



## **ESCUELA SUPERIOR DE INGENIERÍA**

### **INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS**

TUTORIAL EN FORMATO WIKI

SOBRE SISTEMAS OPERATIVOS UNIX

CON PLATAFORMA DE PRUEBA

Noelia Sales Montes

26 de febrero de 2010





ESCUELA SUPERIOR DE INGENIERÍA

INGENIERO TÉCNICO EN INFORMÁTICA DE SISTEMAS

TUTORIAL EN FORMATO WIKI

SOBRE SISTEMAS OPERATIVOS UNIX

CON PLATAFORMA DE PRUEBA

- ◆ Departamento: Lenguajes y sistemas informáticos
- ◆ Director del proyecto: Manuel Palomo Duarte
- ◆ Autor del proyecto: Noelia Sales Montes

Cádiz, 26 de febrero de 2010

Fdo: Noelia Sales Montes



---

## Agradecimientos

Me gustaría agradecer y dedicar este documento...

- ✍ A Pablo y Rosa, que han estado conmigo en los momentos buenos y en los no tan buenos y que me han apoyado siempre.
- ✍ A Manuel, que me sugirió este proyecto que al final ha terminado apasionándome.
- ✍ A Emilio, que cada día me hace aprender algo más del gran mundo de los wikis.
- ✍ A Fabián, que sigue aportando buenas ideas para mejorar el proyecto.
- ✍ A la OSLUCA, que ha accedido a dejar un hueco en su servidor para este proyecto y en particular a Jose, por haberme ayudado cada vez que lo necesitaba.



---

## **Licencia**

Este documento ha sido liberado bajo Licencia GFDL 1.3 (GNU Free Documentation License). Se incluyen los términos de la licencia en inglés al final del mismo.

Copyright (c) 2009 Noelia Sales Montes.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

---

---



---

## Definiciones, acrónimos y abreviaturas

**Bot** programa diseñado para interactuar con otros programas, servicios de Internet u operadores humanos del mismo modo que si fuese una persona real

**Distribución** distribución de software basada en el núcleo Linux que incluye determinados paquetes de software para satisfacer las necesidades de un grupo específico de usuarios, dando así origen a ediciones domésticas, empresariales y para servidores

**Entorno de escritorio** conjunto de software para ofrecer al usuario de una computadora una interacción amigable y cómoda

El software es una solución completa de interfaz gráfica de usuario ofrece iconos, barras de herramientas, e integración entre aplicaciones con habilidades como, arrastrar y soltar.

**Gnome** entorno de escritorio e infraestructura de desarrollo para sistemas operativos Unix y derivados Unix como GNU/Linux, BSD o Solaris, compuesto enteramente de software libre

Nació como una alternativa a KDE bajo el nombre de “GNU Network Object Model Environment”.

**Forja** sitio web dedicado a hospedar proyectos de desarrollo de software

Normalmente, las forjas son gratuitas para proyectos de software libre y/o código abierto[44], pero también existen forjas para entornos comerciales.

**GNU Linux** término genérico referido a los sistemas operativos similares a UNIX (*Unix-like*) basados en el núcleo de Linux

**KDE** entorno de escritorio contemporáneo para estaciones de trabajo Unix

**L<sup>A</sup>T<sub>E</sub>X** sistema de composición de textos, orientado especialmente a la creación de libros, documentos científicos y técnicos que contengan fórmulas matemáticas

**Máquina virtual** software que emula a un ordenador y puede ejecutar programas como si fuese un ordenador real

**MPFC** Memoria Proyecto Fin de Carrera

**OVF** estándar abierto para empaquetar y distribuir servicios virtualizados o de forma más general software a ejecutar en máquinas virtuales.

**PFC** Proyecto Fin de Carrera

**Servicio virtualizado** instancia de una máquina virtual

**UNIX** sistema operativo portable, multitarea y multiusuario, desarrollado en principio en 1969 por un grupo de empleados de los laboratorios Bell de AT&T

**Unix-like** dícese de los sistemas operativos que poseen comportamiento similar al sistema UNIX original

**Wiki** sitio web colaborativo cuyas páginas web pueden ser editadas por múltiples voluntarios a través del navegador web

**Xfce** entorno de escritorio ligero para sistemas tipo Unix como Linux, BSD, Solaris y derivados.

---

---

---

## Notación y formato

A continuación se expone la notación utilizada en este documento para distinguir determinados elementos:

- ✍ Al referirnos a un programa en concreto, utilizaremos la notación: *programa*.
- ✍ Al referirnos a una orden o a una función de un lenguaje, usaremos la notación: *orden*.
- ✍ Para mostrar código empleamos:

FORMATO WIKI: EJEMPLO

---

```
1  {{Root
2  |reboot  [opciones] }}
```

---



# Índice general

---

<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos . . . . .	1
1.2. Contexto: Breve historia de los sistemas UNIX . . . . .	2
1.3. Alcance . . . . .	4
1.4. Visión general . . . . .	4
<b>2. Desarrollo del calendario</b>	<b>7</b>
2.1. Diagrama de Gantt . . . . .	7
2.2. Etapas de desarrollo del proyecto . . . . .	8
<b>3. Descripción general del proyecto</b>	<b>15</b>
3.1. Perspectiva del producto . . . . .	15
3.1.1. Resumen . . . . .	16
3.1.2. UNIX y la máquina virtual . . . . .	17
3.1.3. Lenguaje Python . . . . .	17
3.1.4. Estudio de motores wiki . . . . .	18
3.1.5. Lenguaje $\text{\LaTeX}$ . . . . .	22
3.2. Funciones del producto . . . . .	22
3.3. Características de los usuarios . . . . .	23
3.4. Restricciones generales . . . . .	24
3.5. Requisitos para el usuario . . . . .	24
<b>4. Desarrollo del proyecto</b>	<b>27</b>
4.1. Introducción . . . . .	28
4.2. Metodología de desarrollo . . . . .	28
4.3. Especificación de los requisitos del sistema . . . . .	30
4.3.1. Requisitos de externas y de usuario . . . . .	30
4.3.2. Requisitos funcionales . . . . .	30
4.3.3. Restricciones de diseño . . . . .	31
4.4. Análisis del sistema . . . . .	32
4.4.1. Contenido teórico . . . . .	32
4.4.2. Soporte del contenido . . . . .	34
4.4.3. Soporte de comprobación . . . . .	34
4.5. Diseño del sistema . . . . .	35
4.5.1. Contenido teórico . . . . .	35
4.5.2. Soporte del contenido . . . . .	42
4.5.3. Soporte de comprobación . . . . .	42

4.6. Codificación . . . . .	44
4.6.1. Contenido teórico . . . . .	44
4.6.2. Soporte de comprobación . . . . .	53
4.6.3. Estadísticas . . . . .	69
4.7. Pruebas y validación . . . . .	79
4.7.1. Descripción de Aspectos Generales . . . . .	79
4.7.2. Objetivos . . . . .	79
4.7.3. Entorno . . . . .	79
4.7.4. Proceso de prueba . . . . .	80
4.7.5. Documentación de los casos de prueba . . . . .	81
4.8. Mantenimiento . . . . .	83
4.8.1. Manual . . . . .	83
4.8.2. Automático: Uso de bots . . . . .	83
<b>5. Conclusiones</b>	<b>85</b>
<b>Manual de usuario</b>	<b>87</b>
Introducción . . . . .	87
Interfaz de usuario . . . . .	87
Publicación y modificación de artículos . . . . .	92
Soporte para pruebas . . . . .	105
Scripts de comprobación . . . . .	113
<b>Manual de instalación</b>	<b>117</b>
Introducción . . . . .	117
Requisitos para la instalación . . . . .	117
Instalación de dependencias . . . . .	117
Instalación de MediaWiki . . . . .	118
Cómo restaurar el temario . . . . .	119
Cómo hacer una copia de seguridad . . . . .	119
<b>Software utilizado</b>	<b>121</b>
<b>Lenguajes utilizados</b>	<b>129</b>
<b>Aportaciones a la comunidad</b>	<b>133</b>
<b>Código referenciado</b>	<b>135</b>
<b>Publicaciones</b>	<b>151</b>
<b>Bibliografía</b>	<b>153</b>
<b>GNU Free Documentation License</b>	<b>161</b>

# Índice de figuras

---

1.1. Árbol de las familias UNIX . . . . .	3
2.1. Planificación temporal del desarrollo del proyecto - Hasta octubre . . . . .	7
2.2. Planificación temporal del desarrollo del proyecto - De octubre en adelante . . . . .	8
2.3. Porcentaje de esfuerzo en cada una de las tareas (gráfica) . . . . .	9
2.4. Porcentaje de esfuerzo en la tarea “Conocimientos extra” (gráfica) . . . . .	9
2.5. Porcentaje de esfuerzo en la tarea “Wiki” (gráfica) . . . . .	10
2.6. Porcentaje de esfuerzo en la tarea “Scripts” (gráfica) . . . . .	12
4.1. Captura del esquema básico de información de un artículo . . . . .	29
4.2. Estructura de la página principal . . . . .	36
4.3. Sección Navegación de la Página Principal . . . . .	37
4.4. Sección Avisos de la Página Principal . . . . .	37
4.5. Sección Descargas de la Página Principal . . . . .	38
4.6. Navegación básica . . . . .	39
4.7. Estructura de un artículo principal . . . . .	40
4.8. Plantilla Archivo . . . . .	44
4.9. Plantilla Comando . . . . .	45
4.10. Plantilla Control Final . . . . .	46
4.11. Plantilla Control Intermedio . . . . .	47
4.12. Plantilla Curiosidad . . . . .	47
4.13. Plantilla Ejercicios . . . . .	48
4.14. Plantilla Infobox . . . . .	49
4.15. Plantilla Man . . . . .	50
4.16. Plantilla Resalta . . . . .	50
4.17. Plantilla Root . . . . .	51
4.18. Plantilla Título Minúsculas . . . . .	51
4.19. Plantilla Desambiguación . . . . .	51
4.20. Plantilla Formatear . . . . .	52
4.21. Plantilla ToDo . . . . .	52
4.22. Evolución del contenido . . . . .	70
5.1. Navegación básica general . . . . .	88
5.2. Estructura de la página principal . . . . .	91
5.3. Editor proporcionado por MediaWiki . . . . .	93
5.4. Lo no tan vital . . . . .	94
5.5. Más allá de lo necesario . . . . .	95

5.6. Plantilla Archivo . . . . .	96
5.7. Plantilla Comando . . . . .	96
5.8. Plantilla Control Final . . . . .	97
5.9. Plantilla ControlIntermedio . . . . .	98
5.10. Plantilla curiosidad . . . . .	99
5.11. Plantilla Ejercicios . . . . .	100
5.12. Plantilla Infobox . . . . .	101
5.13. Plantilla man . . . . .	101
5.14. Plantilla Resalta . . . . .	102
5.15. Plantilla Root . . . . .	102
5.16. Plantilla TítuloMinúsculas . . . . .	102
5.17. Plantilla Desambiguación . . . . .	103
5.18. Plantilla Formatear . . . . .	103
5.19. Plantilla ToDo . . . . .	104
5.20. VirtualBox - Importación - paso 1 . . . . .	106
5.21. VirtualBox - Importación - paso 2 . . . . .	107
5.22. VirtualBox - Importación - paso 3 . . . . .	107
5.23. VirtualBox - Importación - paso 4 . . . . .	108
5.24. VirtualBox - Importación - paso 5 . . . . .	109
5.25. VirtualBox - Importación - paso 6 . . . . .	109
5.26. VirtualBox - Importación - paso 7 . . . . .	110
5.27. VirtualBox - Importación - paso 8 . . . . .	110
5.28. VirtualBox - Instantánea - paso 1 . . . . .	111
5.29. VirtualBox - Instantánea - paso 2 . . . . .	111
5.30. VirtualBox - Instantánea - paso 3 . . . . .	112
5.31. VirtualBox - Error . . . . .	113



## Índice de cuadros

---

4.1. Órdenes básicas - Control inicial . . . . .	56
4.2. Órdenes básicas - Primer control intermedio . . . . .	56
4.3. Órdenes básicas - Segundo control intermedio . . . . .	57
4.4. Órdenes básicas - Control final . . . . .	57
4.5. Sistema de archivos - Control inicial . . . . .	58
4.6. Sistema de archivos - Primer control intermedio . . . . .	58
4.7. Sistema de archivos - Segundo control intermedio . . . . .	59
4.8. Sistema de archivos - Control final . . . . .	60
4.9. Órdenes avanzadas - Control inicial . . . . .	61
4.10. Órdenes avanzadas - Primer control intermedio . . . . .	61
4.11. Órdenes avanzadas - Segundo control intermedio . . . . .	63
4.12. Órdenes avanzadas - Control final . . . . .	63
4.13. Arranque y parada del sistema - Control inicial . . . . .	64
4.14. Arranque y parada del sistema - Primer control intermedio . . . . .	64
4.15. Arranque y parada del sistema - Control final . . . . .	65
4.16. Usuarios y grupos - Control inicial . . . . .	66
4.17. Órdenes avanzadas - Primer control intermedio . . . . .	66
4.18. Usuarios y grupos - Segundo control intermedio . . . . .	66
4.19. Usuarios y grupos - Control final . . . . .	68
4.20. Usuarios de WikiUNIX . . . . .	78
5.1. Sintaxis: Lo más básico . . . . .	93
5.2. Sintaxis: Lo básico . . . . .	94



# Introducción

---

Este capítulo consta de las siguientes secciones:

---

<b>1.1. Objetivos</b>	<b>1</b>
<b>1.2. Contexto: Breve historia de los sistemas UNIX</b>	<b>2</b>
<b>1.3. Alcance</b>	<b>4</b>
<b>1.4. Visión general</b>	<b>4</b>

---

## 1.1. Objetivos

El objetivo principal del proyecto es elaborar un sistema informático que ayude en el aprendizaje de los sistemas operativos UNIX a nivel de administración. Además se cuidará la facilidad para ampliar y mejorar su contenido.

El tutorial debe cumplir los siguientes requisitos:

- I. El contenido teórico principal debe estar estructurado en niveles de complejidad, de forma clara, concisa e interrelacionada.
- II. El soporte en el que se publique dicho contenido debe permitir y promover su ampliación y mejora colaborativa, para evitar quede obsoleto.
- III. Debe disponer de ejercicios y de un sistema de control de la resolución correcta de éstos cuando sea posible. De modo que el propio usuario pueda verificar si ha comprendido los conceptos teóricos y sabe aplicarlos de forma práctica.
- IV. El producto final debe estar disponible en Internet para toda persona interesada en el tema.

Es necesario aclarar que el proyecto podía incluir muchísimos temas. Por ello se ha decidido cumplir las necesidades de la parte práctica de la asignatura Administración de Sistemas Operativos impartida en la titulación de Ingeniería Técnica en Informática de Sistemas en la Universidad de Cádiz durante 2010/2011 mediante entrevistas mantenidas con su responsable y tutor de este Proyecto Fin de Carrera.

La estructuración de la temática que debe cubrir el contenido teórico principal es la siguiente:

### **Nivel de principiante:**

- ◆ Características
- ◆ Clasificación
- ◆ Estructura
- ◆ Conceptos básicos

### **Nivel de usuario iniciado:**

- ◆ Órdenes básicas
- ◆ Sistema de ficheros

### **Nivel de usuario habitual:**

- ◆ Editor VI
- ◆ Órdenes avanzadas
- ◆ Instalación de software

### **Nivel de administración:**

- ◆ Arranque y parada del sistema
- ◆ Administración de usuarios y grupos
- ◆ Administración del sistema de ficheros
- ◆ Instalación de un servidor FTP con vsftpd
- ◆ Introducción a la seguridad en un sistema UNIX

El desarrollo del contenido teórico debe realizarse en un lenguaje sencillo y claro para permitir que un estudiante universitario de cualquier Ingeniería Informática pueda seguir sus contenidos sin problema. Aunque este es el usuario principal, también está dirigido a cualquier persona que quiera adentrarse en este área de conocimiento, ya que se parte de un nivel prácticamente cero hasta el nivel de administración.

Por otro lado, el soporte debe permitir una gestión simple y una navegación cómoda por el contenido. Todo lo relacionado con la posible colaboración de otros usuarios debe ser lo más sencilla posible, sin necesidad de herramientas o aplicaciones que supongan un impedimento en lugar de una ayuda.

## **1.2. Contexto: Breve historia de los sistemas UNIX**

Los Laboratorios Bell fueron los responsables del desarrollo de UNIX como un proyecto de investigación privado dirigido por un pequeño grupo de personas que empezó en 1969.<sup>1</sup> El objetivo del grupo fue diseñar un sistema operativo que cumpliera los siguientes requisitos:

---

<sup>1</sup>Para conocer más detalles acerca de estos sistemas accede a la [página correspondiente](#) en el wiki y revisa sus referencias.

- ◆ Que fuera simple y elegante.
- ◆ Que estuviera escrito en un lenguaje de alto nivel.
- ◆ Que permitiera reutilizar el código.

UNIX tenía una relativamente pequeña parte de su código escrita en ensamblador (la que se encargaba de gestionar el hardware, más conocida como *kernel*) y el resto del código escrito en C.

A medida que el desarrollo avanzaba se realizaban pequeñas modificaciones al código del *kernel* (dependiendo de la plataforma) y se desarrollaron muchas utilidades en C. A través de esta evolución el *kernel* y el software asociado se extendieron hasta que un sistema operativo completo se desarrolló “sobre” el *kernel*.

Como evolución de UNIX aparecieron multitud de sistemas derivados, que se basaban en las mismas características (puede visualizarse un esquema en la Figura 1.1 [1]).

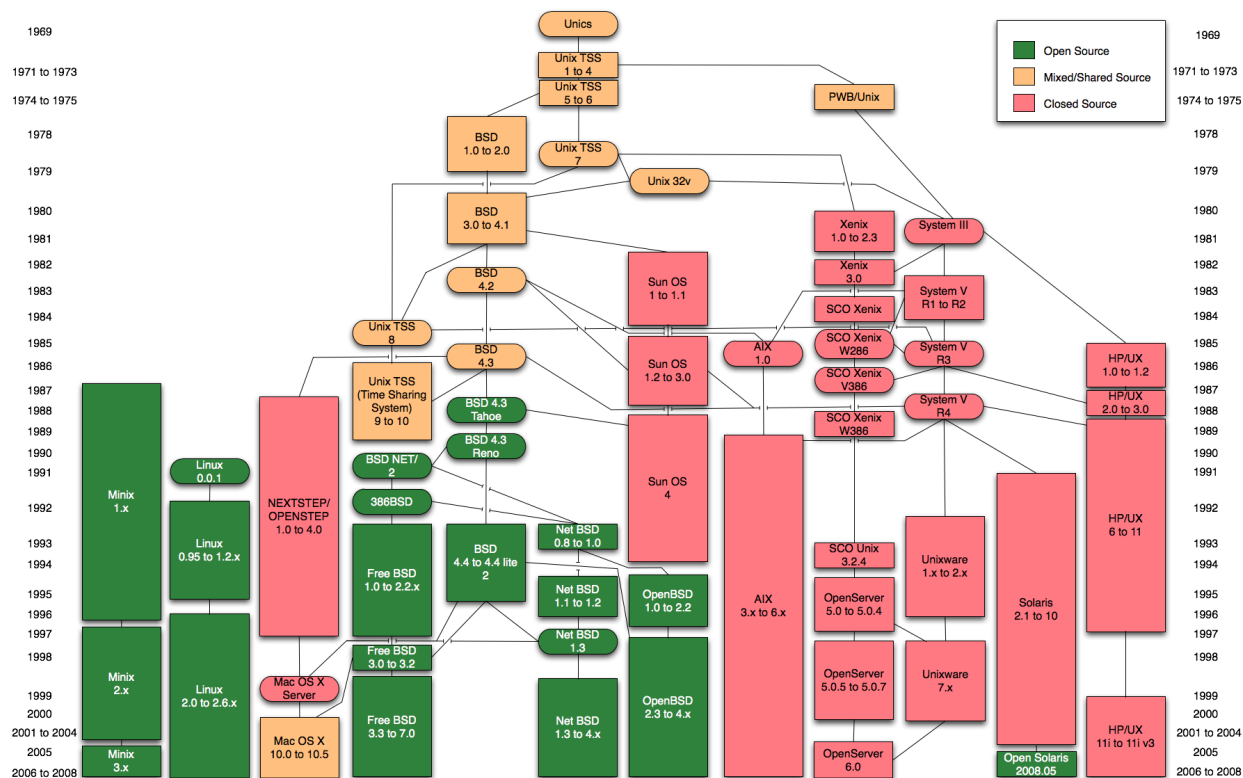


Figura 1.1: Árbol de las familias UNIX

Esta diversidad es muy ventajosa a la hora de elegir un sistema adecuado a nuestras necesidades, pero a la vez dificulta la administración de dichos sistemas dado que cada uno tiene sus particularidades. De ahí que surja la necesidad de una fuente de información unificada.

### 1.3. Alcance

El proyecto *Tutorial en formato wiki sobre sistemas operativos UNIX con plataforma de prueba* dará como resultado el producto *WikiUNIX* que cumplirá con todos los objetivos y especificaciones indicados en el apartado anterior.

El producto *WikiUNIX* se distingue en tres partes bien diferenciadas: el contenido teórico, los ejemplos para asimilar dicho contenido y el soporte de comprobación de ejercicios.

Todo lo que se encuentra en el temario viene descrito a su vez en función de términos más simples hasta donde es necesario (teniendo en cuenta que el usuario ya posee unos conocimientos mínimos básicos y que se proporcionan las referencias necesarias para su entendimiento).

La principal meta del proyecto es que sea un punto de partida robusto que permita la aplicación y mejora del tutorial por parte de todos aquellos interesados en la materia siendo útil y suficiente para el dominio de las técnicas y herramientas que exponemos para la administración de sistemas UNIX.

### 1.4. Visión general

En cuanto a la estructura de esta Memoria del Proyecto Fin de Carrera, tras este capítulo donde se presentan los objetivos y la visión en general del proyecto, se expone el desarrollo del calendario mediante un diagrama de Gantt describiendo con detalle las distintas etapas de desarrollo del proyecto.

El capítulo siguiente contiene la descripción general del proyecto, la perspectiva del producto, las funciones que cumple, los requisitos y las características que deben de cumplir los usuarios para un correcto aprovechamiento del producto. Las restricciones generales, dependencias, requisitos y planificaciones para futuras versiones. También son parte del contenido de esta sección del proyecto. Se incluye una explicación detallada de las herramientas elegidas para la elaboración del proyecto y el porqué de la toma de su elección.

Seguidamente describimos el desarrollo del proyecto, donde se especifica la metodología de desarrollo que hemos seguido, así como los distintos requisitos del sistema para pasar a su posterior análisis. Una vez realizado el proceso de análisis diseñamos el sistema a partir de la información proporcionada, para presentar en el apartado de codificación algunos aspectos relevantes de las distintas implementaciones que hemos llevado a cabo en este proyecto. Como parte del desarrollo, también se abordan las pruebas y el mantenimiento. Se ha de tener en cuenta que cada uno de estos apartados se abarca desde dos puntos de vista distintos: por un lado desde la elaboración del contenido teórico y por otro desde la del soporte para pruebas.

Finalmente, se presentan las conclusiones generales obtenidas una vez realizado el proyecto para pasar inmediatamente a los manuales de usuario y de instalación.

Además se presentan las referencias bibliográficas donde se incluyen las fuentes consultadas para la elaboración de este proyecto, un resumen que engloba las generalidades fundamentales del producto y de la metodología de desarrollo del proyecto, una guía de utilización (manual de usuario), una guía de instalación, un compendio del software utilizado y otro de los lenguajes de programación y finalmente la licencia completa del documento.

Por último, se adjunta un volumen con el contenido del tutorial para complementar esta Memoria del Proyecto Fin de Carrera. Así puede ser consultado en papel para ver el resultado de este proyecto.





# Desarrollo del calendario

Este capítulo consta de las siguientes secciones:

<b>2.1. Diagrama de Gantt</b> . . . . .	<b>7</b>
<b>2.2. Etapas de desarrollo del proyecto</b> . . . . .	<b>8</b>

## 2.1. Diagrama de Gantt

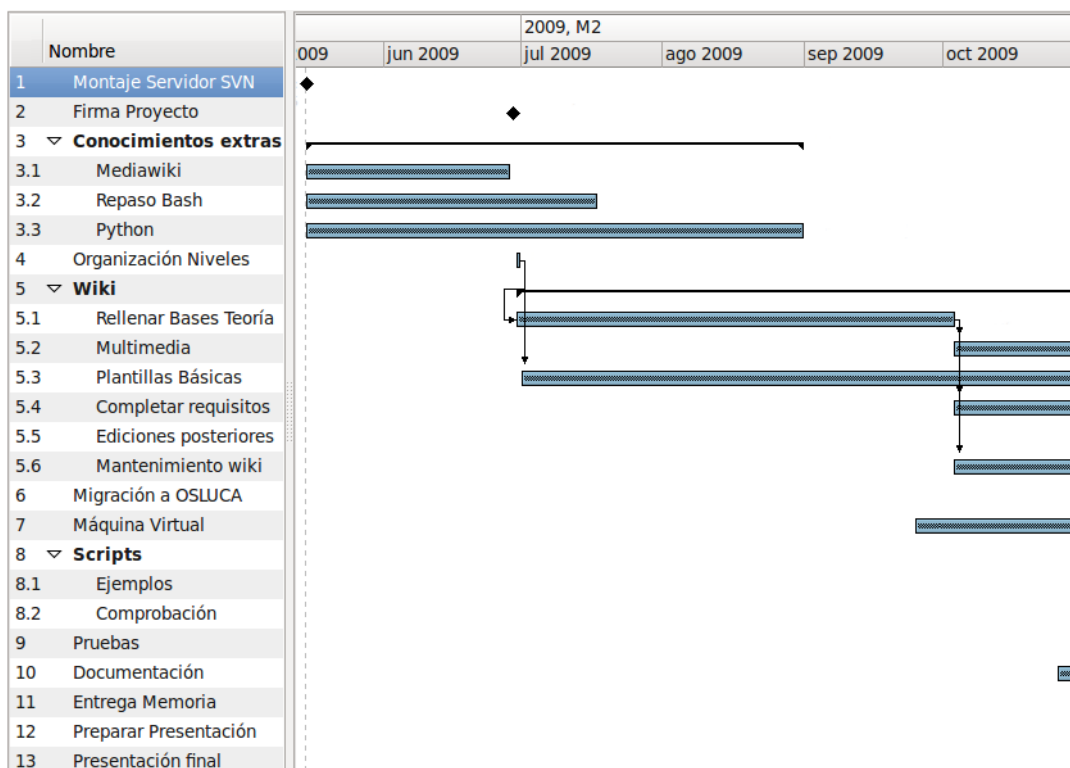


Figura 2.1: Planificación temporal del desarrollo del proyecto - Hasta octubre

En esta sección se muestra el diagrama de Gantt que describe las tareas llevadas a cabo durante el desarrollo completo del proyecto dividido en dos partes: una desde su inicio hasta finales de Octubre de 2009 (Figura 2.1) y otra desde inicios de Octubre de 2009 a su finalización (Figura 2.2).

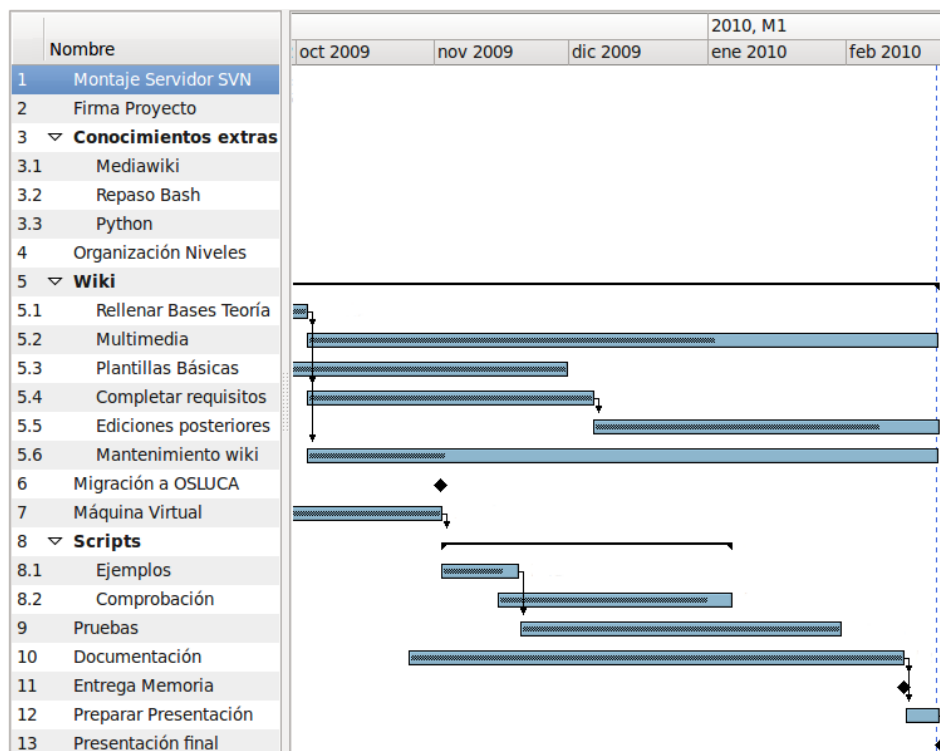


Figura 2.2: Planificación temporal del desarrollo del proyecto - De octubre en adelante

## 2.2. Etapas de desarrollo del proyecto

A continuación, se especifica tanto el proceso de desarrollo como el esfuerzo dedicado<sup>1</sup> en cada una de dichas tareas, junto con los diferentes hitos que marcaron el desarrollo principal de éste:

### 1. MONTAJE DE SERVIDOR *Subversion*:

Se realizaron varias reuniones con el tutor de proyecto para enfocararlo. En éstas sesiones se deciden el tipo de proyecto a realizar, las peculiaridades del mismo y los lenguajes de programación/codificación que serán necesarios.

<sup>1</sup>El esfuerzo aplicado en cada una de las tareas puede visualizarse en los diagramas de sectores que se incluyen en las Figuras 2.3, 2.4, 2.5.

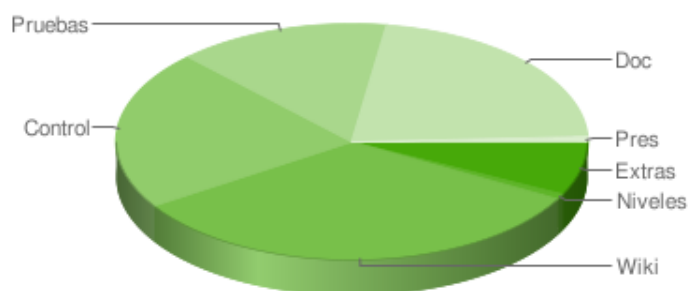


Figura 2.3: Porcentaje de esfuerzo en cada una de las tareas (gráfica)

A partir de ahí se monta un **servidor** en el PC personal de la autora. En él se instala *Subversion* para poder realizar la documentación  $\text{\LaTeX}$ , dado que es lo único claro en ese momento. Se inicia el desarrollo de los primeros **documentos** necesarios para el proyecto.

## 2. (HITO) FIRMA DEL PROYECTO:

Se realiza la firma del proyecto de forma oficial. Unos días más tarde, el proyecto es aceptado y se inicia el periodo de análisis previo.

## 3. CONOCIMIENTOS EXTRAS:

Se realiza un estudio de todas aquellas **herramientas necesarias** para construir y desarrollar el tutorial con el objetivo de facilitar las demás tareas.

Esta etapa de **preparación** es bastante amplia, e incluso se solapa con etapas de desarrollo posteriores, debido a que hay varios temas ante los cuales debemos prepararnos para trabajar con diversas secciones del proyecto.

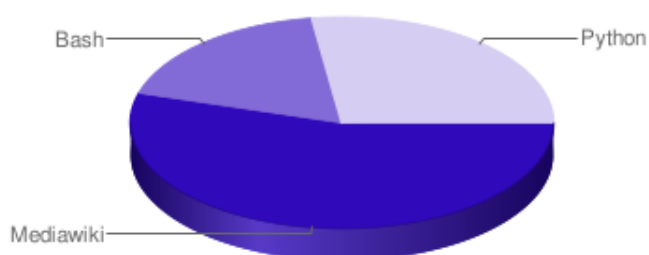


Figura 2.4: Porcentaje de esfuerzo en la tarea "Conocimientos extra" (gráfica)

### 3.1. MediaWiki:

Una vez determinado que el motor wiki escogido es definitivamente *MediaWiki*, por lo que hay que aprender su manejo, tanto a nivel de usuario como de administrador.

Se instalan instancias de wikis en diversas ocasiones durante este periodo. La instalación definitiva en el servidor personal se realizó el día 11 de Junio de 2009.

Se empieza a trabajar en sistemas que utilizan este mismo motor, como pueden ser Wikipedia [21] y Wikimedia Commons [22].

### 3.2. Repaso de Bash:

Repasar lo básico del lenguaje Bash ya estudiado en la carrera y su utilidad en cuanto a administración de sistemas operativos.

Se recuperan y mejoran scripts antiguos para su posible utilidad de cara al proyecto.

Se desarrollan scripts adicionales que puedan ayudar al aprendizaje de los alumnos.

### 3.3. Python:

Se empieza a estudiar la capacidad de este lenguaje en cuanto a administración de sistemas, encontrándose versátil y completo.

Sobre la marcha aprendemos los principios básicos de este lenguaje. Realizando pruebas comprobamos que, aunque no es tan eficiente como otros lenguajes, su simplicidad y su API bastante extensa resultan perfectas para los requisitos expuestos. Aún a pesar de su menor eficiencia, sigue siendo efectivo y rápido (casi instantáneo en todos los casos probados).

## 4. ORGANIZACIÓN EN NIVELES:

Antes de iniciar el desarrollo del proyecto, se aclaran qué necesidades de aprendizaje existen en el proyecto. Se distinguen los niveles de complejidad en función de las necesidades de varios usuarios “tipo” de sistemas Unix.

## 5. WIKI:

Esta tarea engloba la preparación del contenido teórico complejo del wiki. De ahí que sea la que más subtareas posee y la que más tiempo ha consumido.

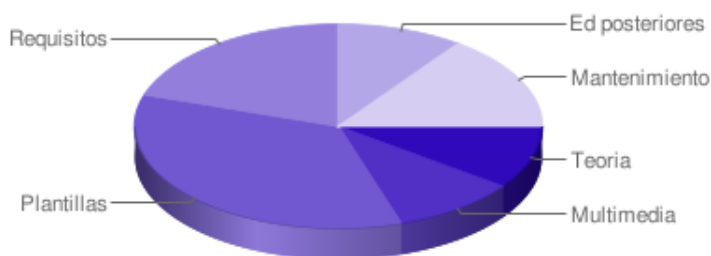


Figura 2.5: Porcentaje de esfuerzo en la tarea “Wiki” (gráfica)

### 5.1. Bases teóricas:

Conforme a los niveles y los artículos estipulados en la tarea anterior, se inicia la recopilación, escritura y formateado de información y su subida al wiki.

### 5.2. Multimedia:

Se crean o buscan las imágenes, esquemas, diagramas, ... que puedan resultar necesarios para una mejor asimilación del contenido que acompañarán.

Se incluyen estos elementos en el wiki, reconociendo su autoría y su licencia en caso de ser necesario (tanto si se obtienen de fuentes libres como si son creadas específicamente por el proyecto).

### 5.3. Adición de plantillas básicas:

Conforme se avanza y amplía el wiki, se detectan carencias o aspectos que pueden generalizarse. De ahí surge la necesidad de crear plantillas internas.

Las plantillas son de utilidad para simplificar el trabajo actual, pero también serán útiles para futuros colaboradores que quieran entrar en el proyecto.

Puede encontrarse una breve guía de estas plantillas en el manual de usuario en la página 87.

### 5.4. Completar requisitos mínimos:

Aunque el wiki estaba formado y ampliándose poco a poco, se necesitaban establecer unos requisitos mínimos para presentar el Proyecto Fin de Carrera.

En esta etapa se establecen dichos requisitos y se completan.

### 5.5. Ediciones posteriores:

Esta etapa continúa después de la entrega del proyecto, puesto que comprende a todas aquellas ediciones que amplían los requisitos mínimos requeridos y que siguen haciéndose. Algunas de estas ediciones son simples correcciones, mientras que otras pueden ser la creación de nuevos artículos.

Se continuarán realizando dichas ediciones dentro del Concurso Universitario de Software Libre y como tarea asociada a la plaza de alumna colaboradora que tiene la autora en el Departamento de Lenguajes y Sistemas Informáticos. Así se intentará fomentar la creación de una comunidad que ayude a la actualización y la mejora del proyecto.

### 5.6. Mantenimiento del wiki:

Al igual que ocurre con las ediciones posteriores, el mantenimiento del wiki es una etapa que no finaliza con la entrega del proyecto. El wiki debe seguir ampliándose por la comunidad y por esto mismo debe ser controlado: los errores deben ser corregidos y los nuevos contenidos deben ser vigilados para evitar información incongruente.

## 6. (HITO) MIGRACIÓN AL SERVIDOR DE LA OSLUCA:

El día 3 de Noviembre de 2009 se trasladó un backup del wiki al servidor de la OSLUCA para preparar la próxima apertura.

Cabe comentar que esta migración tuvo lugar en parte por problemas técnicos, ya que el PC personal de la autora sobre el que estaba implantado el wiki en sus inicios se averió. Este hecho también provocó un retraso en el proyecto, debido a no poder seguir editando el contenido directamente mientras no se restaurara el sistema.

## 7. MÁQUINA VIRTUAL:

Elección y preparación del programa desde el cual se ejecutaría la máquina virtual y de la distribución a utilizar.

## 8. SCRIPTS:

Diseño e implementación de los scripts a incluir en el wiki.

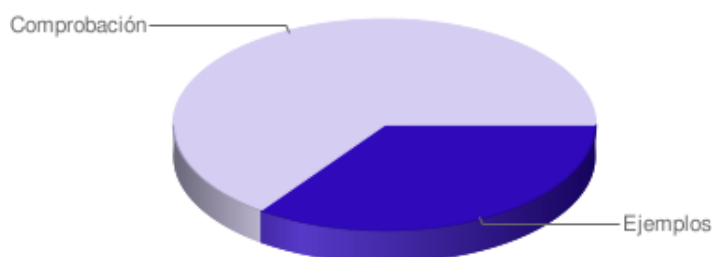


Figura 2.6: Porcentaje de esfuerzo en la tarea “Scripts” (gráfica)

#### 8.1. Ejemplos:

Los scripts de ejemplo son incluidos para que los alumnos vean que hay casos prácticos muy diversos donde la programación en Bash o Python es útil en administración de sistemas.

Se desarrollan scripts de distinto grado de complejidad, algunos inspirados por compañeros que solicitan ayuda en determinados temas relacionados con la administración.

#### 8.2. **Comprobación:** En estos se centra el contenido práctico del proyecto, puesto que sirven de control de los ejercicios que realizarán los alumnos/usuarios implicados.

### 9. PRUEBAS:

Fue necesario controlar tanto los scripts diseñados como la máquina virtual, así como la navegación del wiki. Para ello, se contó con ayuda externa para contrastar posibles problemas que no detectara el desarrollador.

### 10. DOCUMENTACIÓN:

En esta tarea se ha puesto especial hincapié por dos razones:

- ◆ Primero porque los alumnos/usuarios deben tener un buen material como guía o manual para poder trabajar y aprender sin tener dificultades extras.
- ◆ Segundo porque los futuros colaboradores necesitarán información para ocuparse de la administración del wiki y esta debe estar proporcionada en un formato y con un estilo claro, estético y adecuado.

Por esto, se han desarrollado varios documentos afines que ayudarán en ambos sentidos:

- ◆ Memoria y resumen del proyecto
- ◆ Presentaciones del proyecto
- ◆ Manual de usuario
- ◆ Manual de instalación

### 11. (HITO) ENTREGA DE LA MEMORIA:

Una semana antes de la presentación del proyecto se entregan los documentos necesarios para poder realizar la presentación final.

**12. PREPARACIÓN DE LA PRESENTACIÓN:**

Aunque el periodo de preparación de la presentación está restringido a la última semana previa a la presentación final, en realidad la presentación base está preparada desde finales de Diciembre de 2009. Así se ha necesitado más tiempo para el desarrollo de esta tarea.

**13. (HITO) PRESENTACIÓN FINAL:**

La presentación final ante el tribunal pertinente es el último hito del PFC, aunque no del proyecto en sí. Se espera tener continuación y ayuda por parte de la comunidad.





---

# Descripción general del proyecto

---

Este capítulo consta de las siguientes secciones:

---

<b>3.1. Perspectiva del producto</b>	<b>15</b>
3.1.1. Resumen	16
3.1.2. UNIX y la máquina virtual	17
3.1.3. Lenguaje Python	17
3.1.4. Estudio de motores wiki	18
3.1.5. Lenguaje $\text{\LaTeX}$	22
<b>3.2. Funciones del producto</b>	<b>22</b>
<b>3.3. Características de los usuarios</b>	<b>23</b>
<b>3.4. Restricciones generales</b>	<b>24</b>
<b>3.5. Requisitos para el usuario</b>	<b>24</b>

---

## 3.1. Perspectiva del producto

El producto WIKIUNIX nace con la intención de mejorarse y ampliarse más allá de su fecha de entrega. En la actualidad cumple unos requisitos mínimos de contenido teórico y ejercicios prácticos, pero podrá ser ampliado en función de los deseos de la comunidad de usuarios que se interesen por el proyecto.

El producto final viene descrito por los siguientes componentes y características:

◆ **INTERFAZ DE USUARIO:**

El usuario dispondrá del temario desarrollado en una interfaz sencilla tipo web. Esta interfaz es la proporcionada por la personalización del tema por defecto del motor wiki que usemos. Resulta muy amigable e intuitiva, por lo que permite acercar el contenido del producto a cualquier tipo de lector.

◆ **INTERFAZ SOFTWARE:**

El software del producto está compuesto por varias aplicaciones:

Por un lado, tenemos un servidor web sobre el que funciona el motor wiki que será el que soporte el temario del tutorial.

Por otro lado, la parte práctica del proyecto, se refiere a los ejemplos y códigos de control que están escritos en Bash o Python.

Los códigos de control están preparados para ser ejecutados directamente desde el servicio virtualizado habilitado para ello.

### ◆ MODOS DE OPERACIÓN:

- ➔ El primer modo de operación es el de **usuario**, que nos permite hacer modificaciones en la mayoría de los artículos, correcciones y participar en debates.
- ➔ El segundo modo de operación es el de **administrador**. Este perfil de administración sirve para proteger artículos, restringir el acceso a ciertos usuarios y organizar la estructura general del wiki.  
Este administrador será el encargado de realizar las copias de seguridad del sistema periódicamente preservando el histórico de modificaciones pasadas de dicha herramienta.<sup>1</sup>

### ◆ REQUISITOS DE ADAPTACIÓN:

El temario del sistema está preparado para poder ser alojado en cualquier servidor que disponga del motor wiki y el sistema de gestión de base de datos preparado para trabajar con él.

Durante las siguientes secciones se describirán herramientas, lenguajes y otros aspectos utilizados en el desarrollo del proyecto. En los apartados que sea conveniente se especificarán los requisitos necesarios para soportar dicha herramienta.

### 3.1.1. Resumen

El producto consta de un temario elaborado y editado en formato wiki (directamente sobre el sistema ya instalado y personalizado) sobre sistemas operativos UNIX. Este temario es convertido a  $\text{\LaTeX}$  para poder tener una versión estática y fuera de línea del contenido teórico además de la existente en el wiki.

El servidor donde corre el motor de WikiMedia es la versión 2.2.3 de Apache que ha sido preparado para interpretar PHP5 (lenguaje base de la implementación del motor).

En principio el servidor utilizado era uno personal, con características muy pobres para una utilización simultánea de más de 2 personas. Por esta razón se trasladó el sistema completo al servidor de la OSLUCA, donde se ha estimado que funcionará correctamente para un número considerable de usuarios conectados simultáneamente.<sup>2</sup>

---

<sup>1</sup>El propio motor wiki proporciona una recuperación parcial de los artículos dañados voluntaria o involuntariamente y permite una restauración de su contenido a través de la copia de seguridad de la base de datos que compone su contenido.

<sup>2</sup>Se ha probado con un sistema análogo que funciona perfectamente con más de 30 personas conectadas simultáneamente, por lo que cubre nuestras expectativas.

WikiMedia almacena su información en una base de datos. La base de datos está compuesta por los datos introducidos por el motor wiki automáticamente al almacenar en dicho motor el desarrollo de la temática y por el sistema de gestión de base de datos MySQL. Hemos elegido este sistema de gestión de base de datos porque es el más testeado con WikiMedia, lo que nos permite tener un alto grado de confianza en el sistema. La estimación del consumo de recursos puede deducirse siguiendo el mismo razonamiento utilizado con el servidor Apache.

Acerca del material adicional y la visualización correcta del wiki se recomienda la lectura de la Sección 3.4.

A continuación, se expone una visión más general de las herramientas sobre las cuales se cimenta el proyecto.

### 3.1.2. UNIX y la máquina virtual

Para realizar los ejercicios y las pruebas de control necesitamos de una máquina virtual de un sistema con estas características. La opción más lógica en este caso era utilizar una distribución de GNU Linux, debido a ser similar a UNIX (*Unix-like*), a su amplia expansión y a ser de licencia libre. Finalmente, tras un proceso que desarrollaremos en el Capítulo 4, elegimos como distribución base Xubuntu.

A continuación, detallaremos qué características específicas tienen ambos sistemas (tanto UNIX como familia en general como Xubuntu en el papel de distribución específica).

### 3.1.3. Lenguaje Python

La mayoría de los ejemplos y la totalidad de los sistemas de control de ejercicios del material proporcionado con el tutorial están escritos en Python. Python es un lenguaje interpretado, orientado a objetos de propósito general.[45] Entre las características que posee este lenguaje que nos han resultado muy útiles a la hora de codificar encontramos:



- ◆ Permite mantener de forma sencilla interacción con el sistema operativo y resulta muy adecuado para manipular archivos de texto.
- ◆ Python es un lenguaje de alto nivel (se asemeja mucho al pseudocódigo), es decir, los programas son muy compactos: un programa en Python suele ser bastante más corto que sus equivalente en lenguajes como C.
- ◆ Python es muy legible: la sintaxis es muy elegante y permite la escritura de programas cuya lectura resulta fácil, en comparación con otros lenguajes. Así cualquier usuario, aunque no tenga grandes conocimientos del lenguaje puede comprender un ejemplo dado, al menos de forma aproximada.
- ◆ La Biblioteca Estándar de Python (*Python Standard Library*) ofrece una cantidad importante de módulos muy útiles en la administración de sistemas, expuestos API con una navegación

muy simple y cómoda. Existe incluso un apartado de módulos dedicados exclusivamente a los servicios de sistemas operativos.[46]

- ♦ Al ser interpretado, el paso entre la codificación y las pruebas es instantáneo, no hay que compilar el código cada vez que se desee probarlo. Así se agiliza el proceso de desarrollo.
- ♦ La sintaxis obliga a una codificación ordenada.

Por supuesto, hay muchos más detalles entorno a este lenguaje.[47] Su estudio puede ocupar un proyecto entero, por lo que nos hemos limitado a comentar las más relacionadas con el nuestro.

Como todo lenguaje, también tiene sus inconvenientes. Por ejemplo, no es tan eficiente como otros lenguajes, por lo que resulta algo más lento. Sin embargo, tal y como es utilizado en este proyecto, la eficiencia no es un factor clave. Además los códigos de ejemplo se ejecutan casi instantáneamente, por lo que el usuario no percibiría la diferencia respecto a un lenguaje con mayor rendimiento. Por todo esto, tomamos la elección de utilizar Python como lenguaje prioritario en este proyecto.

### 3.1.4. Estudio de motores wiki

Uno de los objetivos del desarrollo del proyecto es que el temario del tutorial sea ampliable y modificable por la comunidad. Para habilitar dicho comportamiento es necesario utilizar un motor wiki que soportase dicho material.

El problema al que nos enfrentamos es qué motor utilizar, porque hay mucha diversidad. A continuación se muestran los motores que se han tenido en cuenta a la hora de tomar la decisión final, con una descripción breve y diversos datos de interés:

#### ♦ MediaWiki<sup>3</sup>

**Documentación:** Gran cantidad de documentación fácil de comprender para usuarios y desarrolladores.

**Comunidad:** Amplia comunidad con wiki y varias listas de mailing.

**Licencia:** GPL

**Última versión:** 1.15.1

**Tamaño:** 10.4 MB

#### ♦ JSPWiki: Motor Wiki de propiedades ricas y expandible construido en base a los componentes estándares J2EE (Java, servlets, JSP).[23]

**Documentación:** Documentación bastante buena para varias versiones de JSPWiki.

**Comunidad:** La comunidad incluye listas de correo y canal IRC.

**Licencia:** Licencia Apache.

**Última versión:** 2.8

**Tamaño:** 7.6 MB

---

<sup>3</sup>Puedes obtener más información en la página 126

- ◆ Moin Moin: Motor Wiki avanzado bastante fácil de utilizar, escrito en Python con una gran comunidad de usuarios.[24]

**Documentación:** Amplia documentación mantenida por la comunidad.

**Comunidad:** Lista de correo y canal IRC.

**Licencia:** GPL

**Última versión:** 1.9.1

**Tamaño:** 5.7 MB

- ◆ PMWiki: Sistema basado en wiki para la creación colaborativa y el mantenimiento de sitios web.[25]

**Documentación:** Buena documentación que describe todas las propiedades de este motor wiki.

**Comunidad:** Posee una lista de discusión y un cookbook escrito por la comunidad.

**Licencia:** GPL

**Última versión:** 2.2.10

**Tamaño:** 380 KB

- ◆ TikiWiki: Se trata de una solución Groupware/CMS<sup>4</sup>, es decir, un sistema de administración de contenido.[26]

**Documentación:** Amplio wiki con documentación.

**Comunidad:** Posee un foro muy activo y canal IRC.

**Licencia:** LGPL

**Última versión:** 4.1

**Tamaño:** 15.4 MB

- ◆ Wacko Wiki: Es un motor Wiki ligero y muy útil.[27]

**Documentación:** Buena documentación disponible en varios idiomas.

**Licencia:** BSD

**Última versión:** 4.3

**Tamaño:** 645 KB

- ◆ DokuWiki: Un wiki, fácil de utilizar, orientado principalmente a la creación de documentación de cualquier tipo. Está enfocado hacia los equipos de desarrolladores, grupos de trabajo y pequeñas compañías. Posee una sintaxis simple pero poderosa que se asegura de que los archivos de información se mantengan legibles aún fuera del wiki y facilita la creación de textos estructurados. Toda la información es almacenada en archivos de texto planos, no se requieren bases de datos.[28]

**Documentación:** Manual compacto que cubre todas las propiedades de este motor wiki.

**Comunidad:** Incluye varios foros activos y canal IRC.

**Licencia:** GPL

---

<sup>4</sup>Content Management System

**Última versión:** 2009-12-25c

**Tamaño:** 1.7 MB

- ◆ XWiki: Wiki de empresa, con muchas propiedades que se necesitan para el uso profesional. Es también una aplicación Wiki que permite la creación de aplicaciones (incluyendo información estructurada y scripting) dentro de la interface Wiki. Se pueden utilizar los lenguajes Velocity y Groovy. Está escrito en Java y utiliza un sistema de base de datos subyacente relacional (MySQL o HSQLDB, entre otros).[29]

**Documentación:** Muy buena documentación, con varias referencias muy útiles.

**Comunidad:** Se pueden encontrar varios foros y listas de correo en la zona de desarrolladores del sitio web.

**Licencia:** GPL

**Última versión:** 2.2

**Tamaño:** 81.7 MB

- ◆ VQWiki: También conocido como *Very Quick Wiki*, es un software de servidor Wiki escrito utilizando tecnología JSPs y Java Servlet. Está diseñado para ser instalado y funcionar con esfuerzo mínimo en el rango de los servidores de aplicaciones Java, así como Tomcat.[30]

**Documentación:** Varios manuales compactos.

**Comunidad:** Varias listas de correo activas.

**Licencia:** GPL

**Última versión:** 2.8.1

**Tamaño:** 2.9 MB

- ◆ WikkaWiki: Motor Wiki flexible, dócil, estándar y de peso liviano escrito en PHP, que utiliza MySQL para almacenar páginas. Elaborado en base a WakkaWikki, está diseñado para tener velocidad, extensibilidad y seguridad.[31]

**Documentación:** Amplia documentación fácil de comprender.

**Comunidad:** Varias listas de correo activas y un canal IRC.

**Licencia:** GPL

**Última versión:** 1.2

**Tamaño:** 804 KB

Finalmente nos decantamos por el motor wiki más potente y común: MediaWiki. Así se facilita que cualquier usuario que haya colaborado en cualquier proyecto de la Fundación Wikimedia[32] participe en nuestro proyecto. Nos hemos basado en la versión 1.15.1, por ser la versión estable más actual y por incluir diversas extensiones por defecto.



Este motor para wikis está desarrollado bajo licencia GPL, programado en PHP usando MySQL y Apache. Fue inicialmente desarrollado para soportar la Wikipedia y otros proyectos de la fundación

Wikimedia[32]. Sin embargo, se popularizó hasta tal punto que numerosos sitios web han sido creados con este motor, normalmente especializados, con diferentes objetivos independientes de dicha fundación.[45]

Algunas de las características más importantes de este motor son:

- ◆ Compatibilidad con el estándar.
- ◆ Lenguaje de edición de contenido claro y sencillo.
- ◆ Capacidad de bloquear usuarios o páginas.
- ◆ Gestión del control de contenido.
- ◆ Sistema de categorías jerárquico.
- ◆ Posible utilización como *groupware* debido a la existencia de diferentes niveles de usuario.
- ◆ Listas de seguimiento para que cada usuario pueda seguir los cambios de los temas de su interés.
- ◆ Personalización de vistas de usuario.
- ◆ Libertad en los nombres de las páginas que componen la wiki.
- ◆ Creación de espacios de nombres separados por páginas de distintos tipos.
- ◆ Páginas de discusión en las cuales se puede debatir sobre el contenido y las modificaciones que deban de realizarse en alguna de las páginas del wiki.
- ◆ Multitud de extensiones diversas, por ejemplo la que proporciona soporte  $\text{T}_{\text{E}}\text{X}$  para fórmulas matemáticas con visualización adaptada a las capacidades del navegador.

Como se ha comentado antes, *MediaWiki* está programado en *PHP* y utiliza como medio de almacenamiento de contenido una base de datos estando orientado a *MySQL*, aunque se puede montar sobre *PostgreSQL* u *Oracle* de manera opcional. Está orientado a los usuarios finales, sobre todo en el ámbito educativo, siendo compatible con todos los sistemas operativos. Es compatible con cualquier servidor que soporte *PHP* y no tiene requerimientos especiales. También dispone soporte comercial.

Tiene varios sistemas de seguridad y *antispam* como los permisos de páginas individuales, autenticación de usuarios, encriptación de email, listas negras. Posee *plugins* que nos ayudan a realizar esta tarea. Además se ha desarrollado un bot para revertir vandalismos llamado AVBOT [33].

Hay más detalles interesantes como la posibilidad de previsualizar artículos antes de guardarlos, controlar errores pequeños, cambios en el índice, histórico de páginas, revisión de artículos, comparación entre diferencias de versiones de un tema, índice general y sistema de ampliación mediante *plugins*[34].

Como capacidades especiales incorpora soporte *unicode*, tiene más de cien traducciones distintas, categorías de organización, espacios de nombres, redirección de páginas, búsqueda y manejo de conflictos.

La creación de artículos se realiza mediante cualquier tipo de enlace interno, no nos restringe la sintaxis y permite realizar enlazados con otros wikis o un banco de imágenes. Soporta fórmulas matemáticas, tablas, resaltado de sintaxis mediante plugins, notas al pie, comentarios, personalización de estilos, todo tipo de estadísticas y de comodidades para la edición.

Produce una salida XHTML 1.0 Transitional Standard. Soporta CSS, temas, RSS, ATOM, generación automática de tablas de contenido, exportación RAW, HTML, XML, PDF.. Incluso podemos adjuntar ficheros de cualquier tipo.

*Mediawiki* es por todas estas características el motor elegido para soportar el contenido.

### 3.1.5. Lenguaje $\text{\LaTeX}$

Para la realización de esta memoria y la extracción de una versión estática del contenido teórico del wiki hemos empleado el lenguaje  $\text{\LaTeX}$ .

$\text{\LaTeX}$  es un lenguaje de marcado para documentos formado por un gran conjunto de macros de  $\text{\TeX}$ , escritas inicialmente por Leslie Lamport ( $\text{\LaTeX}$ ) en 1984, con la intención de facilitar el uso del lenguaje de composición tipográfica creado por Donald Knuth. Es muy utilizado para la composición de artículos académicos, tesis y libros técnicos, dado que la calidad tipográfica de los documentos realizados con  $\text{\LaTeX}$  es comparable a la de una editorial científica de primera línea. Está liberado bajo licencia LPPL.<sup>[45]</sup>

Es un lenguaje de bajo nivel en el sentido de que sus acciones son muy elementales, pero con la ventaja añadida, en palabras de Lamport, de poder aumentar las capacidades de  $\text{\LaTeX}$  utilizando comandos propios de  $\text{\TeX}$  descritos en *The TeXbook*.<sup>[14]</sup> Esto es lo que convierte a  $\text{\LaTeX}$  en una herramienta práctica y útil pues, a su facilidad de uso, se une toda la potencia de  $\text{\TeX}$ . Estas características hicieron que  $\text{\LaTeX}$  se extendiese rápidamente entre un amplio sector científico y técnico, hasta el punto de convertirse en formato preferente en comunicaciones y congresos, y requerido por determinadas revistas a la hora de entregar artículos académicos y/o científicos.

El motivo principal de crear el temario en  $\text{\LaTeX}$  es disponer del contenido del producto en un soporte atemporal que además puede ser fácilmente obtenido de una traducción del formato wiki.

## 3.2. Funciones del producto

Las funciones principales que ofrece el producto WIKIUNIX son las siguientes:

**Aprendizaje por niveles:** El contenido teórico ofrece información útil en función de los conocimientos del usuario. También puede afianzar conocimientos previos (de niveles más básicos), ya que existe un sistema de jerarquías y prerrequisitos que relacionan los artículos entre sí.

**Información completa en castellano:** Toda la información contenida está escrita desde cero, traducida u obtenida de una fuente libre, en términos claros para que los usuarios puedan asimilar fácilmente los conceptos importantes.



**Participación:** El wiki no sirve tan solo para incluir información. Es un sistema de comunicación colaborativo, y como tal, sirve además de canal de comunicación. Permite crear nuevos artículos, mejorar y ampliar los existentes, debatir y comentar ideas en páginas de discusión, entre otras cosas.

**Ejemplos prácticos:** Incluye ejemplos prácticos en bastantes aspectos del contenido teórico para ayudar al usuario a entenderlos.

**Ejercicios y control:** También contiene una batería de ejercicios relacionados con sus respectivos artículos teóricos y por tanto clasificados en niveles, para que el usuario desarrolle sus conocimientos en aplicaciones prácticas.

Estos ejercicios se complementan con un sistema de control para que se asegure de estar realizando los ejercicios correctamente.

### 3.3. Características de los usuarios

El proyecto (tanto la parte teórica como los ejercicios prácticos) está orientado en principio al prototipo de alumno de estudios universitarios de Ingeniería Informática. De todas formas puede ser utilizado por cualquier persona que desee aprender a utilizar estos sistemas.

No se precisa información extra para poder sacar provecho de este proyecto, puesto que toda la información y los ejercicios son autocontenidos, aunque se necesita un mínimo de destreza (utilización básica de un sistema operativo, sea cual sea). Esto quiere decir que cualquier persona que sepa manejar en un entorno de usuario cualquiera puede utilizar este tutorial.

Para participar en el desarrollo del wiki **no se exigirá** ningún tipo de requisito: cualquier usuario, ya sea anónimo o esté registrado, podrá añadir información, editar artículos, participar en los hilos de discusión, etc. No será necesario ningún tipo de registro para realizar este tipo de interacción con el producto.

Existirán unos artículos destacados o importantes que estarán más protegidos: las páginas que se consideren importantes o bien desarrolladas no podrán ser modificadas en ningún sentido (salvo sus hilos de discusión) por ningún usuario, salvo aquellos que sean *burócratas* o *administradores*.

Puede parecer bastante arriesgado dejar el sistema tan “desprotegido”, pero si se ponen barreras tan solo se impide que la comunidad crezca. Los impedimentos solo sirven para que los usuarios que quieran colaborar no lo hagan y los que quieran causar problemas conseguirán hacerlo por muchos controles que deban saltar. De este modo se continúa con la filosofía colaborativa y completamente abierta que se pretendía tomar desde el principio del proyecto.

En conclusión, **cualquier persona** que quiera aprender sobre sistemas operativos UNIX es bienvenida a este proyecto, ya sea como colaborador o como usuario (anónimo o registrado).

### 3.4. Restricciones generales

Se impone una restricción básica en cuanto a la utilización de la máquina virtual. Debe ser un sistema UNIX que cumpla dos condiciones:

- ◆ Ser de licencia libre, dado que todo el proyecto pretende ser liberado bajo licencia libre.
- ◆ Ser un sistema ligero, dado que debe ejecutarse en una máquina virtual

Dentro de estas características se encuentran las distribuciones GNU Linux. Entre ellas podemos encontrar la distribución Xubuntu, una distribución oficial basada en la distribución Ubuntu.

Xubuntu utiliza el entorno de escritorio Xfce, a diferencia de Ubuntu, que utiliza GNOME como entorno de escritorio. Este cambio permite un mejor desempeño del sistema operativo en computadores con recursos limitados de sistema, al ser más ligero y rápido. De la misma manera, Xubuntu también se caracteriza por utilizar aplicaciones GTK+ diseñadas para utilizar menos recursos (como por ejemplo, utiliza Gnome Office en lugar OpenOffice.org).[2]

Por estos detalles, resulta bastante útil a la hora de crear una máquina virtual, por detalles de eficiencia y espacio. La versión empleada para la máquina virtual es la versión estable más reciente a la hora de realizar este proyecto, Xubuntu 9.10 (Karmic Koala), que fue lanzada el 29 de octubre de 2009.

### 3.5. Requisitos para el usuario

El wiki es accesible desde cualquier navegador que cumpla con los estándares web. Las plantillas han sido testeadas en los siguientes navegadores, obteniendo resultados aceptables siempre y cuando la resolución de pantalla fuera suficiente (800x600 o mayor)<sup>5</sup>:

- ◆ Firefox
- ◆ Google Chrome
- ◆ Opera
- ◆ Safari
- ◆ Arora

Con respecto al sistema de realización y control de ejercicios, se necesita tener instalado un programa de virtualización que soporte el estándar abierto OVF. Se aconseja *VirtualBox* para el que se han desarrollado una serie de tutoriales específicos dentro del wiki y de los manuales.

Es requisito indispensable que la versión de *Virtual Box* sea 2.2.0 o superior, a partir de la cual soporta el estándar OVF en el que se distribuye la máquina virtual.[48]

---

<sup>5</sup>Aún así, se recomienda la utilización de Firefox o Chrome ante cualquier anomalía que pueda surgir en la visualización del wiki.

La máquina virtual viene preparada para ser utilizada de forma instantánea, por lo que no hay requisitos que afecten al desarrollo de los ejercicios, a menos que por alguna razón se modifique dicho sistema.

*Descripción*

---

# Desarrollo del proyecto

---

Este capítulo consta de las siguientes secciones:

---

<b>4.1. Introducción</b>	<b>28</b>
<b>4.2. Metodología de desarrollo</b>	<b>28</b>
<b>4.3. Especificación de los requisitos del sistema</b>	<b>30</b>
4.3.1. Requisitos de externas y de usuario	30
4.3.2. Requisitos funcionales	30
4.3.3. Restricciones de diseño	31
<b>4.4. Análisis del sistema</b>	<b>32</b>
4.4.1. Contenido teórico	32
4.4.2. Soporte del contenido	34
4.4.3. Soporte de comprobación	34
<b>4.5. Diseño del sistema</b>	<b>35</b>
4.5.1. Contenido teórico	35
4.5.2. Soporte del contenido	42
4.5.3. Soporte de comprobación	42
<b>4.6. Codificación</b>	<b>44</b>
4.6.1. Contenido teórico	44
4.6.2. Soporte de comprobación	53
4.6.3. Estadísticas	69
<b>4.7. Pruebas y validación</b>	<b>79</b>
4.7.1. Descripción de Aspectos Generales	79
4.7.2. Objetivos	79
4.7.3. Entorno	79
4.7.4. Proceso de prueba	80
4.7.5. Documentación de los casos de prueba	81
<b>4.8. Mantenimiento</b>	<b>83</b>
4.8.1. Manual	83
4.8.2. Automático: Uso de bots	83

---

## 4.1. Introducción

La naturaleza de este proyecto no se adapta a ninguna de las metodologías clásicas de modalidades establecidas. No es un proyecto de desarrollo de aplicaciones software. Consideramos por lo tanto el proyecto como un *proyecto didáctico* para el cuál se han utilizado dos metodologías distintas.<sup>1</sup>

La primera de dichas metodologías se ha seguido para la composición y edición de textos didácticos. La segunda se refiere al desarrollo de software para completar dichos textos, tanto los ejemplos como los sistemas de control de resolución de ejercicios. Cabe comentar respecto al proceso de creación de estos códigos que ha primado el aspecto didáctico sobre la eficiencia o la implementación de los mismos.

## 4.2. Metodología de desarrollo

Como se ha comentado en la introducción, para el desarrollo del proyecto hemos seguido dos metodologías, debido a que las dos partes que componen el proyecto difieren entre sí, aunque se complementan una a la otra y no pueden separarse del todo. Ambas se han basado en un modelo en espiral, que nos permite establecer unos hitos de evaluación y revisión del producto.

En ambos casos se ha llevado a cabo un proceso, en el cual cada nueva fase se compone de las siguientes etapas:

- I. COMUNICACIÓN CON EL CLIENTE: Como cliente tomamos al tutor del Proyecto Fin de Carrera, ya que conoce las necesidades que debe de satisfacer el producto en todos sus aspectos. En esta etapa establecemos todas las nuevas funcionalidades, ampliaciones y modificaciones que haya que realizar al producto. Los elementos a modificar son proporcionados por el hito de la evaluación del cliente.
- II. PLANIFICACIÓN: Una vez propuestos los cambios y ampliaciones se procede a una planificación temporal del desarrollo de los mismos. El tiempo de realización de los mismos es flexible según el interés o carga de trabajo de dichas acciones.
- III. INGENIERÍA, CONSTRUCCIÓN, TERMINACIÓN: En esta fase se elabora el contenido del temario y se desarrolla las aplicaciones demandadas para la elaboración del proyecto.
- IV. EVALUACIÓN DEL CLIENTE: En esta fase se establece una reunión con el cliente para valorar el estado y progreso del proyecto para elaborar un producto de calidad.

La metodología seguida para la creación del contenido teórico utilizado en el proyecto busca potenciar su funcionalidad como herramienta para el estudiante. Las funciones que proporciona son:

- ◆ Presentar el contenido de forma atractiva, incluyendo elementos como texto formateado, imágenes que no sobrecarguen y eviten monotonía.
- ◆ Guiar aprendizaje en función de una serie de niveles de complejidad.

---

<sup>1</sup>Dichas metodologías serán detalladas por separado en los apartados 4.3, 4.4 y 4.5 debido a sus diferencias.

- ◆ Organizar la información de manera clara y concisa.
- ◆ Relacionar información creando conocimiento.

Para completar esta funcionalidad utilizaremos introducciones a los temas, ejemplos y analogías que nos permitan construir el conocimiento. Preguntas y ejercicios que nos permitan relacionar los distintos aspectos del conocimiento potenciando la creación por parte del estudiante.

Las unidades didácticas principales se basan en la siguiente estructura (Figura 4.1):

- ◆ Descripción. Donde se presentan los temas que se tratan en dicha unidad.
- ◆ Introducción o presentación de la unidad. Donde presentamos los aspectos y temas que vamos a tratar.
- ◆ Información básica de la unidad. Una serie de atributos tales como el tiempo estimado a invertir en el estudio de dicha unidad, el nivel de complejidad,...
- ◆ Desarrollo de la unidad. Contenido teórico desarrollado.
- ◆ Referencias de donde se ha extraído la información o en las que nos basamos para construirla desde cero.
- ◆ Integración con ejercicios y ejemplos que ayuden a comprender los conceptos expuestos en dicha unidad didáctica, en caso de que sea una unidad con utilidad práctica.

Ficha de artículo	
{{{descripción}}}	
Tiempo	{{{tiempo}}}
Nivel	{{{nivel}}}
Información extraída de	{{{info}}}
Prerrequisitos	Ninguno

Figura 4.1: Captura del esquema básico de información de un artículo

### 4.3. Especificación de los requisitos del sistema

Como hemos comentado el proyecto tiene dos núcleos fundamentales. El principal es el desarrollo del contenido teórico que compondrá el producto. El segundo núcleo, menos importante que el primero, es el desarrollo de un soporte para pruebas que sirva al estudiante para comprobar personalmente que los ejercicios que realice se han desarrollado correctamente.

A continuación se expone la especificación de requisitos.

#### 4.3.1. Requisitos de externas y de usuario

El contenido debe de ser escrito en una sintaxis clara y concisa. Habrá una serie de temas “principales” que recogerán temas completos y cohesionados para el aprendizaje de una rama relacionado con los sistemas operativos UNIX.

En cuanto al soporte tutorial debe de permitir la navegación entre los temas y ser capaz de mostrar un tema completo en una sólo ventana. Deberá de contener todo lo necesario para el seguimiento completo de la unidad.

En caso de necesitar adjuntar código fuente de ejemplos esté deberá de ser mostrado directamente sobre el mismo sistema, sin tener que acceder a ninguna aplicación externa para su correcta visualización. El interfaz del soporte deberá de ser único para todos los procesos a realizar con el mismo, facilitando el aprendizaje de uso al usuario. El lenguaje de sintaxis de participación del usuario en el soporte debe de ser sencillo y no estar ligado a conceptos técnicos.

Además debe de ser navegable desde cualquier sistema operativo con un navegador web compatible con HTML 4.0 y CSS 2. La organización del contenido debe de ser homogénea para todo el tutorial.

En cuanto a la resolución de ejercicios, tan solo se requiere que el sistema de comprobación sea lo suficientemente simple como para que el usuario pueda centrarse precisamente en los ejercicios y no en cómo comprobarlos. Además aprovechando la interfaz para el contenido teórico, debe precisarse y especificarse cómo debe ser utilizado este sistema, para que no quepan dudas acerca de su funcionalidad.

#### 4.3.2. Requisitos funcionales

El contenido teórico debe contener los recursos suficientes para que el estudiante asimile algunos conceptos introductorios de los sistemas operativos UNIX (siempre y cuando el alumno también esté dispuesto al aprendizaje). Debe tener la funcionalidad de ser un temario de calidad.

El soporte del tutorial debe de aportar las siguientes funcionalidades a los distintos usuarios:

◆ Usuario:



- I. Navegar por el contenido del tutorial.
  - II. Participar en los foros de discusión temáticos.
- ◆ Usuario registrado
  - I. Navegar por el contenido del tutorial.
  - II. Participar en los foros de discusión temáticos.
  - III. Amplicar el contenido del tutorial.
- ◆ Usuario administrador
  - I. Navegar por el contenido del tutorial.
  - II. Participar en los foros de discusión temáticos.
  - III. Amplicar el contenido del tutorial.
  - IV. Modificar el contenido del tutorial
  - V. Administrar las modificaciones de otros usuarios.
  - VI. Gestionar las galerías multimedia.
  - VII. Proteger contenido fundamental.

El sistema de comprobación debe ejecutarse en un entorno controlado y cerrado, al menos en la medida de lo posible. Esto quiere decir que, aunque es conveniente que se puedan realizar los ejercicios en cualquier sistema UNIX, debe proporcionarse un soporte específico en el cual se asegure (en la medida de lo posible) que no se van a producir problemas que desvíen el aprendizaje del alumno.

#### **4.3.3. Restricciones de diseño**

El temario debe estar organizado en unidades didácticas con una estructura bien definida que sea seguida en todo el tutorial.

El soporte del temario (motor wiki) debe de ser compatible con el estándar web HTML 4.0 y CSS 2, propuesto el W3C, para que sea visible en la mayoría de navegadores web. Sería conveniente que el servidor web que soporte dicho motor sea optimizado para el contenido multimedia así como el sistema gestor de base de datos que nos gestione el contenido.<sup>2</sup>

---

<sup>2</sup>En el servidor en que se encuentra actualmente el wiki aún no se ha podido optimizar el sistema.

## 4.4. Análisis del sistema

En este apartado presentamos el análisis del sistema. En nuestro producto hay tres componentes bien diferenciadas: el contenido teórico, el soporte de dicho contenido y el sistema de control. Vamos a proceder al análisis de cada una de estas partes desarrollando en profundidad el análisis de cada una de ellas.

### 4.4.1. Contenido teórico

En el análisis del contenido se dedican los artículos que se van a incluir y los niveles de complejidad en que se van a organizar. Primero se definen los niveles de complejidad como cuatro categorías de menor a mayor nivel de conocimientos:

◆ **Principiante:**

Se refiere a aquellos usuarios que no poseen conocimientos acerca de los sistemas UNIX.

◆ **Usuario iniciado:**

Se refiere a los usuarios que hayan utilizado en alguna ocasión estos sistemas, aunque tan sólo desde el punto de vista del usuario básico de escritorio. Necesitan conocer más a fondo la funcionalidad básica y la forma de trabajar de un usuario común.

◆ **Usuario habitual:**

Este tipo de usuario ya conoce los sistemas UNIX y puede trabajar de forma normal con ellos, aunque necesita de determinados consejos o prácticas para aprovecharlos.

◆ **Administrador:**

El administrador ya no es un usuario común al que le basta con poder trabajar en su propio equipo. Necesita conocimientos mucho más avanzados para poder controlar todo lo que ocurre en los sistemas que administre. Así podrá prevenir, evitar y solucionar los problemas que puedan surgir.

En cada uno de esos niveles se encuentran encuadrados determinados artículos principales, los cuales se detallan a continuación, clasificados en función de dichos niveles.

Dentro del NIVEL PRINCIPIANTE se encuentran los artículos:

1. **Características:**

Introducción a los sistemas operativos que se encuendran bajo la familia UNIX, especificando sus características.

2. **Clasificación:**

Distinción entre las subfamilias de sistemas operativos que derivan del sistema inicial UNIX.

3. **Estructura:**

Descripción de los componentes básicos de la un sistema operativo de la familia Unix: kernel, shell e interfaz.

**4. Conceptos básicos:**

Definición del vocabulario básico y las palabras que se suelen emplear en el ámbito de los sistemas Unix.

Dentro del NIVEL USUARIO INICIADO se encuentran los artículos:

**1. Órdenes básicas:**

Explicación a fondo de las órdenes más básicas que se deben conocer en un sistema Unix por su utilización común.

**2. Sistema de archivos:**

Información de los directorios base que conforman el sistema de archivos de los sistemas UNIX.

Dentro del NIVEL USUARIO HABITUAL se encuentran los artículos:

**1. VI:**

Referencia breve a la utilización del editor VI, que está presente en (casi) cualquier sistema UNIX por defecto.

**2. Órdenes avanzadas:**

Compendio de las órdenes de un sistema UNIX, con su modo de utilización y un enlace al manual del sistema.

**3. Instalación de software:**

Guía explicativa acerca de cómo instalar software.

Dentro del NIVEL ADMINISTRADOR se encuentran los artículos:

**1. Arranque y parada del sistema:**

Explicaciones de los niveles de la forma de comportarse de estos sistemas en su inicio y su apagado, así como los niveles y las formas en que puede hacerse.

**2. Usuarios y grupos:**

Administración y control de los usuarios y los grupos dentro de un sistema UNIX.

**3. Administración del sistema de archivos:**

Dedicada a un nivel más complejo de los sistemas de archivos y a la forma de trabajar de un administrador de sistemas UNIX respecto a ellos.

**4. Instalación de un servidor FTP con vsftpd:**

Proceso de instalación de un servidor FTP

**5. Seguridad:**

Introducción a algunas herramientas útiles que ayudan a que los sistemas UNIX sean más seguro ante ataques y/o eventualidades.

#### 4.4.2. Soporte del contenido

En cuanto al soporte tomamos la etapa de análisis como la fundamental para tomar una decisión acerca sobre el motor que vamos a utilizar para el temario de nuestro tutorial. Sobre los requisitos funcionales llegamos a la conclusión que vamos a utilizar el motor wiki MediaWiki, ya que en general cumplen todos los objetivos marcados en los requisitos del proyecto.

#### 4.4.3. Soporte de comprobación

En el análisis del soporte de comprobación, se especifican qué artículos van a contener ejercicios que deben ser comprobados y verificados en nuestra *plataforma para pruebas*. Además por cada grupo de ejercicios se expone qué se pretende que el alumno aprenda al realizarlos.

Los artículos para los cuales se desarrollarán ejercicios con sus respectivos *scripts* de control son:

◆ ÓRDENES BÁSICAS (NIVEL USUARIO INICIADO):

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes `ls`, `cp`, `cd`, `mv`, `rm`, entre otras.

Debe quedar claro el trabajo básico en la terminal, sobre todo en lo referente a creación, movimiento, eliminación y modificación de archivos y directorios.

◆ SISTEMA DE ARCHIVOS (NIVEL USUARIO INICIADO):

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes `chmod`, `chgrp`, `chown`, entre otras.

Debe quedar clara la relación del sistema de archivos con los permisos de todo su contenido. El usuario debe aprender a modificar según le convenga dichos permisos.

◆ ÓRDENES AVANZADAS (NIVEL USUARIO HABITUAL):

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes `tar`, `find`, `grep`.

Debe quedar clara la utilidad de estas órdenes más complejas. Precisamente su sintaxis más extensa, las hacen mucho más versátiles a la hora de poder sacarles más partido.

◆ ARRANQUE Y PARADA DEL SISTEMA (NIVEL ADMINISTRACIÓN):

Se desea que con este grupo de ejercicios el usuario asimile el comportamiento del sistema al iniciarse y cerrarse.

Debe quedar clara la utilidad de los distintos niveles de arranque y cómo se manejan y modifican sus características.

◆ USUARIOS Y GRUPOS (NIVEL ADMINISTRACIÓN): Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes `useradd`, `userdel`, `usermod`,...

Debe quedar claro cómo se manejan los usuarios y los grupos, sobre todo a la hora de administrar los permisos de cada uno y de controlar su nivel de modificación.

## 4.5. Diseño del sistema

### 4.5.1. Contenido teórico

El diseño del contenido teórico engloba la estructura general que tendrán todos los artículos principales. Cada unidad didáctica tendrá, siempre que sea oportuno, los siguientes apartados:

- I. Ficha de artículo: Contendrá toda la información básica acerca del artículo.
- II. Introducción: Motivación del capítulo que vaya a desarrollarse. Será el primer párrafo del artículo, previo al índice.
- III. Desarrollo: Exposición del contenido.
- IV. Ejemplos: Puede que intercalados en el propio contenido o en un apartado propio, en función del redactor.
- V. Ejercicios: Deberán encontrarse al final del contenido, junto con la solución o la explicación para utilizar el script pertinente según sea necesario.
- VI. Referencias: Fuentes desde donde se ha extraído la información. Se pueden encontrar en un apartado o dentro de la ficha del artículo.

Por otro lado, también se ha de especificar la navegabilidad del wiki en general, así como en las páginas más importantes.

### Navegabilidad en la página principal

Además de la navegación que aporta MediaWiki de por sí, se ha desarrollado una interfaz amigable y completa en la página principal del wiki para facilitar la búsqueda de contenido y ayuda.

Podemos ver en la Figura 4.2 que en esta estructura se distinguen varias secciones claramente diferenciadas:

#### I. BIENVENIDA

Breve introducción acerca de la temática del wiki y lo que se puede esperar de él.

#### II. ÍNDICE DE CAPÍTULO POR NIVEL DE COMPLEJIDAD

Hasta el momento sólo se dispone de doce artículos principales, encuadrados en los cuatro niveles de complejidad descritos anteriormente:

- a. Principiante
- b. Usuario iniciado
- c. Usuario habitual
- d. Unixero profesional



### III. NAVEGACIÓN

- a. Información: acceso directo a la información más importante del wiki (como puede ser lo relacionado con el *soporte para pruebas*).
- b. Búsqueda: además de la búsqueda habitual incluida en el lateral izquierdo del wiki, se han habilitado una serie de accesos directos a categorías de elementos útiles (como por ejemplo un enlace a todos los scripts incluidos en el wiki).
- c. Ayuda: accesos a las páginas de ayuda del wiki, tanto en el sentido estético (estilo en formato wiki) como en el informativo (cambios recientes)



Figura 4.3: Sección Navegación de la Página Principal

### IV. AVISOS

Contiene notas importantes que informan de cuestiones que deberían ser tenidas en cuenta por los usuarios.

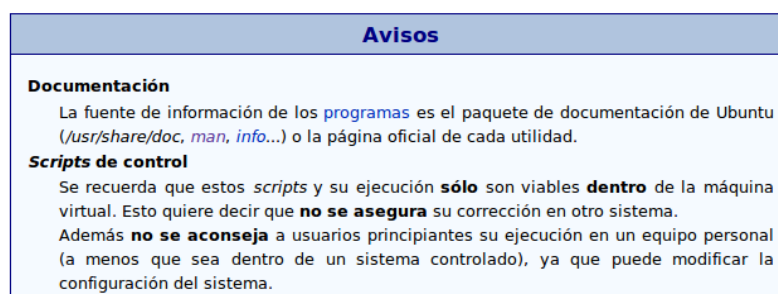


Figura 4.4: Sección Avisos de la Página Principal

## V. DESCARGAS

Sección donde se incluyen los enlaces a las descargas habilitadas, como la de la imagen de la máquina virtual, así como los últimos scripts incluidos en el wiki.

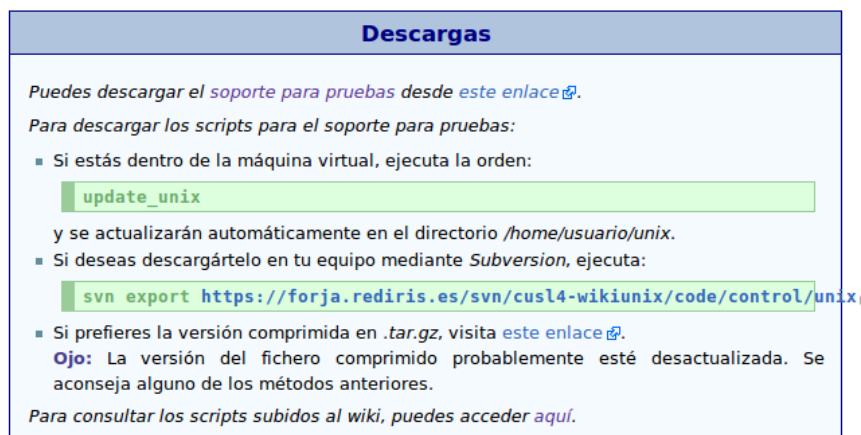


Figura 4.5: Sección Descargas de la Página Principal

## VI. NOTICIAS

Últimas dos noticias en torno al wiki, para mantener informada a la comunidad de manera simple y directa.

### Navegabilidad en cualquier página del wiki

Puede observarse en la Figura 4.6 la forma en que está estructurada la navegación general del wiki:

- I. LOGO DE WIKIUNIX: Permite volver a la portada del wiki en cualquier momento.
- II. BARRA LATERAL: Contiene diversos menús de navegación y búsqueda de artículos.

**Navegación:** Enlaces básicos de navegación. Gracias a ellos se puede acceder a la portada del wiki, al portal de la comunidad (página informativa acerca de la comunidad que desarrolla el wiki), a la sección de actualidad (que incluye las últimas noticias relacionadas con el wiki que también pueden visualizarse en portada), a los cambios recientes (donde se visualizan todos los cambios realizados en el wiki en los últimos instantes), a una página aleatoria (por si quieres tantear el estado de los artículos del wiki o te aburres) y a la ayuda general del wiki.

**Niveles:** Enlaces directos a los niveles de complejidad establecidos, que contienen los artículos principales del wiki.

**Información y descargas:** Enlaces a las páginas de información más importantes del wiki y a aquellas donde se accede a las descargas relacionadas con éste.

**Buscar:** Introduce una palabra para realizar una búsqueda.



Si eliges pulsar el botón **Ir**, en caso de que el título del artículo coincida con lo que hayas escrito, accederás automáticamente al artículo; en caso de que no exista, te mostrará la lista de artículos más probables.

Si pulsas sobre el botón **Buscar**, el servidor buscará en los artículos de WikiUNIX las palabras que hayas escrito, mostrándote una lista con los más probables.



Figura 4.6: Navegación básica

III. **HERRAMIENTAS**: Es un tipo de menú de navegación que nos permite interactuar con ciertos aspectos de la página que se muestra actualmente. Podemos obtener información de la página, subir un archivo y obtener una versión para impresión.

IV. **BARRA SUPERIOR DERECHA**: Si has entrado como usuario registrado, te aparecerá al principio de la página un cuadro con tus propios enlaces.

**Tu nombre de usuario** te lleva a tu página de usuario.

**mi discusión** te lleva a tu página de discusión.

**preferencias** te lleva a la página de configuración de tus preferencias de usuario.

**lista de seguimiento** te permite ver las últimas ediciones de los artículos que has marcado para vigilar.

**mis contribuciones** te muestra una lista de tus contribuciones, ordenadas por fecha.

**salir** te permite salir de tu cuenta de usuario.

- v. PESTAÑAS SUPERIORES: En la zona superior de la ventana, encima del título del artículo actual, se encuentran las pestañas relacionadas con éste.

**artículo** permite ver el artículo de Wikipedia.

**discusión** permite ir a la página de discusión del artículo.

**editar** permite modificar la página que estás viendo.

**historial** permite ver el historial de modificaciones de la página.

**trasladar** permite cambiar de nombre al artículo, dejando una redirección al nuevo nombre en el viejo (sólo para usuarios registrados).

**vigilar o dejar de vigilar** permite añadir o quitar un artículo de la lista de seguimiento (sólo para usuarios registrados).

## Navegabilidad en los artículos principales

### Características

Unix es un sistema operativo portable, multitarea y multiusuario, cuyo desarrollo comenzó en 1969 a manos de un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Ken Thompson, Dennis Ritchie y Douglas McIlroy.

Por normal general, el término "Unix" tradicional suele emplearse para describir a Unix o a un sistema operativo que cuenta con las características de UNIX Versión 7 o UNIX System V.

A continuación, vamos a profundizar en dichas características comunes.

Contenido [ocultar]
1 Portabilidad
2 Multitarea
3 Multiusuario
4 Otras características
4.1 Algoritmo por prioridades o multinivel
4.2 Memoria virtual
4.3 Sistema de archivos jerárquico
4.4 Comunicación con otros ordenadores
4.5 Sistemas de seguridad
5 Filosofía

Características de Unix	
En esta sección se introducen los sistemas operativos que se encuendran bajo la familia Unix, especificando sus características	
Tiempo	1 hora
Nivel	Principiante
Información extraída de	What is UNIX? <a href="#">[?]</a> The Unix Philosophy: A Brief Introduction <a href="#">[?]</a> Libros SO
Prerrequisitos	Unix

Portabilidad

[\[editar\]](#)

Figura 4.7: Estructura de un artículo principal

Los artículos principales tienen unas normas de estilo como se puede observar en la Figura 4.7. Deben contener:

- ◆ Una “ficha de artículo” donde se exponga la información básica:
  - ➔ Título
  - ➔ Descripción breve
  - ➔ Tiempo aproximado a dedicarle al artículo
  - ➔ Fuentes de la información: Referencias de donde hemos obtenido el contenido o en las cuales nos hemos basado para desarrollarlo.

- ➔ Prerrequisitos: Aquellos artículos del wiki que sea necesario leer (o al menos comprender) previo a la lectura de éste.
- ◆ Introducción al contenido del artículo. Sin destacar ni incluir en ninguna sección, para que aparezca antes del índice del documento.
- ◆ Índice en lado izquierdo: El índice se genera automáticamente gracias al motor. Se recomienda no moverlo de posición dado que ya se encuentra la ficha a la izquierda.
- ◆ Contenido del artículo.
- ◆ Referencias: En caso de incluir más aparte de las que se encuentran en la ficha del artículo.
- ◆ Ejercicios: En caso de incluirse deben encontrarse en la página “Nombre del artículo/Ejercicios” y enlazarse desde la página actual en una sección aparte.

## Categorías

Se establecen unas determinadas categorías en las que clasificar los distintos artículos, en función de sus características:

- ◆ Conceptos básicos: Categorizan aquellos artículos que tratan sobre contenidos muy básicos y que deben conocerse para poder trabajar en el wiki.  
Se enlazan desde el artículo “Conceptos básicos”.
- ◆ Control y pruebas: Categorizan aquellos artículos que tratan sobre el soporte para pruebas y los *scripts* de verificación de ejercicios.
- ◆ Curiosidades: Categorizan aquellos artículos que tratan sobre información adicional que no está relacionada directamente con los artículos principales, pero contienen datos interesantes.
- ◆ Desambiguación: Categorizan aquellos artículos cuyo título puede referirse a varios términos.
- ◆ Ejercicios: Categorizan aquellos artículos en que se encuentran los ejercicios de los artículos principales.  
Habitualmente sus títulos son tal que *artículo/Ejercicios*.
- ◆ Formatear: Categorizan aquellos artículos cuyo estilo debe ser formateado según los requerimientos del wiki.
- ◆ Noticias: Categorizan aquellos artículos que contienen noticias relacionadas con el wiki.
- ◆ Programas: Categorizan aquellos artículos que tratan sobre las programas/órdenes de un sistema UNIX.
- ◆ Scripts: Categorizan aquellos artículos que contienen los *scripts* de ejemplo y de control del wiki.
- ◆ ToDo: Categorizan aquellos artículos cuyo contenido debe ser completado o mejorado.

#### 4.5.2. Soporte del contenido

Como se ha comentado anteriormente, el motor elegido es *MediaWiki*. La potencia, fiabilidad, mantenimiento y perspectiva futura han sido los aspectos más relevantes en la decisión tomada. *MediaWiki* es un producto altamente fiable, implantado siempre con éxito en proyectos de distinta envergadura.

Cabe destacar que *MediaWiki* es la base de todos los proyectos hermanos de la Fundación Wikimedia[32] entre los que se encuentran la Wikipedia[35], Wikimedia Commons[22] o Wikibooks[36]. Al ser el motor más conocido, su mantenimiento y futuro están garantizados. Uno de los objetivos de este proyecto (en cuanto al soporte wiki) es luchar contra la temporalidad de ahí que este sea un factor importante de decisión.

#### 4.5.3. Soporte de comprobación

Ya sabemos que vamos a utilizar una máquina virtual, en la cual se realizarán los ejercicios solicitados. Para comprobar que estos ejercicios se han desarrollado de forma correcta, utilizaremos una serie de scripts de comprobación.

La máquina virtual es conveniente que pueda ser ejecutada sin muchas restricciones, por lo que utilizaremos *VirtualBox* para preparar nuestro sistema. Luego podremos exportarlo al formato estándar OVF, el cual puede ser ejecutado desde todos aquellos emuladores que soporten el estándar.

Instalamos y configuramos nuestro sistema virtualizado. Se parte de un Live CD de *Xubuntu*: recién instalado, la imagen en VDI<sup>3</sup> ocupa aproximadamente 3.3 GB. Una vez comprimida y exportada a OVF, ocupa 1,7 GB.

Teniendo en cuenta que esta imagen deberá ser descargada frecuentemente, se necesita que su espacio no sea excesivo, mucho menos si no es necesario. Por tanto, “limpiamos” el sistema virtualizado de todas aquellas aplicaciones que no sean estrictamente necesarias (todas aquellas aplicaciones de escritorio, tipo *OpenOffice*).<sup>4</sup> Una vez eliminadas estas aplicaciones, el sistema virtualizado llega a pesar 1.3 GB en formato OVF, algo más asequible que el tamaño anterior.

Sólo resta definir un esquema de actuación en cuanto a la definición y futura codificación de los scripts a desarrollar. Las pautas a seguir son las siguientes:

- I. Definición precisa del conjunto de ejercicios de la sección afectada
- II. Identificación de los controles intermedios a realizar
- III. Especificación del estado previo y posterior de la máquina en dichos puntos intermedios
- IV. Implementación de los scripts

---

<sup>3</sup>VDI es la extensión de ficheros virtualizados utilizada por *VirtualBox*.

<sup>4</sup>Esto no quita que una vez se descargue y configure el sistema en el equipo del usuario se puedan reinstalar dichas aplicaciones si el usuario lo desea: por algo poseerá todos los permisos en caso de que le sean necesarios.

v. Prueba de los scripts dentro del soporte de pruebas

## 4.6. Codificación

### 4.6.1. Contenido teórico

Una vez aclarado qué información va a incluirse en el wiki y bajo qué criterios organización, tan sólo queda incluir el contenido en formato wiki. Sin embargo, a la hora de empezar a codificar, sería bueno definir nuestras propias plantillas, que nos serían útiles para ayudar con codificación repetitiva o para modificar el estilo de determinados trozos de código.

A continuación, se detallan aquellas plantillas más *superficiales*, es decir, aquellas que se visualizan directamente en los artículos y que se forman en función de otras plantillas más *básicas*).

#### Plantillas habituales

##### ◆ PLANTILLA ARCHIVO

Esta plantilla debe utilizarse para incluir archivos con su respectivo contenido.

Recibe como parámetros el nombre del archivo y su contenido.

Si en formato wiki tenemos:

```
1 {{Archivo
2 |prueba.txt
3 |Monstruo Espagueti Volador
4 Calle de los espagnósticos 123
5 Barrio de Espagnostia
6 Ciudad Pastafari, C.P. 123456}}
```

se visualizará lo siguiente en el wiki:

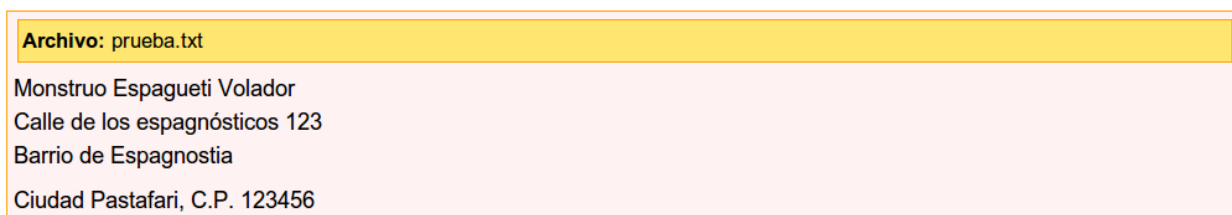


Figura 4.8: Plantilla Archivo

##### ◆ PLANTILLA COMANDO

Esta plantilla debe utilizarse para incluir órdenes que ejecuta en una terminal cualquier usuario con los permisos básicos.

Recibe como único parámetro la línea o las líneas que componen la orden y su respectiva salida en caso de que devolviera algo y se desee mostrarlo.

Si en formato wiki tenemos:

```

1  {{Comando
2  |cp [opciones] origen destino}}
```

se visualizará lo siguiente en el wiki:

```
cp [opciones] origen destino
```

Figura 4.9: Plantilla Comando

#### ◆ PLANTILLA CONTROLFINAL

Esta plantilla debe utilizarse para describir los controles finales de los ejercicios relacionados con los artículos principales del wiki. Solo puede haber uno por conjunto de ejercicios.

Recibe como parámetros:

- ➔ **artículo:** artículo principal relacionado con el conjunto de ejercicios
- ➔ **sudo:** debe ser “sudo” o “ ” (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- ➔ **directorio:** directorio en el cual se esté trabajando
- ➔ **script:** nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio
- ➔ **resumen:** conjunto de órdenes empleadas durante el desarrollo de los ejercicios

Si en formato wiki tenemos:

```

1  {{ControlFinal
2  |sudo=sudo
3  |artículo=Sistema de archivos
4  |directorio=archivos
5  |script=archivos
6  |resumen=[[chmod]] • [[chown]] • [[chgrp]] • otro}}
```

se visualizará lo siguiente en el wiki:

Vuelve al directorio inicial (**archivos**) y ejecuta el control final:

```
sudo python archivos.py
```

Tras la realización de estos ejercicios debes tener clara la utilización y el manejo de las siguientes órdenes:

`chmod` • `chown` • `chgrp` • otro

Si aún sigues teniendo dudas puedes:

1. Releer el artículo [Sistema de archivos](#) de nuevo al completo, o tan sólo las secciones en las que tengas dudas.
2. Volver a desarrollar los ejercicios de nuevo.
3. Revisar los artículos creados en el wiki para cada una de las órdenes.
4. Revisar el manual de las órdenes implicadas (recuerda que tienes los enlaces en los artículos del wiki de cada una y además en está el manual del sistema).

Figura 4.10: Plantilla Control Final

#### ◆ PLANTILLA CONTROLINTERMEDIO

Esta plantilla debe utilizarse para describir los controles intermedios de los ejercicios relacionados con los artículos principales del wiki.

Recibe como parámetros:

- **ordinal**: posición del control en el número total de controles intermedios, a saber *primer*, *segundo*,...
- **número**: número del control en el total de controles intermedios, a saber 1, 2,...
- **sudo**: debe ser "sudo" o " " (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- **directorio**: directorio en el cual se esté trabajando
- **script**: nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio
- **previo**: contenidos desarrollados hasta la aparición del control intermedio actual
- **posterior**: contenidos que se desarrollarán hasta el siguiente control intermedio

Si en formato wiki tenemos:

```
1  {{ControlIntermedio
2  |sudo=sudo
3  |ordinal=segundo
4  |número=2
5  |directorio=archivos
6  |script=archivos
7  |previo=Máscara de permisos y cambio de permisos y usuarios/grupos
8  |propietarios.
9  |posterior=Limpieza final del directorio: restaurar permisos y
10 |propietarios.}}
```



se visualizará lo siguiente en el wiki:

Vuelve al directorio inicial (**archivos**) y ejecuta el segundo control intermedio:

```
sudo python archivos.py -p 2
```

o

```
sudo python archivos.py --partial 2
```

DEBES HABER APRENDIDO A:	QUEDA POR APRENDER:
Máscara de permisos y cambio de permisos y usuarios/grupos propietarios.	Limpieza final del directorio: restaurar permisos y propietarios.

Figura 4.11: Plantilla ControlIntermedio

#### ◆ PLANTILLA CURIOSIDAD

Si en formato wiki tenemos:

```
1  {{Curiosidad
2  |Hay que tener cuidado en la utilización de la orden '''cp''' puesto
3  que puede destruir el contenido de un archivo.
4
5  Esto se debe a que si el archivo ''destino'' existe antes de utilizarla,
6  se sobrescribirá. Si se desea una forma más segura de utilizar esta
7  orden, se debe utilizar de forma interactiva:{{Comando|cp -i}}Así nos
8  preguntará antes de copiar el archivo:
9  {{Comando|$ cp -i a.txt b.txt
10 cp: overwrite 'b.txt'? y}}
11 }}
```

se visualizará lo siguiente en el wiki:

Curiosidad

Hay que tener cuidado en la utilización de la orden **cp** puesto que puede destruir el contenido de un archivo. Esto se debe a que si el archivo *destino* existe antes de utilizarla, se sobrescribirá. Si se desea una forma más segura de utilizar esta orden, se debe utilizar de forma interactiva:

```
cp -i
```

Así nos preguntará antes de copiar el archivo:

```
$ cp -i a.txt b.txt
cp: overwrite 'b.txt'? y
```

Figura 4.12: Plantilla Curiosidad

### ◆ PLANTILLA EJERCICIOS

Esta plantilla debe utilizarse para generar nuevos ejercicios relacionados con los artículos principales. Recoge toda la información básica de dichos ejercicios.

Recibe como parámetros:

- ➔ **sección:** nombre identificativo del artículo con el que se corresponden los ejercicios
- ➔ **resumen:** comentario breve acerca del contenido de los ejercicios
- ➔ **directorio:** directorio en el cual se trabajará
- ➔ **sudo:** debe ser “sudo” o “ ” (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- ➔ **script:** nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio

Si en formato wiki tenemos:

```

1  {{Ejercicios
2  |sección=sistema de archivos
3  |resumen=Permiten la modificación de permisos y propietarios de
4  los ficheros. Son muy útiles para la comprensión a fondo del
5  sistema de permisos que existe en este tipo de sistemas y para
6  valorar un buen control de éstos, así como qué importancia tienen.
7  |directorio=archivos
8  |sudo=sudo
9  |script=archivos}}
```

se visualizará lo siguiente en el wiki:

**Ejercicios de la sección de sistema de archivos**

**ASPECTOS PEDAGÓGICOS:** Permiten la modificación de permisos y propietarios de los ficheros. Son muy útiles para la comprensión a fondo del sistema de permisos que existe en este tipo de sistemas y para valorar un buen control de éstos, así como qué importancia tienen.

**PREPARACIÓN PREVIA:** Para hacer estos ejercicios debes acceder al sistema con el usuario por defecto *usuario* (recuerda que la contraseña es *unix*). A continuación:

- Ve al **directorio casa** del usuario actual. Comprueba que existe un directorio llamado **unix** y dentro de éste un subdirectorio llamado **archivos**. Para ello ejecuta:
 

```
ls unix
```

Debe aparecer el directorio **archivos**. En caso de que no exista alguno de estos elementos, puedes hacer varias cosas:

  1. Descargar el archivo comprimido con la jerarquía de elementos de comprobación que se encuentra en la portada del wiki y descomprimirlo en el directorio casa del usuario actual.
  2. Apagar la máquina virtual y restaurar el *snapshot* que funcionaba bien.
  3. Apagar la máquina virtual y borrarla eliminando el disco duro virtual. A continuación, descargar el disco duro y volver a configurar la máquina de nuevo.
- Ve al subdirectorio **unix/archivos**. Limpia el sistema antes de realizar los demás ejercicios para prevenir posibles incongruencias:
 

```
sudo python archivos.py -c
```

o

```
sudo python archivos.py --clean
```

Figura 4.13: Plantilla Ejercicios

### ◆ PLANTILLA FICHA DE ARTÍCULO

Esta plantilla debe utilizarse para generar nuevos artículos principales. Recoge toda la información básica del artículo.

Recibe como parámetros:

- **nombre:** nombre identificativo del artículo completo
- **descripción:** comentario breve acerca del contenido del artículo
- **tiempo:** duración estimada tanto de la lectura como del aprovechamiento completo del artículo (incluidos artículos secundarios enlazados y ejercicios)
- **nivel:** nivel de complejidad, actualmente *principiante*, *usuario iniciado*, *usuario habitual* y *unixero profesional*
- **info:** enlaces y documentos de los que se ha obtenido información para redactar el artículo
- **prerrequisitos:** páginas del wiki que deben leerse y comprenderse previo a la lectura del artículo

Si en formato wiki tenemos:

```

1  {{Ficha de artículo
2  |nombre=Características de Unix
3  |tiempo=1 hora
4  |descripción=En esta sección se introducen los sistemas operativos que
5  se encuendran bajo la familia Unix, especificando sus características
6  |nivel=[ [Nivel:Principiante|Principiante] ]
7  |info=[http://www.unix.org/what\_is\_unix.html What is UNIX?]<br>
8  [http://www.linfo.org/unix\_philosophy.html The Unix Philosophy: A Brief
9  Introduction]<br>Libros SO
10 |prerrequisitos=[ [Unix] ]}}
```

se visualizará lo siguiente en el wiki:

### Características de Unix

En esta sección se introducen los sistemas operativos que se encuendran bajo la familia Unix, especificando sus características

<b>Tiempo</b>	1 hora
<b>Nivel</b>	Principiante
	<a href="#">What is UNIX?</a>
<b>Información extraída de</b>	<a href="#">The Unix Philosophy: A Brief Introduction</a>
	<a href="#">Libros SO</a>
<b>Prerrequisitos</b>	<a href="#">Unix</a>

Figura 4.14: Plantilla Infobox

### ◆ PLANTILLA LISTAREF

Esta plantilla genera la lista de referencias utilizadas en un artículo.

En formato wiki se indica tal que:

```
1 {{Listaref}}
```

### ◆ PLANTILLA MAN

Esta plantilla genera un enlace al manual de Ubuntu de la orden afectada.

Si en formato wiki tenemos:

```
1 {{man|grep}}
```

se visualizará lo siguiente en el wiki:

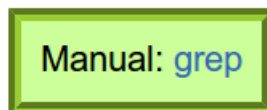


Figura 4.15: Plantilla Man

### ◆ PLANTILLA RESALTA

Esta plantilla destaca el texto contenido sobre el resto del artículo.

Si en formato wiki tenemos:

```
1 {{Resalta|Archivos}}
```

se visualizará lo siguiente en el wiki:

**Archivos**

Figura 4.16: Plantilla Resalta

### ◆ PLANTILLA ROOT

Esta plantilla debe utilizarse para incluir órdenes que ejecuta en una terminal un usuario con permisos de administrador.

Recibe como único parámetro la línea o las líneas que componen la orden y su respectiva salida en caso de que devolviera algo y se desee mostrarlo.

Si en formato wiki tenemos:

```
1 {{Root
2 |reboot [opciones]}}
```

se visualizará lo siguiente en el wiki:



reboot [opciones]

Figura 4.17: Plantilla Root

#### ♦ PLANTILLA TÍTULOMINÚSCULAS

Esta plantilla modifica el título principal del artículo cambiando todas las letras a minúsculas (incluida la primera).

Si en cualquier lugar de la página *Organizar.py* incluimos:

```
1 {{TOCderecha}}
```

se visualizará el título de la página tal que:



Figura 4.18: Plantilla TítuloMinúsculas

#### ♦ PLANTILLA TOCDERECHA

Esta plantilla desplaza el menú del artículo hacia el lado derecho de la página.

### Plantillas de mantenimiento

#### ♦ PLANTILLA DESAMBIGUACIÓN

La plantilla desambiguación se debe colocar en páginas que poseen varios significados o que pueden compartir un mismo título.

Si en formato wiki tenemos:

```
1 {{Desambiguación}}
```

se visualizará lo siguiente en el wiki:

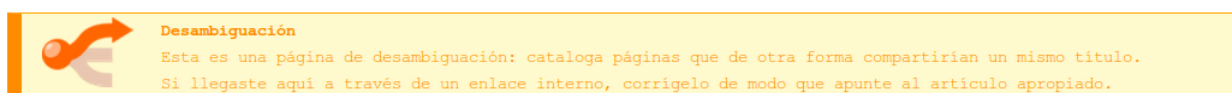


Figura 4.19: Plantilla Desambiguación

◆ PLANTILLA FORMATEAR

Esta plantilla se debe colocar en una página que haya que editar en cuestión de estilo o que no esté completa en este sentido.

Si en formato wiki tenemos:

```
1  {{Formatear}}
```

se visualizará lo siguiente en el wiki:

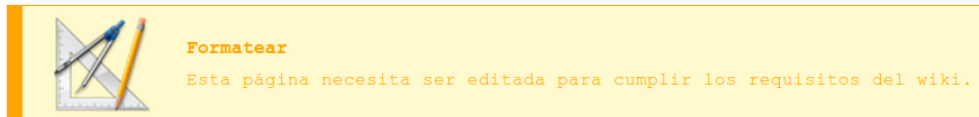


Figura 4.20: Plantilla Formatear

◆ PLANTILA TODO

La plantilla *ToDo* se debe colocar en una página que no esté completa o a la que le falta contenido.

Si en formato wiki tenemos:

```
1  {{ToDo}}
```

se visualizará lo siguiente en el wiki:

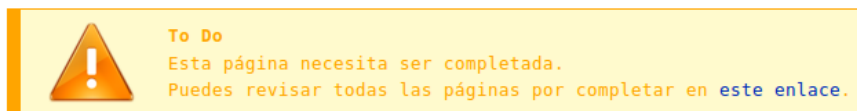


Figura 4.21: Plantilla ToDo

## Listado completo de plantillas

- |                       |                       |
|-----------------------|-----------------------|
| 1. AP                 | 2. Archivo            |
| 3. Artículo           | 4. Caja               |
| 5. Centrado           | 6. Cita web           |
| 7. Clic               | 8. Codigo             |
| 9. Comando            | 10. ControlFinal      |
| 11. ControlIntermedio | 12. CuadroEjercicios  |
| 13. CuadroIntermedio  | 14. Curiosidad        |
| 15. Desambiguación    | 16. Documentación     |
| 17. Documentación/doc | 18. Editar            |
| 19. Ejercicios        | 20. Ficha de artículo |
| 21. Formatear         | 22. Kernel            |

---

23. Listaref	24. Man
25. Noticia	26. Reffist
27. Resalta	28. Root
29. Smallcaps	30. TOCderecha
31. ToDo	32. Título
33. TítuloInterno	34. TítuloMinúsculas

#### 4.6.2. Soporte de comprobación

Una vez hemos definido qué ejercicios vamos a incluir en cada una de las secciones “prácticas”, hemos de especificar dónde vamos a introducir los puntos de control intermedios y qué estado deberá tener la máquina antes y después.

Las secciones que contienen ejercicios con sistemas de comprobación son:

- ◆ Órdenes básicas (Nivel Usuario iniciado)
- ◆ Sistema de archivos (Nivel Usuario iniciado)
- ◆ Órdenes avanzadas (Nivel Usuario habitual)
- ◆ Arranque y parada del sistema (Nivel Unixero profesional)
- ◆ Administración usuarios y grupos (Nivel Unixero profesional)

En esta parte del desarrollo nos encontramos con un obstáculo que salvar. Al principio se pensaba construir un *script* por cada conjunto de ejercicios a verificar. Sin embargo, esto no era factible por varias razones:

- ◆ Escribir código específico para cada grupo de ejercicios es **poco eficiente**, además de una pérdida considerable de tiempo. Por desgracia, a esta conclusión se llegó tras haber desarrollado varios de ellos.
- ◆ Tampoco resultaba fácilmente **ampliable**, ya que obligaba a poseer conocimientos de programación en Python para poder continuar la labor.

Dado que en Unix todo se representa mediante un fichero, finalmente se desarrolló un único *script* (que se puede ver en la página 135) parametrizable para soportar todos los ejercicios. Éste se sirve de una serie de directorios y ficheros de control.

Para ello definimos **directorios de control** que contienen la información básica a comprobar durante el desarrollo de los ejercicios. Éstos separan las etapas de corrección según:

- I. BASE - Directorio que contiene las comprobaciones que deben realizarse siempre. Esto quiere decir que previo a las demás comprobaciones se ejecutarán las que se definan en este directorio.

- II. **CLEAN** - Directorio de limpieza, donde se indican las órdenes y comprobaciones a realizar para dejar el directorio de trabajo del *script* listo para poder realizar los ejercicios.
- III. **PARTIALN** - Siendo  $n$  un número dentro del rango de las comprobaciones a realizar menos 1. Es el directorio que contiene las comprobaciones a realizar en el control intermedio número  $n$ .
- IV. **FINAL** - Directorio que contiene las comprobaciones finales, es decir, las necesarias para verificar que el ejercicio se ha desarrollado correctamente hasta el último punto de control.

Por otro lado, tenemos una serie de ficheros de control, los cuales se distinguen en dos grupos principales: los **ficheros de comprobación**, que tan sólo verifican información sin modificar nada, y los **ficheros de acción**, que se encargan de preparar el entorno para la ejecución de los ejercicios (habitualmente estos ficheros se utilizan dentro de los directorios de limpieza o de comprobación previa).

Los ficheros de comprobación se dividen en:

**direxists** Incluye los directorios de los que se desea comprobar su existencia.

El formato del fichero debe ser un directorio por cada línea.

**fileexists** Incluye los ficheros de los que se desea comprobar su existencia.

El formato del fichero debe ser un fichero por cada línea.

**dirclean** Incluye los directorios de los que se desea comprobar su “no-existencia”, o lo que es lo mismo, de los que se desea comprobar que se hayan borrado.

El formato del fichero debe ser un directorio por cada línea.

**fileclean** Incluye los ficheros de los que se desea comprobar su “no-existencia”.

El formato del fichero debe ser un fichero por cada línea.

**perm** Incluye los permisos que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y de los permisos en formato octal de 7 dígitos.

**own** Incluye los propietarios que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y del identificador del usuario (UID).

**grp** Incluye los grupos propietarios que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y del identificador del grupo (GID).

**mask** Incluye la máscara de permisos que se desea que posea el sistema.

El formato del fichero debe ser una única línea con la máscara en formato octal de 4 dígitos.

**fileequals** Incluye ficheros de los cuales se debe comprobar su igualdad.

El formato del fichero debe ser las parejas de ficheros separados por un espacio.



**direquals** Incluye directorios de los cuales se debe comprobar su igualdad.

El formato del fichero debe ser las parejas de directorios separados por un espacio.

**content** Incluye expresiones regulares que se desean encontrar en un determinado fichero. Se da por válida esta comprobación si se encuentra dicha expresión en una o más líneas del fichero.

El formato del fichero debe ser el nombre del fichero afectado seguido de la expresión regular a buscar.

**links** Incluye enlaces y ficheros enlazados que se desean comprobar.

El formato del fichero debe ser el fichero que contiene el enlace y el fichero que se supone que debe enlazar.

Por su parte, los ficheros de acción son:

**dirremove** Incluye los directorios que se desean eliminar.

El formato del fichero debe ser un directorio por cada línea.

**fileremove** Incluye los ficheros que se desean eliminar.

El formato del fichero debe ser un fichero por cada línea.

**changeperm** Incluye los permisos que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido de los permisos en formato octal de 7 dígitos.

**changeown** Incluye los propietarios que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido del identificador del usuario (UID).

**changegrp** Incluye los grupos propietarios que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido del identificador del grupo (GID).

**changemask** Incluye la máscara de permisos que se desea configurar en el sistema.

El formato debe ser una única línea con los permisos de la máscara en formato octal de 4 dígitos.

**filerestore** Incluye los ficheros que se desean restaurar en el sistema.

El formato debe ser el fichero fuente a restaurar seguido de un espacio y de la ruta de destino.

**commands** Incluye las órdenes que se desean ejecutar en el sistema.

El formato debe ser una línea por cada orden a ejecutar.

A continuación se exponen uno por uno el grupo de ejercicios, junto con sus puntos de control que dan lugar a sus correspondientes ficheros de control.

## Órdenes básicas

### CONTROL INICIAL

COMPROBAR FICHEROS EXISTENTES: antiguo/a.txt, antiguo/b.txt, antiguo/c.txt, antiguo/num/1.txt, antiguo/num/2.txt, antiguo/num/3.txt COMPROBAR DIRECTORIOS EXISTENTES: antiguo, antiguo/num FICHEROS A BORRAR: pwd.res, nuevo.res DIRECTORIOS A BORRAR: nuevo, doble
--

Cuadro 4.1: Órdenes básicas - Control inicial

1. Crea un directorio llamado **nuevo**.
2. Copia el fichero **b.txt** del directorio **antiguo** en el nuevo directorio que acabas de crear.
3. Duplica el fichero **b.txt** en la misma ruta pero con el nombre **bueno.txt**.
4. Copia el directorio **antiguo/num** al directorio **nuevo**.

### PRIMER CONTROL INTERMEDIO

COMPROBAR FICHEROS EXISTENTES: antiguo/a.txt, antiguo/b.txt, antiguo/c.txt, antiguo/num/1.txt, antiguo/num/2.txt, antiguo/num/3.txt, nuevo nuevo/num, nuevo/b.txt, nuevo/bueno.txt, nuevo/-num/1.txt, nuevo/num/2.txt, nuevo/num/3.txt COMPROBAR DIRECTORIOS EXISTENTES: antiguo, antiguo/num, nuevo, nuevo/num COMPROBAR FICHEROS IGUALES: nuevo/b.txt = antiguo/b.txt, nuevo/bueno.txt = antiguo/b.txt COMPROBAR DIRECTORIOS IGUALES: nuevo/num = antiguo/num
--

Cuadro 4.2: Órdenes básicas - Primer control intermedio

5. Borra el fichero **3.txt** dentro de **nuevo/num**.
6. Ve al directorio **nuevo/num**.
7. Ejecuta la orden **pwd** y guarda el resultado en el fichero **pwd.res** en el directorio inicial (**ordenes\_basicas**).
8. Vuelve al directorio **ordenes\_basicas** y crea un directorio **doble** con dos subdirectorios internos: **primero** y **segundo**.
9. Mueve el fichero **1.txt** de **nuevo/num** a **primero**.
10. Enlaza el fichero **enlacea2.txt** de **segundo** al fichero **2.txt** de **nuevo/num**.
11. Lista el directorio **nuevo** y guarda el resultado en el fichero **nuevo.res**.

### SEGUNDO CONTROL INTERMEDIO

COMPROBAR FICHEROS EXISTENTES: antiguo/a.txt, antiguo/b.txt, antiguo/c.txt, antiguo/num/1.txt, antiguo/num/2.txt, antiguo/num/3.txt, nuevo/b.txt, nuevo/bueno.txt, nuevo/num/2.txt, pwd.res, nuevo.res, doble/primero/1.txt, doble/segundo/enlacea2.txt

COMPROBAR DIRECTORIOS EXISTENTES: antiguo, antiguo/num, nuevo, nuevo/num, doble, doble/primero, doble/segundo

COMPROBAR FICHEROS ELIMINADOS: nuevo/num/3.txt

COMPROBAR FICHEROS IGUALES: pwd.res = .correccion/pwd.res, nuevo.res = .correccion/nuevo.res, doble/segundo/enlacea2.txt = .correccion/enlacea2.txt

Cuadro 4.3: Órdenes básicas - Segundo control intermedio

12. Borra todos los ficheros y directorios nuevos que has creado desde que empezaste los ejercicios:

- ◆ El directorio **nuevo**.
- ◆ El fichero **pwd.res**.
- ◆ El fichero **nuevo.res**.
- ◆ El directorio **doble**.

### CONTROL FINAL

COMPROBAR FICHEROS EXISTENTES: antiguo/a.txt, antiguo/b.txt, antiguo/c.txt, antiguo/num/1.txt, antiguo/num/2.txt, antiguo/num/3.txt

COMPROBAR DIRECTORIOS EXISTENTES: antiguo, antiguo/num

COMPROBAR FICHEROS ELIMINADOS: pwd.res, nuevo.res

COMPROBAR DIRECTORIOS ELIMINADOS: nuevo, doble

Cuadro 4.4: Órdenes básicas - Control final

## Sistema de archivos

### CONTROL INICIAL

COMPROBAR FICHEROS EXISTENTES: inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat  
 COMPROBAR DIRECTORIOS EXISTENTES: inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp  
 FICHEROS A BORRAR: foo.bar  
 MODIFICAR PROPIETARIO (UID = 1000): inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat  
 MODIFICAR GRUPO (GID = 1000): inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat  
 MODIFICAR PERMISOS (*rw**xr* – *xr* – –): inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat  
 MÁSCARA DE PERMISOS: 0022

Cuadro 4.5: Sistema de archivos - Control inicial

1. Consulta qué usuarios y grupos posee el directorio inicio y todo su contenido.
2. Consulta qué permisos posee el directorio inicio y todo su contenido.
3. Consulta cuál es la máscara de permisos actual.
4. Modifica la máscara de permisos para que tan sólo puedan leer y modificar ficheros el usuario y grupo propietarios.
5. Crea un archivo foo.bar en el directorio archivos y comprueba que la máscara del sistema funciona.

### PRIMER CONTROL INTERMEDIO

COMPROBAR FICHEROS EXISTENTES: inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat, foo.bar  
 COMPROBAR DIRECTORIOS EXISTENTES: inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp  
 COMPROBAR MÁSCARA DEL SISTEMA: 0018  
 COMPROBAR PERMISOS (*R*-*xR*-*xR*-*X*): foo.bar (*rw*–*r*–*r*–*–*), inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/tmp/root/passwd.dat

Cuadro 4.6: Sistema de archivos - Primer control intermedio

6. Cambia el usuario y el grupo propietarios del directorio inicio/root y todo su contenido por el administrador.

7. Cambia los permisos de los ficheros (únicamente los ficheros, no los directorios) que se encuentran en inicio y en inicio/valores para que todos los usuarios puedan leer y ejecutar los ficheros, pero ninguno pueda modificarlos.
8. Cambia el grupo propietario del directorio inicio/tmp y todo su contenido por www-data.
9. Dale permisos de lectura (únicamente) a todos los usuarios al fichero passwd.dat que se encuentra en inicio/tmp/root.
10. Cambiar el usuario propietario de los ficheros contenidos en inicio/valores por invitado y el grupo propietario por users.
11. Añade permisos de ejecución para el usuario propietario al fichero foo.bar.

### SEGUNDO CONTROL INTERMEDIO

COMPROBAR FICHEROS EXISTENTES: inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat, nuevo/b.txt, nuevo/bueno.txt, nuevo/num/2.txt, pwd.res, nuevo.res, doble/primero/1.txt, doble/segundo/enlacea2.txt

COMPROBAR DIRECTORIOS EXISTENTES: inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp, nuevo, nuevo/num, doble, doble/primero, doble/segundo

COMPROBAR FICHEROS ELIMINADOS: nuevo/num/3.txt

COMPROBAR FICHEROS IGUALES: pwd.res = correccion/pwd.res, nuevo.res = correccion/nuevo.res, doble/segundo/enlacea2.txt = correccion/enlacea2.txt

COMPROBAR PROPIETARIOS (UID): inicio/root/ (0), inicio/root/passwd.dat (0), inicio/valores/1.txt (1001), inicio/valores/2.txt (1001), inicio/valores/3.txt (1001)

COMPROBAR GRUPOS PROPIETARIOS (GID): inicio/root/ (0), inicio/root/passwd.dat (0), inicio/tmp (33), inicio/tmp/root (33), inicio/tmp/root/passwd.dat (33), inicio/valores/1.txt (100), inicio/valores/2.txt (100), inicio/valores/3.txt (100)

Cuadro 4.7: Sistema de archivos - Segundo control intermedio

12. Borra el fichero foo.bar que ya no es necesario.
13. Restaura el usuario y el grupo propietario para el directorio inicio y todo su contenido.
14. Restaura los permisos básicos para el directorio inicio y todo su contenido.
15. Restaura la máscara de permisos de usuario.

### CONTROL FINAL

COMPROBAR FICHEROS EXISTENTES: inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat

COMPROBAR DIRECTORIOS EXISTENTES: inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/root, inicio/valores/user, inicio/valores/grp

*Continúa en la siguiente página...*

*...continúa desde la página anterior.*

COMPROBAR FICHEROS ELIMINADOS: foo.bar COMPROBAR PROPIETARIOS (UID = 1000): inicio, inicio/root, inicio/tmp , inicio/valores, inicio/tmp/-root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat COMPROBAR GRUPOS (GID = 1000): inicio, inicio/root, inicio/tmp , inicio/valores, inicio/tmp/-root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat COMPROBAR PERMISOS: ( <i>rw</i> <i>xr</i> – <i>xr</i> – <i>x</i> ) inicio, inicio/root, inicio/tmp, inicio/valores, inicio/tmp/-root, inicio/valores/user, inicio/valores/grp, inicio/a.txt, inicio/b.txt, inicio/c.txt, inicio/valores/1.txt, inicio/valores/2.txt, inicio/valores/3.txt, inicio/root/passwd.dat, inicio/tmp/root/passwd.dat COMPROBAR MÁSCARA DEL SISTEMA: 0022
--

Cuadro 4.8: Sistema de archivos - Control final

## Órdenes avanzadas

### CONTROL INICIAL

#### ÓRDENES A EJECUTAR:

```
find /usr/share/doc -name "*.pdf" >.correccion/stats-pdf
find /usr/share/doc -name "*.txt" - size +100 >.correccion/stats-txt
egrep (d|D)evic /var/log/dmesg >.correccion/dispositivos.txt
grep eth /var/log/dmesg | grep -v driver >.correccion/red.txt
grep -i pci /var/log/dmesg >.correccion/pecei.txt
```

#### DIRECTORIOS A BORRAR: prueba

FICHEROS A BORRAR: stats-pdf, stats-txt, backup.tgz, /tmp/dispositivos.txt, /tmp/red.txt, /tmp/pecei.txt

Cuadro 4.9: Órdenes avanzadas - Control inicial

1. Deseamos hacer una contabilidad de la información de ayuda incluida en nuestro sistema dentro del directorio /usr/share/doc.

Para ello almacene en un fichero llamado stats-pdf, el nombre de todos los ficheros con extensión pdf que estén ubicados dentro de él.

Realice la operación recursivamente.

2. Después almacene en un fichero llamado stats-txt en el nombre de todos los ficheros con extensión txt que ocupen más de 100 bytes que estén ubicados dentro de él.

Realice la operación recursivamente.

### PRIMER CONTROL INTERMEDIO

#### ÓRDENES A EJECUTAR:

```
find /usr/share/doc -name "*.pdf" >.correccion/stats-pdf
find /usr/share/doc -name "*.txt" - size +100 >.correccion/stats-txt
egrep (d|D)evic /var/log/dmesg >.correccion/dispositivos.txt
grep eth /var/log/dmesg | grep -v driver >.correccion/red.txt
grep -i pci /var/log/dmesg >.correccion/pecei.txt
```

COMPROBACIÓN DE FICHEROS EXISTENTES: stats-pdf, .correccion/stats-pdf, stats-txt, .correccion/stats-txt

COMPROBACIÓN DE FICHEROS IGUALES: stats-pdf= .correccion/stats-pdf, stats-txt = .correccion/stats-txt

Cuadro 4.10: Órdenes avanzadas - Primer control intermedio

3. También sería bueno tener una copia de todos los ficheros alojados en /usr/share/doc/lsb-release. Hágala con tar+gz y almacene el resultado en un fichero llamado backup.tgz dentro del directorio actual.

4. Después restaure la información en un directorio llamado "prueba".





## SEGUNDO CONTROL INTERMEDIO

## ÓRDENES A EJECUTAR:

```
find /usr/share/doc -name "*.pdf" >.correccion/stats-pdf
find /usr/share/doc -name "*.txt" - size +100 >.correccion/stats-txt
egrep (d|D)evic /var/log/dmesg >.correccion/dispositivos.txt
grep eth /var/log/dmesg | grep -v driver >.correccion/red.txt
grep -i pci /var/log/dmesg >.correccion/pecei.txt
```

COMPROBACIÓN DE DIRECTORIOS EXISTENTES: prueba

COMPROBACIÓN DE FICHEROS EXISTENTES: backup.tgz .correccion/backup.tgz

COMPROBACIÓN DE DIRECTORIOS IGUALES: prueba = .correccion/prueba

COMPROBACIÓN DE FICHEROS IGUALES: backup.tgz = .correccion/backup.tgz

Cuadro 4.11: Órdenes avanzadas - Segundo control intermedio

5. Queremos analizar el proceso de arranque de nuestro sistema. Para ello analizaremos la información en /var/log/dmesg y generaremos ficheros que lo resuman en el directorio tmp de la raíz del sistema de ficheros:

- ◆ Copie todas las líneas que contengan Device o device a un fichero llamado "dispositivos.txt".
- ◆ Almacene todas las líneas que tienen la cadena eth pero no la palabra driver en red.txt"
- ◆ Almacene todas las líneas que tengan la cadena PCI en cualquier combinación de mayúsculas y minúsculas en "pecei.txt"

## CONTROL FINAL

## ÓRDENES A EJECUTAR:

```
find /usr/share/doc -name "*.pdf" >.correccion/stats-pdf
find /usr/share/doc -name "*.txt" - size +100 >.correccion/stats-txt
egrep (d|D)evic /var/log/dmesg >.correccion/dispositivos.txt
grep eth /var/log/dmesg | grep -v driver >.correccion/red.txt
grep -i pci /var/log/dmesg >.correccion/pecei.txt
```

COMPROBACIÓN DE FICHEROS EXISTENTES: .correccion/dispositivos.txt, /tmp/dispositivos.txt, .correccion/red.txt, /tmp/red.txt, .correccion/pecei.txt, /tmp/pecei.txt

COMPROBACIÓN DE FICHEROS IGUALES: .correccion/dispositivos.txt = /tmp/dispositivos.txt, .correccion/red.txt = /tmp/red.txt, .correccion/pecei.txt = /tmp/pecei.txt

Cuadro 4.12: Órdenes avanzadas - Control final

## Arranque y parada del sistema

### CONTROL INICIAL

ÓRDENES A EJECUTAR:

```
ls -la etc/init.d/rc4.d >.correccion/rc4
find etc/init.d/rc4.d/ -name "*cupsexec rm
find etc/init.d/rc4.d/ -name "*pcmciautilsexec rm
find etc/init.d/rc4.d/ -name "*anacronexec rm
```

FICHEROS A RESTAURAR: .correccion/inittab → etc/inittab

Cuadro 4.13: Arranque y parada del sistema - Control inicial

1. Compruebe cuál es el modo de arranque por defecto de su sistema.
2. Deseamos configurar el nivel de ejecución 4 para pruebas del sistema. Por eso deseamos que en dicho nivel:
  - ◆ No se ejecuten los servicios “cups” ni “pcmciautils”
  - ◆ El servicio “anacron” se ejecute inmediatamente después de la ejecución de “atd” (no modifique “atd” para ello, sólo “anacron”)

### PRIMER CONTROL INTERMEDIO

ÓRDENES A EJECUTAR:

```
ls -la etc/init.d/rc4.d >.correccion/rc4
```

VERIFICAR CONTENIDO DE LOS FICHEROS: .correccion/rc4 (K.\*cups), .correccion/rc4 (K.\*pcmciautils), .correccion/rc4 (S25.\*anacron)

Cuadro 4.14: Arranque y parada del sistema - Primer control intermedio

3. Establezca el nivel de ejecución 4 como modo por defecto del sistema.
4. Asegúrese de que al pulsar *Ctrl+Alt+Del* en dicho nivel no se reinicie el sistema, sino que se almacena la hora de pulsación en el fichero */var/log/ctrl.log* (cree este fichero si no existe y asegúrese de que sólo puede leerlo o modificarlo el root).

### CONTROL FINAL

ÓRDENES A EJECUTAR:

```
ls -la etc/init.d/rc4.d >.correccion/rc4
```

VERIFICAR CONTENIDO DE LOS FICHEROS: etc/inittab (initdefault.\*4), etc/inittab (ctrlalt-del.\*date.\*var/log/ctrl.log)

COMPROBAR PERMISOS (*rwx* ----): var/log/ctrl.log

COMPROBAR PROPIETARIOS (UID = 0): var/log/ctrl.log

*Continúa en la siguiente página...*

...continúa desde la página anterior.

Cuadro 4.15: Arranque y parada del sistema - Control final

## Administración de usuarios y grupos

### CONTROL INICIAL

DIRECTORIOS A BORRAR: /srv/development, /etc/skel/repos FICHEROS A RESTAURAR: .correction/restore/passwd → /etc/passwd, .correction/restore/group → /etc/group, .correction/restore/shadow → /etc/shadow MODIFICAR PERMISOS ( $r-xr--r--$ ): /etc/passwd, /etc/group, /etc/shadow MODIFICAR PROPIETARIOS (UID = 0): /etc/passwd, /etc/group, /etc/shadow MODIFICAR GRUPOS (GID = 0): /etc/passwd, /etc/group, /etc/shadow
---

Cuadro 4.16: Usuarios y grupos - Control inicial

1. Cree un grupo de usuarios denominado “progs”.
2. Cree un usuario llamado “juan” cuyo grupo primario sea “progs”.
3. Cree un grupo de usuarios denominado “externos”.
4. Cree un usuario llamado “luis” cuyo grupo primario sea “externos” y grupo secundario “progs”.

### PRIMER CONTROL INTERMEDIO

VERIFICAR CONTENIDO EN LOS FICHEROS: /etc/passwd (juan*progs), /etc/passwd (luis*externos), /etc/group (progs*luis)
---

Cuadro 4.17: Órdenes avanzadas - Primer control intermedio

5. Cree un directorio llamado “desarrollo” en /srv (compruebe si existe este directorio en la máquina virtual)
6. Ponga los permisos adecuados para que luis pueda leer, escribir y acceder a él, pero que juan sólo pueda acceder y leer su contenido.
7. Queremos que a partir de ahora todas las cuentas de usuario que se creen en el sistema tengan un directorio llamado “repos”.

### SEGUNDO CONTROL INTERMEDIO

COMPROBAR DIRECTORIOS EXISTENTES: /srv/development, /etc/skel/repos COMPROBAR PERMISOS ( $rwxr-x---$ ): /srv/development COMPROBAR PROPIETARIOS (UID = 1003): /etc/development COMPROBAR GRUPOS (GID = 1002): /etc/development
---

Cuadro 4.18: Usuarios y grupos - Segundo control intermedio

8. Asegúrese de que todos los usuarios cambian su contraseña cada 30 días como máximo.

9. Cambie la contraseña del usuario luis, ponga como contraseña "SuJh\_4".
10. Cambie el shell por defecto de juan, que use /bin/zsh.

### CONTROL FINAL

VERIFICAR CONTENIDO EN LOS FICHEROS: /etc/shadow (luis:!!:!:30:!!:!:!), /etc/shadow (juan:!!:!:30:!!:!:!), /etc/passwd (juan:*/bin/zsh)
---

Cuadro 4.19: Usuarios y grupos - Control final

### 4.6.3. Estadísticas

Una vez terminada la parte de codificación, se utilizó *StatMediaWiki* [37] para extraer estadísticas acerca de las ediciones realizadas en el wiki desde su instalación (en el mes de Junio de 2009) hasta la fecha de realización de esta memoria (Febrero de 2010).<sup>5</sup>

A continuación, se muestran los datos más interesantes tanto de estas estadísticas como de las proporcionadas en las páginas especiales de *MediaWiki*.

#### Información general

- ◆ Sitio web: [WikiUNIX](#)
- ◆ Fecha de generación de las estadísticas: 8 de Febrero de 2010
- ◆ Periodo: 1 de Junio de 2009 - 28 de Febrero de 2010
- ◆ Número total de páginas: 375
- ◆ Número de artículos: 227
- ◆ Número total de ficheros: 35
- ◆ Usuarios: 12

#### Ediciones y visualizaciones

- ◆ Ediciones en páginas desde que WikiUNIX fue instalado: 1.975
- ◆ Ediciones en artículos: 1.234
- ◆ Media de ediciones por página: 5,27
- ◆ Visitas totales: 11.445
- ◆ Visitas por edición: 6,16
- ◆ Página más vista: *Página principal* - 1.832 ediciones

#### Contenido

En gráfica 4.22 se muestra la evolución del contenido del wiki.

---

<sup>5</sup>Se pueden visualizar las colaboraciones en diversos sistemas soportados por MediaWiki en el anexo de la página 133.

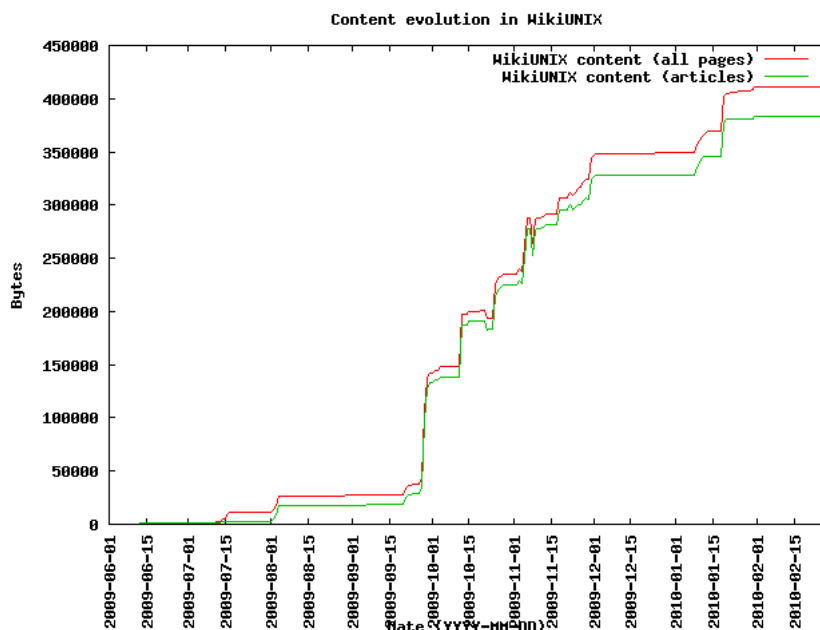


Figura 4.22: Evolución del contenido

## Páginas

Hay un total de 292 páginas (sumado contenido real y redirecciones).

A continuación se muestra el listado total de páginas de contenido real:

- |  |   |
|--|---|
| 1. /etc/group  | 2. /etc/passwd  |
| 3. /etc/shadow   | 4. /home  |
| 5. 2009-07-13 Añadidas nuevas plantillas               | 6. 2009-08-01 Recuperada la wiki                        |
| 7. 2009-10-24 Primeros scripts                         | 8. 2009-10-24 Primeros scripts subidos... <sup>6</sup>  |
| 9. 2009-11-03 Wiki migrada                             | 10. 2009-11-03 Wiki migrada al servidor... <sup>7</sup> |
| 11. 2010-02-13 Descargas habilitadas                   | 12. AWK   |
| 13. Acceso servidor utilizando SSH                     | 14. Admon. del sistema de archivos                      |
| 15. Admon. sistema de archivos/Ejercicios <sup>8</sup> | 16. Admon. sistema de archivos/Ejemplos <sup>9</sup>    |
| 17. Administrador                                      | 18. Agregar maquina.sh                                  |
| 19. Anacron  | 20. Apropos   |
| 21. Ar   | 22. Arch  |
| 23. Arranque   | 24. Arranque y parada del sistema                       |
| 25. Arranque y parada del sistema/Ejercicios           | 26. Asignación de cuotas de disco                       |
| 27. At   | 28. Atq   |
| 29. Atrm   | 30. Awk   |

<sup>6</sup>El nombre real de este artículo es "2009-10-24 Primeros scripts subidos a la wiki"

<sup>7</sup>El nombre real de este artículo es "2009-11-03 Wiki migrada al servidor de la OSLUCA"

<sup>8</sup>El nombre real de este artículo es "Administración del sistema de archivos/Ejercicios"

<sup>9</sup>El nombre real de este artículo es "Administración del sistema de archivos/Ejemplos"



- 
- |                                       |  |
|---------------------------------------|--|
| 31. Ayuda                             | 32. Ayuda en UNIX                            |
| 33. Backup-remoto.sh                  | 34. Backup.py                                |
| 35. Backup.sh                         | 36. Banner                                   |
| 37. Basename                          | 38. Bash                                     |
| 39. Batch                             | 40. Bunzip2                                  |
| 41. Bzip2                             | 42. Cal                                      |
| 43. Caracteres comodines              | 44. Características                          |
| 45. Carácter comodín                  | 46. Cat                                      |
| 47. Cd                                | 48. Cdp                                      |
| 49. Chage                             | 50. Chext.py                                 |
| 51. Chfn                              | 52. Chgrp                                    |
| 53. Chmod                             | 54. Chown                                    |
| 55. Chpasswd                          | 56. Chroot                                   |
| 57. Chsh                              | 58. Chvt                                     |
| 59. Clasificación                     | 60. Clear                                    |
| 61. Cmp                               | 62. Col                                      |
| 63. Colrm                             | 64. Column                                   |
| 65. Comm                              | 66. Conceptos básicos                        |
| 67. Conceptos básicos/Ejercicios      | 68. Consola                                  |
| 69. Control.py                        | 70. Copias de seguridad                      |
| 71. Copias de seguridad automatizadas | 72. Cp                                       |
| 73. Cpio                              | 74. Cron                                     |
| 75. Crond                             | 76. Csplit                                   |
| 77. Cut                               | 78. Date                                     |
| 79. Dd                                | 80. Deallocvt                                |
| 81. Deamon                            | 82. Demonio                                  |
| 83. Df                                | 84. Diff                                     |
| 85. Dir                               | 86. Directorio Padre                         |
| 87. Directorio casa                   | 88. Directorio de entrada                    |
| 89. Directorio de inicio              | 90. Directorio de trabajo                    |
| 91. Directorio oculto                 | 92. Directorio padre                         |
| 93. Directorio raíz                   | 94. Directorios casa                         |
| 95. Dirname                           | 96. Du                                       |
| 97. Echo                              | 98. Ed                                       |
| 99. Egrep                             | 100. Ejecutar script automáticamente         |
| 101. Env                              | 102. Espacio montado.sh                      |
| 103. Estado maquinas.sh               | 104. Estructura                              |
| 105. Estructura/Ejercicios            | 106. FHS                                     |
| 107. Familia UNIX                     | 108. Fgrep                                   |
| 109. Fichero oculto                   | 110. File                                    |
| 111. Find                             | 112. Finger                                  |
| 113. Free                             | 114. GRUB                                    |
| 115. Gawk                             | 116. Gestionar espacio de swap <sup>10</sup> |
| 117. Grep                             | 118. Groupadd                                |
| 119. Groupdel                         | 120. Groupmod                                |
| 121. Groups                           | 122. Gunzip                                  |

---

<sup>10</sup>El nombre real de este artículo es "Gestionar espacio de memoria de intercambio"

123. Gzip	124. Halt
125. Head	126. Help
127. History	128. Histórico del sistema
129. Horas.sh	130. Hostname
131. Id	132. Identificador de proceso
133. Igawk	134. Info
135. Init	136. Instalación de software
137. Instalación de un servidor FTP con vsftpd	138. Ipreal.py
139. Jerarquía de directorios	140. Join
141. Kernel	142. Kill
143. LILO	144. Last
145. Latex.sh	146. Ldd
147. Less	148. Liberar memoria cache
149. Ln	150. Login
151. Logname	152. Look
153. Ls	154. Mail
155. Mailx	156. Main Page
157. Malas prácticas en un sistema UNIX	158. Malas prácticas y consejos útiles
159. Man	160. Manpath
161. Marca UNIX	162. Mc
163. Mesg	164. Mev
165. Mkdir	166. Mknod
167. Modos de funcionamiento	168. Monitorización de recursos
169. More	170. Mv
171. Namei	172. Newgrp
173. Newusers	174. Nice
175. Nivel:Principiante	176. Nivel:Unixero profesional
177. Nivel:Usuario habitual	178. Nivel:Usuario iniciado
179. Nl	180. Nohup
181. Od	182. Optimizar el sistema de archivos ext3
183. Ordenes basicas	184. Organizar.py
185. PID	186. Padre
187. Passwd	188. Paste
189. Portada	190. Portada:Avisos
191. Poweroff	192. Pr
193. Printenv	194. Prompt
195. Ps	196. Pwd
197. Pygments script.py	198. Página Principal
199. Página principal	200. RFC
201. Reboot	202. Red
203. Referencias	204. Registros del sistema
205. Reset	206. Rev
207. Rm	208. Rmail
209. Rmdir	210. Root
211. Ruta absoluta	212. Ruta relativa
213. Rutas absolutas	214. Rutas relativas
215. Script	216. Sed
217. Seguridad	218. Setterm

---

219. Sg	220. Sh
221. Shell	222. Shell Bash
223. Shutdown	224. Sistema de archivos
225. Sistema de archivos/Ejercicios	226. Sistema operativo UNIX
227. Skill	228. Sleep
229. Sln	230. Soporte para pruebas
231. Sort	232. Split
233. Stty	234. Su
235. Sudo	236. Sum
237. Superusuario	238. Syslogd
239. Tail	240. Tar
241. Tee	242. Telinit
243. Terminal	244. Test
245. Time	246. Top
247. Touch	248. Tr
249. Trabajando en la consola	250. Trap
251. Tty	252. UNIX
253. Umask	254. Uname
255. Uniq	256. Unix
257. Unzip	258. Useradd
259. Userdel	260. Usermod
261. Users	262. Usuarios y grupos
263. Usuarios y grupos/Ejercicios	264. VI
265. Vdir	266. Vi
267. Vim	268. Vivo.py
269. W	270. Wall
271. Watch	272. Wc
273. Whatis	274. Whereis
275. Who	276. Whoami
277. Write	278. Xargs
279. Yes	280. Zcat
281. Zcmp	282. Zdiff
283. Zforce	284. Zip
285. Zmore	286. Znew
287. Zona de pruebas	288. Órdenes
289. Órdenes/Ejemplos	290. Órdenes/Ejercicios
291. Órdenes avanzadas	292. Órdenes básicas
293. Órdenes básicas/Ejercicios	

Y este es el listado de redirecciones (páginas que no poseen contenido real, sino que envían al usuario a otra página):

1. Main Page → Página principal
2. Directorio de entrada → Directorio casa
3. Directorio oculto → Fichero oculto
4. Portada → Página Principal
5. Malas prácticas en un sistema UNIX → Malas prácticas y consejos útiles

6. Arranque → Arranque y parada del sistema
7. 2009-11-03 Wiki migrada al servidor de la OSLUCA → 2009-11-03 Wiki migrada
8. 2009-10-24 Primeros scripts subidos a la wiki → 2009-10-24 Primeros scripts
9. Ayuda → Ayuda en UNIX
10. VI → Vi
11. Plantilla:Ficha de articulo → Plantilla:Ficha de artículo
12. PID → Identificador de proceso
13. Awk → AWK
14. Página Principal → Página principal
15. Superusuario → Administrador
16. Mailx → Mail
17. Consola → Terminal
18. Ordenes basicas → Órdenes básicas
19. Directorio de inicio → Directorio casa
20. Egrep → Grep
21. Fgrep → Grep
22. Wikiunix:Acerca de → Wikiunix:Portal de la comunidad
23. Bunzip2 → Bzip2
24. Unzip → Zip
25. Padre → Directorio padre
26. Directorio Padre → Directorio padre
27. FHS → Jerarquía de directorios
28. Directorios casa → Directorio casa
29. Rutas absolutas → Ruta absoluta
30. Rutas relativas → Ruta relativa
31. /etc/passwd → Usuarios y grupos
32. /etc/group → Usuarios y grupos
33. /etc/shadow → Usuarios y grupos
34. Ed → Red
35. Caracteres comodines → Carácter comodín
36. Shell Bash → Bash
37. UNIX → Unix
38. Deamon → Demonio
39. Crond → Cron
40. Sysklogd → Registros del sistema
41. Histórico del sistema → Registros del sistema
42. Modos de funcionamiento → Arranque y parada del sistema
43. Categoría:Órdenes → Categoría:Programas
44. Órdenes → Órdenes avanzadas
45. LILO → Arranque y parada del sistema
46. GRUB → Arranque y parada del sistema
47. Vim → Vi

## Categorías

- ◆ Conceptos básicos (13 artículos)

- 
- |                                      |                             |
|--------------------------------------|-----------------------------|
| 1. Administrador                     | 2. Ayuda en UNIX            |
| 3. Directorio casa                   | 4. Directorio de trabajo    |
| 5. Directorio padre                  | 6. Directorio raíz          |
| 7. Fichero oculto                    | 8. Identificador de proceso |
| 9. Malas prácticas y consejos útiles | 10. Ruta absoluta           |
| 11. Ruta relativa                    | 12. Terminal                |
| 13. Unix                             |                             |

◆ Control y pruebas (2 artículos)

- ➔ control.py
- ➔ Soporte para pruebas

◆ Curiosidades (10 artículos)

1. Acceso servidor utilizando SSH
2. Asignación de cuotas de disco
3. Copias de seguridad automatizadas
4. Ejecutar script automáticamente
5. Gestionar espacio de memoria de intercambio
6. Liberar memoria cache
7. Malas prácticas y consejos útiles
8. Monitorización de recursos
9. Optimizar el sistema de archivos ext3
10. Trabajando en la consola

◆ Desambiguación (2 artículos)

- ➔ Root
- ➔ Unix

◆ Ejercicios (7 artículos)

- ➔ Administración del sistema de archivos/Ejercicios
- ➔ Arranque y parada del sistema/Ejercicios
- ➔ Conceptos básicos/Ejercicios
- ➔ Sistema de archivos/Ejercicios
- ➔ Usuarios y grupos/Ejercicios
- ➔ Órdenes básicas/Ejercicios
- ➔ Órdenes/Ejercicios

◆ Formatear (0 artículos)

◆ Noticias (5 artículos)

- ➔ Wikiunix:Actualidad

- ➔ 2009-07-13 Añadidas nuevas plantillas
- ➔ 2009-08-01 Recuperada la wiki
- ➔ 2009-10-24 Primeros scripts
- ➔ 2009-11-03 Wiki migrada
- ➔ 2010-02-13 Descargas habilitadas

◆ Programas (157 artículos)

- |              |              |               |
|--------------|--------------|---------------|
| 1. Apropos   | 2. Ar        | 3. Arch       |
| 4. At        | 5. Atq       | 6. Atrm       |
| 7. Banner    | 8. Basename  | 9. Batch      |
| 10. Bzip2    | 11. Cal      | 12. Cat       |
| 13. Cd       | 14. Cdp      | 15. Chage     |
| 16. Chfn     | 17. Chgrp    | 18. Chmod     |
| 19. Chown    | 20. Chpasswd | 21. Chroot    |
| 22. Chsh     | 23. Chvt     | 24. Clear     |
| 25. Cmp      | 26. Col      | 27. Colrm     |
| 28. Column   | 29. Comm     | 30. Cp        |
| 31. Cpio     | 32. Csplit   | 33. Cut       |
| 34. Date     | 35. Dd       | 36. Dealocvt  |
| 37. Df       | 38. Diff     | 39. Dir       |
| 40. Dirname  | 41. Du       | 42. Echo      |
| 43. Ed       | 44. Env      | 45. File      |
| 46. Find     | 47. Finger   | 48. Free      |
| 49. Gawk     | 50. Grep     | 51. Groupadd  |
| 52. Groupdel | 53. Groupmod | 54. Groups    |
| 55. Gunzip   | 56. Gzip     | 57. Halt      |
| 58. Head     | 59. Help     | 60. History   |
| 61. Hostname | 62. Id       | 63. Igawk     |
| 64. Info     | 65. Init     | 66. Join      |
| 67. Kill     | 68. Last     | 69. Ldd       |
| 70. Less     | 71. Ln       | 72. Login     |
| 73. Logname  | 74. Look     | 75. Ls        |
| 76. Mail     | 77. Man      | 78. Manpath   |
| 79. Mc       | 80. Mesg     | 81. Mev       |
| 82. Mkdir    | 83. Mknod    | 84. More      |
| 85. Mv       | 86. Namei    | 87. Newgrp    |
| 88. Newusers | 89. Nice     | 90. Nl        |
| 91. Nohup    | 92. Od       | 93. Passwd    |
| 94. Paste    | 95. Poweroff | 96. Pr        |
| 97. Printenv | 98. Ps       | 99. Pwd       |
| 100. Reboot  | 101. Red     | 102. Reset    |
| 103. Rev     | 104. Rm      | 105. Rmail    |
| 106. Rmdir   | 107. Script  | 108. Setterm  |
| 109. Sg      | 110. Sh      | 111. Shutdown |
| 112. Skill   | 113. Sleep   | 114. Sln      |
| 115. Sort    | 116. Split   | 117. Stty     |

118. Su	119. Sudo	120. Sum
121. Tail	122. Tar	123. Tee
124. Telinit	125. Test	126. Time
127. Top	128. Touch	129. Tr
130. Trap	131. Tty	132. Umask
133. Uname	134. Uniq	135. Useradd
136. Userdel	137. Usermod	138. Users
139. Vdir	140. W	141. Wall
142. Watch	143. Wc	144. Whatis
145. Whereis	146. Who	147. Whoami
148. Write	149. Xargs	150. Yes
151. Zcat	152. Zcmp	153. Zdiff
154. Zforce	155. Zip	156. Zmore
157. Znew		

◆ Scripts (14 artículos)

1. agregar_maquina.sh	2. backup-remoto.sh
3. backup.py	4. backup.sh
5. chext.py	6. control.py
7. espacio_montado.sh	8. estado_maquinas.sh
9. horas.sh	10. ipreal.py
11. latex.sh	12. organizar.py
13. pygments_script.py	14. vivo.py

◆ ToDo (1 artículos)

- ➔ Administración del sistema de archivos/Ejercicios
- ➔ Wikiunix:Derechos de autor
- ➔ Wikiunix:Limitación general de responsabilidad
- ➔ Wikiunix:Política de protección de datos
- ➔ Wikiunix:Portal de la comunidad

## Usuarios

Usuario	Ediciones	Ed. Artículos	Bytes añadidos	Bytes en artículos	Subidas
Nessa	1265 (80.37 %)	1080 (87.52 %)	632068 (90.78 %)	612380 (96.88 %)	34
213.96.160.52	114 (7.24 %)	29 (2.35 %)	23759 (3.41 %)	916 (0.14 %)	0
95.61.154.27	61 (3.88 %)	32 (2.59 %)	12075 (1.73 %)	2062 (0.33 %)	0
84.122.51.73	33 (2.10 %)	9 (0.73 %)	5922 (0.85 %)	119 (0.02 %)	0
Manuel.palomo	32 (2.03 %)	32 (2.59 %)	7719 (1.11 %)	7719 (1.22 %)	0
Emijrp	23 (1.46 %)	14 (1.13 %)	1616 (0.23 %)	54 (0.01 %)	0
79.108.33.78	14 (0.89 %)	14 (1.13 %)	2321 (0.33 %)	2321 (0.37 %)	0
84.122.16.248	9 (0.57 %)	6 (0.49 %)	8237 (1.18 %)	4270 (0.68 %)	0
continúa en la siguiente página...					

<i>continúa desde la página anterior...</i>					
Usuario	Ediciones	Ed. Artículos	Bytes añadidos	Bytes en artículos	Subidas
Fsero	7 (0.44 %)	4 (0.32 %)	1491 (0.21 %)	1166 (0.18 %)	0
127.0.0.1	5 (0.32 %)	5 (0.41 %)	249 (0.04 %)	249 (0.04 %)	0
0:0:0:0:0:0:1	4 (0.25 %)	4 (0.32 %)	63 (0.01 %)	63 (0.01 %)	0
84.122.27.171	3 (0.19 %)	3 (0.24 %)	255 (0.04 %)	255 (0.04 %)	0
Noelia	2 (0.13 %)	0 (0.00 %)	0 (0.00 %)	0 (0.00 %)	1
MediaWiki default	1 (0.06 %)	1 (0.08 %)	494 (0.07 %)	494 (0.08 %)	0
84.122.21.159	1 (0.06 %)	1 (0.08 %)	29 (0.00 %)	29 (0.00 %)	0

Cuadro 4.20: Usuarios de WikiUNIX

Cabe destacar que la mayoría de las IP's incluidas como usuarios son de las primeras pruebas, realizadas en el equipo personal de la autora.



## 4.7. Pruebas y validación

Un buen plan de pruebas es una piedra angular en el desarrollo de un proyecto. Nos permite desarrollar un producto de mejor calidad que si no estableciéramos estos requisitos de pruebas. La planificación de pruebas es especial para este proyecto. Tenemos tres elementos en el proyecto: el motor wiki, el contenido teórico y el soporte para pruebas.

Lógicamente, no tenemos que realizar prueba alguna sobre el motor wiki *MediaWiki* ni sobre *VirtualBox*, ya que el proceso de desarrollo de estas herramientas garantizan la fiabilidad del sistema. Por tanto, no tiene sentido que desarrollemos un plan de pruebas para esta herramientas. Para el contenido se ha establecido un plan de pruebas, mejora y mantenimiento específico. En este caso ambas tareas serán llevadas a cabo conjuntamente.

### 4.7.1. Descripción de Aspectos Generales

El temario del tutorial está compuesto de un desarrollo teórico y una serie de ejercicios y ejemplos. El plan de pruebas del desarrollo teórico se define estableciendo unos hitos periódicos que nos obligan a revisar el contenido del proyecto. El revisar el contenido del proyecto tiene varias fases.

La primera fase es sintáctica: revisamos que el tutorial no tenga errores ortográficos; mientras que la segunda fase es semántica, centrándonos en cumplir los criterios de calidad que hemos establecido sobre el temario. En caso de que el desarrollo no cumpla con los mínimos exigidos pasaremos a una tarea de mejora del producto. Replanteamos el diseño de la temática concreta que estemos revisando y volvemos a establecer el tema con el nuevo contenido.

El plan de pruebas para los ejercicios, los ejemplos y el soporte de comprobación será idéntico para todos por tratarse del mismo sistema parametrizado.

### 4.7.2. Objetivos

El objetivo de los planes de prueba es controlar que se cumplen los compromisos de calidad adquiridos en la planificación de este producto.

### 4.7.3. Entorno

Se realizarán pruebas a aquellos varios tipos de elementos.

- ◆ Se revisará todo el contenido del tutorial en busca de errores.
- ◆ Se probarán aquellos listados que sean complejos.

#### 4.7.4. Proceso de prueba

##### Cuestión 1: La estructura

*¿La estructura de los artículos principales cumple lo acordado inicialmente? ¿Es suficiente con los artículos incluidos?*

Sí, el contenido teórico principal del wiki mantiene la estructura planteada al principio del proyecto. Los artículos también coinciden.

##### Cuestión 2: Las plantillas

*¿Se visualizan correctamente las plantillas desarrolladas? ¿Cumplen la función para la cual se han diseñado? ¿Ayudan al usuario a acceder a diversas secciones del wiki?*

Sí, todas las plantillas visuales<sup>11</sup> han sido comprobadas en diversos navegadores (Firefox, Opera, Chrome,...). Cumplen su funcionalidad y están enfocadas a la simplicidad y a mejorar la navegación del usuario.

##### Cuestión 3: Los artículos

*¿Existen páginas huérfanas “reales”? ¿Hay páginas a las cuales no se puede acceder a través de la página principal?*

En MediaWiki el concepto de “artículo huérfano” es aquel que no esté enlazado con otros artículos. Para nosotros es aquel artículo al cual no se pueda acceder a menos que sea a través de su URL.

En WikiUNIX no hay páginas huérfanas: todos los artículos son accesibles a través de uno o dos enlaces desde la página principal.

##### Cuestión 4: Las categorías

*¿Al realizar una búsqueda por categoría aparecen los artículos correctamente?*

Sí, todas las búsquedas funcionan correctamente.

##### Cuestión 5: El sistema de control

*¿Se puede importar la máquina virtual? ¿Este sistema virtual actualiza el sistema de control de ejercicios automáticamente?*

---

<sup>11</sup>No todas las plantillas se ven en la pantalla: algunas tan sólo simplifican el uso de otras plantillas.

Sí, la máquina virtual puede importarse en sistemas actualizados<sup>12</sup> con unas características básicas, debido a que el entorno XFCE es bastante ligero. Además se ha modificado la máquina virtual para que se actualice automáticamente al iniciarse la sesión<sup>13</sup>.

#### **Cuestión 6: Los scripts de control**

*¿Verifican los scripts los ejercicios para los cuales están diseñados? ¿Limpian el sistema antes de iniciar el desarrollo de los ejercicios? ¿Verifican todos los controles intermedios además del final?*

Sí, los *scripts* funcionan correctamente. Verifican que se han realizado bien los ejercicios e informan de los errores.

#### **Cuestión 7: Ampliabilidad**

*¿El contenido teórico es fácilmente modificable y ampliable? ¿Pueden desarrollarse más ejercicios? ¿Y más sistemas de control para éstos?*

Sí, el contenido teórico, gracias al motor MediaWiki, es modificable fácilmente. Además la corrección de los ejercicios es ampliable incluso para usuarios que no posean conocimientos de programación.

#### **4.7.5. Documentación de los casos de prueba**

No se han registrado incidentes en la realizaciones de las pruebas sobre el software desarrollado como material del proyecto. Durante la revisión del contenido tampoco se han producido incidentes que hayan afectado a las pruebas.

En cuanto al histórico de pruebas vamos a distinguir, de nuevo entre la revisión del contenido teórico y la prueba de software.

La teoría ha sido revisada encontrando multitud de errores de sintaxis en sus primeras versiones. No hemos creído conveniente el crear un histórico con estos detalles, ya que se trata de errores no sistemáticos que no tienen interés alguno su almacenamiento.<sup>14</sup>

En cuanto al software el caso es similar. Después de numerosas pruebas no se han encontrado problemas en el software. El software no tiene operaciones críticas.

Realizamos pruebas con nuestro producto como por ejemplo:

- I. Hemos modificado el contenido del temario comprobando que las operaciones se realizaban correctamente.

<sup>12</sup>Como se ha comentado anteriormente, se necesita que la versión de VirtualBox sea superior a la 2.2, aunque es preferible que sea superior a la 3.0.

<sup>13</sup>Para la actualización se necesita tener conexión a internet.

<sup>14</sup>En la sección de mantenimiento (página 83 se encuentra una descripción completa acerca de cómo se solucionan estos problemas.)

- II. Hemos intentado modificar páginas que estaban protegidas y como era de esperar no hemos podido llevar a cabo esta operación.
- III. Las conexiones entre los distintos elementos del sistema han sido establecidas y probadas.
- IV. Se comprueba el acceso desde distintos navegadores.
- V. Las búsquedas y en tiempo de presentación se realizan dentro de un tiempo adecuado.

## 4.8. Mantenimiento

Dentro del soporte wiki tenemos actualmente una colección de información controlada. Sin embargo, precisamente por ser completamente abierta y modificable, esta puede crecer sobremanera en poco tiempo.

Ante la posibilidad de una gran cantidad de nuevas aportaciones que haya que controlar, podemos encontrarnos con dos vertientes para que el proyecto siga teniendo la misma calidad que actualmente: el mantenimiento manual y el automático. No hay que escoger uno u otro: es preferible complementar ambos para poder asegurar que el sistema se encuentra en buen estado.

### 4.8.1. Manual

El mantenimiento manual es el que deben realizar los administradores del wiki a la hora de controlar las nuevas ediciones realizadas en el sistema. *MediaWiki* proporciona una gran cantidad de información y estadísticas tomada del control que el propio sistema procesa a través de las modificaciones de los usuarios.

Los administradores deberán controlar los **cambios recientes** realizados en el sistema y, según su criterio, decidir si esas modificaciones son válidas. En caso de que observen alguna anomalía en alguna edición determinada, tan sólo deben revertirla, para devolver el artículo a la versión anterior que se suponía correcta.

### 4.8.2. Automático: Uso de bots

El trabajo del administrador de un wiki puede llegar a ser muy tedioso, más aún a medida que el sistema crece. Entonces se hace necesario desarrollar sistemas que funcionen de manera automática, tanto para tareas de control y corrección de errores como para tareas repetitivas como modificación de categorías o similares.

Para sistemas basados en el motor *MediaWiki* se puede utilizar *pywikipedia*, la API para Wikipedia en Python, que puede utilizarse para automatizar o simplificar tareas de edición en cualquier proyecto de la textitFundación Wikimedia. Es necesario realizar algunas modificaciones e incluir una configuración extra para poder utilizarla para un wiki externo a este proyecto.

En el caso de WIKIUNIX resultó muy útil a la hora de generar artículos de las órdenes de sistemas UNIX a partir del manual de *Ubuntu*. La idea se basaba en leer un listado de órdenes para cada una de las cuales se seguían los siguientes pasos:

- ◆ Se obtiene la dirección URL de la orden dentro del repositorio de páginas de manual de Ubuntu [3]
- ◆ Se accede a dicha página y se procesa la información buscada. Dicha información era la descripción y la sintaxis de uso de la orden.
- ◆ Con los datos obtenidos, se construye el contenido que deseamos incluir en el wiki.

- ◆ Se edita el artículo del wiki de la orden actual para introducir el contenido generado.

Dado que cabía la posibilidad de que el script (que se encuentra en la página [147](#)) no funcionara en determinados casos, bien por problemas de conexión o de procesamiento erróneo del contenido, se incluye un sistema de estadísticas durante la ejecución del mismo. Así se controlan los artículos que ya estaban creados, los que han fallado y los que se han editado de forma correcta.

Este es solo un ejemplo muy simple de lo mucho que pueden ayudar los bots a nivel de ejecución de tareas repetitivas o de control. Otro bot que puede evitar tareas repetitivas revirtiendo vandalismos es AVBOT [\[38\]](#). Podemos adaptarlo a nuestras necesidades en caso de que fuera necesario.

## Conclusiones

---

### El trabajo realizado

Terminado ya el proyecto *Tutorial en formato wiki sobre Sistemas Operativos UNIX con soporte para pruebas* se han cubierto los siguientes objetivos propuestos:

- ◆ Hemos conseguido un sistema editable por un grupo de trabajo de manera sencilla con una interfaz amigable.
- ◆ El contenido de dicho sistema está bien estructurado y clasificado en niveles de dificultad.
- ◆ Se ha construido un sistema donde realizar ejercicios y donde comprobar su resolución de forma automática.
- ◆ El sistema es accesible a cualquiera que desee acceder a él a través de la dirección:

`http://osl.uca.es/wikiunix`

Ha sido un trabajo duro e intenso que será liberado de forma oficial bajo licencia libre una vez presentado el proyecto ante el tribunal para su evaluación. Este hito supondrá la culminación de esta primera etapa del proyecto, a partir de la cual podrá ser leído y trabajado por todo aquel que lo considere interesante. La siguiente etapa será de revisión, mejora y ampliación del contenido y contará con una participación externa abierta (espero que abundante).

### El futuro del proyecto

El producto resultado de este PFC es un producto evolutivo. Por su naturaleza está ideado para que la comunidad logre una evolución propia de su utilización. La posibilidad de que los usuarios modifiquen, completen y agreguen temas es una característica intrínseca al proyecto por lo que las futuras versiones parten del interés de los usuarios.

Sería bastante interesante crear otras máquinas virtuales. En este proyecto se ha utilizado una máquina con la distribución *Xubuntu*, que tiene como entorno de escritorio XCFE. En un futuro podría incluirse otra distribución que utilice KDE por ejemplo, que es más actual y extendido.

En este sentido también sería muy didáctico ampliar la batería de ejercicios y artículos en torno a los entornos gráficos de escritorio, puesto que actualmente los ejercicios se centran en la terminal.

Por otro lado, sería conveniente seguir ampliando el mantenimiento del wiki con nuevos bots. Sobre todo pensando en que la wiki va a ser accesible a cualquiera, habrá que desarrollar algunos bots de control, para que ayuden a los administradores a controlar las nuevas entradas.



# Manual de usuario

---

## Introducción

Este es el manual de WikiUNIX enfocado a los usuarios que vayan a utilizar dicha plataforma educativa. Como antecedentes al propio manual hay que explicar que el tutorial está desarrollado en formato wiki<sup>1</sup> soportado por el motor MediaWiki<sup>2</sup>.

Los usuarios de esta plataforma pueden distinguirse en dos tipos:

- ✍ Usuario final: Cualquiera que tan sólo desee ver el contenido del wiki y/o ejecutar sus scripts de comprobación en el soporte para pruebas.

En este caso no se necesitarán conocimientos de edición en formato wiki ni conocimientos avanzados de ningún tipo: todo lo necesario para realizar esta labor está autocontenido en el wiki, aún así puede ser consultado en las siguientes páginas.

Lo único que sí es necesario saber en este aspecto es navegar por el entorno y conocer el entorno para pruebas, para lo cual deben revisar las páginas [88](#) y [105](#).

- ✍ Miembro de la comunidad: Aquellos usuarios que además pretendan colaborar con el proyecto y que por tanto necesiten saber cómo modificar y/o ampliar de forma adecuada el contenido existente.

Estos usuarios deben prestar especial atención a partir de la página [92](#).

Este manual contiene información acerca de cómo editar contenido en formato wiki, sin embargo, no es una fuente completa ni única en ese sentido. Esto quiere decir que para cualquier información adicional acerca de ediciones en estos ámbitos puede ser válido cualquier otro manual o ayuda soportado por la comunidad relacionada con MediaWiki.

---

<sup>1</sup>Se llama *wiki* a las páginas web con enlaces, imágenes y cualquier tipo de contenido que puede ser visitada y editada por cualquier persona con permisos para ello. De esta forma se convierte en una herramienta Web que nos permite crear colectivamente documentos obteniendo una gran riqueza de la comunidad.

<sup>2</sup>MediaWiki es uno de los Sistemas Gestores de Contenidos más populares que se especializan en wikis; fue desarrollado inicialmente para Wikipedia, pero luego fue hecho público bajo la Licencia GNU GPL. Este motor permite varios tipos de interacción con el producto que pasamos a detallar.

## Interfaz de usuario

Se puede acceder al sitio donde se aloja el tutorial utilizando un navegador web compatible con los estándares HTML 4.0 y CSS de W3C<sup>3</sup>.

La navegación general en todas las páginas se puede distinguir en los siguientes bloques:



Figura 5.1: Navegación básica general

- I. LOGO DE WIKIUNIX: Permite volver a la portada del wiki en cualquier momento.
- II. BARRA LATERAL: Contiene diversos menús de navegación y búsqueda de artículos.

**Navegación:** Enlaces básicos de navegación. Gracias a ellos se puede acceder a la portada del wiki, al portal de la comunidad (página informativa acerca de la comunidad que desarrolla el wiki), a la sección de actualidad (que incluye las últimas noticias relacionadas con el wiki que también pueden visualizarse en portada), a los cambios recientes (donde se visualizan todos los cambios realizados en el wiki en los últimos instantes), a una página aleatoria (por si quieres tantear el estado de los artículos del wiki o te aburres) y a la ayuda general del wiki.

<sup>3</sup>Con navegadores que no cumplan esta premisa no se asegura la correcta visualización de los componentes.

**Niveles:** Enlaces directos a los niveles de complejidad establecidos, que contienen los artículos principales del wiki.

**Información y descargas:** Enlaces a las páginas de información más importantes del wiki y a aquellas donde se accede a las descargas relacionadas con éste.

**Buscar:** Introduce una palabra para realizar una búsqueda.

Si eliges pulsar el botón **Ir**, en caso de que el título del artículo coincida con lo que hayas escrito, accederás automáticamente al artículo; en caso de que no exista, te mostrará la lista de artículos más probables.

Si pulsas sobre el botón **Buscar**, el servidor buscará en los artículos de WikiUNIX las palabras que hayas escrito, mostrándote una lista con los más probables.

III. **HERRAMIENTAS:** Es un tipo de menú de navegación que nos permite interactuar con ciertos aspectos de la página que se muestra actualmente. Podemos obtener información de la página, subir un archivo y obtener una versión para impresión.

IV. **BARRA SUPERIOR DERECHA:** Si has entrado como usuario registrado, te aparecerá al principio de la página un cuadro con tus propios enlaces.

**Tu nombre de usuario** te lleva a tu página de usuario.

**mi discusión** te lleva a tu página de discusión.

**preferencias** te lleva a la página de configuración de tus preferencias de usuario.

**lista de seguimiento** te permite ver las últimas ediciones de los artículos que has marcado para vigilar.

**mis contribuciones** te muestra una lista de tus contribuciones, ordenadas por fecha.

**salir** te permite salir de tu cuenta de usuario.

V. **PESTAÑAS SUPERIORES:** En la zona superior de la ventana, encima del título del artículo actual, se encuentran las pestañas relacionadas con éste.

**artículo** permite ver el artículo de Wikipedia.

**discusión** permite ir a la página de discusión del artículo.

**editar** permite modificar la página que estás viendo.

**historial** permite ver el historial de modificaciones de la página.

**trasladar** permite cambiar de nombre al artículo, dejando una redirección al nuevo nombre en el viejo (sólo para usuarios registrados).

**vigilar o dejar de vigilar** permite añadir o quitar un artículo de la lista de seguimiento (sólo para usuarios registrados).

## Página principal

Además de la navegación que aporta MediaWiki de por sí, se ha desarrollado una interfaz amigable y completa en la página principal del wiki para facilitar la búsqueda de contenido y ayuda.<sup>4</sup>

---

<sup>4</sup>Esta página no debe ser modificable por nadie, excepto por los usuarios administradores o burócratas.

Podemos ver en la Figura 5.2 que en esta estructura se distinguen varias secciones claramente diferenciadas:



- I. BIENVENIDA: Breve introducción acerca de la temática del wiki y lo que se puede esperar de él.
- II. INDICE DE CAPÍTULOS POR NIVEL DE COMPLEJIDAD:
  - a. Principiante
  - b. Usuario iniciado
  - c. Usuario habitual
  - d. Unixero profesional
- III. NAVEGACIÓN:
  - a. Información: acceso directo a la información más importante del wiki (como puede ser lo relacionado con el *soporte para pruebas*).
  - b. Búsqueda: además de la búsqueda habitual incluida en el lateral izquierdo del wiki, se han habilitado una serie de accesos directos a categorías de elementos útiles (como por ejemplo un enlace a todos los scripts incluidos en el wiki).
  - c. Ayuda: accesos a las páginas de ayuda del wiki, tanto en el sentido estético (estilo en formato wiki) como en el informativo (cambios recientes)
- IV. AVISOS: notas importantes que informan de cuestiones que deberían ser tenidas en cuenta por los usuarios
- V. DESCARGAS: sección donde se incluyen los enlaces a las descargas habilitadas, así como los últimos scripts incluidos en el wiki
- VI. NOTICIAS: últimas dos noticias en torno al wiki

## Publicación y modificación de artículos

Para crear nuevos artículos o modificar los existentes tan solo se necesitan los conocimientos que se deseen añadir y conocer la sintaxis del formato wiki.

Para editar un artículo que ya esté creado sólo hay que pulsar en la pestaña *Editar* en la página de dicho artículo, apareciendo un editor con el código de la página, tal y como se ve en la Figura 5.3.

Este editor es muy completo a pesar de su simplicidad. En torno a él se encuentran una barra de herramientas (que contiene toda la sintaxis básica del formato wiki), una caja para incluir un resumen de tus modificaciones<sup>5</sup> y una serie de acciones, tales como **Grabar la pagina** (guarda los cambios que has realizado), **Mostrar previsualización** (actualiza la página, mostrándote cómo se vería con tus modificaciones ya guardadas, pero sin guardarlos realmente) y **Mostrar cambios** (expone las modificaciones que has realizado con respecto a la versión anterior del artículo).

---

<sup>5</sup>Este resumen es muy importante para compaginar el trabajo con otros usuarios y poder hacer un seguimiento simple de tu trabajo.



Figura 5.3: Editor proporcionado por MediaWiki

Si quieres crear un nuevo artículo basta con crear un enlace interno al título que desees que posea la página. El wiki automatiza la operación de creación.

## Sintaxis básica

En esta sección se expone la notación a utilizar cuando se emplea el formato wiki. Podéis encontrar en esta misma información en la [guía de estilo](#) y además puede ser completada con cualquier guía acerca de formato en wikis cuyo motor sea *MediaWiki*.

Para facilitar su comprensión, se ha estructurado la sintaxis en cuatro tablas de menor a mayor complejidad y frecuencia (normalmente la notación más simple es la más utilizada, como poner un texto **en negrita**).

Si escribes esto al editar...	Este será el resultado...
<code>'''Hola'''</code>	<b>Hola</b>
<code>''Adiós''</code>	<i>Adiós</i>
<code>''''Wikiunix''''</code>	<b><i>Wikiunix</i></b>

Cuadro 5.1: Sintaxis: Lo más básico

Si escribes esto al editar...	Este será el resultado...
<code>&lt;span style=çolor:purple&gt;Morado&lt;/span&gt;</code>	<b>Morado</b>
<code>&lt;u&gt;Una persona&lt;/u&gt;</code>	<u>Una persona</u>
<code>&lt;center&gt;Extrañamente&lt;/center&gt;</code>	Extrañamente
<code>&lt;!--Cualquier palabra--&gt;</code>	
<code>[[Portada]]</code>	Portada
<code>[[Portada La portada]]</code>	La portada
<code>http://es.wikipedia.org</code>	http://es.wikipedia.org
<code>[http://es.wikipedia.org]</code>	[1]
<code>[http://es.wikipedia.org Wikipedia]</code>	Wikipedia

Cuadro 5.2: Sintaxis: Lo básico

Lo no tan vital	
<i>Si escribes esto al editar...</i>	<i>Este será el resultado...</i>
<code>[[wikipedia:Ejemplo]]</code>	wikipedia:Ejemplo
<code>[[wikipedia:Second Life Segunda Vida]]</code>	Segunda Vida
<ul style="list-style-type: none"> <li>* Hola 1</li> <li>* Hola 2</li> <li>* Hola 3</li> </ul>	<ul style="list-style-type: none"> <li>■ Hola 1</li> <li>■ Hola 2</li> <li>■ Hola 3</li> </ul>
<ul style="list-style-type: none"> <li># One</li> <li># Two</li> <li># Three</li> </ul>	<ol style="list-style-type: none"> <li>1. One</li> <li>2. Two</li> <li>3. Three</li> </ol>
----	-----
<pre>:Espacio 1 ::Espacio 2 :::Espacio 3 ::::Espacio 4</pre>	<pre>Espacio 1   Espacio 2     Espacio 3       Espacio 4</pre>
<ul style="list-style-type: none"> <li>* Uno</li> <li>**sub-uno</li> <li>**sub-uno2</li> <li>* Dos</li> </ul>	<ul style="list-style-type: none"> <li>■ Uno <ul style="list-style-type: none"> <li>■ sub-uno</li> <li>■ sub-uno2</li> </ul> </li> <li>■ Dos</li> </ul>
<pre>#primero ##primero.primero ##primero.segundo #segundo</pre>	<ol style="list-style-type: none"> <li>1. primero <ol style="list-style-type: none"> <li>1. primero.primero</li> <li>2. primero.segundo</li> </ol> </li> <li>2. segundo</li> </ol>

Figura 5.4: Lo no tan vital



Más allá de lo necesario										
<i>Si escribes esto al editar...</i>	<i>Este será el resultado...</i>									
<pre>{  border=1 style="float:right; margin:0 0 1em 1em"  + Leyenda ! Columna1 !! Columna2 !! Columna3  -   Celda A1    Celda A2    Celda A3  -   Celda B1    Celda B2    Celda B3  }</pre>	<div>Leyenda</div> <table><tr><th>Columna1</th><th>Columna2</th><th>Columna3</th></tr><tr><td>Celda A1</td><td>Celda A2</td><td>Celda A3</td></tr><tr><td>Celda B1</td><td>Celda B2</td><td>Celda B3</td></tr></table>	Columna1	Columna2	Columna3	Celda A1	Celda A2	Celda A3	Celda B1	Celda B2	Celda B3
Columna1	Columna2	Columna3								
Celda A1	Celda A2	Celda A3								
Celda B1	Celda B2	Celda B3								

Figura 5.5: Más allá de lo necesario

## Plantillas adicionales

Además de la sintaxis básica del formato wiki, existen algunas plantillas que pueden servir de utilidad a la hora de crear nuevo contenido. A continuación, se detallan cada una de las plantillas más superficiales (las que más se visualizan en los artículos y que se forman en función de otras plantillas a más *bajo nivel*), junto con un ejemplo tanto en código como en visualización final.

### PLANTILLA ARCHIVO

Esta plantilla debe utilizarse para incluir archivos con su respectivo contenido.

Recibe como parámetros el nombre del archivo y su contenido.

Si en formato wiki tenemos:

```

1  {{Archivo
2  |prueba.txt
3  |Monstruo Espagueti Volador
4  Calle de los espagnósticos 123
5  Barrio de Espagnostia
6  Ciudad Pastafari, C.P. 123456}}
```

se visualizará lo siguiente en el wiki:

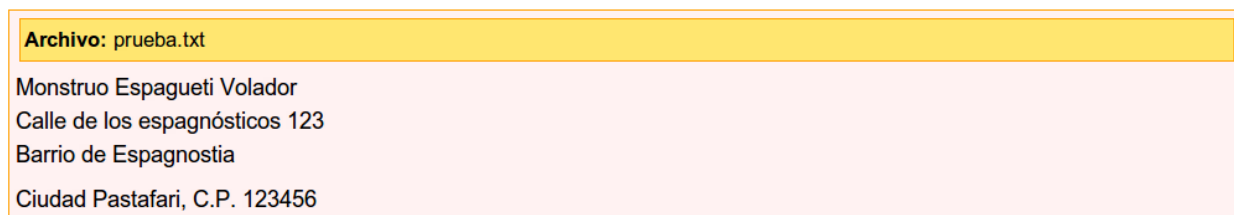


Figura 5.6: Plantilla Archivo

#### ✍ PLANTILLA COMANDO

Esta plantilla debe utilizarse para incluir órdenes que ejecuta en una terminal cualquier usuario con los permisos básicos.

Recibe como único parámetro la línea o las líneas que componen la orden y su respectiva salida en caso de que devolviera algo y se desee mostrarlo.

Si en formato wiki tenemos:

```
1 {{Comando
2 |cp [opciones] origen destino}}
```

se visualizará lo siguiente en el wiki:

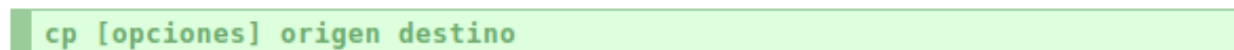


Figura 5.7: Plantilla Comando

#### ✍ PLANTILLA CONTROLFINAL

Esta plantilla debe utilizarse para describir los controles finales de los ejercicios relacionados con los artículos principales del wiki. Solo puede haber uno por conjunto de ejercicios.

Recibe como parámetros:

- ✱ **artículo:** artículo principal relacionado con el conjunto de ejercicios
- ✱ **sudo:** debe ser “sudo” o “ ” (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- ✱ **directorio:** directorio en el cual se esté trabajando
- ✱ **script:** nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio
- ✱ **resumen:** conjunto de órdenes empleadas durante el desarrollo de los ejercicios

Si en formato wiki tenemos:

```

1  {{ControlFinal
2  |sudo=sudo
3  |artículo=Sistema de archivos
4  |directorio=archivos
5  |script=archivos
6  |resumen=[[chmod]] • [[chown]] • [[chgrp]] • otro}}

```

se visualizará lo siguiente en el wiki:

Vuelve al directorio inicial (**archivos**) y ejecuta el control final:

```
sudo python archivos.py
```

Tras la realización de estos ejercicios debes tener clara la utilización y el manejo de las siguientes órdenes:

`chmod` • `chown` • `chgrp` • `otro`

Si aún sigues teniendo dudas puedes:

1. Releer el artículo [Sistema de archivos](#) de nuevo al completo, o tan sólo las secciones en las que tengas dudas.
2. Volver a desarrollar los ejercicios de nuevo.
3. Revisar los artículos creados en el wiki para cada una de las órdenes.
4. Revisar el manual de las órdenes implicadas (recuerda que tienes los enlaces en los artículos del wiki de cada una y además en está el manual del sistema).

Figura 5.8: Plantilla Control Final

## ✍ PLANTILLA CONTROLINTERMEDIO

Esta plantilla debe utilizarse para describir los controles intermedios de los ejercicios relacionados con los artículos principales del wiki.

Recibe como parámetros:

- \* **ordinal**: posición del control en el número total de controles intermedios, a saber *primer*, *segundo*,...
- \* **número**: número del control en el total de controles intermedios, a saber 1, 2,...
- \* **sudo**: debe ser “sudo” o “ ” (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- \* **directorio**: directorio en el cual se esté trabajando
- \* **script**: nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio
- \* **previo**: contenidos desarrollados hasta la aparición del control intermedio actual
- \* **posterior**: contenidos que se desarrollarán hasta el siguiente control intermedio

Si en formato wiki tenemos:

```

1  {{ControlIntermedio
2  |sudo=sudo
3  |ordinal=segundo
4  |número=2
5  |directorio=archivos
6  |script=archivos
7  |previo=Máscara de permisos y cambio de permisos y usuarios/grupos
8  propietarios.
9  |posterior=Limpieza final del directorio: restaurar permisos y
10 propietarios.}}
```

se visualizará lo siguiente en el wiki:

Vuelve al directorio inicial (**archivos**) y ejecuta el segundo control intermedio:

```
sudo python archivos.py -p 2
```

O

```
sudo python archivos.py --partial 2
```

DEBES HABER APRENDIDO A:	QUEDA POR APRENDER:
Máscara de permisos y cambio de permisos y usuarios/grupos propietarios.	Limpieza final del directorio: restaurar permisos y propietarios.

Figura 5.9: Plantilla ControlIntermedio

## ✍ PLANTILLA CURIOSIDAD

Si en formato wiki tenemos:

```

1  {{Curiosidad
2  |Hay que tener cuidado en la utilización de la orden '''cp''' puesto
3  que puede destruir el contenido de un archivo.
4
5  Esto se debe a que si el archivo ''destino'' existe antes de utilizarla,
6  se sobrescribirá. Si se desea una forma más segura de utilizar esta
7  orden, se debe utilizar de forma interactiva:{{Comando|cp -i}}Así nos
8  preguntará antes de copiar el archivo:
9  {{Comando|$ cp -i a.txt b.txt
10 cp: overwrite 'b.txt'? y}}
11 }}
```

se visualizará lo siguiente en el wiki:

**Curiosidad**

Hay que tener cuidado en la utilización de la orden **cp** puesto que puede destruir el contenido de un archivo. Esto se debe a que si el archivo *destino* existe antes de utilizarla, se sobrescribirá. Si se desea una forma más segura de utilizar esta orden, se debe utilizar de forma interactiva:

```
cp -i
```

Así nos preguntará antes de copiar el archivo:

```
$ cp -i a.txt b.txt
cp: overwrite 'b.txt'? y
```

Figura 5.10: Plantilla curiosidad

### ✍ PLANTILLA EJERCICIOS

Esta plantilla debe utilizarse para generar nuevos ejercicios relacionados con los artículos principales. Recoge toda la información básica de dichos ejercicios.

Recibe como parámetros:

- ✱ **sección:** nombre identificativo del artículo con el que se corresponden los ejercicios
- ✱ **resumen:** comentario breve acerca del contenido de los ejercicios
- ✱ **directorio:** directorio en el cual se trabajará
- ✱ **sudo:** debe ser “sudo” o “ ” (espacio, la cadena vacía no funciona), en caso de que script deba ejecutarse con *sudo* delante o no
- ✱ **script:** nombre (sin la extensión *.py*) del script que ejecuta la comprobación de resolución del ejercicio

Si en formato wiki tenemos:

```
1  {{Ejercicios
2  |sección=sistema de archivos
3  |resumen=Permiten la modificación de permisos y propietarios de
4  los ficheros. Son muy útiles para la comprensión a fondo del
5  sistema de permisos que existe en este tipo de sistemas y para
6  valorar un buen control de éstos, así como qué importancia tienen.
7  |directorio=archivos
8  |sudo=sudo
9  |script=archivos}}
```

se visualizará lo siguiente en el wiki:

## Ejercicios de la sección de sistema de archivos

**ASPECTOS PEDAGÓGICOS:** Permiten la modificación de permisos y propietarios de los ficheros. Son muy útiles para la comprensión a fondo del sistema de permisos que existe en este tipo de sistemas y para valorar un buen control de éstos, así como qué importancia tienen.

**PREPARACIÓN PREVIA:** Para hacer estos ejercicios debes acceder al sistema con el usuario por defecto *usuario* (recuerda que la contraseña es *unix*). A continuación:

- Ve al **directorio casa** del usuario actual. Comprueba que existe un directorio llamado **unix** y dentro de éste un subdirectorio llamado **archivos**. Para ello ejecuta:

```
ls unix
```

Debe aparecer el directorio **archivos**. En caso de que no exista alguno de estos elementos, puedes hacer varias cosas:

1. Descargar el archivo comprimido con la jerarquía de elementos de comprobación que se encuentra en la **portada** del wiki y descomprimirlo en el directorio casa del usuario actual.
2. Apagar la máquina virtual y restaurar el *snapshot* que funcionaba bien.
3. Apagar la máquina virtual y borrarla eliminando el disco duro virtual. A continuación, descargar el disco duro y volver a configurar la máquina de nuevo.

- Ve al subdirectorio **unix/archivos**. Limpia el sistema antes de realizar los demás ejercicios para prevenir posibles incongruencias:

```
sudo python archivos.py -c
```

o

```
sudo python archivos.py --clean
```

Figura 5.11: Plantilla Ejercicios

## ✍ PLANTILLA FICHA DE ARTÍCULO

Esta plantilla debe utilizarse para generar nuevos artículos principales. Recoge toda la información básica del artículo.

Recibe como parámetros:

- ✳ **nombre:** nombre identificativo del artículo completo
- ✳ **descripción:** comentario breve acerca del contenido del artículo
- ✳ **tiempo:** duración estimada tanto de la lectura como del aprovechamiento completo del artículo (incluidos artículos secundarios enlazados y ejercicios)
- ✳ **nivel:** nivel de complejidad, actualmente *principiante*, *usuario iniciado*, *usuario habitual* y *unixero profesional*
- ✳ **info:** enlaces y documentos de los que se ha obtenido información para redactar el artículo
- ✳ **prerrequisitos:** páginas del wiki que deben leerse y comprenderse previo a la lectura del artículo

Si en formato wiki tenemos:

```
1  {{Ficha de artículo
2  |nombre=Características de Unix
3  |tiempo=1 hora
4  |descripción=En esta sección se introducen los sistemas operativos que
5  se encuendran bajo la familia Unix, especificando sus características
6  |nivel=[[Nivel:Principiante|Principiante]]
7  |info=[[http://www.unix.org/what_is_unix.html What is UNIX?]]<br>
8  [[http://www.linfo.org/unix_philosophy.html The Unix Philosophy: A Brief
9  Introduction]]<br>Libros SO
10 |prerrequisitos=[[Unix]]}}
```

se visualizará lo siguiente en el wiki:

Características de Unix	
En esta sección se introducen los sistemas operativos que se encuendran bajo la familia Unix, especificando sus características	
<b>Tiempo</b>	1 hora
<b>Nivel</b>	Principiante
<b>Información extraída de</b>	What is UNIX? <a href="#">↗</a>
	The Unix Philosophy: A Brief Introduction <a href="#">↗</a>
	Libros SO
<b>Prerrequisitos</b>	Unix

Figura 5.12: Plantilla Infobox

#### ✍ PLANTILLA LISTAREF

Esta plantilla genera la lista de referencias utilizadas en un artículo.

En formato wiki se indica tal que:

```
1 {{Listaref}}
```

#### ✍ PLANTILLA MAN

Esta plantilla genera un enlace al manual de Ubuntu de la orden afectada.

Si en formato wiki tenemos:

```
1 {{man|grep}}
```

se visualizará lo siguiente en el wiki:

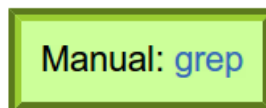


Figura 5.13: Plantilla man

#### ✍ PLANTILLA RESALTA

Esta plantilla destaca el texto contenido sobre el resto del artículo.

Si en formato wiki tenemos:

```
1 {{Resalta|Archivos}}
```

se visualizará lo siguiente en el wiki:

## Archivos

Figura 5.14: Plantilla Resalta

### ✍ PLANTILLA ROOT

Esta plantilla debe utilizarse para incluir órdenes que ejecuta en una terminal un usuario con permisos de administrador.

Recibe como único parámetro la línea o las líneas que componen la orden y su respectiva salida en caso de que devolviera algo y se desee mostrarlo.

Si en formato wiki tenemos:

```
1 {{Root
2 |reboot [opciones]}}
```

se visualizará lo siguiente en el wiki:


A screenshot of a terminal window with a light red background. The text 'reboot [opciones]' is displayed in a monospaced font.

Figura 5.15: Plantilla Root

### ✍ PLANTILLA TÍTULOMINÚSCULAS

Esta plantilla modifica el título principal del artículo cambiando todas las letras a minúsculas (incluida la primera).

Si en cualquier lugar de la página *Organizar.py* incluimos:

```
1 {{TOCderecha}}
```

se visualizará el título de la página tal que:


A screenshot of a terminal window showing the title 'organizar.py' in a large, bold, monospaced font. Above the title, there are two buttons: 'página' (highlighted in blue) and 'discusión' (highlighted in red).

Figura 5.16: Plantilla TítuloMinúsculas

### ✍ PLANTILLA TOCDERECHA

Esta plantilla desplaza el menú del artículo hacia el lado derecho de la página.



## Plantillas de mantenimiento

### ✍ PLANTILLA DESAMBIGUACIÓN

La plantilla desambiguación se debe colocar en páginas que poseen varios significados o que pueden compartir un mismo título.

Si en formato wiki tenemos:

```
1  {{Desambiguación}}
```

se visualizará lo siguiente en el wiki:

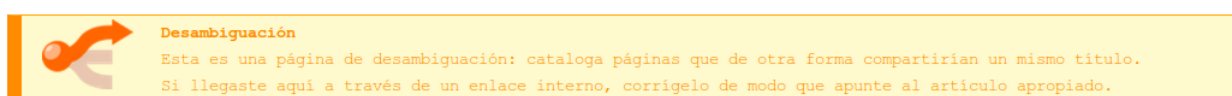


Figura 5.17: Plantilla Desambiguación

### ✍ PLANTILLA FORMATEAR

Esta plantilla se debe colocar en una página que haya que editar en cuestión de estilo o que no esté completa en este sentido.

Si en formato wiki tenemos:

```
1  {{Formatear}}
```

se visualizará lo siguiente en el wiki:

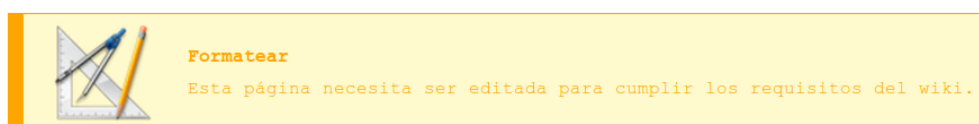


Figura 5.18: Plantilla Formatear

### ✍ PLANTILA TODO

La plantilla *ToDo* se debe colocar en una página que no esté completa o a la que le falta contenido.

Si en formato wiki tenemos:

```
1  {{ToDo}}
```

se visualizará lo siguiente en el wiki:

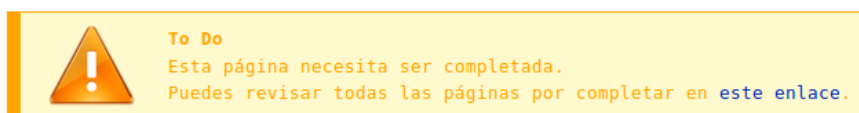


Figura 5.19: Plantilla ToDo

## Normas generales

Es importante seguir unas normas de edición a la hora de crear un artículo. Divide tus creaciones en secciones lógicamente que permitan al lector tener un esquema claro de la temática que allí se quiere exponer.

Usa las negritas y las cursivas para resaltar texto pero sin abusar. Complementa tus artículos con imágenes y demás contenido multimedia que harán que el saber sea más divertido para todos.

## Soporte para pruebas

El soporte para pruebas del wiki se basa en la utilización de una máquina virtual que simula un sistema donde se instala una distribución Linux, en el cual los usuarios desarrollarán los ejercicios y comprobarán su resolución utilizando los scripts de comprobación adecuados.

## Máquina virtual

La máquina virtual se distribuye bajo el formato OVF<sup>6</sup> estándar de servicios virtualizados, que puede ser utilizado desde diversas aplicaciones.

En este manual se explicará su utilización, importación y manejo desde *VirtualBox*, aunque cualquier otra que soporte este formato es válida.

## Instalación en sistemas Debian

Simplemente ejecutamos como root:

```
apt-get install virtualbox
```

Aunque no es necesario, también es conveniente instalar el paquete *dkms* (no disponible para Debian) para asegurar que los módulos del kernel de VirtualBox (*vboxdrv*, *vboxnetflt* y *vboxnetadp*) se actualizan si cambia la versión del kernel de linux al actualizar el sistema en un futuro.

## Descarga de la imagen

La imagen de disco duro virtual puede ser descargada desde el siguiente enlace:

<http://osl.uca.es/wikiunix/archivos/Xubuntu.tar.gz>

El fichero a descargar en un *tar.gz* con un directorio llamado *Xubuntu*, dentro del cual se encuentran dos ficheros más: uno es el fichero de configuración de extensión OVF (que será el que se abra desde *VirtualBox*) y otro es el sistema virtualizado comprimido en VMDK.<sup>7</sup>

## Importación de la imagen

Si iniciamos *VirtualBox* por primera vez nos encontraremos con una pantalla, la cual está dividida en dos zonas principales: la de la izquierda es donde aparecerá la lista de las máquinas virtuales de nuestro equipo y la de la derecha es donde se especificarán los detalles de la que esté seleccionada

---

<sup>6</sup>OVF es un estándar abierto para empaquetar y distribuir servicios virtualizados o de forma más general software a ejecutar en máquinas virtuales.

<sup>7</sup>Asegúrate de que los dos continúan en la misma carpeta al descomprimirlos para poder importar el sistema.

actualmente. Para introducir nuestra imagen de Xubuntu como una máquina virtual en VirtualBox tendremos que pulsar en *Archivo > Importar Servicio Virtualizado*, en la parte superior izquierda.

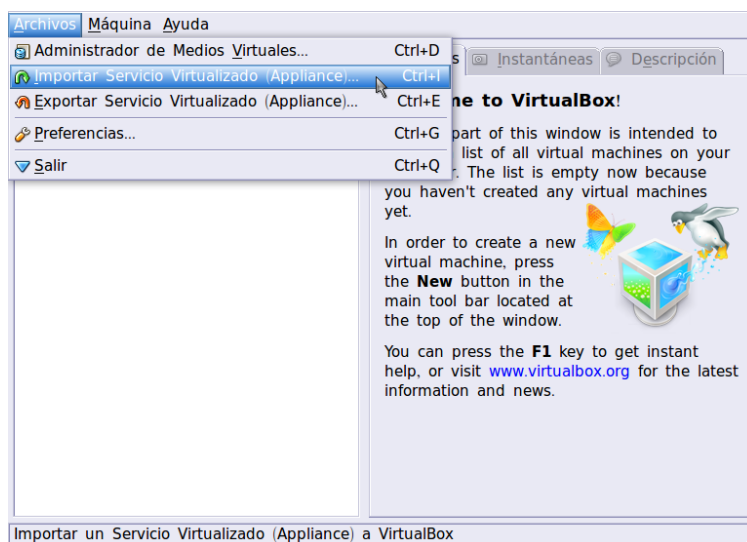


Figura 5.20: VirtualBox - Importación - paso 1

Se abrirá el asistente de importación de servicios centralizados, que nos guiará a cada paso. Pulsamos el botón **Elegir** y buscamos nuestro fichero OVF.



Figura 5.21: VirtualBox - Importación - paso 2

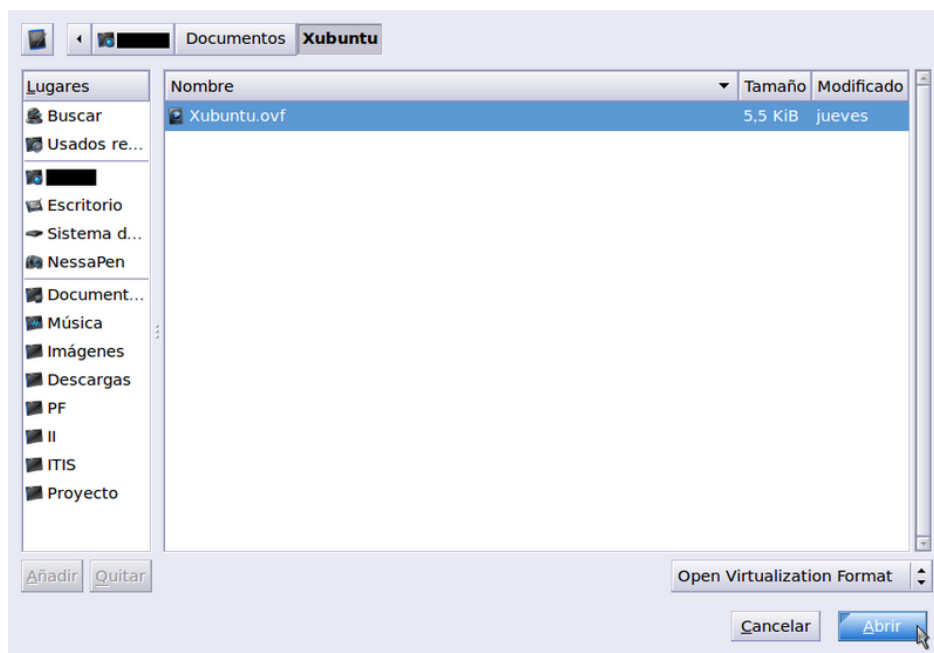


Figura 5.22: VirtualBox - Importación - paso 3

El asistente mostrará el fichero que se haya seleccionado. En caso de que sea correcto pulsamos sobre **Siguiente**.

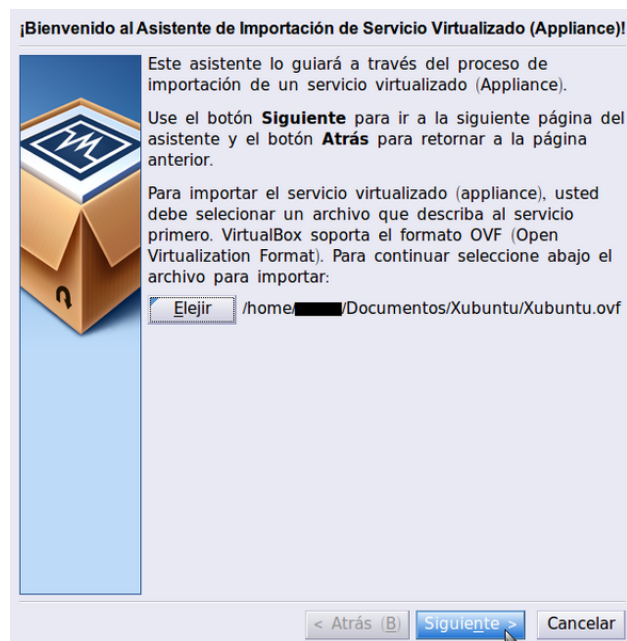


Figura 5.23: VirtualBox - Importación - paso 4

Aparecerá una pantalla con un resumen de la configuración de los servicios que se hayan detectado en el fichero OVF que hemos elegido. Sólo tenemos que pulsar en **Importar**.

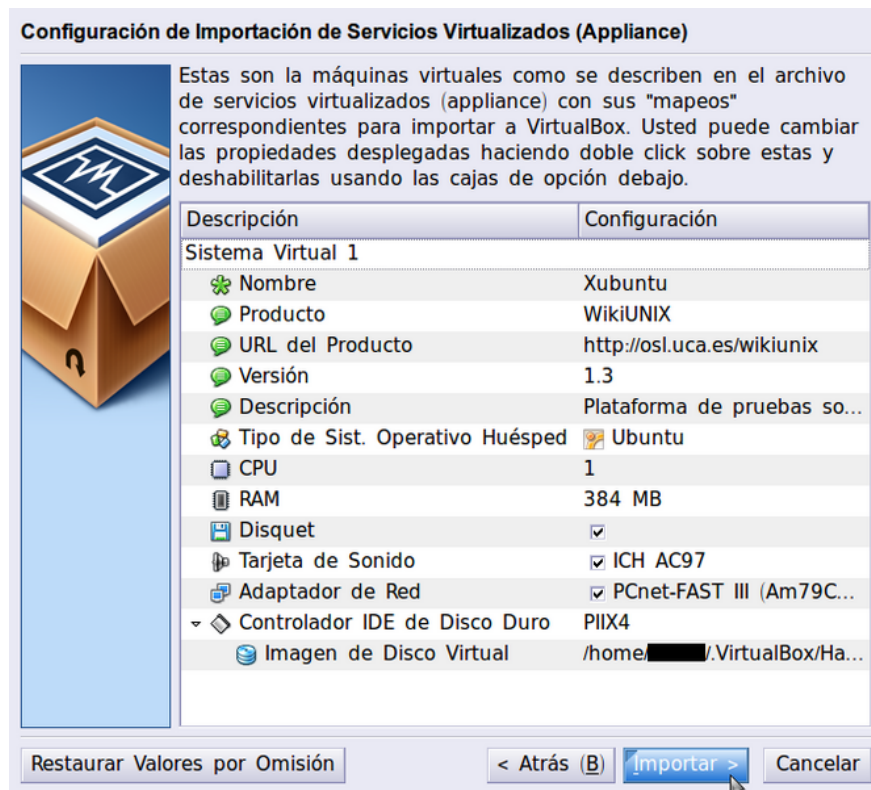


Figura 5.24: VirtualBox - Importación - paso 5

Lo siguiente es aceptar la licencia bajo la que se distribuye el sistema pulsando en el botón correspondiente. En nuestro caso es GPL. Si quieres leer la licencia completa, revisa <http://www.gnu.org/copyleft/gpl.html>

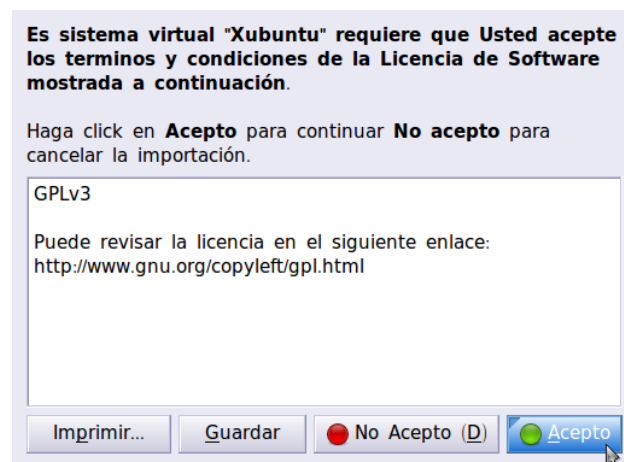


Figura 5.25: VirtualBox - Importación - paso 6

Se abrirá una nueva ventana con una barra de progreso que indica el tanto por ciento del sistema

importado, así como el tiempo restante.

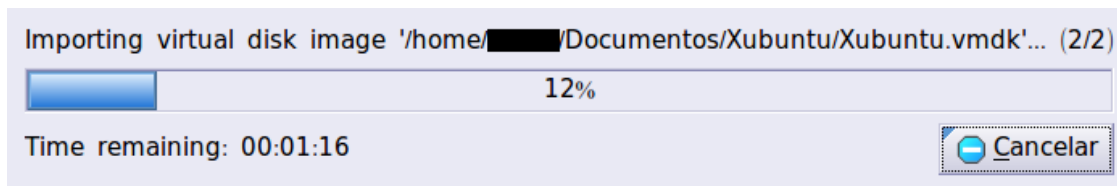


Figura 5.26: VirtualBox - Importación - paso 7

Espera a que la barra se complete. Esto último nos devuelve a la pantalla principal del programa, donde podemos observar que ya aparece la máquina virtual que acabamos de importar a la izquierda y los detalles de dicha configuración y algunos parámetros más a la derecha.

En principio no hay necesidad de modificar la configuración por defecto, por lo que pulsamos sobre **Iniciar**. Ya podemos trabajar en nuestro nuevo sistema virtualizado.

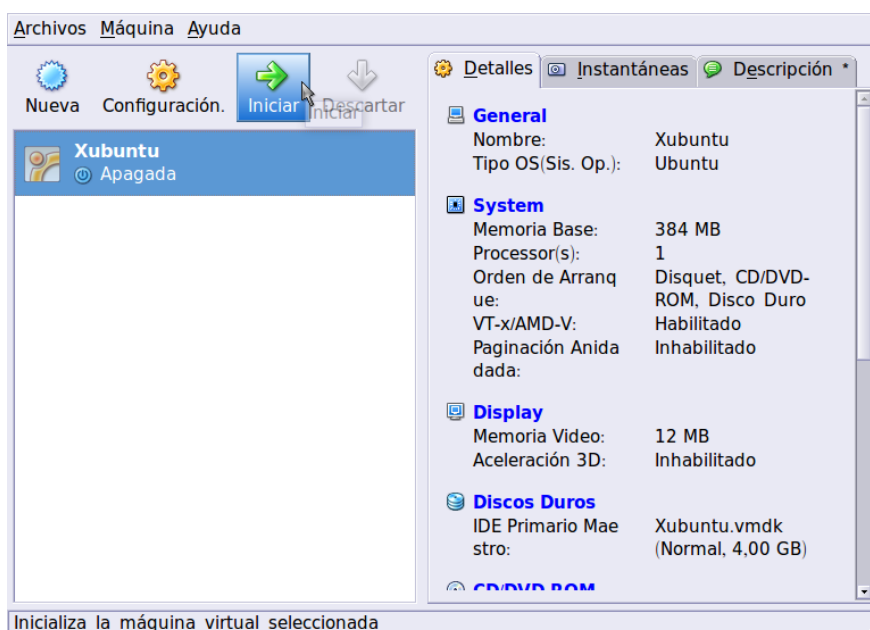


Figura 5.27: VirtualBox - Importación - paso 8

## Creación de una instantánea

Para evitar futuros problemas al probar el sistema o para que no nos sea necesario volver a configurarlo desde el principio en caso de que haya que restaurar el sistema inicial, crearemos una instantánea en cuanto acabemos de configurarlo por primera vez.

Sólo tenemos que seleccionar la máquina virtual en la zona de la izquierda y pulsar sobre la pestaña



**Instantáneas** en la zona de la derecha. A continuación, seleccionamos **Tomar instantánea** (*Ctrl + Shift + S*):

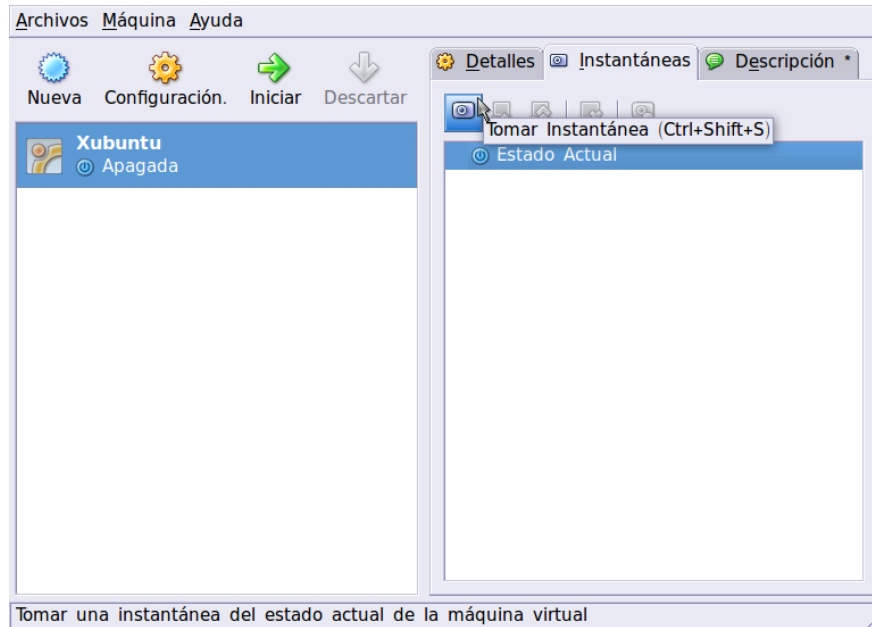


Figura 5.28: VirtualBox - Instantánea - paso 1

Luego introducimos el nombre y la descripción de la instantánea (a ser posible con algo significativo, como la fecha u otro dato que la distinga):

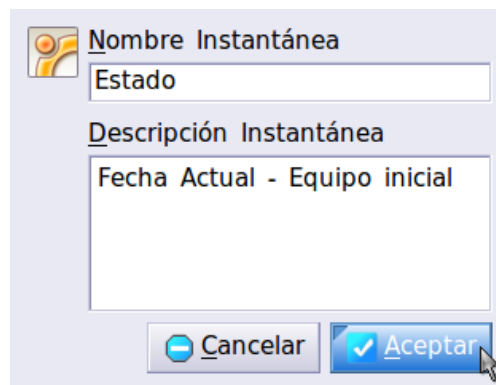


Figura 5.29: VirtualBox - Instantánea - paso 2

Si en un futuro quisiéramos restaurar el estado de la máquina virtual al de la instantánea tomada, solo tenemos que pulsar sobre ella y seleccionar **Revertir a la Instantánea Actual** (*Ctrl + Shift + R*):

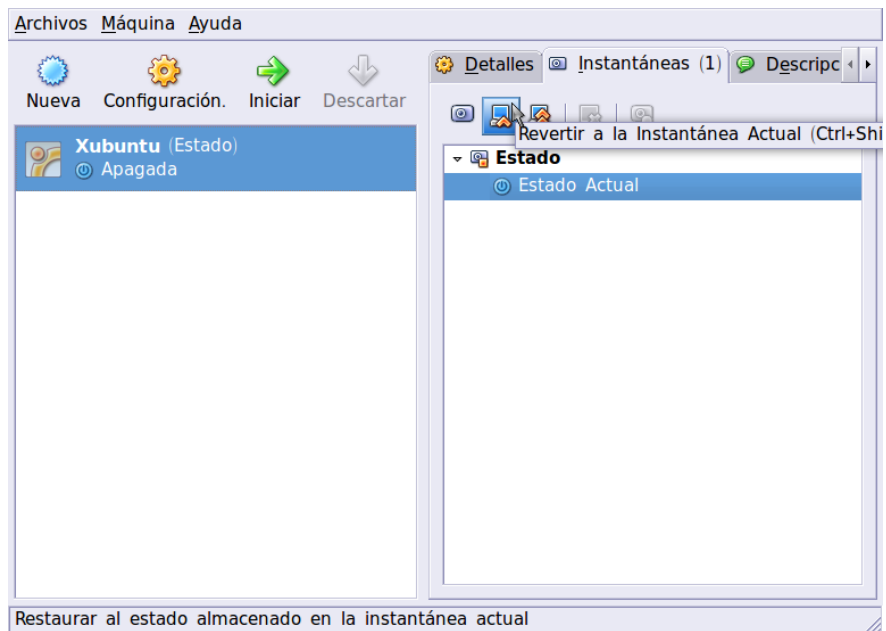


Figura 5.30: VirtualBox - Instantánea - paso 3

### Información básica de la imagen

- ✍ Nombre de usuario habitual: *usuario*
- ✍ Contraseña: *unix*
- ✍ Nombre de usuario sin privilegios: *unixero*
- ✍ Contraseña: *wikiunix*
- ✍ Contraseña para el administrador: *unixero*
- ✍ Contraseña para realizar tareas administrativas: *unix*

Para más información, entra en la página [Soporte para pruebas del wiki](#).

### Problemas

Puede ser que al instalar los paquetes para mantener el kernel actualizar en la máquina virtual, aparezca un mensaje de error (Figura 5.31) cuando se actualice el kernel.

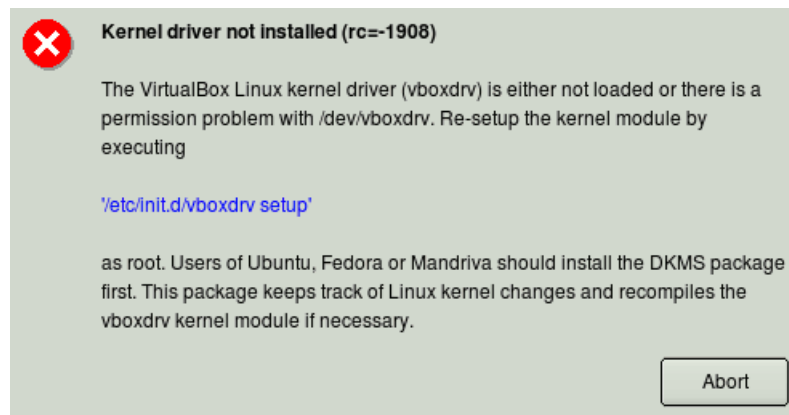


Figura 5.31: VirtualBox - Error

Se soluciona instalando el paquete DKMS y configurando el servicio *vboxdrv*:

```
sudo apt-get install dkms
/etc/init.d/vboxdrv setup
```

## Scripts de comprobación

### Utilizando los *scripts* existentes

Los scripts que se han desarrollado tienen como única dependencia el metapaquete *python2.6*, que ya viene instalado por defecto en sistemas Debian y derivados.<sup>8</sup>

Sus funciones básicas son las siguientes:

#### ✍ LIMPIEZA Y COMPROBACIÓN PREVIA DEL SISTEMA.

Para utilizar esta función debes llamar al script con la opción “c”/“clean”.

```
python script.py -c
```

O

```
python script.py --clean
```

#### ✍ CONTROLES INTERMEDIOS.

Para utilizar esta función debes llamar al script con la opción “p”/“partial” acompañada del número de comprobación actual.

---

<sup>8</sup>Habitualmente en otro tipo de sistemas también vienen instalados, aunque en caso de tener algún tipo de problema se debe consultar la [página oficial de Python](#).

```
python script.py -p <num>
```

o

```
python script.py --partial <num>
```

#### CONTROL FINAL.

Para utilizar esta función debes llamar al script sin indicar ninguna opción extra (es su comportamiento por defecto).

```
python script.py
```

## Creando nuevos scripts

Para crear nuevos scripts tan sólo se necesita “copiar” el “script” *control.py* (puedes encontrarlo en la forja y en el wiki) y modificar los nombres de los ficheros de comprobación según queramos.

A continuación se detallan los tipos de ficheros de comprobación, junto con los nombres actuales. Tenemos una serie de ficheros de control, los cuales se distinguen en dos grupos principales: los **ficheros de comprobación**, que tan sólo verifican información sin modificar nada, y los **ficheros de acción**, que se encargan de preparar el entorno para la ejecución de los ejercicios (habitualmente estos ficheros se utilizan tan sólo dentro de los directorios de limpieza o de comprobación previa).<sup>9</sup>

Los ficheros de comprobación se dividen en:

**direxists - dir\_existentes.dat** Incluye los directorios de los que se desea comprobar su existencia.

El formato del fichero debe ser un directorio por cada línea.

**fileexists - fich\_existentes.dat** Incluye los ficheros de los que se desea comprobar su existencia.

El formato del fichero debe ser un fichero por cada línea.

**dirclean - dir\_eliminados.dat** Incluye los directorios de los que se desea comprobar su “no-existencia”, o lo que es lo mismo, de los que se desea comprobar que se hayan borrado.

El formato del fichero debe ser un directorio por cada línea.

**fileclean - fich\_eliminados.dat** Incluye los ficheros de los que se desea comprobar su “no-existencia”.

El formato del fichero debe ser un fichero por cada línea.

**perm - permisos.dat** Incluye los permisos que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y de los permisos en formato octal de 7 dígitos.

---

<sup>9</sup>Téngase en cuenta que debe existir un directorio para cada tipo de comprobación y dentro de ellos es donde deben estar estos ficheros. Para entenderlo mejor, puede observar la estructura de los sistemas ya desarrollados en la forja.

**own - propietarios.dat** Incluye los propietarios que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y del identificador del usuario (UID).

**grp - grupos.dat** Incluye los grupos propietarios que se deben comprobar de determinados elementos.

El formato del fichero debe ser el elemento (fichero o directorios) seguido de un espacio y del identificador del grupo (GID).

**mask - mascara.dat** Incluye la máscara de permisos que se desea que posea el sistema.

El formato del fichero debe ser una única línea con la máscara en formato octal de 4 dígitos.

**fileequals - fich\_iguales.dat** Incluye ficheros de los cuales se debe comprobar su igualdad.

El formato del fichero debe ser las parejas de ficheros separados por un espacio.

**direquals - dir\_iguales.dat** Incluye directorios de los cuales se debe comprobar su igualdad.

El formato del fichero debe ser las parejas de directorios separados por un espacio.

**content - contenido.dat** Incluye expresiones regulares que se desean encontrar en un determinado fichero. Se da por válida esta comprobación si se encuentra dicha expresión en una o más líneas del fichero.

El formato del fichero debe ser el fichero afectado seguido de la expresion regular a buscar.

**links - enlaces.dat** Incluye enlaces y ficheros enlazados que se desean comprobar.

El formato del fichero debe ser el fichero que contiene el enlace y el fichero que se supone que debe enlazar.

Por su parte, los ficheros de acción son:

**dirremove - dir\_a\_borrar.dat** Incluye los directorios que se desean eliminar.

El formato del fichero debe ser un directorio por cada línea.

**fileremove - fich\_a\_borrar.dat** Incluye los ficheros que se desean eliminar.

El formato del fichero debe ser un fichero por cada línea.

**changeperm - cambiar\_permisos.dat** Incluye los permisos que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido de los permisos en formato octal de 7 dígitos.

**changeown - cambiar\_propietarios.dat** Incluye los propietarios que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido del identificador del usuario (UID).

**changegrp - cambiar\_grupos.dat** Incluye los grupos propietarios que se desean modificar a determinados elementos.

El formato debe ser el elemento (fichero o directorio) seguido del identificador del grupo (GID).

**changemask - cambiar\_mascara.dat** Incluye la máscara de permisos que se desea configurar en el sistema.

El formato debe ser una única línea con los permisos de la máscara en formato octal de 4 dígitos.

**filerestore - fich\_a\_restaurar.dat** Incluye los ficheros que se desean restaurar en el sistema.

El formato debe ser el fichero fuente a restaurar seguido de un espacio y de la ruta de destino.

**commands - ordenes.dat** Incluye las órdenes que se desean ejecutar en el sistema.

El formato debe ser una línea por cada orden a ejecutar.

# Manual de instalación

---

## Introducción

En las siguientes secciones vamos a detallar la instalación que se debe de realizar para restaurar el contenido de WIKIUNIX en otro equipo.

## Requisitos para la instalación

Además del software en sí mismo, una instalación estándar de MediaWiki tiene los siguientes requisitos:

- ✍ PHP para ejecutar el software.
- ✍ Un servidor de bases de datos para archivar las páginas y los datos.
- ✍ Un servidor web para enviar las páginas generadas al navegador web.

Existen otras dependencias opcionales necesarias si se desean utilizar ciertas características avanzadas.

Los prerequisites pueden ser instalados de forma individual o a través del instalador de Xampp. Es importante leer la documentación de cada componente por separado antes de instalarlos. Si tu web se encuentra en un host (es decir, si no tienes control sobre ella), contacta con los administradores del servidor o con la compañía de hosting para confirmar que estén instalados.

Ten en cuenta que aunque MediaWiki usa un script de instalación que establece la configuración base de los elementos más importantes, es necesario algún conocimiento de PHP para personalizaciones más detalladas.

En este manual se detalla cómo instalar PHP5, MySQL y Apache2, aunque podrían emplearse otras herramientas.<sup>10</sup>

---

<sup>10</sup>Para más información acerca de estos requisitos, accede a la documentación de MediaWiki en torno a los requisitos de instalación ([http://www.mediawiki.org/wiki/Manual:Installation\\_requirements/es](http://www.mediawiki.org/wiki/Manual:Installation_requirements/es))

## Instalación de dependencias

A continuación se detalla la instalación en una distribución tipo Debian. Para información acerca de otros sistemas, accede al siguiente enlace:

[http://www.mediawiki.org/wiki/Manual:OS\\_specific\\_help](http://www.mediawiki.org/wiki/Manual:OS_specific_help)

Te será necesario instalar los metapaquetes *apache2*, *mysql-server*, *php5* y *php5-mysql*:

```
sudo apt-get update && sudo apt-get upgrade && sudo apt-get dist-upgrade
sudo apt-get install apache2 mysql-server php5 php5-mysql
```

Adicionalmente si quieres ejecutar órdenes PHP desde la línea de comandos (útil para búsqueda de errores y ejecutar scripts de mantenimiento), instala PHP-cli:

```
sudo apt-get install php5-cli
```

Si además quieres utilizar la biblioteca GD (en lugar de ImageMagic para redimensionar imágenes):

```
sudo apt-get install libgd2-xpm libgd2-xpm-dev php5-gd
```

Si no has configurado la clave de administrador de MySQL durante la instalación debes utilizar:

```
mysqladmin -u root password "<passwd>"
```

La configuración PHP no es necesaria: MediaWiki funcionará correctamente sin modificar nada. Si no sabes qué variables modificar mejor no toques nada, o acude a la página de instalación de MediaWiki ([http://www.mediawiki.org/wiki/Manual:Running\\_MediaWiki\\_on\\_Ubuntu](http://www.mediawiki.org/wiki/Manual:Running_MediaWiki_on_Ubuntu)) u otras páginas de referencia similares.

## Instalación de MediaWiki. Configuración.

En la carpeta *instalar* del material proporcionado hay un fichero comprimido con la versión 1.15.1 de MediaWiki (sobre la cual se ha desarrollado el wiki). Para realizar la instalación debes hacer lo siguiente:

- I. Descomprime el fichero en algún directorio accesible para el servidor web; por ejemplo, en */var/www/wikiunix*<sup>11</sup>:

```
cd /var/www
tar xvfz mediawiki-1.15.1.tar.gz
mv mediawiki-1.15.1 wikiunix
```

- II. Crea la base de datos donde alojaremos el contenido del wiki. Para ello ejecutamos:

---

<sup>11</sup>Es recomendable darle un nombre característico a este directorio por si ese servidor debe alojar más wikis en un futuro.



```
mysqladmin -u root -p create wikidb
mysql -u root -p
mysql> grant all privileges on wikidb.* to wikiuser@localhost identified by 'passwd';
mysql> flush privileges;
```

Se crea la base de datos *wikidb* y se le dan permisos a un usuario *wikiuser* nuevo, con contraseña *passwd*.

- III. Ve a la dirección donde has alojado el wiki para que se ejecute el script de configuración. De forma general es:

<http://localhost/wiki/config>

Nos pedirá una serie de datos de configuración. Luego se restaurarán los del tutorial, pero es importante establecer los datos del administrador de la base de datos MySQL correctamente.

- IV. Restaura la información en el wiki. Esta tarea viene detallada en la siguiente sección.

Como complemento a MediaWiki hemos instalado diversas extensiones:

- ✍ Geshi: Un resaltador de sintaxis para código de diversos lenguajes

[http://www.mediawiki.org/wiki/Extension:SyntaxHighlight\\_GeSHi](http://www.mediawiki.org/wiki/Extension:SyntaxHighlight_GeSHi)

- ✍ Cite: Da soporte a las citas y referencias del wiki

<http://www.mediawiki.org/wiki/Extension:Cite>

- ✍ Wiki2 $\text{\LaTeX}$ : Convierte los artículos del wiki a formato  $\text{\LaTeX}$

<http://www.mediawiki.org/wiki/Extension:Wiki2LaTeX>

En la carpeta *instalar* del material también se encuentran estas extensiones comprimidas. Si existe alguna duda acerca de cómo instalarlas, siempre se puede consultar la documentación oficial.

## Cómo restaurar el temario

Para restaurar el temario lo primero que tenemos que hacer es recuperar la base de datos en MySQL. Para ello vamos a utilizar la utilidad *mysqldump*. La sentencia que tenemos que introducir en consola para realizar esta recuperación es:

```
mysqldump --user=wikiuser --password=wikipasswd wikidb < wikidump.sql
```

Una vez restaurada la base de datos debemos de reestablecer, mediante el tarball que facilitamos con el contenido de la documentación *wikiuniz.tar.gz*, la estructura de ficheros de nuestro sitio wiki. Para ello sustituimos el contenido de la carpeta *wikiunix* por el árbol de ficheros que proporcionamos como material del proyecto donde se encuentran todos los recursos necesarios para el tutorial.

## Cómo hacer una copia de seguridad

Los pasos para realizar para obtener una copia de seguridad del wiki es el inverso al que realizamos para restaurarlo. En este caso tenemos que guardar la estructura de carpetas del servidor donde tenemos alojado el wiki. Puedes hacerlo de la manera que más te convenga<sup>12</sup>: simplemente tienes que asegurarte que guardas todo el árbol de directorios.

En cuanto a la base de datos tan sólo hay que utilizar la orden *mysqldump* en sentido inverso: si antes usábamos el operador inserción para restaurar la copia de seguridad, ahora utilizaremos el de extracción o redireccionamiento para sacar una copia de la base de datos. La orden quedaría de la siguiente manera:

```
mysqldump --user=wikiuser --password=wikipasswd wikidb > wikidump.sql
```

Con estos sencillos pasos tenemos garantizada una copia de seguridad<sup>13</sup> fiable de todo el producto. Es muy importante la realización de copias de seguridad periódicas en un sistema que sea modificable por un gran número de usuarios. Debemos prevenir los cambios erróneos que puedan provocar usuarios inexpertos o vándalos.

---

<sup>12</sup>En el wiki tienes diversos scripts para realizar backups que pueden ser útiles en este sentido.

<sup>13</sup>Realizar copias de seguridad es una buena técnica que proporciona estabilidad y fiabilidad al sistema.

# Software utilizado

---

En este anexo se incluyen los elementos utilizados para el desarrollo de este proyecto, ya sean sistemas operativos o herramientas.

## ✍ Sistemas operativos:

- \* GNU Linux. Ubuntu. Versión 9.10.
- \* (MÁQUINA VIRTUAL) GNU Linux. Xubuntu. Versión 9.10.

## ✍ Herramientas para el desarrollo:

- \* Compilador  $\text{\LaTeX}$ : *pdflatex*
- \* Editores:
  - ✧ GNU Emacs. Versión 23.1.50.1EXTENSIONES ADICIONALES:
  - ✧ *AucTeX*: entorno de desarrollo integrado  $\text{\LaTeX}$
  - ✧ *eshell*: emulador de terminal
  - ✧ *flyspell*: corrector ortográfico
  - ✧ *org-mode*: planificación general
  - ✧ *pylint*:
  - ✧ *python-mode*:
- ✧ Vi. Versión 7.2.245
- \* Editor de diagramas: *Dia*
- \* Editor de gráficas: *Google Chart*
- \* Editor de imágenes ráster: *The Gimp*. Versión 2.2.13
- \* Editor de imágenes vectoriales: *Inkscape*. Versión 0.45.1
- \* Planificación de proyectos: *Planner*
- \* Realización de un *screencast*: *XVidCap*
- \* Sistema de control de versiones: *Subversion* [62]

## ✍ Plataforma del producto:

- \* Wiki:
  - ✧ Servidor web: *Apache 2*. Versión 2.2.3

- ✧ Motor Wiki: *MediaWiki*. Versión 1.15.1
- EXTENSIONES:
  - ✧ *Cite*: citas y referencias bibliográficas
  - ✧ *GeSHi Syntax Highlighter*: resaltador de sintaxis
- ✧ *PHP*. Versión 5.2.5
- ✧ *MySQL*. Versión 5.0.26
- ✧ Máquina virtual:
  - ✧ *VirtualBox*. Versión 3.0.8<sup>14</sup>

A continuación se incluyen las descripciones completas acerca de las herramientas mencionadas.

## Apache2

El servidor elegido para las pruebas del sitio wiki es *Apache*. El servidor HTTP *Apache* es un software (libre) servidor HTTP de código abierto para plataformas UNIX, Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

*Apache* presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

En este sentido, podríamos haber optado por utilizar *Cherokee*, un servidor web libre multiplataforma, disponible bajo la licencia GPL. Es bastante rápido y también soporta las funcionalidades más comunes de servidor. Está escrito completamente en C, es escalable y puede usarse como un Sistema integrado.

Apache tiene amplia aceptación en la red: en 2005, Apache fue el servidor HTTP más usado, siendo el servidor empleado en el 48 % de los sitios web en el mundo. Sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años.

## Dia

*Dia* es una aplicación informática de propósito general para la creación de diagramas, desarrollada como parte del proyecto GNOME. Está concebido de forma modular, con diferentes paquetes de formas para diferentes necesidades.

Se puede utilizar para dibujar diferentes tipos de diagramas. Actualmente se incluyen diagramas entidad-relación, diagramas UML, diagramas de flujo, diagramas de redes, diagramas de circuitos eléctricos, etc. Nuevas formas pueden ser fácilmente agregadas, dibujándolas con un subconjunto de SVG e incluyéndolas en un archivo XML.

El formato para leer y almacenar gráficos es XML (comprimido con gzip, para ahorrar espacio). Puede producir salida en los formatos EPS, SVG y PNG.

---

<sup>14</sup>La versión mínima requerida de VirtualBox es la 2.2.0

Para más información acerca de *Dia*, consulte [50]

## Emacs

*Emacs* es uno de los programas de edición más usados por desarrolladores de software, ya que es bastante versátil admitiendo gran cantidad de extensiones que permiten ampliar aún más sus funcionalidades.

### ✍ AucTeX

AucTeX es una extensión que proporciona un entorno de desarrollo integrado para TeX, LaTeX, TeXinfo y ConTeXt, agregando muchas funciones importantes que no están incluidos en el modo de TeX estándar. Así, la redacción de artículos científicos se facilita enormemente.

Puede ejecutar compiladores de TeX y programas complementarios, tales como BibTeX, y visores de PDF y DVI.

Incluye vista previa de LaTeX, generadas con PostScript y muestra directamente en el editor tablas, fórmulas, e imágenes.

Junto con el `reftex-mode`, que es parte de Emacs, puede manipular con facilidad referencias cruzadas, y también bibliográficas.

AucTeX es software libre utilizando la Licencia Pública General de GNU.

### ✍ eshell

Esta extensión viene instalada por defecto en la versión de *emacs* empleada. Es muy útil puesto que se puede realizar cualquier orden como si estuviéramos en una terminal real.

Por ejemplo, para ejecutar en la línea de comandos *make* y mostrar la salida tan solo debemos utilizar `M-! make RET` e introducir la orden.

### ✍ flyspell

Emacs incluye *ispell* para la corrección automática de texto. *Flyspell* tiene el mismo efecto que *ispell*, pero este se realiza en tiempo real, a medida que vamos escribiendo cualquier palabra “on the fly”.

Para ejecutar *flyspell* tan sólo debemos invocarlo mediante `M-x flyspell-mode`, aunque en nuestro caso hemos modificado el fichero *.emacs* para que se ejecute automáticamente al cargar ficheros TeX.

### ✍ org-mode

Esta extensión sirve para guardar notas, mantener listas *ToDo*, haciendo la planificación de proyectos y la redacción en modo texto simples, rápidas y eficaces.[51]

Entre sus ventajas tenemos:

- ✱ Simplicidad: Aprender los principios básicos no lleva más de 5 minutos.
- ✱ Portabilidad: Los ficheros son de texto plano, por lo que se pueden ver y modificar desde cualquier editor.

- ✱ Profundidad y flexibilidad: Sirve para hacer muchas cosas, tengan o no que ver con la planificación. Incluso documentos en PDF, sin tener que utilizar código  $\text{\LaTeX}$  directamente (aunque no con tanta eficacia lógicamente).
- ✱ Soporte: Hay multitud de tutoriales y la comunidad es enorme.
- ✱ Es software libre.

### 🔪 **pylint**

Pylint es un instrumento de Python que comprueba si un módulo satisface un estándar de codificación. Es similar a PyChecker, pero ofrece más rasgos, como la comprobación de la longitud del código de línea, la comprobación de si nombres de variables son gramaticalmente correctos según su estándar de codificación, o la comprobación de si las interfaces declaradas realmente son puestas en práctica, entre otras cosas.[52]

Puede comprobar la lista completa de comprobaciones desde [53]

### 🔪 **python-mode**

Es una extensión que proporciona un entorno de desarrollo integrado para Python. Posee un gran número de características de edición y búsqueda de fallos disponible para los programadores de Python que usen Emacs. Sus características básicas son:

- ✱ resaltado de sintaxis
- ✱ soporte de indentado y “desindentado”
- ✱ ejecución de código en otro *buffer*
- ✱ marcado de clase/función
- ✱ integración con el *debugger* de Python

Para más información acerca de *Emacs* y sus extensiones, consulte [54] [55]

## **Gimp**

GIMP (*GNU Image Manipulation Program*) es un programa de edición de imágenes digitales en forma de mapa de bits, tanto dibujos como fotografías. Es un programa libre y gratuito. Está englobado en el proyecto GNU y disponible bajo la Licencia pública general de GNU.

La primera versión de GIMP se desarrolló inicialmente en sistemas Unix y fue pensada especialmente para GNU/Linux. Se incluye en muchas distribuciones GNU/Linux. También se ha portado a otros sistemas operativos, haciéndolo el programa de manipulación de gráficos disponible en más sistemas operativos.

Para más información acerca de *Gimp*, consulte [56]

## **Google Chart**

Google Chart es un instrumento sumamente simple que permite crear una gráfica de algunos datos e integrarlo en una página web. Se integran los datos y parámetros a formatear en una petición HTTP,

y se devuelve una imagen PNG de la gráfica.

Al principio esto era un instrumento interno para apoyar la integración rápida de gráficas dentro de los propios usos de Google (como Finanzas de Google por ejemplo). Google imaginó que esto sería un instrumento útil para desarrolladores web.

Se lanzó oficialmente el 6 de diciembre de 2007. Actualmente están soportados varios tipos de diagramas, como los diagramas lineales, los diagramas de barras, los mapas o los diagramas de Venn.

Para más información acerca de Google Chart, consulte [\[57\]](#)

## GNU Make

*GNU Make* es el programa de recompilación y de control de dependencias por excelencia. Se puede utilizar para compilar proyectos software en diversos códigos, o como en el caso de este documento, para compilar documentos  $\text{\LaTeX}$  con diversas opciones.

Para más información acerca de GNU Make, consulte [\[58\]](#)

## Planner

*Planner* es una herramienta para planear, programar y seguir proyectos para el escritorio GNOME. Es una aplicación GTK+ escrita en C y licenciada bajo GPL (versión 2 o posterior).

Planner puede almacenar sus datos en ficheros XML o en una base de datos postgresql. Los proyectos pueden ser imprimidos en PDF o exportados a HTML para una visualización simple desde cualquier navegador web.

El programa permite:

- ✍ Gestión de calendarios
- ✍ Gestión de recursos
- ✍ Seguimiento del avance del proyecto
- ✍ Enlazar tareas
- ✍ Exportación a diferentes formatos

## Inkscape

*Inkscape* es una herramienta de dibujo libre y multiplataforma para gráficos vectoriales SVG. Este programa surgió de una bifurcación del proyecto *Sodipodi*. Las características de SVG soportadas incluyen formas básicas, trayectorias, texto, canal alfa, transformaciones, gradientes, edición de nodos, exportación de SVG a PNG, agrupación de elementos, etc.

La principal motivación de Inkscape es proporcionar a la comunidad una herramienta de código abierto de dibujo SVG que cumpla completamente con los estándares XML, SVG y CSS.

Inkscape se encuentra desarrollado principalmente para el sistema operativo GNU/Linux, pero es una herramienta multiplataforma que funciona en Windows, Mac OS X, y otros sistemas derivados de UNIX.

Esta aplicación tiene una buena compatibilidad multilingüe, particularmente para sistemas de escritura complejos (árabe, hebreo...)

Para más información acerca de *Inkscape*, consulte [\[59\]](#) [\[60\]](#)

## MediaWiki

MediaWiki es un motor para wikis bajo licencia GNU, programado en PHP. A pesar de haber sido creado y desarrollado para Wikipedia y otros proyectos de la fundación Wikimedia, ha tenido una gran expansión desde el año 2005, existiendo un gran número de wikis basados en este software que nada tienen que ver con dicha fundación, aunque si comparten la idea de la generación de contenidos de manera colaborativa. La mayoría de ellos se dedican a la documentación de software o a temas especializados.

Puede ser instalado sobre servidores Web Apache o IIS y puede usar como motor de base de datos MySQL o PostgreSQL.

Para más información acerca de MediaWiki, consulte [\[39\]](#) [\[34\]](#) [\[40\]](#)

## Subversion

*Subversion* es un software de sistema de control de versiones diseñado específicamente para reemplazar al popular CVS, el cual posee varias deficiencias. Es software libre bajo una licencia de tipo Apache/BSD y se le conoce también como *svn* por ser ese el nombre de la herramienta de línea de comandos. Una característica importante de Subversion es que, a diferencia de CVS, los archivos versionados no tienen cada uno un número de revisión independiente. En cambio, todo el repositorio tiene un único número de versión que identifica un estado común de todos los archivos del repositorio en cierto punto del tiempo.

*Subversion* puede acceder al repositorio a través de redes, lo que le permite ser usado por personas que se encuentran en distintos ordenadores. A cierto nivel, la capacidad para que varias personas puedan modificar y administrar el mismo conjunto de datos desde sus respectivas ubicaciones fomenta la colaboración. Se puede progresar mas rápidamente sin un único conducto por el cual deban pasar todas las modificaciones. Y puesto que el trabajo se encuentra bajo el control de versiones, no hay razón para temer por que la calidad del mismo vaya a verse afectada por la pérdida de ese conducto único—si se ha hecho un cambio incorrecto a los datos, simplemente deshaga ese cambio.

Para más información acerca de *Subversion*, consulte [\[61\]](#) [\[62\]](#) [\[63\]](#)



## pywikipedia

*Pywikipedia* es la API para Wikipedia en Python. Sirve para instalar y utilizar un sistema de bots, e incluso desarrollar otros propios, para automatizar o simplificar tareas de edición en cualquier proyecto basado en un motor *MediaWiki*.<sup>[41]</sup>

El proyecto *pywikipedia* ha desarrollado una librería de recursos en Python que implementa las funciones necesarias para interactuar con el software de MediaWiki; el software incluye varios bots, capaces de realizar un gran número de tareas. Si bien otros bots especializados se emplean ocasionalmente, la mayoría de los bots activos en Wikilibros utilizan este software.

Para más información acerca de *pywikipedia*, consulte <sup>[42]</sup>

## Vi

*Vi* (Visual) es un simple editor de texto, que no lo formatea en absoluto, pues no centra ni justifica párrafos pero permite mover, copiar, eliminar o insertar caracteres por medio del búfer permaneciendo la información ahí hasta que los cambios en el archivo se hayan guardado o bien hasta que termine la ejecución de la aplicación sin haber guardado las modificaciones.

*Vi* fue originalmente escrito por Bill Joy en 1976, tomando recursos de *ed* y *ex*, dos editores de texto deficientes para Unix, que trataban de crear y editar archivos, de ahí, la creación de *vi*.

Hay una versión mejorada que se llama *Vim*, pero *Vi* es un editor de texto que se encuentra en (casi) todo sistema de tipo UNIX, de forma que conocer rudimentos de *Vi* es una salvaguarda ante operaciones de emergencia en diversos sistemas operativos.

Para más información acerca de *Vi*, consulte <sup>[7]</sup> <sup>[8]</sup> <sup>[9]</sup>

## VirtualBox

*VirtualBox* es un software de virtualización para arquitecturas x86 que fue desarrollado originalmente por la empresa alemana Innotek GmbH, pero que pasó a ser propiedad de la empresa Sun Microsystems en febrero de 2008 cuando ésta compró a innotek. Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como “sistemas invitados”, dentro de otro sistema operativo “anfitrión”, cada uno con su propio ambiente virtual.

La aplicación fue inicialmente ofrecida bajo una licencia de software privado, pero en enero de 2007, después de años de desarrollo, surgió VirtualBox OSE (Open Source Edition) bajo la licencia GPL 2. Actualmente existe la versión privada, VirtualBox, que es gratuita únicamente bajo uso personal o de evaluación, y esta sujeta a la licencia de “Uso Personal y de Evaluación VirtualBox” (VirtualBox Personal Use and Evaluation License o PUEL) y la versión Open Source, VirtualBox OSE, que es software libre, sujeta a la licencia GPL.

En comparación con otras aplicaciones privadas de virtualización, como VMware Workstation, Vir-

VirtualBox carece de algunas funcionalidades, pero provee de otras como la ejecución de máquinas virtuales de forma remota, por medio del Remote Desktop Protocol (RDP), soporte iSCSI.

En cuanto a la emulación de hardware, los discos duros de los sistemas invitados son almacenados en los sistemas anfitriones como archivos individuales en un contenedor llamado Virtual Disk Image, incompatible con los demás software de virtualización.

Otra de las funciones que presenta es la de montar imágenes ISO como unidades virtuales de CD o DVD, o como un disco floppy.

Para más información acerca de VirtualBox, consulte [\[64\]](#) [\[65\]](#)

## XVidCap Screen Capture

*XVidCap* es una aplicación de software libre utilizada para grabar screencast o grabación digital de eventos de escritorio X Window con narración en audio. Las herramientas de captura de pantalla son muy útiles para los desarrolladores para documentar procedimientos de instalación y del software. También pueden ser utilizados por educadores para ayudarles a preparar tutoriales sobre otro software.

Pretendía ser una alternativa estándar, de código abierto a otras aplicaciones similares de software comercial como por ejemplo Lotus ScreenCam para plataformas UNIX.

Para más información acerca de *XVidCap*, consulte [\[66\]](#)

# Lenguajes utilizados

---

En este anexo se incluyen los lenguajes de programación y de sintaxis utilizados en el desarrollo de este proyecto.

## ✍ Lenguajes de programación:

- \* Bash y AWK para la elaboración de ejemplos prácticos.
- \* HTML y CSS para la personalización del interfaz wiki y la inclusión de nuevas plantillas.
- \* PHP para la modificación y personalización de aspectos del motor wiki.
- \* Python para la elaboración de ejemplos y scripts de control.

## ✍ Lenguajes de sintaxis:

- \*  $\text{\LaTeX}$  para la elaboración de todo el material documentativo del proyecto, así como una instancia estática del wiki en PDF.
- \* Lenguaje Wiki para la edición directa en el wiki.

## AWK

AWK es un lenguaje de programación diseñado para procesar datos basados en texto, ya sean ficheros o flujos de datos. El nombre AWK deriva de los apellidos de los autores: Alfred Aho, Peter Weinberger, y Brian Kernighan. `awk`, cuando está escrito todo en minúsculas, hace referencia al programa de Unix o Plan 9 que interpreta programas escritos en el lenguaje de programación AWK.

AWK es ejemplo de un lenguaje de programación que usa ampliamente el tipo de datos de listas asociativas (es decir, listas indexadas por cadenas clave), y expresiones regulares. El poder, brevedad y limitaciones de los programas de AWK y los guiones de sed inspiraron a Larry Wall a escribir Perl. Debido a su densa notación, todos estos lenguajes son frecuentemente usados para escribir programas de una línea.

AWK fue una de las primeras herramientas en aparecer en Unix (en la versión 3) y ganó popularidad como una manera de añadir funcionalidad a las tuberías de Unix. La implementación de alguna versión del lenguaje AWK es estándar en casi todo sistema operativo tipo unix moderno. AWK es mencionado en las Single UNIX Specification (especificaciones básicas de unix) como una de las utilidades necesarias de todo sistema operativo Unix. Se pueden instalar implementaciones de AWK en casi todos los demás sistemas operativos.

Para más información acerca de *AWK*, consulte [10]

## Bourne-Again Shell

*Bash* está basado en la shell de Unix y es compatible con POSIX. Fue escrito para el proyecto GNU y es el intérprete de comandos por defecto en la mayoría de las distribuciones de Linux. Su nombre es un acrónimo de *Bourne-Again Shell* (otro shell bourne) — haciendo un juego de palabras (“born-again” significa renacimiento) sobre el *Bourne shell* (sh), que fue uno de los primeros intérpretes importantes de Unix.

Para más información acerca de *Bash*, consulte [11] [12]

## HTML y CSS

HTML (*HyperText Markup Language*) es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de “etiquetas”, rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

Por otro lado, CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

Para más información acerca de HTML y CSS, consulte [67] [68] [69]

## L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X es un sistema de composición de textos, orientado especialmente a la creación de libros, documentos científicos y técnicos que contengan fórmulas matemáticas.

Está formado por un gran conjunto de macros de T<sub>E</sub>X, escrito por Leslie Lamport en 1984, con la intención de facilitar el uso del lenguaje de composición tipográfica, creado por Donald Knuth. Es muy utilizado para la composición de artículos académicos, tesis y libros técnicos, dado que la calidad tipográfica de los documentos realizados con L<sup>A</sup>T<sub>E</sub>X es comparable a la de una editorial científica de primera línea.

L<sup>A</sup>T<sub>E</sub>X es software libre bajo licencia LPPL.

Para más información acerca de  $\text{\LaTeX}$ , consulte [15] [16] [14] [17] [18] [19] [20]

## PHP

*PHP* es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

Es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

Para más información acerca de PHP, consulte [43] [70]

## Python

*Python* es un lenguaje de programación interpretado creado por Guido van Rossum en el año 1991.

Se compara habitualmente con Tcl, Perl, Scheme, Java y Ruby. En la actualidad Python se desarrolla como un proyecto de código abierto, administrado por la Python Software Foundation.

Python es considerado como la .ºposición leal.<sup>a</sup> Perl, lenguaje con el cual mantiene una rivalidad amistosa. Los usuarios de Python consideran a éste mucho más limpio y elegante para programar.

Python permite dividir el programa en módulos reutilizables desde otros programas Python. Viene con una gran colección de módulos estándar que se pueden utilizar como base de los programas (o como ejemplos para empezar a aprender Python). También hay módulos incluidos que proporcionan E/S de ficheros, llamadas al sistema, sockets y hasta interfaces a GUI (interfaz gráfica con el usuario) como Tk, GTK, Qt entre otros.

Python se utiliza como lenguaje de programación interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar funciones durante el desarrollo del programa.

El nombre del lenguaje proviene de la afición de su creador original, Guido van Rossum, por los humoristas británicos Monty Python. El principal objetivo que persigue este lenguaje es la facilidad, tanto de lectura, como de diseño.

Para más información acerca de Python, consulte [71] [72] [73] [74] [75] [13]



# Aportaciones a la comunidad

---

Para complementar el desarrollo del proyecto, se ha trabajado en otros sistemas que trabajan con el motor *MediaWiki*.

Entre las ediciones realizadas en estos sistemas, algunas tienen mucho que ver con el proyecto, puesto que fueron realizadas gracias a que hacía falta utilizar dicho contenido en la documentación.

A continuación se muestran algunos de los artículos en los que ha participado la autora:

✍ WIKIPEDIA. *La enciclopedia libre*.

- I. Open Virtualization Format
  - ✳ Creación del artículo en español de *Open Virtualization Format*
- II. OVF
  - ✳ Redirección al artículo “Open Virtualization Format”
- III. Planner
  - ✳ Creación del artículo en español de *Planner*
- IV. Unix
  - ✳ Enlace roto eliminado
  - ✳ Adición del artículo a WIKIUNIX
- V. XVidCap
  - ✳ Creación del artículo en español de *XVidCap*

✍ WIKIPEDIA. *The Free Encyclopedia*.

- I. Open Virtualization Format
  - ✳ Adición de interwiki: enlace al nuevo artículo en Español
- II. Planner (project management)
  - ✳ Creación del artículo en inglés de *Planner*
- III. XVidCap
  - ✳ Adición de interwiki: enlace al nuevo artículo en Español

✍ *MediaWiki.org*

- I. Category:Navigation templates/es

- \* Creación de la categoría

II. Installation/es

- \* Actualización de los enlaces (obsoletos)

III. Manual:Installation requirements/es

- \* Creación de la versión en español para el artículo “Installation\_requirements”
- \* Adición de la sección “database server” traducida al español
- \* Traducción de nota
- \* Adición de la sección “webserver” traducida al español
- \* Adición de la sección “hardware requirements” traducida al español
- \* Adición de la sección “optional dependencies” traducida al español

IV. Template:InstallationNav/es

- \* Creación de la versión en español para la plantilla “InstallationNav”

v. Template:Warning/es

- \* Creación de la versión en español para la plantilla “Warning”



## Código referenciado

---

En este anexo se incluye una muestra de los códigos incluidos y relacionados con el proyecto, que han sido referenciados a lo largo de la Memoria del Proyecto Fin de Carrera.

Todos los códigos contenidos contienen el siguiente mensaje indicando su licencia:

### LICENCIA GPL

---

```
1  # Creado por Noelia Sales Montes
2
3  # This program is free software: you can redistribute
4  # it and/or modify it under the terms of the GNU General
5  # Public License as published by the Free Software
6  # Foundation, either version 3 of the License, or any
7  # later version.
8
9  # This program is distributed in the hope that it will
10 # be useful, but WITHOUT ANY WARRANTY; without even the
11 # implied warranty of MERCHANTABILITY or FITNESS FOR A
12 # PARTICULAR PURPOSE. See the GNU General Public License
13 # for more details.
14
15 # You should have received a copy of the GNU General
16 # Public License along with this program. If not, see
17 # <http://www.gnu.org/licenses/>.
18
19 # Contacto:
20 #          noelia.salesmontes@gmail.com
```

---

## Scripts de control

### CONTROL: CONTROL.PY

---

```
1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
```

## Código referenciado

---

```
3
4 # USO CORRECTO: sudo python script.py [-v] [-h/-c/-p <num>]
5 # * -v -> modo chivato activado
6 # * -h o --help -> ayuda
7 # * -c o --clean -> limpieza previa
8 # * -p <n> o --partial <n> -> comprobación intermedia
9 #                               número n
10 # * -s o --show -> muestra los ejercicios a realizar
11 # Debe ser ejecutado con sudo o bajo permisos de admi-
12 # nistrador o no funcionará correctamente
13
14 # Creado por Noelia Sales Montes
15
16 # This program is free software: you can redistribute
17 # it and/or modify it under the terms of the GNU General
18 # Public License as published by the Free Software
19 # Foundation, either version 3 of the License, or any
20 # later version.
21
22 # This program is distributed in the hope that it will
23 # be useful, but WITHOUT ANY WARRANTY; without even the
24 # implied warranty of MERCHANTABILITY or FITNESS FOR A
25 # PARTICULAR PURPOSE. See the GNU General Public License
26 # for more details.
27
28 # You should have received a copy of the GNU General
29 # Public License along with this program. If not, see
30 # <http://www.gnu.org/licenses/>.
31
32 # Contacto:
33 #         noelia.salesmontes@gmail.com
34
35 """
36 Script de comprobación del proyecto WikiUNIX
37 Comprueba la resolución de ejercicios
38 """
39
40 import os
41 import getopt
42 import sys
43 import filecmp
44 import stat
45 import re
46 import shutil
47
48 # Directorios de comprobación
49 __base__ = ".correccion/base"
50 __clean__ = ".correccion/limpia"
```

---

```

51 __partial__ = ".correccion/parcial"
52 __final__ = ".correccion/final"
53 # Fichero que contiene datos a mostrar
54 __exercises__ = ".correccion/ejercicios.dat"
55 __usage__ = ".correccion/uso.dat"
56
57 # Tipos de ficheros de comprobación
58 __direxists__ = "dir_existentes.dat"
59 __fileexists__ = "fich_existentes.dat"
60 __dirclean__ = "dir_eliminados.dat"
61 __fileclean__ = "fich_eliminados.dat"
62 __perm__ = "permisos.dat"
63 __own__ = "propietarios.dat"
64 __grp__ = "grupos.dat"
65 __mask__ = "mascara.dat"
66 __fileequals__ = "fich_iguales.dat"
67 __direquals__ = "dir_iguales.dat"
68 __content__ = "contenido.dat"
69 __links__ = "enlaces.dat"
70
71 # Tipos de ficheros de acciones
72 __dirremove__ = "dir_a_borrar.dat"
73 __fileremove__ = "fich_a_borrar.dat"
74 __changeperm__ = "cambiar_permisos.dat"
75 __changeown__ = "cambiar_propietarios.dat"
76 __changegrp__ = "cambiar_grupos.dat"
77 __changemask__ = "cambiar_mascara.dat"
78 __filerestore__ = "fich_a_restaurar.dat"
79 __dirrestore__ = "dir_a_restaurar.dat"
80 __commands__ = "ordenes.dat"
81
82 # Funciones y modos
83 __verbose__ = False
84 __fun__ = __final__
85
86 # -----
87 #                               FUNCIONES AUXILIARES
88 # -----
89
90 def usage():
91     """
92     Impresión de la utilización del script
93     """
94     if os.path.exists(__usage__):
95         with open(__usage__, 'r') as __f:
96             for __l in __f.read().split('\n'):
97                 print __l
98         __f.close()

```

---

```
99     else:
100         error("No existe el fichero de muestra de uso")
101
102 def error(problem):
103     """
104     Función auxiliar para simplificar la salir con error
105     """
106     print "Error: %s" % problem
107     sys.exit(2)
108
109 def verb(__line):
110     """
111     Función auxiliar para simplificar el modo chivato
112     """
113     if __verbose__ == True:
114         print __line
115
116 def show():
117     """
118     Función auxiliar que se encarga de mostrar todos los ejercicios relacionados
119     con este script
120     """
121     if os.path.exists(__exercises__):
122         with open(__exercises__, 'r') as __f:
123             for __l in __f.read().split('\n'):
124                 print __l
125             __f.close()
126     else:
127         error("No existe el fichero de muestra de ejercicios")
128
129 def remove(__file):
130     """
131     Función auxiliar para simplificar el borrado de un único fichero
132     """
133     if os.path.exists(__file):
134         try:
135             os.remove(__file)
136             verb("    -> Borrando fichero %s <-" % __file)
137         except os.error:
138             verb("No se ha podido borrar el fichero %s" % __file)
139
140 def to_decimal(__int):
141     """
142     Conversión de una cadena en octal a un número en decimal
143     """
144     ind = len(__int) - 1
145     dec = 0
146     for __l in __int:
```

```
147         dec += int(__l) * 8**ind
148         ind -= 1
149     return dec
150
151 def removedir(__dir):
152     """
153     Función auxiliar para simplificar el borrado de un directorio
154     """
155     if os.path.exists(__dir):
156         try:
157             os.removedirs(__dir)
158             verb("    -> Borrando directorio %s <-" % __dir)
159         except os.error:
160             verb("No se ha podido borrar el directorio %s" % __dir)
161
162
163 def direxists(__dir):
164     """
165     Función auxiliar para simplificar la comprobación de existencia de un
166     directorio
167     """
168     if os.path.exists(__dir) and os.path.isdir(__dir):
169         return True
170     else:
171         return False
172
173 def notdirexists(__dir):
174     """
175     Función auxiliar para simplificar la comprobación de no existencia de un
176     directorio
177     """
178     return not direxists(__dir)
179
180 def fileexists(__file):
181     """
182     Función auxiliar para simplificar la comprobación de existencia de un
183     fichero
184     """
185     if os.path.exists(__file) and os.path.isfile(__file):
186         return True
187     else:
188         return False
189
190 def notfileexists(__file):
191     """
192     Función auxiliar para simplificar la comprobación de no existencia
193     de un fichero
194     """
```

```
195     return not fileexists(__file)
196
197
198 def comp_perm(__elem, __perm):
199     """
200     Comprueba si un elemento (fichero o directorio) tiene unos determinados
201     permisos (estrictamente)
202     """
203     if os.path.exists(__elem):
204         try:
205             # Permisos en octal y en forma de cadena
206             mode = str(oct(os.stat(__elem)[stat.ST_MODE]))
207         except os.error:
208             error("No se han podido leer los permisos de %s" % __elem)
209
210         if mode == __perm:
211             return True
212         else:
213             return False
214     else:
215         verb("No existe %s" % __elem)
216         return False
217
218 def comp_user(__elem, __user):
219     """
220     Comprueba si un elemento (fichero o directorio) es poseído por un usuario
221     en función de su identificador
222     """
223     if os.path.exists(__elem):
224         try:
225             __st = os.stat(__elem)
226             __uid = __st[stat.ST_UID]
227         except os.error:
228             error("No se ha podido leer el usuario de %s" % __elem)
229
230         if __uid == __user:
231             return True
232         else:
233             return False
234     else:
235         return False
236
237 def comp_grp(__elem, __grp):
238     """
239     Comprueba si un elemento (fichero o directorio) es poseído por un grupo en
240     función de su identificador
241     """
242     if os.path.exists(__elem):
```

---

```

243     try:
244         __st = os.stat(__elem)
245         gid = __st[stat.ST_GID]
246     except os.error:
247         error("No se ha podido leer el grupo de %s" % __elem)
248
249     if gid == __grp:
250         return True
251     else:
252         return False
253 else:
254     return False
255
256 def comp_mask(__mask):
257     """
258     Comprueba la máscara de permisos del sistema
259     """
260     try:
261         mas = os.umask(__mask)
262     except os.error:
263         error("No se ha podido leer la máscara de permisos")
264
265     if mas == __mask:
266         return True
267     else:
268         os.umask(__mask)
269         return False
270
271 def comp_content(__fich, __er):
272     """
273     Comprueba que una determinada expresión regular se encuentra en una de las
274     líneas del fichero
275     """
276     if os.path.exists(__fich):
277         with open(__fich, 'r') as __f:
278             for __l in __f.read().split('\n'):
279                 exp = re.compile(__er)
280                 match = exp.match(__l)
281                 if match:
282                     return True
283             __f.close()
284             # Si no se encuentra coincidencia
285             return False
286
287 def comp_link(__elem, __src):
288     """
289     Comprueba el enlace de un fichero
290     """

```

---

```
291     if os.path.exists(__elem):
292         return (os.readlink(__elem) == __src)
293
294
295 def chg_own(__elem, __user):
296     """
297     Cambia el propietario de un elemento (fichero o directorio)
298     """
299     if os.path.exists(__elem):
300         __st = os.stat(__elem)
301         gid = __st[stat.ST_GID]
302         try:
303             os.chown(__elem, int(__user), gid)
304         except os.error:
305             error("No se ha podido modificar el usuario de %s" % __elem)
306
307 def chg_grp(__elem, __grp):
308     """
309     Cambia el grupo de un elemento (fichero o directorio)
310     """
311     if os.path.exists(__elem):
312         __st = os.stat(__elem)
313         uid = __st[stat.ST_UID]
314         try:
315             os.chown(__elem, uid, int(__grp))
316         except os.error:
317             error("No se ha podido modificar el usuario de %s" % __elem)
318
319 def chg_perm(__elem, __perm):
320     """
321     Cambiar los permisos de un elemento (fichero o directorio)
322     """
323     if os.path.exists(__elem):
324         try:
325             os.chmod(__elem, to_decimal(__perm))
326         except os.error:
327             error("No se han podido modificar los permisos de %s" % __elem)
328
329 def chg_mask(__mask):
330     """
331     Cambia la máscara de permisos de archivos
332     """
333     try:
334         os.umask(int(__mask))
335     except os.error:
336         error("No se ha podido modificar la máscara del sistema")
337
338
```



---

```

339 def check_simple(__funcion, __actual, __mensaje, __verbose):
340     """
341     Comprobación básica que recibe una función y un único argumento
342     """
343     if fileexists(__actual):
344         verb("%s (%s)" % (__verbose, __actual))
345         # Abrimos el fichero para realizar la comprobación pertinente
346         with open(__actual, 'r') as __f:
347             for __l in __f.read().split('\n'):
348                 if not __funcion(__l):
349                     error("%s %s" % (__mensaje, __l))
350             __f.close()
351
352 def check_doble(__funcion, __actual, __mensaje, __verbose):
353     """
354     Comprobación doble que recibe una función y dos argumentos
355     """
356     if fileexists(__actual):
357         verb("%s (%s)" % (__verbose, __actual))
358         # Abrimos el fichero para realizar la comprobación pertinente
359         with open(__actual, 'r') as __f:
360             for __l in __f.read().split('\n'):
361                 __elema = __l[:__l.find(' ')]
362                 __elemb = __l[__l.find(' ') + 1:]
363                 if not __funcion(__elema, __elemb):
364                     error("%s %s (%s)" % (__mensaje, __elema, __elemb))
365             __f.close()
366
367
368 def change_simple(__funcion, __actual, __mensaje, __verbose):
369     """
370     Modificación simple que recibe una función y un único argumento
371     """
372     if fileexists(__actual):
373         verb("%s (%s)" % (__verbose, __actual))
374         with open(__actual, 'r') as __f:
375             for __l in __f.read().split('\n'):
376                 __funcion(__l)
377             __f.close()
378
379 def change_doble(__funcion, __actual, __mensaje, __verbose):
380     """
381     Modificación doble que recibe una función y dos argumentos
382     """
383     if fileexists(__actual):
384         verb("%s (%s)" % (__verbose, __actual))
385         with open(__actual, 'r') as __f:
386             for __l in __f.read().split('\n'):

```

---

```
387         __elema = __l[:__l.find(' ')]
388         __elemb = __l[__l.find(' ') + 1:]
389         __funcion(__elema, __elemb)
390     __f.close()
391
392
393 # -----
394 #             FUNCIÓN GENERAL DE LIMPIEZA Y COMPROBACIÓN
395 # -----
396 def check(__comp):
397     """
398     Se pasa el nombre del tipo de comprobación a realizar (que debe
399     corresponderse al directorio de comprobación adecuado) y se ejecuta dicha
400     verificación
401     """
402     # Comprobaciones de existencia
403     check_simple(direxists, os.path.join(__comp, __direxists__),
404                 "No existe el directorio",
405                 " - Comprobación de existencia de directorios")
406
407     check_simple(fileexists, os.path.join(__comp, __fileexists__),
408                 "No existe el fichero",
409                 " - Comprobación de existencia de ficheros")
410
411     check_simple(notdirexists, os.path.join(__comp, __dirclean__),
412                 "Aún existe el directorio",
413                 " - Comprobación de no existencia de directorios")
414
415     check_simple(notfileexists, os.path.join(__comp, __fileclean__),
416                 "Aún existe el fichero",
417                 " - Comprobación de no existencia de ficheros")
418
419     # Comprobaciones de permisos
420     check_doble(comp_perm, os.path.join(__comp, __perm__),
421                 "No coinciden los permisos de",
422                 " - Comprobación de los permisos de los ficheros")
423
424     check_doble(comp_user, os.path.join(__comp, __own__),
425                 "No coinciden los permisos de",
426                 " - Comprobación del propietario de los ficheros")
427
428     check_doble(comp_grp, os.path.join(__comp, __grp__),
429                 "No coinciden los permisos de",
430                 " - Comprobación del grupo de los ficheros")
431
432     check_simple(comp_mask, os.path.join(__comp, __mask__),
433                 "No coincide la máscara del sistema -",
434                 " - Comprobación de la máscara del sistema")
```

```
435
436     check_doble(comp_content, os.path.join(__comp, __content__),
437             "No se encuentra coincidencia",
438             " - Comprobación de contenido de ficheros")
439
440     # Comprobación de enlaces
441     check_doble(comp_link, os.path.join(__comp, __links__),
442             "No coincide el enlace de ",
443             " - Comprobación de enlaces")
444
445     # Comprobaciones de igualdades
446     check_doble(filecmp.cmp, os.path.join(__comp, __fileequals__),
447             "No coincide el fichero",
448             " - Comprobación de ficheros adicionales")
449
450     check_doble(filecmp.dircmp, os.path.join(__comp, __direquals__),
451             "No coincide el directorio",
452             " - Comprobación de directorios adicionales")
453
454     # Borrados
455     change_simple(removedir, os.path.join(__comp, __dirremove__),
456             "Borrando directorio",
457             " - Borrado de directorios")
458
459     change_simple(remove, os.path.join(__comp, __fileremove__),
460             "Borrando fichero",
461             " - Borrado de ficheros")
462
463     # Restaurados
464     change_doble(shutil.copy, os.path.join(__comp, __filerestore__),
465             "Restaurando fichero",
466             " - Restaurando ficheros")
467
468     change_doble(shutil.copytree, os.path.join(__comp, __dirrestore__),
469             "Restaurando directorio",
470             " - Restaurando directorios")
471
472     # Modificaciones de permisos
473     change_doble(chg_perm, os.path.join(__comp, __changeperm__),
474             "Modificando permisos",
475             " - Modificación de permisos")
476
477     change_doble(chg_own, os.path.join(__comp, __changeown__),
478             "Modificando propietario de ",
479             " - Modificación de propietarios")
480
481     change_doble(chg_grp, os.path.join(__comp, __changegrp__),
482             "Modificando grupo propietario de ",
```

```
483         " - Modificación de grupos")
484
485     change_simple(chg_mask, os.path.join(__comp, __changemask__),
486                 "Modificando máscara",
487                 " - Modificación de la máscara del sistema")
488
489     change_simple(os.popen, os.path.join(__comp, __commands__),
490                 "Ejecutando orden",
491                 " - Ejecución de órdenes")
492
493
494 # -----
495 #                               FUNCIÓN PRINCIPAL
496 # -----
497 if __name__ == "__main__":
498     try:
499         __opts__, __args__ = getopt.getopt(sys.argv[1:],
500                                         "vhcp:s",
501                                         ["__verbose__", "help", "clean",
502                                          "partial", "show"])
503     except getopt.GetoptError, err:
504         # Imprimir error y ayuda y salir
505         print str(err)
506         usage()
507         sys.exit(2)
508
509     # Identificar opciones
510     for o, a in __opts__:
511         if o == "-v":
512             __verbose__ = True
513         elif o in ("-h", "--help"):
514             usage()
515             sys.exit()
516         elif o in ("-c", "--clean"):
517             __fun__ = __clean__
518         elif o in ("-p", "--partial"):
519             __fun__ = __partial__ + a
520             print "PARCIAL: %s" % __fun__
521         elif o in ("-s", "--show"):
522             show()
523             sys.exit()
524         else:
525             assert False, "Opción no manejada"
526
527     # Comprobaciones
528     check(__base__)
529     check(__fun__)
530
```

---

```

531     print "Verificación correcta"
532     # Mensajes finales
533     if __fun__ == __clean__:
534         print "    -> Puedes iniciar el desarrollo de los ejercicios"
535     if __fun__ == __final__:
536         print "    -> Has completado los ejercicios: ¡BIEN HECHO!"

```

---

## Scripts de mantenimiento

BOT: ORDENES.PY

---

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4  """
5  Script que utiliza pywikipedia para generar páginas automáticamente a partir
6  de las páginas del manual de Ubuntu
7  """
8
9  import wikipedia
10 import re
11 import urllib
12
13 # Lista de comandos a generar
14 __comandos__ = ['ar', 'arch', 'basename', 'cdp', 'cat', 'chroot',
15                 'chvt', 'colrm', 'column', 'cp', 'cpio', 'dd',
16                 'deallocvt', 'df', 'diff', 'dir', 'du', 'egrep',
17                 'env', 'expr', 'fgrep', 'file', 'flex', 'free',
18                 'gawk', 'getopt', 'grep', 'groups', 'gunzip',
19                 'gzip', 'hostname', 'id', 'igawk', 'last', 'ldd',
20                 'ln', 'login', 'logname', 'look', 'make', 'manpath',
21                 'mc', 'mev', 'mkdir', 'mkfifo', 'mknod', 'more',
22                 'mv', 'namei', 'newgrp', 'nice', 'nm', 'objdump',
23                 'od', 'passwd', 'paste', 'pr', 'printenv', 'ps',
24                 'pwd', 'reset', 'rev', 'rm', 'rmdir', 'script',
25                 'setterm', 'skill', 'sleep', 'sln', 'strip', 'stty',
26                 'su', 'tar', 'telnet', 'test', 'time', 'tload',
27                 'top', 'touch', 'uname', 'users', 'w', 'wall',
28                 'whatis', 'watch', 'whereis', 'who', 'whoami',
29                 'write', 'xargs', 'yes', 'ytree', 'zcat', 'zcmp',
30                 'zdiff', 'zforce', 'zgrep', 'zmore', 'znew',
31                 'banner', 'history', 'exit', 'poweroff', 'mailx',
32                 'news', 'uname', 'df', 'pack', 'unpack', 'pcat',
33                 'cmp', 'comm', 'dircmp', 'tree', 'sum', 'uniq',

```

---

```
34         'cut', 'tr', 'nl', 'wc', 'split', 'paste', 'join',
35         'tail', 'head', 'od']
36
37 __fallo__ = 0
38 __creado__ = 0
39 __exito__ = 0
40 __errores__ = []
41
42 for com in __comandos__:
43     print '\n\n%s\nComando %s\n' % ('-'*50, com)
44     # Buscando la página web del manual de "com"
45     f = urllib.urlopen("http://manpages.ubuntu.com/\
46     manpages/lucid/es/man1/%s.1.html" % com)
47
48     s = f.read()
49     try:
50         s = unicode(s, "utf-8")
51     except:
52         pass
53     f.close()
54
55     entradas = s.splitlines()
56
57     re_desc = re.compile("<h4><b>NOMBRE</b></h4><pre>")
58     re_sint = re.compile("<h4><b>SINOPSIS</b></h4><pre>")
59
60     # Búsqueda de NOMBRE y SINOPSIS
61     i = 0
62     for e in entradas:
63         if re_desc.search(e):
64             print "Encontrado en %s" % str(i)
65             descrip = entradas[i+1]
66             print descrip
67         if re_sint.search(e):
68             print "Encontrado en %s" % str(i)
69             syntax = entradas[i+1]
70             print syntax
71     i += 1
72
73     # Limpiar descripción
74     descrip = descrip.split('-')[1]
75     # Introducción completa
76     intro = u"La orden '%s' %s." % (com, descrip)
77
78     # Limpiar sintaxis
79     syntax = re.sub(ur"(|</?[bu]>)", ur"", syntax)
80     # Sintaxis completa
81     sintaxis = ur"La sintaxis es: {{Comando|s}}" % syntax
```

---

---

```

82
83     page = wikipedia.Page(wikipedia.Site("wikiunix", "osluca"), com)
84     if page.exists():
85         print "La página ya existe"
86         __creado__ = __creado__ + 1
87     else:
88         texto = u"{{TítuloMinúsculas}}{{man|s}}\n%s\n\n%s\
89 \n<noinclude>[[Categoría:Órdenes]]</noinclude>" % (com, intro, sintaxis)
90         print texto
91
92         wikipedia.output(com)
93         page.put(texto, u"BOT - Creando página automática para %s" % com)
94         __exito__ += 1
95
96         # Comprobando creación de página
97         if not page.exists():
98             print "ERROR"
99             __fallo__ += 1
100             __errores__.append(com)
101
102
103 print "\n\nESTADÍSTICAS: "
104 print " * total = %s" % str(len(__comandos__))
105 print " * __errores__ = %s" % str(__fallo__)
106 print " * ya creados = %s" % str(__creado__)
107 print " * __exito__ = %s" % str(__exito__)
108 print "\n\nREGISTRO DE ERRORES: "
109 for e in __errores__:
110     print "%s " % e

```

---

## Actualización de la máquina virtual

UPDATE\_UNIX.PY

---

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4  # Necesita tener instalado pysvn:
5  #     sudo apt-get install python-svn
6
7  """
8  Script en Python que exporta las rutas de repositorios indicadas
9  """
10

```

---

```
11 import pysvn
12
13 # Incluir tuplas por cada repositorio//ruta de la siguiente forma:
14 # (ruta del repositorio, ruta donde exportar)
15 REPOS = [("https://forja.rediris.es/svn/cusl4-wikiunix/code/control/unix",
16          "/home/nessa/unix")]
17
18 def error(cad):
19     """
20     Imprime el error del programa
21     """
22     print "ERROR: %s" % cad
23
24 if __name__ == "__main__":
25     # Cliente de Subversion
26     __client__ = pysvn.Client()
27     # Exportar todas las rutas
28     for ruta_export, ruta in REPOS:
29         revision = __client__.export(ruta_export, ruta, True)
```

---



## Publicaciones

---

En este anexo se incluyen las publicaciones en las que ha participado la autora relacionadas con este proyecto.

- ✍ Tecnologías Wiki y Conocimiento Abierto en la Universidad.

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

Actas de la V Conferencia Internacional de Software Libre, páginas 16-19. Cáceres, Diciembre 2009. I.S.B.N.: 978-84-692-8739-2

- ✍ Tecnologías wiki en la docencia de Ingeniería Informática

*Manuel Palomo Duarte, Emilio J. Rodríguez Posada, Inmaculada Medina Buló y Noelia Sales Montes*

XVI Jornadas de Enseñanza Universitaria de la Informática (JENUI2010)

Enviado a la espera de aceptación.



# Bibliografía UNIX

---

- [1] ERASERHEAD1, Árbol cronológico de la familia UNIX,  
Publicada en Wikimedia Commons bajo licencia Creative Commons Attribution Share-Alike:  
[http://commons.wikimedia.org/wiki/File:Unix\\_history-simple.png](http://commons.wikimedia.org/wiki/File:Unix_history-simple.png).
- [2] Página Oficial de la distribución Linux Xubuntu, <http://www.xubuntu.com/>.
- [3] Ubuntu Manpages Repository, <http://manpages.ubuntu.com/>.
- [4] C. SCHRODER, Linux Cookbook,  
Ed. O'Reilly. 2005.  
I.S.B.N.: 0-596-00640-3.
- [5] M. G. SOBELL, Manual práctico de Linux,  
Ed. Anaya. 2005.  
I.S.B.N.: 978-84-415-2350-0.
- [6] MATTHIAS KALLE DALHEIMER, MATT WELSH, Guía de referencia y aprendizaje Linux,  
Ed. O'Reilly. 2006.  
I.S.B.N.: 84-415-2031-3.
- [7] WILLIAM JOY, MARK HORTON, An introduction to display editing with Vi,  
<http://docs.freebsd.org/44doc/usd/12.vi/paper.html>.
- [8] Editor de texto VI, <http://www.dc.fi.udc.es/~afyanez/info-vi/index.html>.
- [9] Vi Lovers Home Page, <http://thomer.com/vi/vi.html>.
- [10] Manuales de Administración en sistemas GNU/Linux en [alcancelibre.org](http://www.alcancelibre.org),  
<http://www.alcancelibre.org/staticpages/index.php/manuales-indice>.
- [11] Bourne-Again Shell FAQ, <ftp://ftp.cwru.edu/pub/bash/FAQ>.
- [12] Bourne-Again Shell Home Page, <http://www.gnu.org/software/bash/bash.html>.
- [13] J. M. J. NOAH GIFT, Python para administración de sistemas Unix y Linux,  
Ed. Anaya, O'Reilly. 2008.  
I.S.B.N.: 978-84-415-2540-5.



## Bibliografía LaTeX

---

- [14] The T<sub>E</sub>Xbook,  
[net.ytu.edu.cn/share/%D7%CA%C1%CF/texbook.pdf](http://net.ytu.edu.cn/share/%D7%CA%C1%CF/texbook.pdf).
- [15] L<sup>A</sup>T<sub>E</sub>X. A document preparation system., <http://www.latex-project.org/>.
- [16] TikZ & PGF Examples, <http://www.texample.net/tikz/examples/>.
- [17] The Comprehensive L<sup>A</sup>T<sub>E</sub>XSymbol List,  
<http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf>.
- [18] BibTeX and Bibliography Style,  
<http://hugo.csie.ntu.edu.tw/~jih/html/BibliographyStyle.html>.
- [19] The multibib package, <http://www.tex.ac.uk/tex-archive/macros/latex/contrib/multibib/multibib.pdf>.
- [20] The LaTeX Beamer Class, <http://latex-beamer.sourceforge.net/>.



## Bibliografía Wiki

---

- [21] Wikipedia, la enciclopedia libre, <http://es.wikipedia.org>.
- [22] Wikimedia Commons, <http://commons.wikimedia.org/>.
- [23] JSPWiki, <http://www.jspwiki.org/>.
- [24] MoinMoin, <http://moinmo.in/>.
- [25] PMWiki, <http://www.pmwiki.org/wiki/PmWikiEs/PmWikiEs>.
- [26] TikiWiki, <http://info.tikiwiki.org/tiki-index.php>.
- [27] WackoWiki, <http://wackowiki.org/>.
- [28] Dokuwiki, <http://www.dokuwiki.org/es:dokuwiki>.
- [29] XWiki, <http://www.xwiki.org/xwiki/bin/view/Main/WebHome>.
- [30] VQWiki, <http://www.vqwiki.org/>.
- [31] WikkaWiki, <http://wikkawiki.org/HomePage>.
- [32] Fundación Wikimedia, <http://wikimediafoundation.org/wiki/Portada>.
- [33] EMILIO JOSÉ RODRÍGUEZ POSADA, Blog de AntiVandalBOT,  
<http://avbot.blogspot.com/>.
- [34] Plugins disponibles para MediaWiki,  
[http://www.mediawiki.org/wiki/Category:All\\_extensions](http://www.mediawiki.org/wiki/Category:All_extensions).
- [35] Wikipedia, the free encyclopedia, <http://en.wikipedia.org>.
- [36] Wikibooks, <http://en.wikibooks.org/>.
- [37] EMILIO JOSÉ RODRÍGUEZ POSADA, Forja del proyecto StatMediaWiki,  
<https://forja.rediris.es/projects/statmediawiki/>.
- [38] EMILIO JOSÉ RODRÍGUEZ POSADA, Página de usuario en Wikipedia de AVBOT,  
<http://es.wikipedia.org/wiki/Usuario:AVBOT>.
- [39] MediaWiki, <http://www.mediawiki.org/wiki/MediaWiki/es>.

## Bibliografía

---

- [40] MediaWiki. Dudas Usualmente Frecuentes.,  
<http://www.mediawiki.org/wiki/Manual:FAQ/es>.
- [41] Forja de pywikipedia, <http://pywikipediabot.sourceforge.net/>.
- [42] Tutorial Pywikipedia,  
[http://es.wikibooks.org/wiki/Tutorial\\_de\\_bots/Usar\\_pywikipedia](http://es.wikibooks.org/wiki/Tutorial_de_bots/Usar_pywikipedia).
- [43] Manual LocalSettings para MediaWiki,  
<http://www.mediawiki.org/wiki/Manual:LocalSettings.php>.

Puedes ver todas las referencias utilizadas navegando en el wiki directamente:

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



## Bibliografía General

---

- [44] Forja de RedIRIS, <http://forja.rediris.es>.
- [45] ANTONIO GARCÍA ALBA, Tutorial de LibSDL para la Programación en Videojuegos.
- [46] PYTHON SOFTWARE FOUNDATION, The Python Standard Library,  
<http://docs.python.org/library/>.
- [47] PYTHON SOFTWARE FOUNDATION, Python Documentation,  
<http://docs.python.org/3.1/>.
- [48] SUN MICROSYSTEMS, INC, VirtualBox Changelog v2.2 - Soporte para OVE,  
<http://www.virtualbox.org/wiki/Changelog-2.2>.
- [49] ERIC S. RAYMOND, The cathedral and the bazaar,  
Ed. O'Reilly. 2001.  
I.S.B.N.: 978-0-596-00108-7.
- [50] DIA, Dia's homepage,  
<http://live.gnome.org/Dia>.
- [51] Org-Mode: Your Life in Plain Text, <http://orgmode.org/>.
- [52] Pylint Project Page, <http://www.logilab.org/project/name/pylint>.
- [53] Pylint Features, <http://www.logilab.org/card/pylintfeatures>.
- [54] Emacs Wiki, <http://www.emacswiki.org/>.
- [55] GNU Emacs, <http://www.gnu.org/software/emacs/>.
- [56] Gimp's homepage, <http://www.gimp.org/>.
- [57] API de Google Chart. Guía del desarrollador,  
<http://code.google.com/intl/es-ES/apis/chart/>.
- [58] G. ABURRUZAGA GARCÍA, Make. Un programa para controlar la recompilación,  
<http://www.uca.es/softwarelibre/publicaciones/make.pdf>.
- [59] Inkscape. Dibuja libremente.,  
<http://www.inkscape.org/?lang=es>.

- [60] J. HERRERA GOÁS, El blog de Joaclint Istgud,  
<http://joaclintistgud.wordpress.com/>.
- [61] Página oficial de *Subversion*, <http://subversion.tigris.org/>.
- [62] BEN COLLINS-SUSSMAN, BRIAN W. FITZPATRICK, C. MICHAEL PILATO,  
Version Control with Subversion,  
<http://svnbook.red-bean.com/nightly/en/index.html>, 2004,  
distribuido bajo licencia Creative Commons.
- [63] ROSA DURANTE, LEANDRO PASTRANA, PABLO RECIO, NOELIA SALES,  
Sistemas para el Control de Versiones,  
<http://aulavirtual.uca.es/moodle/file.php/7587/estudiosInformes/trabajo.pdf>.
- [64] SUN MICROSYSTEMS, INC, Sun VirtualBox. User Manual.,  
<http://www.virtualbox.org/manual/UserManual.html>.
- [65] SUN MICROSYSTEMS, INC, Documentación de Virtual Box,  
<http://www.virtualbox.org/wiki/Documentation>.
- [66] VideoTutorial: [XVidCap] Cómo hacer un screencast,  
<http://www.youtube.com/watch?v=kR9R-bQx-tg>.
- [67] HTML 4.01 Specification, <http://www.w3.org/TR/html4/>.
- [68] Manual de HTML, <http://www.webestilo.com/html/>.
- [69] Guía breve de CSS, <http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>.
- [70] Página oficial de PHP, <http://php.net/index.php>.
- [71] R. GONZÁLEZ DUQUE, *Python para todos*,  
distribuido bajo licencia Creative Commons Reconocimiento 2.5.
- [72] ANDRÉS MARZAL, ISABEL GARCÍA, Introducción a la programación con Python,  
<http://aulavirtual.uca.es/moodle/file.php/7587/python/python-jaume1.pdf>.
- [73] The Python Language Reference, <http://docs.python.org/reference/index.html>.
- [74] The Python Standard Library, <http://docs.python.org/library/index.html>.
- [75] ALEX MARTELLI, ANNA MARTELLI RAVENSCROFT, DAVID ASCHER, *Python Cookbook. Recipes from the Python Community*,  
Ed. O'Reilly. 2005.  
I.S.B.N.: 978-0-596-00797-3.

# GNU Free Documentation License

---

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.

`<http://fsf.org/>`

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “**Document**”, below, refers to any such manual or work. Any member of the public is a

licensee, and is addressed as “**you**”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “**Modified Version**” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “**Secondary Section**” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “**Invariant Sections**” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “**Cover Texts**” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “**Transparent**” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “**Opaque**”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “**Title Page**” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “**publisher**” means any person or entity that distributes copies of the Document to the public.

A section “**Entitled XYZ**” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “**Acknowledgements**”, “**Dedications**”, “**Endorsements**”, or “**History**”). To “**Preserve the Title**” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

## 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

- K. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled “Endorsements”. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## **5. COMBINING DOCUMENTS**

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements”.

## 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

## 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.



## 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## 11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

## **ADDENDUM: How to use this License for your documents**

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”.

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with ... Texts.” line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.