

OPENPlanForm

Manual del Desarrollador

Índice de contenido

Instalación.....	3
Requisitos.....	3
Obtener el código.....	4
Mediante la interfaz web de github.com.....	4
Mediante la consola del sistema servidor.....	4
Instalación del Framework OWL.....	5
Instalación de OPENPlanForm.....	6
Configuración.....	7
El archivo application.ini.....	7
Possibles problemas durante la instalación.....	9
Desarrollo.....	10

Instalación

La instalación de *OPENPlanForm* se puede realizar de diferentes maneras, dependiendo de cómo se haya obtenido el código fuente. En este manual se abarcarán los diferentes tipos de instalaciones, como así también los pasos necesarios para poner en pleno funcionamiento la aplicación.

La aplicación implementa un framework de desarrollo llamado OWL (*Open Web Library*), por lo que para poner en funcionamiento la misma, será necesario disponer de OWL instalado en el servidor. Tanto *OPENPlanForm* como OWL son software libre.

Requisitos

OPENPlanForm ha sido desarrollado bajo entornos *LAMP* (*Linux, Apache, MySql y php*), pero no se descarta su ejecución bajo otras plataformas. Como requisitos mínimos para su funcionamiento se destaca (En negrita el software recomendado):

- Servidor web: **Apache** (<http://www.apache.org>), Concretamente el desarrollo se ha hecho bajo la versión 2.2.9, aunque cualquier servidor web que soporte **mod_rewrite** (http://httpd.apache.org/docs/2.2/mod/mod_rewrite.html) puede ser utilizado en su lugar.
- **Php 5.1 o Mayor:** (<http://www.php.net>), El desarrollo se ha realizado bajo la versión 5.2.6, pero es posible que funcione con cualquier versión dentro de la rama 5, sin problemas.
- **MySQL 5.1** (<http://www.mysql.com>), No se utiliza ninguna extensión propia de mysql, pero la capa modelo sólo funciona con este tipo de base de datos. Se cree que su funcionamiento será normal con cualquier versión 4.x o mayor.

Obtener el código

Mediante la interfaz web de github.com

El código puede obtenerse directamente desde internet. Concretamente desde el sitio (<http://www.github.com>). A continuación detallamos los repositorios que contienen la aplicación. Si el desarrollador no se encuentra familiarizado con el sistema de control de versiones Git, puede obtener los paquetes de código mediante la interfaz web de github.com:

The screenshot shows the GitHub web interface for the repository `openplanform / OPENPlanForm`. The main navigation bar includes links for `Dashboard`, `Inbox`, `Configuración`, and `Cerrar Sesión`. Below the navigation is a search bar and a sidebar with links for `Explorar GitHub`, `Gist`, `Blog`, and `Ayuda`.

The repository page displays basic information such as the number of commits, pull requests, and issues. It also shows the current branch as `master`. A prominent button labeled `Descargas` (Downloads) is visible.

Under the `Fuente` tab, there are links for `SSH`, `HTTP`, and `Git sólo lectura` (Read-only Git). The URL listed is `git@github.com:openplanform/OPENPlanForm.git`. A note indicates that this URL has permissions for `Lectura y escritura` (Read and Write).

The commit history shows a single commit by `Nicolas Daniel Palumbo` made "1 day ago". The commit message is: "- Eliminados los controladores y vistas que no estaban siendo utilizados." The commit hash is `f5b5c1bee6bf9eef4e5`, the author is `Abol`, and the parent hash is `b613e0ecf13f9c7eccf0`.

The commit details section shows a table with columns for `Nombre` (Name), `edad` (Age), and `mensaje` (Message). The table contains five rows corresponding to the files `app/`, `public/`, `.gitignore`, `.htaccess`, and `owl`.

At the bottom of the commit history, a note states: "No hemos podido encontrar un archivo README para este repositorio; se recomienda crearlo. Para más detalles sobre qué formatos podemos mostrar, visita [github/markup](#)".

At the very bottom of the page, there are links for `Bloq`, `Ayuda`, `Training`, `Tablero de trabajos`, `Tienda`, `Contacto`, `API`, and `Estado`. On the right side, there is a logo for "Powered by the Dedicated Servers and" followed by a small icon.

<https://github.com/openplanform/Owl> – (Open Web Library) Framework de desarrollo

<https://github.com/openplanform/OPENPlanForm> – (OPENPlanForm) Aplicación

<https://github.com/openplanform/doc.openplanform> – Documentación

Mediante la consola del sistema servidor

Si el desarrollador así lo desea, puede obtener el código de los diferentes paquetes utilizando git:

Para el framework:

```
#git clone git://github.com/openplanform/Owl.git /var/www/owl
```

Para la aplicación:

```
#git clone git://github.com/openplanform/OPENPlanForm.git /var/www/[DIR]
```

Donde [DIR] es el directorio donde será servida la aplicación.

Instalación del Framework OWL

Una vez obtenido el código de OWL, el mismo se debe colocar en un directorio cualquiera de nuestro servidor, la ubicación recomendada/por defecto es **/var/www/owl**, aunque este directorio puede cambiar dependiendo en donde se tenga definido el *server document root*.

Luego de colocar el framework en el directorio adecuado (en los ejemplos asumiremos que el mismo es **/var/www/owl**), debemos descomprimir Zend Framework que viene incluido con OWL:

```
# tar -xvf /var/www/owl/extlib/Zend.tar.gz
```

En este momento, el desarrollador ya tiene el framework OWL disponible. Este manual no explicará el funcionamiento interno del framework, ya que para dicho fin se encuentra disponible la completa documentación del API en el siguiente repositorio:

<https://github.com/openplanform/doc.openplanform>

A modo de resumen, podemos decir que OWL es un framework de desarrollo web bajo licencia GPL. Implementa el patrón de diseño MVC, ofrece una capa de modelo con completa abstracción a bases de datos, por el momento solo MySQL y una librería para acelerar el desarrollo de tareas comunes. A su vez incluye el conocido framework de Zend para algunas rutinas, como lo son el envío de mails, lectura de archivos de configuración, etc. En la documentación se puede encontrar el archivo **PhpDoc - API Desarrollo.tar.gz** que contiene el phpdoc de todas las clases de OWL.

Instalación de OPENPlanForm

Una vez que hemos descargado el código de la aplicación, deberemos crear un host virtual para el despliegue de la misma, en todos nuestros ejemplos, lo ubicaremos en **/var/www/extranet.planespime.es** y copiaremos todos los archivos a ese directorio. Siguiendo con nuestro ejemplo el archivo de definición del host virtual debería ser algo así:

```
<VirtualHost *:80>

    ServerAdmin webmaster@localhost
    ServerName extranet.in.planespime.es

    DocumentRoot /var/www/extranet.planespime.es
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/extranet.planespime.es>
        Options -Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
    </Directory>

</VirtualHost>
```

Lo siguiente es crear la base de datos. El archivo con el código sql necesario se encuentra en **/var/www/extranet.planespime.es/app/dbase/**. El sql inserta los datos necesarios para el primer funcionamiento como así también algunos de ejemplo. Una vez que se ha insertado el sql se debe editar el archivo de configuración de la aplicación, el mismo se encuentra en **/var/www/extranet.planespime.es/app/configs/application.ini**. En él definiremos algunos aspectos esenciales para el correcto funcionamiento, como lo son los datos de acceso a la base de datos, el servidor de mail, la codificación y un largo etc. En este manual nos dedicaremos a configurar solo las claves necesarias para que nuestra aplicación inicie.

Configuración

El archivo application.ini

Todo tipo de configuración relacionada con el funcionamiento interno de la aplicación se efectúa en este archivo, el formato utilizado es ini. Todo lo que se encuentre en production afectará a la aplicación en producción, mientras que develop extiende production y sobreescribe aquellas claves que se encuentren definidas en esa sección.

Cabe destacar que para que el sistema detecte que se encuentra en producción o desarrollo, la url de desarrollo debe ser del tipo: .in.algo.com, (debe contener el prefijo .in.). A continuación mostramos solo las claves que deben ser modificadas (en negrita) para configurar correctamente nuestra aplicación

```
[production]

; --- Configuración base
app.name = "extranet.planespime"
app.hostname = extranet.planespime.es ;dominio en producción
app.default_controller = index
app.default_action = index
app.default_lang = es
app.encoding = UTF-8 ;codificación deseada
app.basetitle = "PlanesPime"

; --- Layout a utilizar por defecto
app.default_layout = principal

; --- Canonizar nombres
app.canonize = no

; --- Description y keywords generales de la aplicación
app.description = "Una descripción de la aplicación" ;Meta description
app.keywords = "keyword, keyword, keyword" ;Meta keywords

; --- Configuración de base de datos
system.db.host = localhost
system.db.username = username ;Nombre de usuario BBDD
system.db.password = password ;Password del usuario BBDD
system.db.port = default ;Por defecto 3306
system.db dbname = planespime ;Nombre de la base de datos
```

```

system.owl_debug = no
;system.mantenimiento = no

; --- Solo podrá acceder a la aplicación la ip especificada
;system.allow = 127.0.0.1

; --- Directorios de captcha
system.captcha.creationdir = default
system.captcha.outputdir = default

; --- Configuración del mailer
system.mailer.mode = plain ;Modo de envío (plain | html)
system.mailer.server-type = smtp ;Tipo de servidor (smtp | mail)
system.mailer.target.mail = dir@correo.es
system.mailer.target.name = Nombre Apellidos
system.mailer.server = mail.planespime.es ;Servidor de mail
system.mailer.username = username; Nombre de usuario SMTP
system.mailer.password = password; Passowrd usuario SMTP
system.mailer.force-utf = no
system.mailer.subject = ""

[develop : production]

;Aquí se debe sobreescribir cualquier configuración que solo afecte al
;servidor de desarrollo

```

Una vez configurada la aplicación, si todo ha ido bien y siguiendo con nuestra url de ejemplo al acceder a <http://extranet.in.planespime.es> deberíamos ver el login:



El nombre del usuario administrador por defecto es **admin** y su contraseña es: **clave1**.

Posibles problemas durante la instalación

- Si se obtiene un “Internal Server Error”, seguramente no se dispone del módulo de apache **mod_rewrite**.
- Los errores más comunes se encuentran documentados, para el caso de cada excepción.
- IMPORTANTE: el directorio **owl/** debe ser un enlace simbólico que apunte al framework OWL instalado en nuestro servidor. Siguiendo nuestro ejemplo, el mismo debería apuntar a **/var/www/owl**

Desarrollo

Directorios a tener en cuenta a la hora de extender/modificar la aplicación:

app/configs: archivos de configuración, definición de constantes y globales

app/controllers: controladores, clases encargadas de la lógica de negocio

app/models: los modelos de cada tabla se encuentran aquí

app/modules: módulos dinámicos comunes, como lo son el menú, barra de herramientas, breadcrumbs, etc.

app/doc: contiene restos de documentación no relevante utilizada en el desarrollo

app/layouts: plantillas html y plantillas de email

app/views: capa de vista, aquí encontraremos una vista por cada acción de la aplicación

owl/: enlace simbólico al framework

public/css: todas las hojas de estilo se almacenan aquí

public/js: archivos javascript

public/docs: este directorio debe tener permisos de escritura para el usuario del servidor (generalmente www-data), aquí se almacenan los documentos subidos desde la aplicación.

public/tmp: directorio temporal, debe tener permisos de escritura para el usuario del servidor, aquí se genera la documentación. El contenido de este directorio debe ser vaciado diariamente.