:
 - ◆ https://developer.mozilla.org/en-US/Marketplace/Options/Hosted_apps

◆ Características

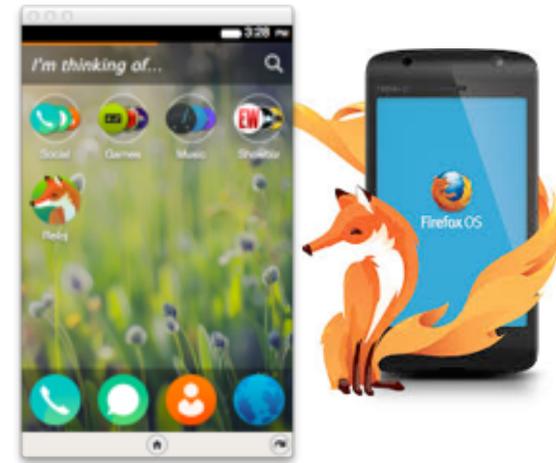
- Necesitan conexión a Internet para ser ejecutadas
- Solo puede haber una hosted WebApp por dominio Web

◆ Se gestionan a través del objeto predefinido **navigator.mozApps**

- con métodos **install(...)** o **checkInstalled(...)**



Packaged WebApps



◆ Webapps empaquetadas en un fichero ZIP

- Pueden ser publicadas en una tienda
 - ◆ para su instalación en dispositivos Firefox OS
- Mas info:
 - ◆ https://developer.mozilla.org/en-US/Marketplace/Options/Packaged_apps

◆ Se pueden publicar en tiendas o en servidores

- Solamente se pueden instalar
 - ◆ No se pueden ejecutar directamente en un navegador
- Pueden ser autonomas
 - ◆ Funcionando sin conexión a Internet

◆ Se instalan a través del objeto predefinido **navigator.mozApps**

- con método **installPackage(....)**

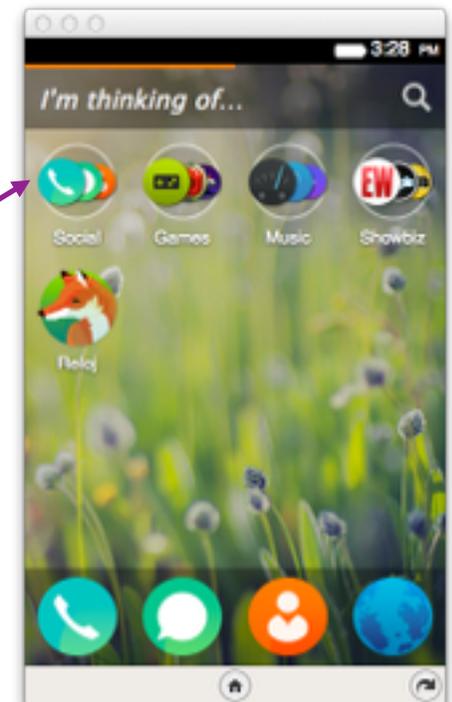
Firefox Marketplace



- ◆ Firefox Marketplace permite crear
 - tiendas de aplicaciones alojadas en servidores
 - ◆ <https://developer.mozilla.org/en-US/Marketplace>

- ◆ Los Marketplace FirefoxOS pueden competir entre si
 - serán más o menos populares
 - ◆ dependiendo de la aceptación de los usuarios
 - serán más o menos seguras
 - ◆ dependiendo del control de la seguridad realizado

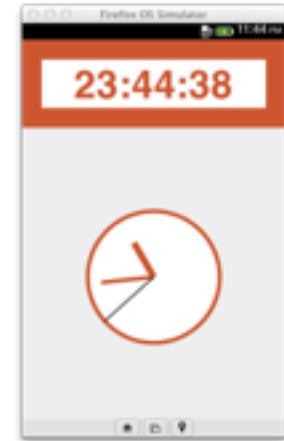
- ◆ Market places con apps de Mozilla
 - <https://marketplace.firefox.com>





Final del tema
Muchas gracias!



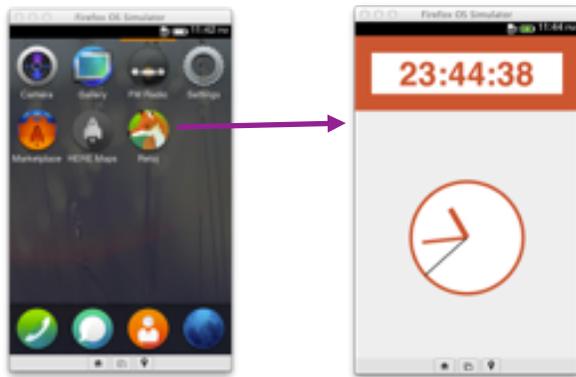


Reloj SVG como App FirefoxOS

Reloj SVG como App

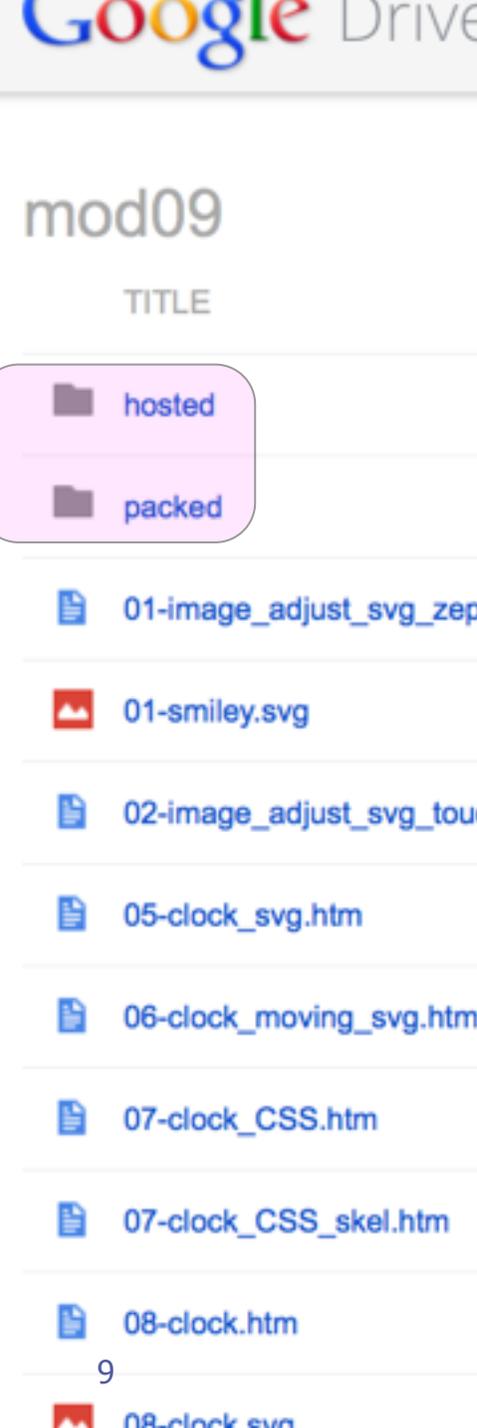
◆ Incluimos el reloj SVG como

- **hosted App**
- **packaged App**



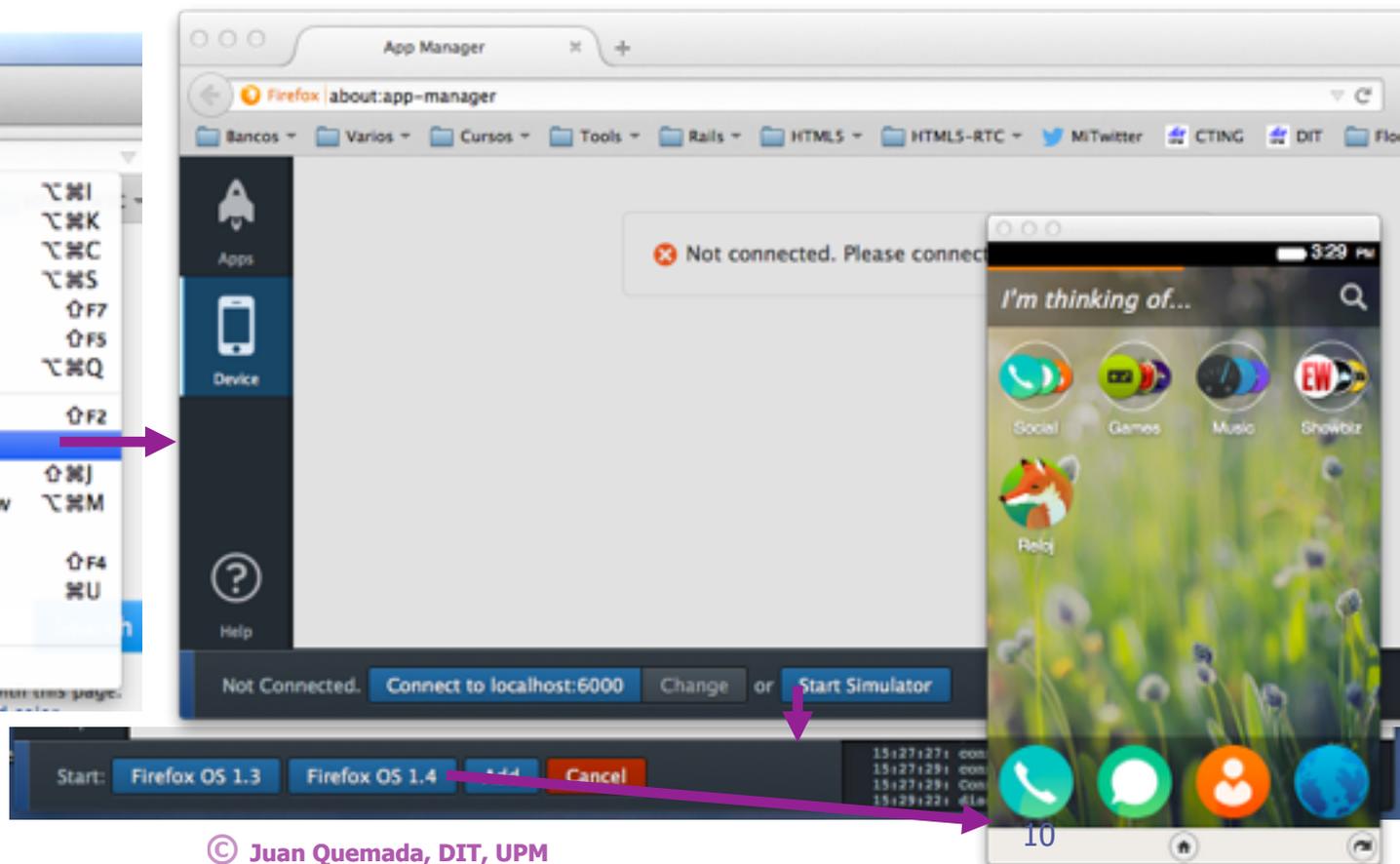
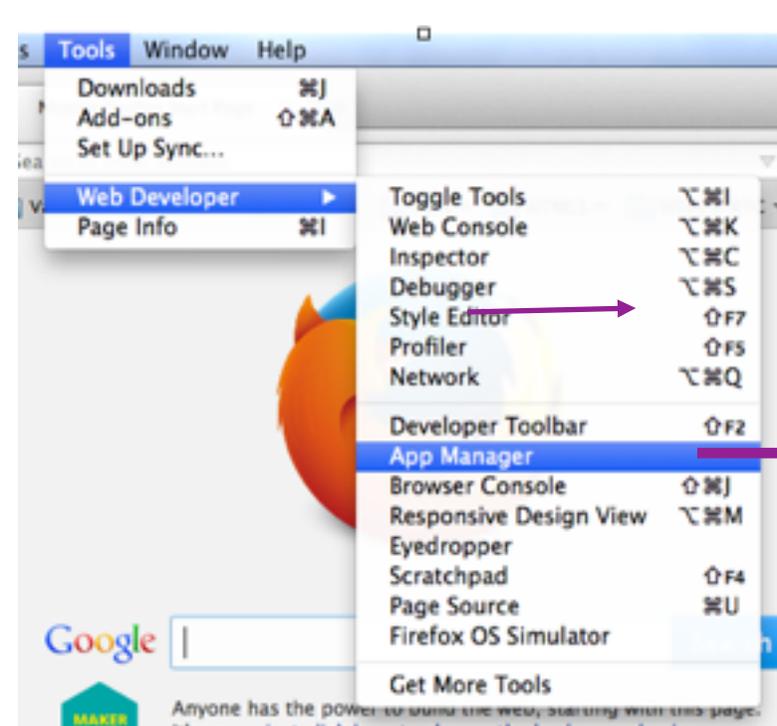
◆ Ambas están publicadas en

- **Google Drive de asignatura**
 - ◆ <https://googledrive.com/host/0B48KCWfVwCIEMjFhUHM4d3FnSTg/>



Arrancar el simulador de FirefoxOS en Firefox

- ◆ Las apps se instalan en el simulador incluido en Firefox
 - Firefox incluye a partir de Firefox 30.0 un simulador de FirefoxOS
 - ◆ Se arranca seleccionando “Tools -> Web Developer -> App Manager”
 - Después se arranca (**Start Simulator**) y se selecciona la versión 1.4 (**Firefox 1.4**)
 - ◆ **OJO!** La app con el Reloj dada está probada en Firefox 1.4 y 1.3 y pueden no funcionar en otras versiones



Instalar hosted App en simulador FirefoxOS

2. Introducir URL del instalador en el navegador y clicar (ejecutar instalador)

<https://googledrive.com/host/oB48KCWfVwCIEMjFhUHM4d3FnSTg/hosted/install.htm>

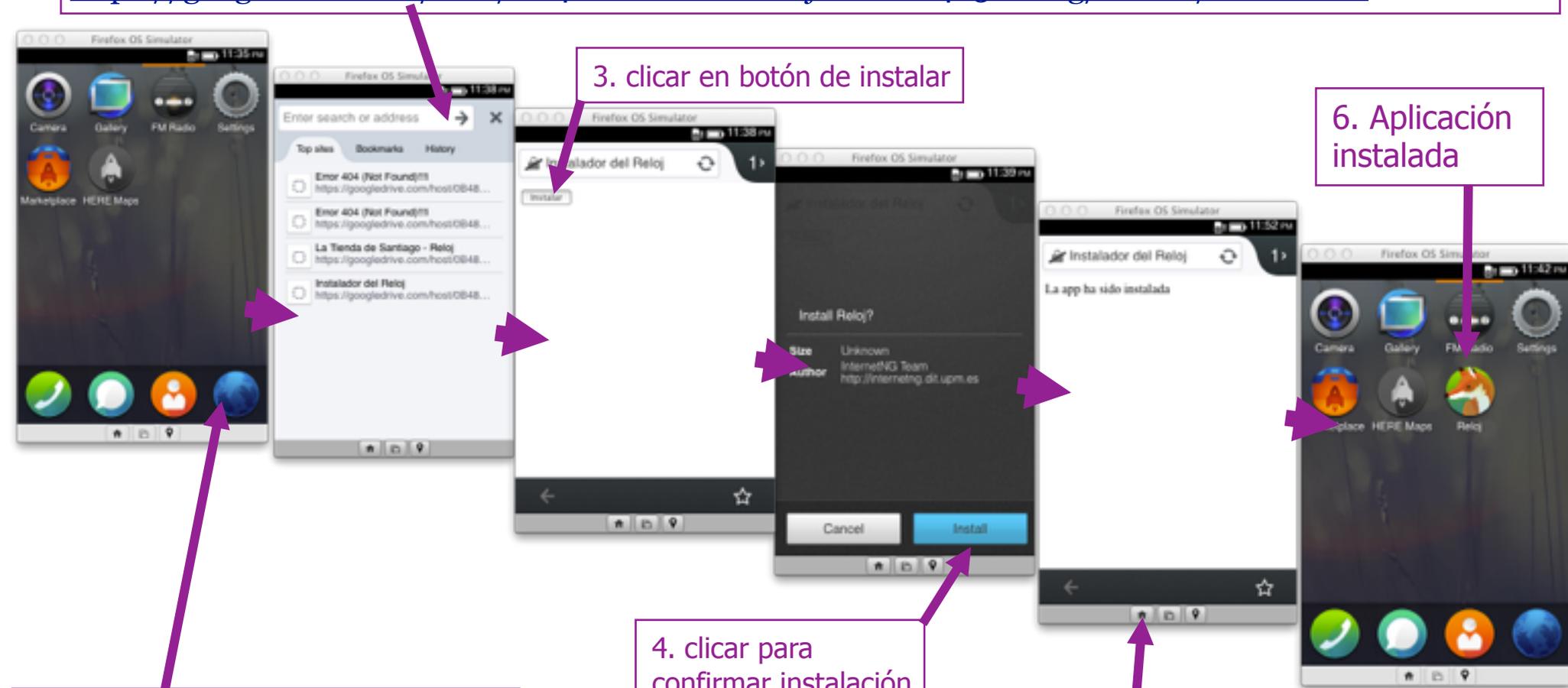
3. clicar en botón de instalar

6. Aplicación instalada

1. clicar para desplegar el navegador y poder instalar la hosted App

4. clicar para confirmar instalación

4. clicar para volver a escritorio



Reloj SVG: hosted App

◆ Para crear una hosted App con Reloj SVG se debe añadir a **clock.htm**:

- **Iconos**

- ◆ que se mostrarán en el escritorio

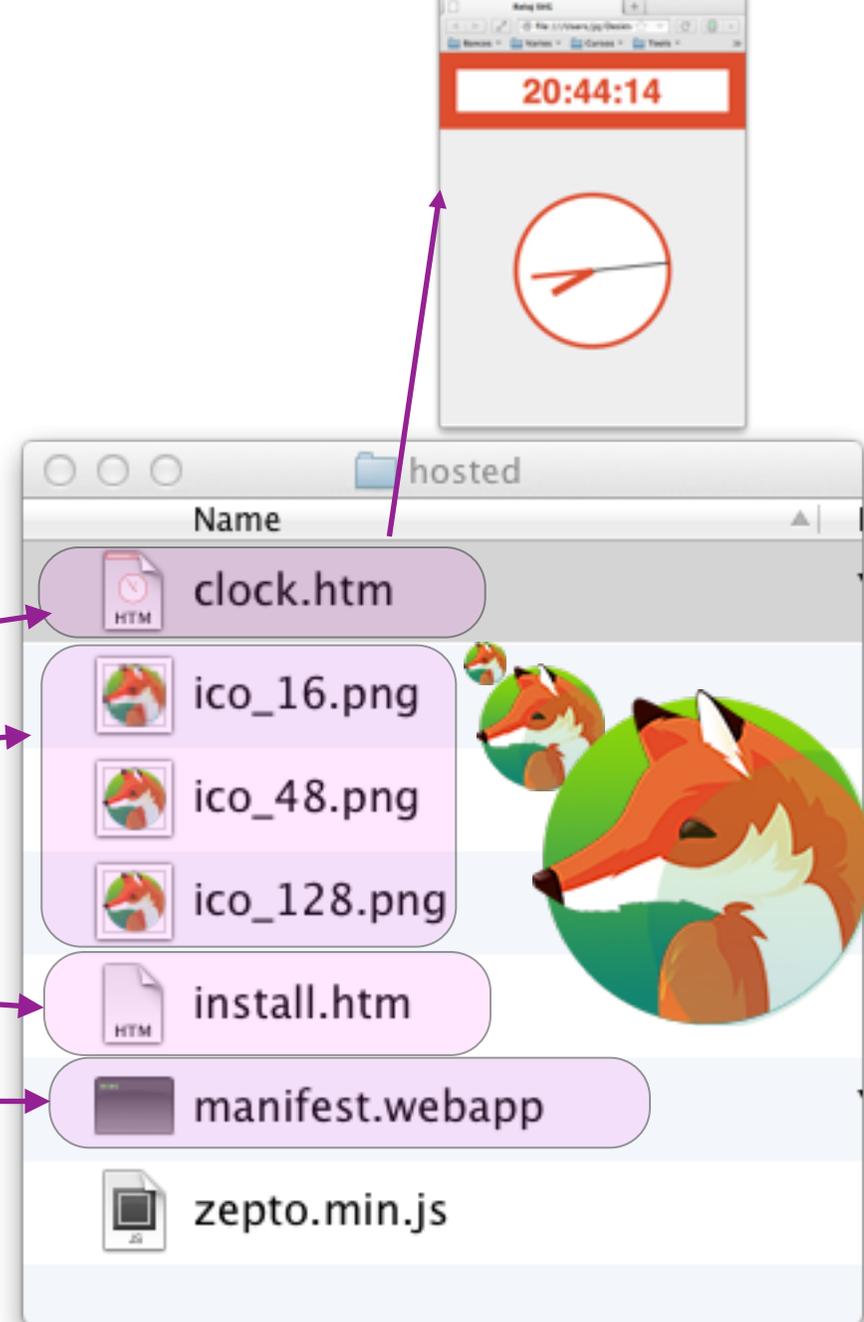
- **Instalador**

- ◆ de la WebApp en el escritorio

- **Manifiesto**

- ◆ con información de instalación

- El ejemplo suministrado incluye además estilos y tipografías



manifest.webapp

- ◆ Fichero en formato JSON (JavaScript Object Notation)
 - Con información de instalación, recursos, diseñador, ...
 - Los URLs son relativos con path absoluto
- ◆ JSON es un formato muy habitual para manifiestos

The image shows a code editor window titled 'manifest.webapp' containing a JSON object. Several fields are highlighted with pink rounded rectangles, and arrows point from these boxes to descriptive labels in white boxes with purple borders. The labels are: 'versión' (pointing to 'version'), 'nombre que aparece en el escritorio' (pointing to 'name'), 'descripción para tienda' (pointing to 'description'), 'URL de WebApp en servidor' (pointing to 'launch_path'), 'iconos escritorio' (pointing to the 'icons' object), and 'equipo/empresa desarrolladora' (pointing to the 'developer' object).

```
{
  "version": "5.0",
  "name": "Reloj",
  "description": "Reloj Analogico y Digital",
  "launch_path": "/host/0B48KCwfVwCIEUDVFY1dwSEYwbHM/hosted/clock.htm",
  "icons": {
    "16": "/host/0B48KCwfVwCIEUDVFY1dwSEYwbHM/hosted/ico_16.png",
    "48": "/host/0B48KCwfVwCIEUDVFY1dwSEYwbHM/hosted/ico_48.png",
    "128": "/host/0B48KCwfVwCIEUDVFY1dwSEYwbHM/hosted/ico_128.png"
  },
  "developer": {
    "name": "InternetNG Team",
    "url": "http://internetng.dit.upm.es"
  }
}
```

© Juan Quemada, DIT, UPM

install.html

```

<!DOCTYPE html>
<html><head><title>Instalador del Reloj</title>
<meta charset="UTF-8">
<script type="text/javascript" src="http://zeptojs.com/zepto.min.js" ></script>
</head>
<body>
  <div id="install">Estado de la instalación del Reloj 5.0</div>
<script>
  $(document).ready(function() {
    var gManifestName = location.href.replace("install.htm","") + 'manifest.webapp';
    var request = navigator.mozApps.checkInstalled(gManifestName);

    request.onsuccess = function() {
      if (request.result) { $("#install").text("La app ya esta instalada"); // App instalada
      else { // App no instalada
        $("#install").html("<input type='button' value='Instalar'>");

        $("#install").click(function() {
          var req = navigator.mozApps.install(gManifestName);
          req.onsuccess = function(data) {
            $("#install").text("La app ha sido instalada").unbind('click');
          }
          req.onerror = function(errObj) {
            alert("Error de instalacion: " + this.error.name);
          }
        });
      }
    };

    request.onerror = function() { alert('Error de instalacion: ' + this.error.message); }
  });
</script>
</body>
</html>

```

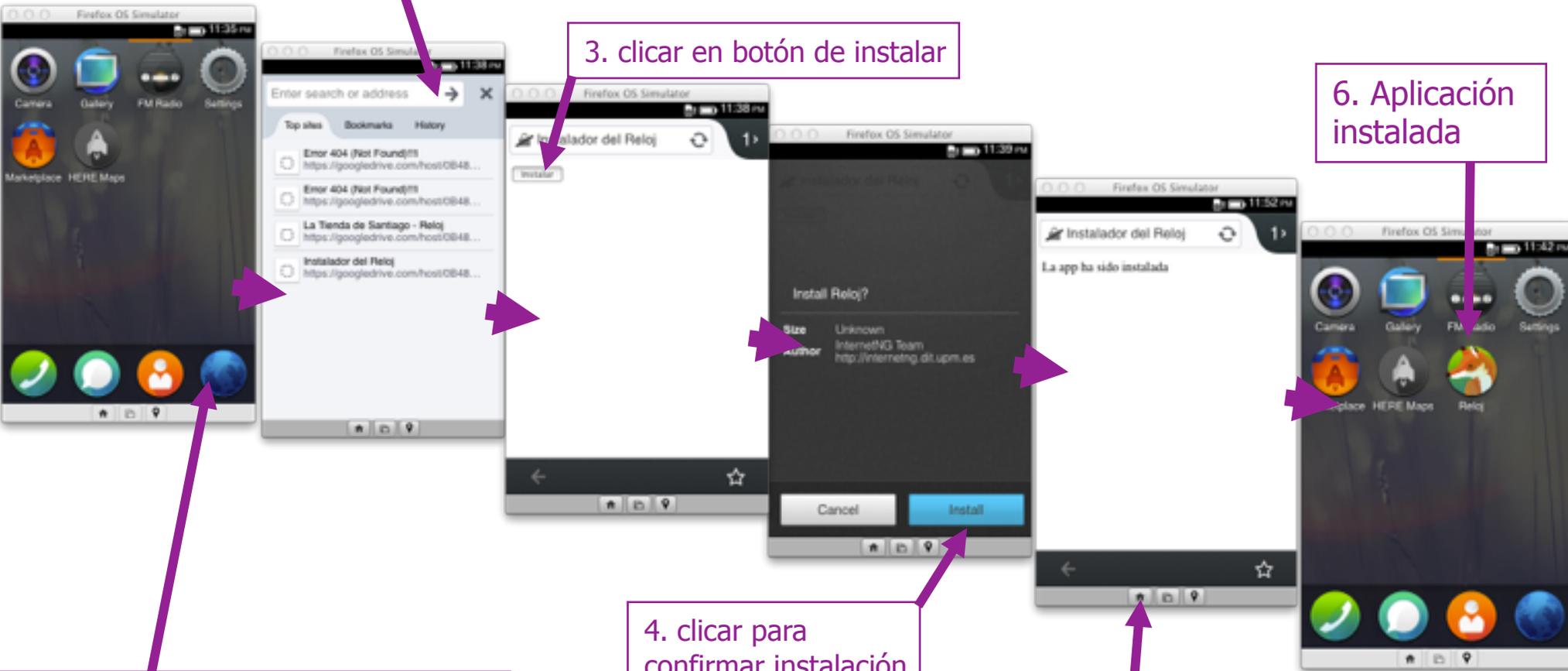
Instalar hosted App en simulador FirefoxOS

2. Introducir URL del instalador en el navegador y clicar (ejecutar instalador)

<https://googledrive.com/host/oB48KCWfVwCIEMjFhUHM4d3FnSTg/hosted/install.htm>

3. clicar en botón de instalar

6. Aplicación instalada



1. clicar para desplegar el navegador y poder instalar la hosted App

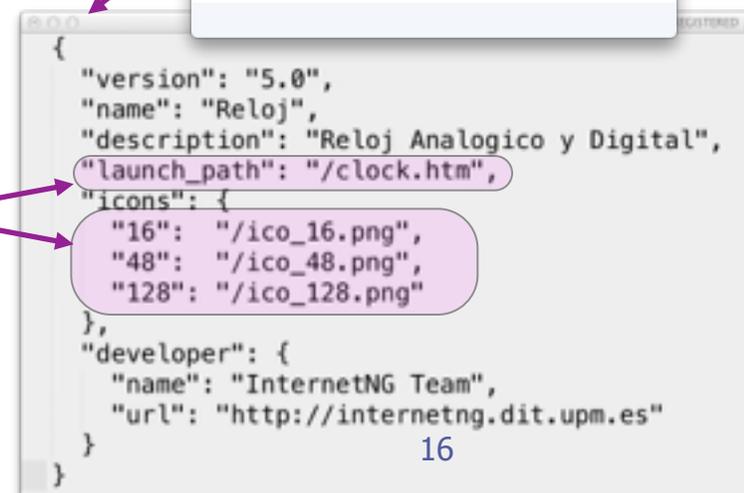
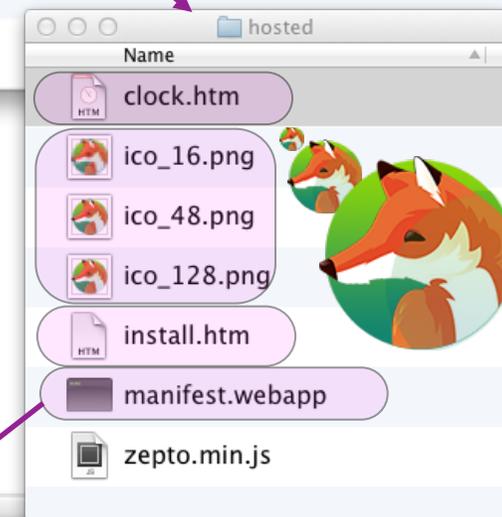
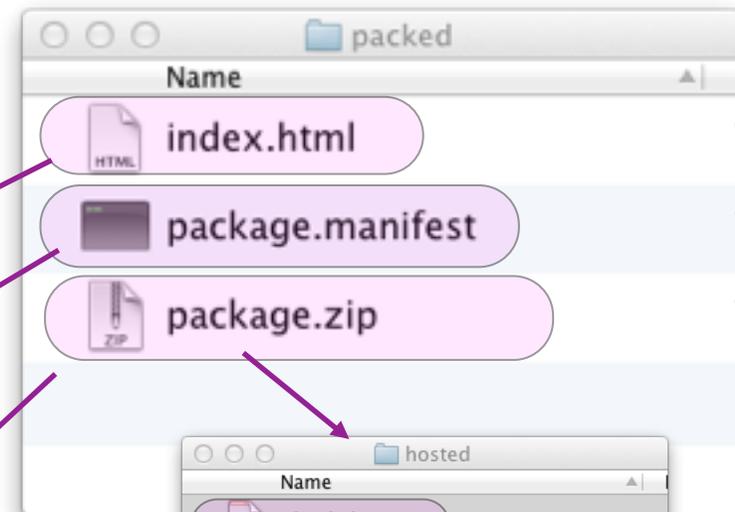
4. clicar para confirmar instalación

4. clicar para volver a escritorio

Reloj SVG: packaged App

◆ Componentes

- **Instalador**
 - ◆ de la WebApp en el escritorio
- **Manifiesto**
 - ◆ con información de instalación del paquete
- **App empaquetada**
 - ◆ **Zip** con todos los ficheros de la App
 - los mismos ficheros de la hosted App
 - Salvo manifiesto: URLs relativos
 - ◆ **OJO**: Los ficheros deben empaquetarse sin más, sin estar en ningún directorio



package.manifest

- ◆ Fichero en formato JSON (JavaScript Object Notation)
 - Con información sobre la webapp empaquetada
- ◆ JSON representa datos como literales de objetos JavaScript
 - Todos los datos están serializados en el fichero como un string

The image shows a code editor window titled "package.manifest" with the following JSON content:

```
{  
  "version": "5.0",  
  "name": "Reloj",  
  "package_path": "https://googledrive.com/host/0B48KCwfVwCIEUDVFY1dwSEYwbHM/packed/package.zip",  
  "developer": {  
    "name": "InternetNG Team",  
    "url": "http://internetng.dit.upm.es"  
  }  
}
```

Annotations with arrows pointing to the corresponding fields in the JSON:

- version (igual al del manifest.webapp) points to "5.0"
- nombre (igual al del manifest.webapp) points to "Reloj"
- URL a paquete en tienda points to the package_path string
- equipo/empresa desarrolladora (igual al del manifest.webapp) points to the developer object

```
<!DOCTYPE html>
<html>
<head>
  <title>La Tienda de Santiago - Reloj</title>
  <meta charset="UTF-8">
</head>
```

```
<body>
  <p>Página de Auto-Instalación del Reloj 5.0</p>
```

```
<script>
  (function() {
    var manifestUrl = location.href.replace("index.html","") +'package.manifest';

    if (! navigator.mozApps.installPackage) {
      alert("ERROR: Esta aplicación no es compatible con tu dispositivo.");
      return;
    }

    var req = navigator.mozApps.installPackage(manifestUrl);
    req.onsuccess = function() {
      alert("Instalacion completada."+this.result.origin);
    };
    req.onerror = function() {
      alert("Error de instalación: "+this.error.name);
    };
  }
)();
</script>
```

```
</body>
</html>
```

Instalar packed App en simulador FirefoxOS

2. Introducir URL del instalador en el navegador y clicar (ejecutar instalador)
<https://googledrive.com/host/0B48KCWfVwCIEMjFhUHM4d3FnSTg/packed/>

3. clicar en botón de instalar

6. Aplicación instalada

1. clicar para desplegar el navegador y poder instalar la hosted App

4. clicar para confirmar instalación

4. clicar para volver a escritorio





Final del tema
Muchas gracias!



Ejercicio

- ◆ Empaquetar como App FirefoxOS, tanto hosted, como packaged, el reloj animado basado en SVG o en CANVAS (solo uno, el que se prefiera) propuestos como ejercicios P2P opcionales en los temas SVG y CANVAS.
- ◆ Publicar dichas apps en Google Drive para que se puedan instalar por terceros.