



`sudo` can log both successful and unsuccessful attempts (as well as errors) to `syslog(3)`, a log file, or both. By default `sudo` will log via `syslog(3)` but this is changeable at configure time or via the `sudoreg` file.

## OPTIONS

`sudo` accepts the following command line options:

- A Normally, if `sudo` requires a password, it will read it from the current terminal. If the `-A` (`askpass`) option is specified, a helper program is executed to read the user's password and output the password to the standard output. If the `SUDO_ASKPASS` environment variable is set, it specifies the path to the helper program. Otherwise, the value specified by the `askpass` option in `sudoreg(5)` is used.
- a `type` The `-a` (`authentication type`) option causes `sudo` to use the specified authentication type when validating the user, as allowed by `/etc/login.conf`. The system administrator may specify a list of sudo-specific authentication methods by adding an "auth-sudo" entry in `/etc/login.conf`. This option is only available on systems that support BSD authentication.
- b The `-b` (`background`) option tells `sudo` to run the given command in the background. Note that if you use the `-b` option you cannot use shell job control to manipulate the process.
- C `fd` Normally, `sudo` will close all open files except `stdin`, `stdout`, and `stderr`. Values less than three are not permitted. `fd` specifies the file descriptor to use for the standard error (file descriptor three).
- c `class` The `-c` (`class`) option causes `sudo` to use the specified `class`. It indicates that the command should be run from a shell that is already running as root, or the `sudo` command must be run from a shell that is already running as root. Specifying a `class` of "root" indicates that the command should be run as root, or the `sudo` command must be run from a shell that is already running as root.
- E The `-E` (`preserve environment`) option tells `sudo` to preserve the user's environment. If the user is authorized by `sudoreg` the `EDITOR` environment variable is used.
- e The `-e` (`edit`) option indicates that the user wants to edit a file. The following steps are taken:
  1. Temporary copies are made of the file to be edited.
  2. The editor specified by the `EDITOR` environment variable is used.
  3. If they have been modified, the file is saved.
 If the specified file does not exist, the user will receive a message. If the user is unable to update a file with its edited version, the user will receive a message.

# Tutorial en formato wiki sobre sistemas operativos UNIX con plataforma de prueba

Noelia Sales Montes  
DIRECTOR: Manuel Palomo Duarte  
I.T. Informática de Sistemas  
Febrero 2010

## MANUALS

You can order printed copies of the GNU Emacs reference manual. Your local Emacs maintainer might also have copies. The manual is also included in the Emacs source distribution.

## FILES

`/usr/share/info` - files for the Info documentation browser. The complete text of the Emacs reference manual is included in a convenient tree structured form. Also includes the Emacs Lisp Reference Manual, useful to anyone wishing to write programs in the Emacs Lisp extension language.

`/usr/share/emacs/SVERSION/lisp` - Lisp source files and compiled files that define most editing commands. Some are preloaded; others are autoloaded from this directory when used.

`/usr/lib/emacs/SVERSION/arch` - various programs that are used with GNU Emacs.

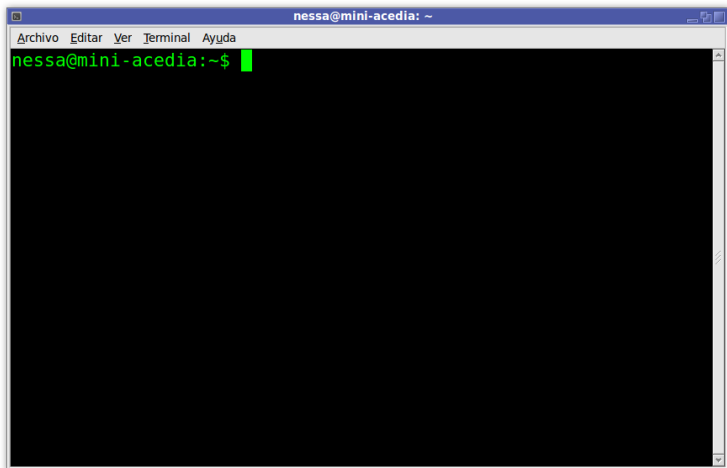
`/usr/share/emacs/SVERSION/etc` - various files of information.

`/usr/share/emacs/SVERSION/etc/DOC.*` - contains the documentation strings for the Lisp primitives and preloaded Lisp functions of GNU Emacs. They are stored here to reduce the size of Emacs proper.

`/usr/share/emacs/SVERSION/etc/SERVICE` lists people offering various services to assist users of GNU Emacs, including education, troubleshooting, porting and customization.

## BUGS

There is a mailing list, `bug-gnu-emacs@gnu.org`, for reporting Emacs bugs and fixes. But before reporting something as a bug, please try to be sure that it really is a bug, not a misunderstanding or a deliberate feature. We ask you to read the section "Reporting Emacs Bugs" near the end of the reference manual (or Info system) for hints on how and when to report bugs. Also, include the version number of the Emacs you are running. Bugs that are actually features can be isolated, so it is in your interest to report them in such a way that they can be easily reproduced.



# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación

## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión



## ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación

## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# OBJETIVOS



Elaborar un sistema informático que ayude en el **aprendizaje** de los sistemas operativos **Unix** a nivel de administración

## REQUISITOS:

- Contenido teórico **estructurado**
- Sistema **ampliable** de forma colaborativa
- Ejercicios y sistema de **control**
- **Disponible** a través de internet



# OBJETIVOS



Elaborar un sistema informático que ayude en el **aprendizaje** de los sistemas operativos **Unix** a nivel de administración

## REQUISITOS:

- 1 Contenido teórico **estructurado**
- 2 Sistema **ampliable** de forma colaborativa
- 3 Ejercicios y sistema de **control**
- 4 **Disponible** a través de internet



# OBJETIVOS



Elaborar un sistema informático que ayude en el **aprendizaje** de los sistemas operativos **Unix** a nivel de administración

## REQUISITOS:

- 1 Contenido teórico **estructurado**
- 2 Sistema **ampliable** de forma colaborativa
- 3 Ejercicios y sistema de **control**
- 4 **Disponible** a través de internet



# OBJETIVOS



Elaborar un sistema informático que ayude en el **aprendizaje** de los sistemas operativos **Unix** a nivel de administración

## REQUISITOS:

- 1 Contenido teórico **estructurado**
- 2 Sistema **ampliable** de forma colaborativa
- 3 Ejercicios y sistema de **control**
- 4 **Disponible** a través de internet





# OBJETIVOS



Elaborar un sistema informático que ayude en el **aprendizaje** de los sistemas operativos **Unix** a nivel de administración

## REQUISITOS:

- 1 Contenido teórico **estructurado**
- 2 Sistema **ampliable** de forma colaborativa
- 3 Ejercicios y sistema de **control**
- 4 **Disponible** a través de internet



## ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- **Motivaciones**
- Planificación

## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# ¿POR QUÉ ESTE PROYECTO?

Cobertura práctica para ASO - Curso 2010/2011

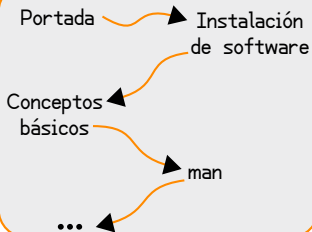


# ¿POR QUÉ ESTE PROYECTO?

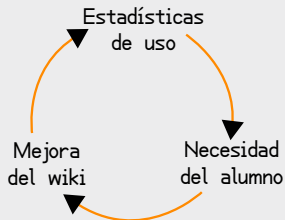
## PROPORCIONA AYUDA A...

- los alumnos
- los profesores
- **cualquier** usuario interesado que desee aprender

### ALUMNOS



### PROFESOR



# ¿POR QUÉ UN WIKI?

## MOTOR WIKI

Sistema sobre el que se sustenta un wiki

**WIKI** sitio web que puede ser modificado fácilmente a través de un navegador

## ¿POR QUÉ MONTAR UN WIKI NUEVO?

- No se puede utilizar *Wikipedia* para contener este proyecto:
  - *Wikipedia* es una enciclopedia
  - no incluye ejercicios
  - el nivel de explicaciones es más general

# ¿POR QUÉ UN WIKI?

## MOTOR WIKI


Sistema sobre el que se sustenta un wiki

**WIKI** sitio web que puede ser modificado fácilmente a través de un navegador

## ¿POR QUÉ MONTAR UN WIKI NUEVO?

- No se puede utilizar *Wikipedia* para contener este proyecto:
  - ➔ *Wikipedia* es una enciclopedia
  - ➔ no incluye ejercicios
  - ➔ el nivel de explicaciones es más general

# ¿POR QUÉ MEDIAWIKI?

- 
- Trabajo colaborativo
    - ➞ facilita ediciones y mejoras
  - Motor muy maduro<sup>1</sup>
    - ➞ más de 50.000 wikis
    - ➞ más de 42 millones de artículos
    - ➞ más de 1.000 millones de modificaciones
    - ➞ soporta los estándares HTML 4.0 y CSS 2.1
  - Comunidad muy amplia
    - ➞ más de 100.000 usuarios activos
    - ➞ usuarios en potencia (Fundación Wikimedia)

---

<sup>1</sup> [s23.org/wikistats](https://s23.org/wikistats)



# ¿POR QUÉ MEDIAWIKI?

- Trabajo colaborativo
  - ➞ facilita ediciones y mejoras
- Motor muy maduro<sup>1</sup>
  - ➞ más de 50.000 wikis
  - ➞ más de 42 millones de artículos
  - ➞ más de 1.000 millones de modificaciones
  - ➞ soporta los estándares HTML 4.0 y CSS 2.1
- Comunidad muy amplia
  - ➞ más de 100.000 usuarios activos
  - ➞ usuarios en potencia (Fundación Wikimedia)

---

<sup>1</sup>[s23.org/wikistats](https://s23.org/wikistats)



# ¿POR QUÉ MEDIAWIKI?

- Trabajo colaborativo
  - ➞ facilita ediciones y mejoras
- Motor muy maduro<sup>1</sup>
  - ➞ más de 50.000 wikis
  - ➞ más de 42 millones de artículos
  - ➞ más de 1.000 millones de modificaciones
  - ➞ soporta los estándares HTML 4.0 y CSS 2.1
- Comunidad muy amplia
  - ➞ más de 100.000 usuarios activos
  - ➞ usuarios en potencia (Fundación Wikimedia)

---

<sup>1</sup>[s23.org/wikistats](http://s23.org/wikistats)

## ÍNDICE



## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- **Planificación**

## 2 DESARROLLO

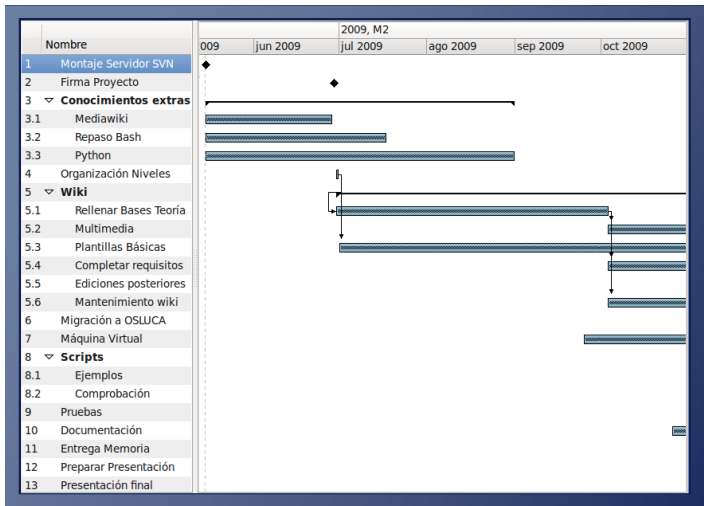
- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

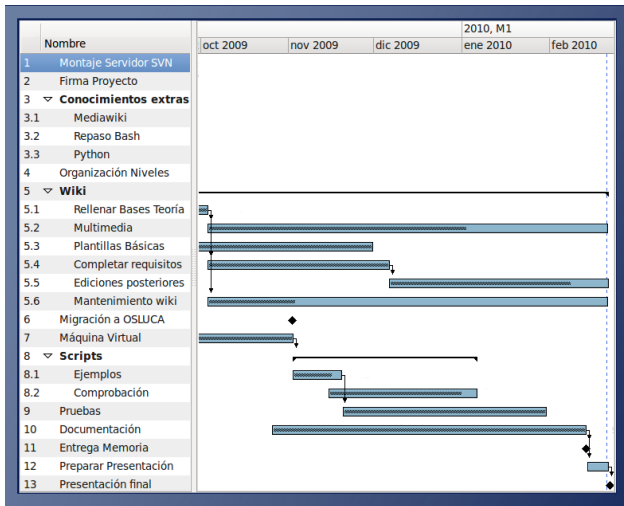
- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión



# DIAGRAMA DE GANTT [1/2]



# DIAGRAMA DE GANTT [2/2]



## ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación


## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES


- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# ESTRUCTURA DEL WIKI

- 
- Artículos principales para ASO
    - **niveles** de aprendizaje
    - información base  
por ejemplo, *Órdenes avanzadas* y `grep`
  - *Scripts* de ejemplo
  - Anexos: *curiosidades* e información *práctica*
  - *Ayuda* y navegación



# NIVELES DE APRENDIZAJE

- 
- 1 Usuario principiante
  - 2 Usuario iniciado
  - 3 Usuario habitual
  - 4 Unixero profesional



# ARTÍCULOS PRINCIPALES [1/2]



## USUARIO PRINCIPIANTE

- Características de los sistemas Unix
- Clasificación de la familia Unix
- Estructura de los sistemas Unix
- Conceptos básicos

## USUARIO INICIADO

- Órdenes básicas
- Sistema de archivos





# ARTÍCULOS PRINCIPALES [2/2]

## USUARIO HABITUAL

- Editor Vi
- Órdenes avanzadas
- Instalación de software

## UNIXERO PROFESIONAL

- Arranque y parada del sistema
- Administración de usuarios y grupos
- Administración del sistema de archivos
- Instalación de un servidor FTP con *vsftpd*
- Introducción a la seguridad en sistemas Unix

# CATEGORÍAS

*Agrupación de páginas que comparten algún tema en común*

## AGRUPACIÓN Y BÚSQUEDA

- Conceptos básicos
- Control y pruebas
- Curiosidades
- Desambiguación
- Ejercicios
- Noticias
- **Programas**
- Scripts

## MANTENIMIENTO

- Formatear
- ToDo

# CATEGORÍAS

*Agrupación de páginas que comparten algún tema en común*

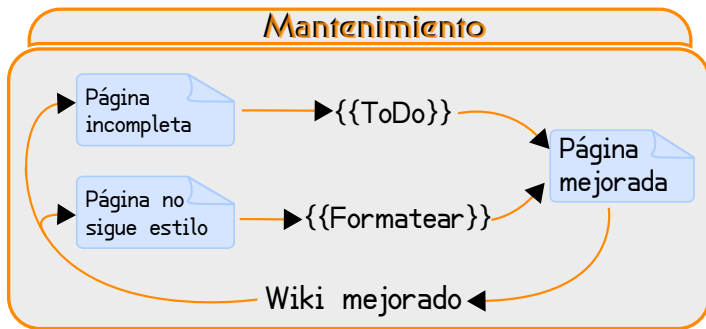
## AGRUPACIÓN Y BÚSQUEDA

- Conceptos básicos
- Control y pruebas
- Curiosidades
- Desambiguación
- Ejercicios
- Noticias
- **Programas**
- Scripts

## MANTENIMIENTO

- Formatear
- ToDo

# CATEGORÍAS



# ASPECTOS A DESTACAR

## ADAPTAR MEDIAWIKI AL PROYECTO

### Plantillas:

- Más de 30 plantillas
- Simplifican la edición
- Establecen pautas

{{man|grep}}

Manual: [grep](#)

# ASPECTOS A DESTACAR

## CONTENIDO LIBRE

Licencia dual (compatible proyectos Fundación Wikimedia)

- 1 Búsqueda de bibliografía fiable en español
- 2 Búsqueda de bibliografía fiable en inglés y traducción
- 3 Creación de información

# ASPECTOS A DESTACAR

## MODIFICACIONES EXTENSAS Y REPETITIVAS

### BOT

- programa que vigila un wiki (24x7)
- realiza mantenimiento repetitivo y pesado

*pywikipediabot*

La biblioteca para MediaWiki en Python.

- Generación de artículos de programas
- Traslado de categorías
- Correcciones ortográficas/semánticas leves
- Inclusión de nuevas plantillas en artículos de una categoría

# ASPECTOS A DESTACAR

## MODIFICACIONES EXTENSAS Y REPETITIVAS

### BOT

- programa que vigila un wiki (24x7)
- realiza mantenimiento repetitivo y pesado

### *pywikipediabot*

La biblioteca para MediaWiki en Python.

- Generación de artículos de programas
- Traslado de categorías
- Correcciones ortográficas/semánticas leves
- Inclusión de nuevas plantillas en artículos de una categoría



# ASPECTOS A DESTACAR

## MODIFICACIONES EXTENSAS Y REPETITIVAS

### BOT

- programa que vigila un wiki (24x7)
- realiza mantenimiento repetitivo y pesado

### *pywikipediabot*

La biblioteca para MediaWiki en Python.

- Generación de artículos de programas
- Traslado de categorías
- Correcciones ortográficas/semánticas leves
- Inclusión de nuevas plantillas en artículos de una categoría

# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación

## 2 DESARROLLO

- Contenido teórico
- **Plataforma de prueba**
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# LA MÁQUINA VIRTUAL

## MÁQUINA VIRTUAL

- software que **emula** a un ordenador
- puede ejecutar programas sobre un sistema operativo

## ¿POR QUÉ VIRTUALBOX?

- Es libre
- Es multiplataforma
- Utilización de instantáneas
- Soporte estándar OVF

# LA MÁQUINA VIRTUAL

## MÁQUINA VIRTUAL

- software que **emula** a un ordenador
- puede ejecutar programas sobre un sistema operativo

## ¿POR QUÉ VIRTUALBOX?

- Es libre
- Es multiplataforma
- Utilización de instantáneas
- Soporte estándar OVF

# LA MÁQUINA VIRTUAL

## MÁQUINA VIRTUAL

- software que **emula** a un ordenador
- puede ejecutar programas sobre un sistema operativo

## ¿POR QUÉ VIRTUALBOX?

- Es libre
- Es multiplataforma
- Utilización de instantáneas
- Soporte estándar OVF

# LA MÁQUINA VIRTUAL

## MÁQUINA VIRTUAL

- software que **emula** a un ordenador
- puede ejecutar programas sobre un sistema operativo

## ¿POR QUÉ VIRTUALBOX?

- Es libre
- Es multiplataforma
- Utilización de instantáneas
- Soporte estándar OVF

# LA MÁQUINA VIRTUAL

## MÁQUINA VIRTUAL

- software que **emula** a un ordenador
- puede ejecutar programas sobre un sistema operativo

## ¿POR QUÉ VIRTUALBOX?

- Es libre
- Es multiplataforma
- Utilización de instantáneas
- Soporte estándar OVF

# LA MÁQUINA VIRTUAL

- *Ubuntu*: distribución Debian GNU/Linux con entorno GNOME
- *Xubuntu*: distribución Ubuntu con entorno XFCE

## MÁQUINA VIRTUAL: *Xubuntu*

- Sistema seguro  
→ control + libertad
- Escritorio gráfico ligero  
→ requisitos menores
- Imagen del disco duro de menos peso  
→ en OVF: 1,3 GB



# LA MÁQUINA VIRTUAL

- *Ubuntu*: distribución Debian GNU/Linux con entorno GNOME
- *Xubuntu*: distribución Ubuntu con entorno XFCE

## MÁQUINA VIRTUAL: *Xubuntu*

- Sistema seguro
  - ➔ control + libertad
- Escritorio gráfico ligero
  - ➔ requisitos menores
- Imagen del disco duro de menos peso
  - ➔ en OVF: 1,3 GB

# LOS SCRIPTS DE VERIFICACIÓN

## SCRIPTS DE VERIFICACIÓN

- Genérico (un *script* real)
- Directorios y ficheros de comprobación
- COMPORTAMIENTO:
  - ① Muestra los ejercicios
  - ② Limpia el sistema antes de empezar
  - ③ Verifica controles intermedios y final
  - ④ Informa de los “errores” cometidos

¿POR QUÉ NO VARIOS *scripts*?

- Facilita al desarrollo de nuevos ejercicios
- Los usuarios no necesitan saber programar para ello

# LOS SCRIPTS DE VERIFICACIÓN

## SCRIPTS DE VERIFICACIÓN

- Genérico (un *script* real)
- Directorios y ficheros de comprobación
- COMPORTAMIENTO:
  - ① Muestra los ejercicios
  - ② Limpia el sistema antes de empezar
  - ③ Verifica controles intermedios y final
  - ④ Informa de los “errores” cometidos

¿POR QUÉ NO VARIOS *scripts*?

- Facilita al desarrollo de nuevos ejercicios
- Los usuarios no necesitan saber programar para ello

# EL SISTEMA DE CONTROL



## ACTUALIZACIÓN AUTOMÁTICA

- Se actualiza el directorio donde se realizan los ejercicios desde la forja
- Cada vez que se inicie la sesión<sup>2</sup>

---

<sup>2</sup>Siempre que tenga conexión a internet



# DEMOSTRACIÓN



## DEMOSTRACIÓN DEL WIKI Y DE LA PLATAFORMA DE PRUEBAS



# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación


## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# ESTADÍSTICAS DE WIKIUNIX

- 
- Número total de páginas: 375
  - Número total de artículos: 292
  - Ediciones en páginas: 1.975
  - Media de ediciones por página: 5,27
  - Ediciones en artículos: 1.234
  - Visitas totales: 11.445
  - Visitas por edición: 6,16



# APORTACIONES A LA COMUNIDAD

## FUNDACIÓN WIKIMEDIA

WIKIPEDIA. *La enciclopedia libre.*

➞ OVF, Unix, Planner,...

WIKIPEDIA. *The free encyclopedia.*

➞ Open Virtualization Format, Planner, XVIDCAP,...

## MEDIAWIKI.ORG

➞ Category:Navigation template/es, Installation/es, Manual:Installation requirements/es,...



# APORTACIONES A LA COMUNIDAD

## IV CONCURSO UNIVERSITARIO DE SOFTWARE LIBRE

### VISIBILIDAD DEL PROYECTO

WIKI <http://osl.uca.es/wikiunix>

PÁGINA WEB <http://cusl4-wikiunix.forja.rediris.es>

FORJA <https://forja.rediris.es/projects/cusl4-wikiunix/>

BLOG <http://wikiunix.wordpress.com/>

TWITTER <http://twitter.com/wikiunix>

# PUBLICACIONES

## V CONFERENCIA INTERNACIONAL DE SOFTWARE LIBRE

**TÍTULO** Tecnologías Wiki y Conocimiento  
Abierto en la Universidad

**AUTORES** *Manuel Palomo, Inmaculada Medina,  
Emilio J. Rodríguez, Noelia Sales*

**SEDE** Cáceres

**ACEPTADO** Diciembre 2009

**PÁGINAS** 16-19

**ISBN** 978-84-692-4211-7

# PUBLICACIONES

## XVI JORNADAS DE ENSEÑANZA UNIVERSITARIA DE LA INFORMÁTICA

**TÍTULO** Tecnologías wiki en la docencia  
de Ingeniería Informática

**AUTORES** *Manuel Palomo, Inmaculada Medina,  
Emilio J. Rodríguez, Noelia Sales*

**Enviado** a la espera de aceptación

# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación

## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión

# LENGUAJES Y BIBLIOTECAS

## LENGUAJES

- Bash (ejemplos)
- Python

## BIBLIOTECAS DE PYTHON

- os/sys/shutil...
- pysvn

## HERRAMIENTAS

- python-mode
- pylint: calidad del código (control = 10/10)

# LENGUAJES Y BIBLIOTECAS



## LENGUAJES

- Bash (ejemplos)
- Python

## BIBLIOTECAS DE PYTHON

- os/sys/shutil...
- pysvn

## HERRAMIENTAS

- python-mode
- pylint: calidad del código (control = 10/10)



# LENGUAJES Y BIBLIOTECAS

## LENGUAJES

- Bash (ejemplos)
- Python

## BIBLIOTECAS DE PYTHON

- os/sys/shutil...
- pysvn

## HERRAMIENTAS

- python-mode
- pylint: calidad del código (control = 10/10)

# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación

## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto

## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- Conclusión



# APERTURA DEL WIKI

- Apertura completa: sin restricciones
  - ➞ seguimiento de la evolución del wiki
- Versión estática de los artículos principales
  - ➞ información fiable
  - ➞ versión en PDF exportada a  $\text{\LaTeX}$

# ÍNDICE

## 1 INTRODUCCIÓN

- Objetivos
- Motivaciones
- Planificación


## 2 DESARROLLO

- Contenido teórico
- Plataforma de prueba
- Aportaciones del proyecto


## 3 CONCLUSIONES

- Lenguajes y herramientas utilizados
- El futuro del proyecto
- **Conclusión**


# OBJETIVOS CUMPLIDOS

- 
- 1 Contenido estructurado en niveles de aprendizaje
  - 2 Sistema wiki ampliable de forma colaborativa
  - 3 Sistema de verificación de ejercicios
  - 4 Disponible a través de internet

# OBJETIVOS CUMPLIDOS


- 
- 1 Contenido estructurado en niveles de aprendizaje
  - 2 Sistema wiki ampliable de forma colaborativa
  - 3 Sistema de verificación de ejercicios
  - 4 Disponible a través de internet

# OBJETIVOS CUMPLIDOS

- 
- 1 Contenido estructurado en niveles de aprendizaje
  - 2 Sistema wiki ampliable de forma colaborativa
  - 3 Sistema de verificación de ejercicios
  - 4 Disponible a través de internet



# OBJETIVOS CUMPLIDOS

- 
- 1 Contenido estructurado en niveles de aprendizaje
  - 2 Sistema wiki ampliable de forma colaborativa
  - 3 Sistema de verificación de ejercicios
  - 4 Disponible a través de internet

# EXPERIENCIA

He aprendido...

- mucho acerca de los sistemas Unix
- a trabajar en comunidades wiki
- a programar en un nuevo lenguaje (Python)
- a planificar y desarrollar un proyecto de forma individual
- a afrontar decisiones
- a trabajar de cara a la comunidad

# FUTURAS AMPLIACIONES

- Otras distribuciones con entornos gráficos diferentes
  - ➞ KDE
- Incluir y ampliar artículos
- Nuevos *bots* de mantenimiento
  - ➞ correcciones ortográficas y semánticas



# BIBLIOGRAFÍA



*Mark G. Sobell,*

MANUAL PRÁCTICO DE LINUX. Ed. Anaya, 2005.

I.S.B.N.: 978-84-415-2350-0



*N. Gift, J. M. Jones,*

PYTHON PARA ADMINISTRACIÓN DE SISTEMAS UNIX Y  
LINUX. Ed. Anaya, O'Reilly, 2008.

I.S.B.N.: 978-84-415-2540-5



*M. K. Dalheimer, M. Welsh,*

GUÍA DE REFERENCIA Y APRENDIZAJE LINUX. Ed. O'Reilly,  
2006.

I.S.B.N.: 84-415-2031-3



*A. Martelli, A. M. Ravenscroft, D. Ascher,*

PYTHON COOKBOOK. Ed. O'Reilly, 2005.

I.S.B.N.: 978-0-596-00797-3

# PREGUNTAS



Gracias por su atención

# ¿Preguntas?

WikiUNIX:

<http://osl.uca.es/wikiunix>



# ESTA PRESENTACIÓN ES LIBRE

Copyright 2009 Noelia Sales Montes

Este trabajo se publica bajo la siguiente licencia:  
Creative Commons Attribution-Share Alike License

<http://creativecommons.org/licenses/by-sa/2.5/>

Usted es libre de:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

**ATRIBUCIÓN** Debes reconocer la autoría de la obra en los términos especificados por el propio autor o licenciante.

**LICENCIAMIENTO RECÍPROCO** Si alteras, transformas o creas una obra a partir de esta obra, solo podrás distribuir la obra resultante bajo una licencia igual a ésta.