



EBook Gratis

APRENDIZAJE plone

Free unaffiliated eBook created from
Stack Overflow contributors.

#plone

Tabla de contenido

Acerca de	1
Capítulo 1: Empezando con Plone	2
Observaciones.....	2
Examples.....	2
Instalación o configuración.....	2
Capítulo 2: Crear complemento	3
Examples.....	3
Descripción general de la estructura de archivos.....	3
Mínimo esqueleto.....	3
Componentes opcionales.....	3
Un cuento sobre la creación de un complemento de Plone.....	3
Preámbulo	3
Estructura de un mínimo de Plone-Add-on	4
Módulo de pitón.....	4
Huevo de pitón.....	4
Intérprete de Python.....	5
Haciendo un complemento instalable dentro de un sitio Plone.....	5
Creditos	7

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [plone](#)

It is an unofficial and free plone ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official plone.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con Plone

Observaciones

Esta sección proporciona una descripción general de qué es plone y por qué un desarrollador puede querer usarlo.

También debe mencionar cualquier tema grande dentro de plone, y vincular a los temas relacionados. Dado que la Documentación para plone es nueva, es posible que deba crear versiones iniciales de los temas relacionados.

Examples

Instalación o configuración

Instrucciones detalladas sobre cómo configurar o instalar plone.

Lea Empezando con Plone en línea: <https://riptutorial.com/es/plone/topic/8657/empezando-con-plone>

Capítulo 2: Crear complemento

Examples

Descripción general de la estructura de archivos

Mínimo esqueleto

```
mynamespace.myaddon | The container-directory of the add-on.

  setup.py           | Register this directory to the Python-interpreter of
                    | the Zope-instance.

  mynamespace       | The namespace-directory of the add-on.

    __init__.py     | Makes this directory a Python-module and contains
                    | the lines for the namespace-magic.

  myaddon           | The base-directory of the add-on.

    __init__.py     | Empty file, make this directory a Python-module.

  myfile.py         | Example-file: If it contained a method called
                    | 'myMethod', it would be importable of any other
                    | registered egg like this:
                    |     from mynamespace.myaddon.myfile import myMethod
```

Componentes opcionales

El directorio 'mynamespace.myaddon / mynamespace / myaddon' puede contener:

```
configure.zcml | Register all following files and directories, here.

setuphandlers.py | Make an addon available to Zope-instance.

profiles        | Make an addon installable to Plone-site and generally
                | hold GenericSetup-xml-files.

skins           | The "old-school"-way, where resources like templates,
                | stylesheets, etc., live.

browser         | The "new-school"-way, where resources live.

content         | Holds the Python-files for defining custom content-types.

subscriber.py   | Subscribers listen to events, describe here what to do
                | when an event happens, e.g.: "An item has been added to the
                | site, notify admin about it via e-mail-notifica."
```

Un cuento sobre la creación de un complemento de Plone

Preámbulo

TL; TR: A modo de ilustración, las líneas de este documento que comienzan con 'TL; TR:' van seguidas de una línea de comando que hace referencia a los scripts de ayuda de Plone del huevo 'adi.devgen'. Si ejecuta el comando dado después de 'TL; TR:', se crearán todos los archivos que se explican en el siguiente capítulo. Hay muchas herramientas alternativas de script de ayuda, sin embargo, en caso de que desee utilizar esta herramienta, puede instalarla rápidamente como esta en la línea de comandos:

```
pip install adi.devgen
```

Estructura de un mínimo de Plone-Add-on

TL; TR: `devgen addBase mynamespace.myaddon`

Al igual que Plone consta de más de 200 huevos de Python, un Plone-Add-On es también un huevo de Python y un huevo es un módulo de Python con espacios de nombre. Cualquier huevo estará disponible para el intérprete Python de Zope. Para hacer un huevo instalable en un sitio de Plone, también se necesita un llamado 'perfil'. Lo siguiente explicará estos términos.

Módulo de pitón

Un módulo de Python es un directorio que contiene al menos un archivo que debe llamarse `__init__.py`. El nombre es un indicador para que el intérprete de Python reconozca este directorio como un módulo, lo que significa que su ruta se puede importar de otros scripts de Python, por ejemplo, suponiendo que tenemos un directorio llamado 'myaddon', que contiene un archivo llamado 'myfile' y ese archivo contiene una definición llamada 'myDefinition', puede importar la definición de otro archivo Python como este:

```
from myaddon.myfile import myDefinition
```

Huevo de pitón

Un Python-egg es un módulo Python con espacios de nombres, un espacio de nombres puede ser cualquier nombre, pero es mejor que no lo utilices, en caso de que quieras compartir tu complemento con el mundo.

El motivo para el espacio de nombres es lo más importante para excluir la posibilidad de que dos o más módulos puedan tener el mismo nombre (en este ejemplo denominado 'myaddon'), ya que podría generar conflictos cuando un script de Python intente importar la ruta, no siendo único nunca más.

Aquí hay una descripción general de la estructura del directorio y los archivos involucrados, antes

de continuar explicando sus propósitos en detalle más adelante:

```
mynamespace.myaddon
  setup.py
  mynamespace
    __init__.py
    myaddon
      __init__.py
      myfile.py
```

Para asignar un nombre a un complemento, primero cree un directorio llamado 'mynamespace.myaddon'.

Este directorio debe contener el archivo 'setup.py', hace que el huevo se pueda registrar en el intérprete de Python de la instancia de Zope, esto se explica con más detalle en el siguiente capítulo [TODO].

Luego, en ese directorio, cree otro directorio con el nombre de su espacio de nombres: 'mynamespace'. Ahora, hacemos de este directorio una ruta de módulo importable, agregando el archivo con nombre de palabra clave `__init__.py` en él. También debe contener las siguientes líneas, que hacen que suceda la magia de espacio de nombres, explicada en el último párrafo:

```
[TODO: namespace-magic-lines]
```

Casi terminamos, por último, en este nuevo directorio colocamos el módulo del capítulo anterior: el directorio 'myaddon', que contiene ' `__init__.py`' y ' `myfile.py`' con ' `myDefinition`'.

Ahora tenemos un huevo registrable para la instancia de Zope y podríamos hacer referencia a sus métodos de cualquier otro huevo registrado, usando esta notación de ruta:

```
from mynamespace.myaddon.myfile import myDefinition
```

Es posible que vea la magia, los dos primeros directorios se omiten, *no* lo es:

```
from mynamespace.mynamespace.myaddon.myfile import myDefinition
```

Intérprete de Python

[QUE HACER:

- Explicar `setup.py`
- Agregue un capítulo anterior sobre buildout (= instalando Plone) como referencia.]

Haciendo un complemento instalable dentro de un sitio Plone

Un 'perfil' hará que el complemento se muestre en el panel de control de complementos de un

sitio, por lo que un administrador puede (des) activarlo allí para el sitio.

Además, como un sitio Plone es siempre un elemento secundario de ZOPE-instance y una instancia ZOPE puede contener varios sitios, es posible que no desee que se instalen componentes de forma complementaria de nuestro complemento en otros sitios Plone, por lo tanto, los vinculamos a un perfil'.

TL; TR: `devgen addProfile mynamespace.myaddon`

Esto nos lleva a las partes realmente interesantes, la arquitectura de componentes de ZOPE, controlada por los archivos que terminan con '.zcml', que significa 'lenguaje de marcas de componentes de ZOPE'. Con él puedes registrar un perfil, vincular vistas a interfaces y mucho más. De hecho, merece un capítulo principal propio y un subcapítulo para cada uno de los directorios llamados "perfil", "máscaras", "navegador" y "contenido".

Lea **Crear complemento en línea**: <https://riptutorial.com/es/plone/topic/8751/crear-complemento>

Creditos

S. No	Capítulos	Contributors
1	Empezando con Plone	Community
2	Crear complemento	Ida Ebkes