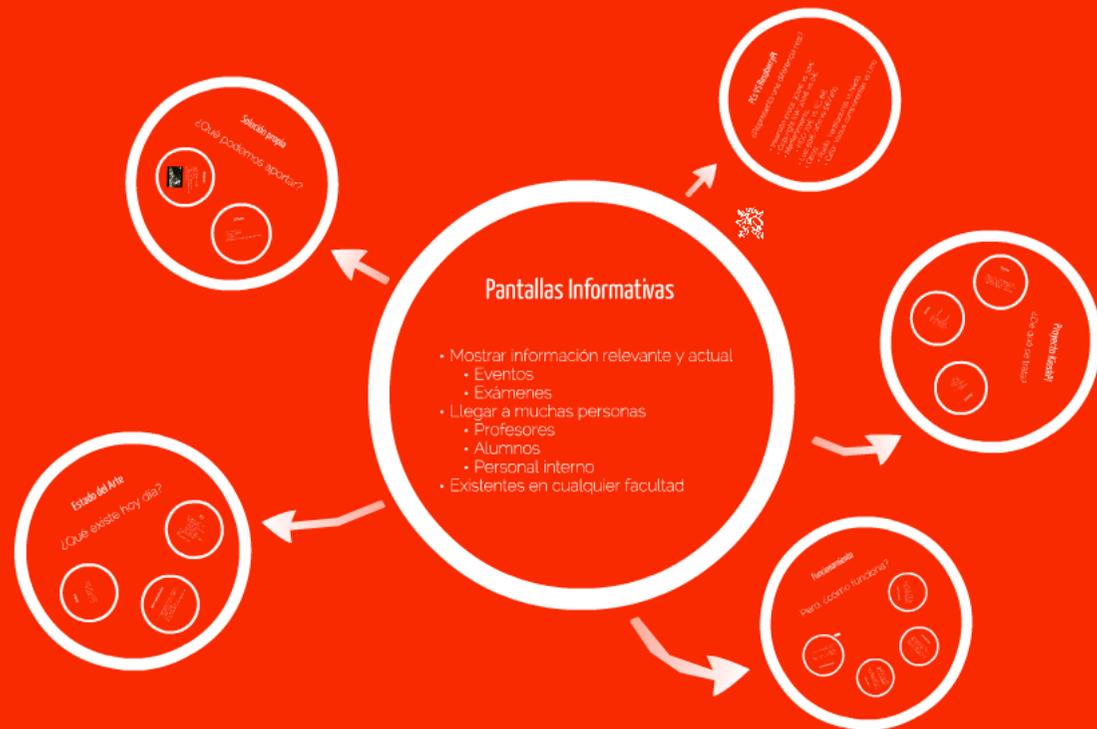


**[KioskPI]**

# KioskPI: Solución centralizada para paneles de información



**Fernando García Díaz-Calvo**

# Pantallas Informativas

- Mostrar información relevante y actual
  - Eventos
  - Exámenes
- Llegar a muchas personas
  - Profesores
  - Alumnos
  - Personal interno
- Existentes en cualquier facultad

# Estado del Arte

## ¿Qué existe hoy día?

### Kioskos

- PCs "disfranzados"
- Menos desventajas
- Aun mayores costes



### PCs

- Inversión inicial alta
- Mantenimiento
- SPCO actualizado
- Nuevo SW, HW
- Resguardo HW, original
- Copyright, SW, original
- Puro constante
- Poca fiabilidad

### Soluciones privadas

- Diseños específicos y cerrados
- Compatibilidad nula o muy poca
- Muy alto coste
- Publicidad intrusiva
- Poco aprovechamiento del terminal



# PC's

- Inversión inicial alta
- Mantenimiento
  - SSOO actualizado
  - Nuevo SW
  - Requisito HW
  - Copyright SW original
- Ruido constante
- Luz
- Poca fiabilidad

# Kioskos

- PC's 'disfrazados'
- Mismas desventajas
- Aún mayores costes



## Soluciones privadas

- Diseños específicos y cerrados
- Compatibilidad nula o muy poca
- Muy alto coste
- Publicidad incrustada
- Poco aprovechamiento del terminal

# Solución propia

## ¿Qué podemos aportar?



**Hardware**

- Placa Raspberry Pi
- Bajo coste
- Poco mantenimiento
- Proyectos que necesitan pocos recursos



**Software**

- SW libre, Linux
- Código abierto
- Gratuito
- Extensa comunidad de usuarios

ula



# Hardware

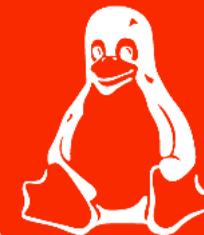
- Placa RaspberryPI
- Bajo coste
- Poco mantenimiento
- Proyectos que necesiten pocos recursos



BBC



YouTube



# Software

- SW libre Linux
- Código abierto
- Gratuito
- Extensa comunidad de usuarios



universidad  
de león

## PCs VS RaspberryPI

¿Representa una diferencia real?

- Inversión inicial: 300€ vs 50€
- Copyright SW: 100€ vs 0€
- Mantenimiento:
  - HDD 70€ vs SD 8€
- Luz: 50€/año vs 5€/año
- Otros:
  - Ruido: Ventiladores vs Nada
  - Calor: Varios componentes vs Uno

# Proyecto KioskPI

¿De qué se trata?

## Objetivo

- Crear un sistema de gestión de pantallas informativas centralizado
- Delegar la reproducción de presentaciones en RaspberryPI

UCLM



## Cliente(s)

- Hardware
- RaspberryPI
- Software
- Raspbian
- Omniclayer
- Scripts



## Servidor

- Hardware Virtual
- Máquina
- PHP
- MySQL
- Synchrony
- NFS
- Fundación



# Objetivo

- Crear un sistema de gestión de pantallas informativas centralizado
- Delegar la reproducción de presentaciones en RaspberryPI





# Servidor

- Hardware
  - Máquina Virtual
- Software
  - PHP
  - MySQL
  - FoundationZurb
  - Symfony
  - NFS

# Cliente(s)

- Hardware
  - RaspberryPI
- Software
  - RaspBian
  - Omxplayer
  - Scripts



# Funcionamiento

## Pero, ¿cómo funciona?



### Crear presentación

El usuario crea una presentación normalmente en HTML y cuando ha hecho el vídeo.  
Como paso adicional la guarda en formato vídeo.

### Archivar vídeo

El usuario accede a uno de sus ficheros y selecciona el vídeo que desea archivar.  
La Universidad Archiva siempre basándose en el fichero para proporcionar el enlace para acceder a los vídeos y con ellos la reproducción.

### Administración

Una vez se han creado los ficheros de vídeo se genera una lista de los vídeos que se han creado para poder administrar los vídeos que se han creado en la plataforma de vídeo.

### Dar de alta vídeo

El usuario accede al vídeo y genera un vídeo que se genera en la plataforma de vídeo.  
El usuario genera un vídeo que se genera en la plataforma de vídeo.

## Crear presentación

El usuario crea una presentación normalmente, tal y como lo hacía hasta ahora.

Como paso adicional la guarda en formato vídeo.



## Dar de alta vídeo

El usuario accede al portal habilitado para ello y da de alta el nuevo vídeo, seleccionando lo(s) kioskos en los que quiera reproducirlo.



## Activar vídeo

El usuario accede a uno de sus monitores y selecciona qué vídeo pasará a reproducir desde ese momento.

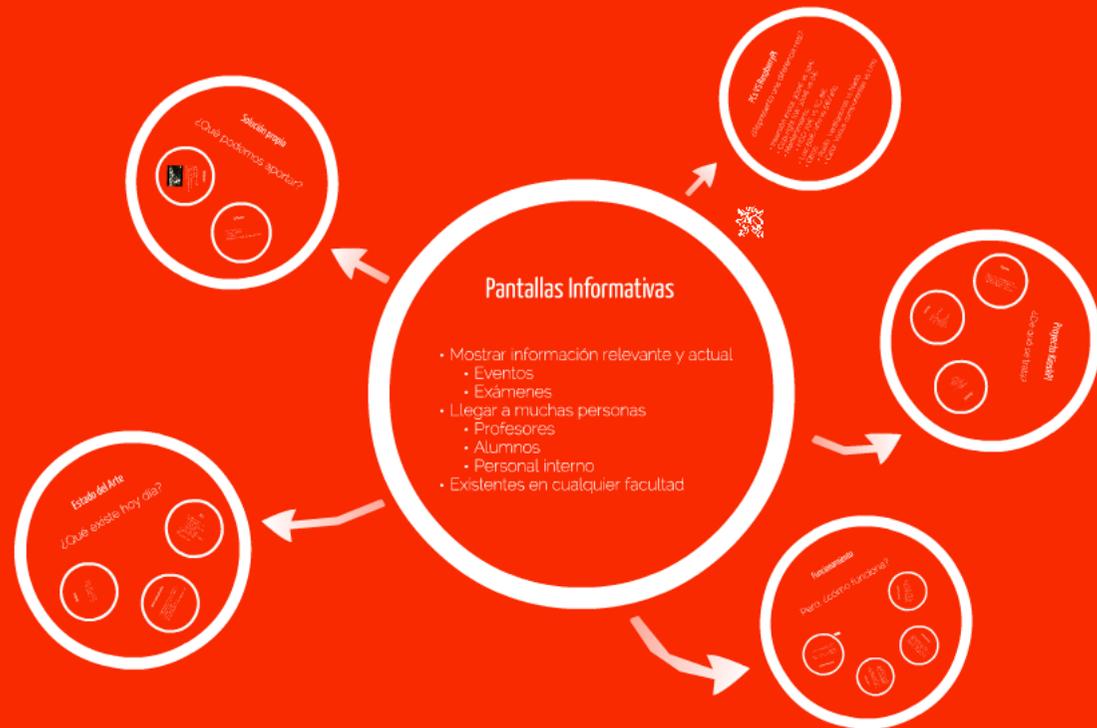
La RaspberryPI está siempre buscando nuevos vídeos para reproducir. Detecta el cambio y comienza la reproducción.



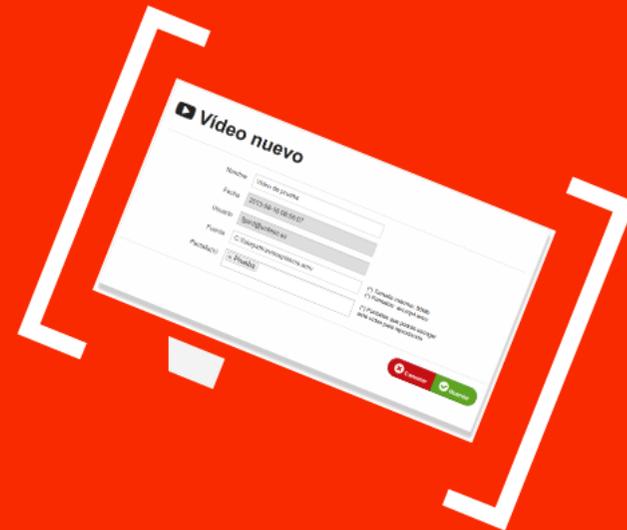
# Administración

Los administradores pueden comprobar qué usuarios suben los vídeos, qué monitores son accesibles para qué usuario(s), logs de la aplicación, dar de alta nuevos kioskos, etc, todo a través del portal.

# KioskPI: Solución centralizada para paneles de información



**Fernando García Díaz-Calvo**





avisosplasma.ppt [Modo de compatibilidad] - PowerPoint ? - □ ×

Iniciar sesión

# Exportar

- Información
- Nuevo
- Abrir
- Guardar
- Guardar como
- Guardar como Adobe PDF
- Imprimir
- Compartir
- Exportar**
- Cerrar
- Cuenta
- Opciones

- Crear Adobe PDF
- Crear documento PDF/XPS
- Crear un vídeo**
- Empaquetar presentación para CD
- Crear documentos
- Cambiar el tipo de archivo

intervalos, las narraciones y los gestos del puntero láser grabados

- Preserva las animaciones, las transiciones y los medios

Obtenga ayuda para grabar el vídeo de su presentación con diapositivas en un DVD o para cargarla en Internet

Pantallas de...  
Para ver en u...

Usar narracio...  
Las diapositiv...

Segundos de duración de cada diapositiva: 05,00

**Crear vídeo**

## Vídeo nuevo

Nombre

Fecha

Usuario

Fuente

Pantalla(s)

(\* Tamaño máximo: 50Mb  
(\* Formatos: avi,mp4,wmv

(\* Pantallas que podrán escoger  
este vídeo para reproducirlo

 Cancelar

 Guardar



## Pantallas

Usted tiene acceso a cambiar las presentaciones de las pantallas de este apartado, pudiendo seleccionar qué vídeo reproducir en cada una de ellas de los añadidos anteriormente en el menú Vídeos.

Cuando cambie el vídeo de una pantalla, la visualización no comenzará hasta que el vídeo anterior haya concluido.

Para más información, consultar el **Manual de la aplicación**.

10

#	Nombre	Edificio	Vídeo
1	Kiosko conferencias	Biblioteca de San Isidoro	programacion (17/09/13 09:14)

#

Mostrando pantallas del 1 al 1 de un total de 1 pantallas

- Ninguna
- programacion (17/09/13 09:14)
- prueba (16/09/13 14:06)



universidad  
de león

```
OPEN FILES
  • untitled
FOLDERS
  Kioskpi
    app
    components
    src
    vendor
    web
  .gitignore
  FFmpegx264.md
  LICENSE
  README.md
  Resumen.md
  composer.json
  script.sh
untitled
1 #!/bin/bash
2
3 #echo "[$(date +%m/%d/%y - %H:%M:%S)] - Initializing script..." >> /home/pi/log.txt
4 while true; do
5   #echo "[$(date +%m/%d/%y - %H:%M:%S)] - Getting IP..." >> /home/pi/log.txt
6   SLEEP=10
7   NIGHT=22
8   IP=$(hostname -I | tr -d ' ' | awk '{print $1}')
9   #echo "[$(date +%m/%d/%y - %H:%M:%S)] - My IP is $IP..." >> /home/pi/log.txt
10  if [ $(mount | grep -c "193.146.97.11") -gt 0 ]; then
11    #echo "[$(date +%m/%d/%y - %H:%M:%S)] - NFS is mounted, checking config..." >> /home/pi/log.txt
12    if [ -f /home/pi/videos/SIP/config.txt ]; then
13      SLEEP=$(cat /home/pi/videos/SIP/config.txt | grep "sleep" | awk -F':' '{print $2}')
14      #echo "[$(date +%m/%d/%y - %H:%M:%S)] - The new sleep time is $SLEEP..." >> /home/pi/log.txt
15      NIGHT=$(cat /home/pi/videos/SIP/config.txt | grep "night" | awk -F':' '{print $2}')
16      #echo "[$(date +%m/%d/%y - %H:%M:%S)] - The new night limit is $NIGHT..." >> /home/pi/log.txt
17    fi
18    if [ -f /home/pi/video.mp4 ]; then
19      #echo "[$(date +%m/%d/%y - %H:%M:%S)] - /home/pi/video.mp4 exists..." >> /home/pi/log.txt
20      if [ -f /home/pi/videos/SIP/video.mp4 ]; then
21        #echo "[$(date +%m/%d/%y - %H:%M:%S)] - /home/pi/videos/SIP/video.mp4 exists too..." >> /home/pi/log.txt
22        CLIENT=$(md5sum /home/pi/video.mp4 | awk '{print $1}')
23        SERVER=$(md5sum /home/pi/videos/SIP/video.mp4 | awk '{print $1}')
24        if [ "$CLIENT" != "$SERVER" ]; then
25          #echo "[$(date +%m/%d/%y - %H:%M:%S)] - They are different, copying..." >> /home/pi/log.txt
26          cp /home/pi/videos/SIP/video.mp4 /home/pi/video.mp4
27        else
28          #echo "[$(date +%m/%d/%y - %H:%M:%S)] - They are the same. Doing nothing..." >> /home/pi/log.txt
29          fi
30        else
31          #echo "[$(date +%m/%d/%y - %H:%M:%S)] - /home/pi/videos/SIP/video.mp4 does not exists, removing /home/pi/video.mp4..." >> /home/pi/log.txt
32          rm -f /home/pi/video.mp4
33          fi
34        else
35          #echo "[$(date +%m/%d/%y - %H:%M:%S)] - /home/pi/video.mp4 does not exists..." >> /home/pi/log.txt
36          if [ -f /home/pi/videos/SIP/video.mp4 ]; then
37            #echo "[$(date +%m/%d/%y - %H:%M:%S)] - /home/pi/videos/SIP/video.mp4 exists, copying..." >> /home/pi/log.txt
38            cp /home/pi/videos/SIP/video.mp4 /home/pi/video.mp4
39          fi
40        fi
41      fi
42      if [ -f /home/pi/video.mp4 ]; then
43        #echo "[$(date +%m/%d/%y - %H:%M:%S)] - Playing /home/pi/video.mp4..." >> /home/pi/log.txt
44        omxplayer /home/pi/video.mp4
45      fi
46      TIME=$(date +%H)
47      #echo "[$(date +%m/%d/%y - %H:%M:%S)] - Checking time..." >> /home/pi/log.txt
48      if test $TIME -ge $NIGHT; then
49        SLEEP=36000
50      fi
51      #echo "[$(date +%m/%d/%y - %H:%M:%S)] - Sleeping $SLEEP..." >> /home/pi/log.txt
52      sleep $SLEEP
53    done
54  done
```



```
OPEN FILES
untitled
•
• untitled
FOLDERS
kioskpi
  app
  components
  src
  vendor
  web
.gitignore
FFMPEGx264.md
LICENSE
README.md
Resumen.md
composer.json
script.sh
1 #!/bin/bash
2
3 #echo "[$(date +%m/%d/%y - %H:%M:%S')] - Initializing script..." >> /home/pi/log.txt
4 while true; do
5   #echo "[$(date +%m/%d/%y - %H:%M:%S')] - Getting IP..." >> /home/pi/log.txt
6   SLEEP=10
7   NIGHT=22
8   IP=$(hostname -I | tr -d ' ' | awk '{print $1}')
9   #echo "[$(date +%m/%d/%y - %H:%M:%S')] - My IP is $IP..." >> /home/pi/log.txt
10  if [ $(mount | grep -c "193.146.97.11") -gt 0 ]; then
11    #echo "[$(date +%m/%d/%y - %H:%M:%S')] - NFS is mounted, checking config..." >> /home/pi/log.txt
12    if [ -f /home/pi/videos/$IP/config.txt ]; then
13      SLEEP=$(cat /home/pi/videos/$IP/config.txt | grep "sleep" | awk -F':' '{print $2}')
14      #echo "[$(date +%m/%d/%y - %H:%M:%S')] - The new sleep time is $SLEEP..." >> /home/pi/log.txt
15      NIGHT=$(cat /home/pi/videos/$IP/config.txt | grep "night" | awk -F':' '{print $2}')
16      #echo "[$(date +%m/%d/%y - %H:%M:%S')] - The new night limit is $NIGHT..." >> /home/pi/log.txt
17    fi
18    if [ -f /home/pi/video.mp4 ]; then
19      #echo "[$(date +%m/%d/%y - %H:%M:%S')] - /home/pi/video.mp4 exists..." >> /home/pi/log.txt
20      if [ -f /home/pi/videos/$IP/video.mp4 ]; then
21        #echo "[$(date +%m/%d/%y - %H:%M:%S')] - /home/pi/videos/$IP/video.mp4 exists too..." >> /home/pi/log.txt
22        CLIENT=$(md5sum /home/pi/video.mp4 | awk '{print $1}')
23        SERVER=$(md5sum /home/pi/videos/$IP/video.mp4 | awk '{print $1}')
24        if [ "$CLIENT" != "$SERVER" ]; then
25          #echo "[$(date +%m/%d/%y - %H:%M:%S')] - They are different, copying..." >> /home/pi/log.txt
26          cp /home/pi/videos/$IP/video.mp4 /home/pi/video.mp4
27        else
28          #echo "[$(date +%m/%d/%y - %H:%M:%S')] - They are the same. Doing nothing..." >> /home/pi/log.txt
29          fi
30        else
31          #echo "[$(date +%m/%d/%y - %H:%M:%S')] - /home/pi/videos/$IP/video.mp4 does not exists, removing /home/pi/video.mp4..." >> /home/pi/log.txt
32          rm -f /home/pi/video.mp4
33          fi
34        else
35          #echo "[$(date +%m/%d/%y - %H:%M:%S')] - /home/pi/video.mp4 does not exists..." >> /home/pi/log.txt
36          if [ -f /home/pi/videos/$IP/video.mp4 ]; then
37            #echo "[$(date +%m/%d/%y - %H:%M:%S')] - /home/pi/videos/$IP/video.mp4 exists, copying..." >> /home/pi/log.txt
38            cp /home/pi/videos/$IP/video.mp4 /home/pi/video.mp4
39          fi
40        fi
41      fi
42      if [ -f /home/pi/video.mp4 ]; then
43        #echo "[$(date +%m/%d/%y - %H:%M:%S')] - Playing /home/pi/video.mp4..." >> /home/pi/log.txt
44        omxplayer /home/pi/video.mp4
45      fi
46      TIME=$(date +%H)
47      #echo "[$(date +%m/%d/%y - %H:%M:%S')] - Checking time..." >> /home/pi/log.txt
48      if test $TIME -ge $NIGHT; then
49        SLEEP=36000
50      fi
51      #echo "[$(date +%m/%d/%y - %H:%M:%S')] - Sleeping $SLEEP..." >> /home/pi/log.txt
52      sleep $SLEEP
53    done
54  fi
55 done
```

**[KioskPI]**