ÁREA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

PROGRAMA ACADÉMICO DE MULTIMEDIA Y COMERCIO ELECTRÓNICO

UNIVERSIDAD TECNOLÓGICA DE LEÓN CAMPUS ACÁMBARO

|  |  |
| --- | --- |
| Asignatura | Profesor |
| Administración de Proyectos II | Sergio Alan Orarla Silva |
| Desarrollo de Aplicaciones Web | Rodolfo Martínez Puente |
| Optativa II | Rodolfo Martínez Puente |
| Integradora I | Christian Emanuel Lozano Hernández |

**ACTIVIDAD 5:** Puesta en marcha del sitio web

****

**PRESENTAN:**

|  |  |  |
| --- | --- | --- |
| Matrícula | Nombre | Correo electrónico |
| 12002026 | José Sánchez Rosales | josesanchezutl@gmail.com |
| 12002080 | José Francisco López Ruiz | francklr@gmail.com |
| 11002513 | Daniel Ávila Chávez | 89rommel@gmail.com |
| 13001705 | Brayan Dylan Cuevas Mora | ing\_brayan@outlook.com |

**GENERACIÓN:** 2015-2017

ACÁMBARO, GUANAJUATO. 02 DE JULIO DEL 2016

**TABLA DE CONTENIDO**

[INTRODUCCIÓN 3](#_Toc455259961)

[COMPETENCIA-OBJETIVO 4](#_Toc455259962)

[1. SERVIDORES GRATUITOS DE INTERNET 5](#_Toc455259963)

[2. PLATAFORMAS DE COMERCIO ELECTRÓNICO 12](#_Toc455259964)

[3. LIBRERÍAS DE CODEIGNITER (Technology, 2016) 23](#_Toc455259965)

[4. CONTINGENCIA Y MITIGACIÓN 32](#_Toc455259966)

[5. ESTUDIO DE SALARIOS 33](#_Toc455259967)

[6. WBS 38](#_Toc455259968)

[7. CONEXIÓN A REPOSITORIOS DEL PROYECTO 39](#_Toc455259969)

[8. FLUJO DE TRABAJO COLABORATIVO 40](#_Toc455259970)

[9. DOMINIO 41](#_Toc455259971)

[10. APLICACIÓN WEB 42](#_Toc455259972)

[MANUAL DE COMO CARGAR LOS ARCHIVOS EN EL SERVIDOR 42](#_Toc455259973)

[**MANUAL DE COMO SE GENERA LA BASE DE DATOS EN EL SERVIDOR** 51](#_Toc455259974)

[11. INNOVACIÓN 52](#_Toc455259975)

[ANALÍTICO-SINTÉTICO (CONCLUSIÓN) 53](#_Toc455259976)

[AUTO Y CO-EVALUACIÓN 54](#_Toc455259977)

[BIBLIOGRAFÍA 55](#_Toc455259978)

[ANEXOS 56](#_Toc455259979)

[DIAGRAMA DE ACTIVIDADES. 57](#_Toc455259980)

# INTRODUCCIÓN

A lo largo del tiempo en cualquier desarrollo de un proyecto sin importar las dimensiones de la empresa, existen riesgos a los cuales el proyecto siempre se encuentra a expensas de ser afectado y ser propenso a tener demoras o hasta inclusive el incumplimiento del mismo proyecto.

Es por tales motivos la importancia de generar un plan de respuesta a los riesgos, con el objetivo de identificar los riesgos en las diferentes etapas que comprende el proyecto para tener oportunidad de analizarlos a tiempo y generas planes de contingencias o mitigación de los riesgos para minimizarlos a tiempo y que no afecten el correcto desarrollo del proyecto.

La correcta identificación de los riesgos más relevantes se llevan en las etapas tempranas de ahí la priorización y categorización de los riesgos para crear planes de contingencia a tiempo tener una correcta respuesta para que durante el transcurso del desarrollo y al finalizar el proyecto no se encuentre esté en peligro.

Los planes de respuesta a los riegos no pueden ir ayudando en la experiencia personal para obtener lecciones aprendidas y acrecentar nuestros conocimientos de cómo es posible confrontar un riesgo y de qué manera es más conveniente afrontar un riesgo, de tal manera siempre estaremos en un proceso de mejora continua en el desarrollo de proyectos.

# COMPETENCIA-OBJETIVO

El alumno de las materias de Optativa II e Integradora I durante las unidades II y de las materias de Desarrollo de Aplicaciones Web y Administración de Proyectos II durante la unidad III, será competente para implementar un proyecto de aplicación web, de tal forma que en la elaboración de una evaluación de procesos de solución de problemas compuesta por los parámetros de instalación de la aplicación web y su correspondiente programación, conformada por al menos un módulo por alumno, obtenga como mínimo un 80% de desempeño, de acuerdo a los criterios de la rúbrica de evaluación 5.

.

# SERVIDORES GRATUITOS DE INTERNET

Dentro de este apartado encontraremos lo que son los servidores web, los diferentes servidores web que ofrecen hosting gratuito, asimismo indicaremos cual es el mejor para nosotros y con el dominio que tenemos.

**¿Qué son los servidores web?**

Es un programa informático que procesa una aplicaciones del lado del servidor, realizando conexiones bidireccionales (la información fluye en una sola dirección) o unidireccionales (se ajusta a las condiciones del receptor) y síncronas o asíncronas con el cliente, generando una respuesta en cualquier lenguaje.

El código recibido por el cliente suele ser compilado y ejecutado por el navegador web. Por lo general se usa el protocolo HTTP (Permite las transferencias de información en la world wie web) para estas comunicaciones, pertenecientes a la capa de aplicaciones del modelo OSI, también se empela el termino para referirse al ordenador que ejecuta el programa.

**ALGUNOS SERVIDORES WEB**

**Hostinger**

Fundado en Chipre en el año 2004 Hostinger es un proveedor popular de servicios de hosting con presencia en España desde el 2012. En su página web no aparece mucha información sobre la empresa pero mencionan que son un grupo global con presencia en varios países y cuentan con 60.000 clientes en España.

En Hostinger afirman que les encanta la idea de que Internet sea de acceso libre y por eso ofrecen planes de hosting gratuitos. El servicio de hosting también tiene una cuenta gratis en la que podrás alojar tu página web, eso sí con algunas limitaciones.

**000webhost**

Link: <https://www.000webhost.com/>

Es una empresa líder en la prestación de servicios de primera clase en alojamiento web gratuito. No hay coste culto, no cuenta con anuncios y no términos restrictivos. Velocidad rápida y máxima fiabilidad y apoyo a los usuarios son algunas de las características de los que recibirá con el servicios

**Zymic**

Link: <http://www.zymic.com/>

Zymic es una empresa consolidada dentro del sector del hosting gratis, ya que ofrece su servicio desde hace 7 años, además de ofrecer alojamiento web gratuito también ofrece una variedad de plantillas (themes o templates) para poder crear tu web más fácilmente sin tener que diseñarla desde cero. También dispone de una sección de tutoriales enfocados al diseño y programación web, CSS y HTML y Adobe Photoshop para la parte del diseño, así como PHP y MySQL en la parte de programación web.



**Byethost**

Link: <https://byet.host/>

Desde sus inicios han implementado una serie de redes de servidores de alojamiento gratuito avanzados para proporcionar la mejor relación costo conveniente, clave de soluciones de alojamiento y mucho más. Ha sido incesante búsqueda del éxito que ha sido capaces de desarrollar, toda una extensa lista de las infraestructuras de red de alojamiento únicas y soluciones que brinda la plataforma de servicios.

La red basada en clúster cuenta con cientos de nodos de servidores utilizando el software adecuado para el trabajo correcto, impulsado por los sistemas operativos Linux y Unix. El hardware en toda la red se compone principalmente de Quad CPU procesadores Intel Xeon, que se mantienen firmes ante las peticiones más exigentes de la red, tales como procesamiento de peticiones del usuario final

**Podserver**

Link: <http://www.podserver.info/>

A diferencia de otros planes de alojamiento gratuito, este incluye acceso completo a FTP, PHP, MySQL, Apache y el panel de control (panel de Vista). También tendrá la posibilidad de ampliar su sitio mediante la adición de subdominios adicionales y sí, incluso apoyar la adición de su propio nombre de dominio de nivel superior.

**TABLA COMPARATIVA**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *Servidores Web* | *Límite de espacio* | *Ancho de banda* | *Publicidad* | *Soporte FTP* | *Restricciones de subida de archivos* | *Copia de seguridad* | *Panel de Control* |
| 000webhost | 1500 MB | 100 GB | No | Si | 5 MB | Si | Personalizado |
| Hostinger | 2000 MB | 100 GB | No | Si | 8 MB | Si | Personalizado |
| Podserver | 300 MB | 10 GB | Si | Si | 5 MB | Si | Personalizado |
| Zymic | 2000 MB | 50 GB | No | Si | 15 MB | Si | Personalizado |
| Byethost | 5000 MB | 200 GB | No | Si | 4 MB | Si | Personalizado |

**ELECCIÓN DEL SERVIDOR WEB**

El servidor web por el que optamos al principio fue 000webhots, pero tuvimos algunos inconvenientes con la base de datos, así que decidimos utilizar Hostinger, ya que nos brinda 2 GB en el disco, igualmente nos ofrece un espacio de datos en el disco duro del servidor de 100 GB al mes, así mismo contamos con dominios ilimitados.

Otra parte que nos pareció perfecto es que no cuenta con publicidad por parte de Hostinger, a menos de que dejemos de utilizar el servicio por un largo tiempo, además nos permite rentabilizar nuestro sitio con anuncios propios. Cuenta con copia de seguridad y protocolo FTP el cual nos brida tranquilidad al momento de usar sus servicios.

Algo que también nos llamó la atención y queremos destacar, es que cuentan con atención al cliente, el cual brinda soporte al usuario proporcionando una respuesta que va de 1 a 4 días en español, lo cual es una ventaja extra ya que muchos de los servidores no cuentan con este servicio.

Este servidor web es relativamente nuevo para los integrantes del equipo, a comparación de 000webhost el cual facilitaría mayor control y manejo del servicio, por otra parte el registro de Hostinger es bueno, el cual brinda al instante nuestro sitio. Del mismo modo la respuesta al servidor es buena a comparación de otros servidores web que ofrecen el mismo servicio.

A demás basándonos en experiencias de otros usuarios que utilizaron Hostinger mencionan que corre bien en páginas HTML y se puede usar en proyectos personales pequeños sin necesidad de comprar el hosting.

Es por estos puntos por el cual elegimos este servidor web llamado Hostinger y nuestro dominio <http://refaccionarianvw.esy/index.php.Frontend>

# PLATAFORMAS DE COMERCIO ELECTRÓNICO

Dentro de este apartado analizaremos con información recopilada sobre 5 plataformas de comercio electrónico, igualmente indicaremos sus ventajas y desventajas, además haremos mención sobre una de las plataformas la cual creemos y consideramos que es la mejor.

Pero antes de continuar ¿Qué es una plataforma de comercio electrónico? Es un software que ofrece la creación de una tiendas online para la venta de productos a través de internet. En algunos casos, no es necesario tener altos conocimientos en programación o diseño. (Castillo Sequera & Navarro Huerga, 2015)

La mayoría pone a disposición de cualquier usuario el código fuente (código abierto) y cuenta con una versión gratuita, así que tampoco se necesita grandes cantidades de dinero, lo que beneficia a los usuarios que decena crear una tienda online, es suficiente con pagar el hosting y en algunos casos pagar módulos adicionales para funciones especiales.



**PrestaShop**

Link: <https://www.prestashop.com/es/>

Es un sistema gestor de contenidos (CMS) libre y de código abierto. Su principal aplicación es la creación de tiendas online de comercio electrónico. Prestashop es compatible con DirecPay, Google CheckOut, Authorize.net, Moneybookers, PayPal. También se ofrecen módulos de pago adicionales en el mercado. Actualmente es la aplicación para la creación de tiendas online más utilizada, por delante de otras aplicaciones populares como Magento o OpenCart.

**Ventajas:**

* Seguridad integrada con SSL.
* Personalización el back
* Es perfecta para proyectos pequeños
* El panel del administrador es muy fácil de manejar y comprender.
* Fácil de instalar.
* Funciona muy bien en servidores compartidos.
* Tiene muchas plantillas con aspectos profesionales.
* Las platillas son aptas para cualquier dispositivo.
* Multilingüe.
* Permite integrar ERPS.

**Desventajas:**

* Cuenta con muchos módulos pero los mojes y los que tienen mejor funcionalidades son de pago.
* Algunas traducciones están mal hechas, por lo que deberás repasarlas y cambiarlas.
* No funciona bien cundo el catálogo es muy extenso.
* No es escalable (indica su habilidad para reaccionar y adaptarse sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos.)
* Su ficha de productos es muy sencilla.
* Algunos TPV (Terminal de Punto de Venta) fallan.



**Shopify**

Link: <https://es.shopify.com/?ref=lancetalent>

Es una plataforma que permite crear tiendas de comercio electrónico sin necesidad de tener conocimientos técnicos informáticos. [Shopify](http://www.davidrabaez.com/ir/shopify" \t "_blank) es una plataforma [SaaS](http://es.wikipedia.org/wiki/Software_como_servicio" \o "Software como servicio) (Software as a Service), es decir, una herramienta que se puede utilizar sin necesidad de comprar una licencia, únicamente se paga un alquiler o renta mensual por el uso del servicio.

Si contamos con un amplio catálogo de productos, estamos pensando en internacionalizar nuestra tienda online y no deseamos o no podemos hacer una fuerte inversión en tecnología, [Shopify](http://www.davidrabaez.com/ir/shopify" \t "_blank) puede ser la plataforma ideal.

**Ventajas**

* Internacionalización de tus productos. Por un lado, podrás vender en varios idiomas. Por otro, el carrito de compra de Shopify permite a tus clientes poder pagar con cualquier moneda. Tu producto o servicio no tendrá fronteras.
* Es muy bueno para desarrollar pequeños comercios electrónicos.
* Es un software especializado en comercio electrónico.
* No se necesita comprar un servidor, porque es un software en la nube.
* Fácil de utilizar
* Las plantillas son muy profesionales y estéticas.
* Para pequeños negocios que desean internacionalizarse es ideal.

**Desventajas**

* Una de las grandes desventajas es personalizar el código
* Los idiomas no funcionan muy bien.
* Si quieres ampliar las funcionalidades, tendremos que integrar los módulos de pago (algunos son gratis, pero son muy pocos).
* Los servidores están en estados unidos.
* No es escalables (indica su habilidad para reaccionar y adaptarse sin perder calidad).
* Catálogos pequeños.
* No multitienda.

**Magento.**

Link: <https://magento.com/>

Es un gestor de contenido web de código abierto para comercio electrónico. Es una solución flexible y escalable con la que se pueden desarrollar prácticamente todo tipo de proyectos de comercio electrónico.

Ofrece muchas funcionalidades entre las que destaca especialmente la gran flexibilidad en el diseño y el potente panel de control. Además está respaldado por una de las empresas más importantes de internet, Ebay que en el 2012 adquirió la empresa Magento.

**Ventajas:**

* Es ideal para proyectos grades.
* Ideal para multitiendas.
* Multilingüe y multimoneda.
* Puede poner infinidad de forma de pago la cual funciona muy bien.
* Es un software nativo de comercio electrónico, por lo que tiene infinidad de módulos buenos.
* Contiene muchos módulos, de hecho es raro que necesites añadir nada.
* Contiene un montón de plantillas gratuitas, las cuales contienes un aspecto muy profesional.
* Casi todas las platillas son responsivas.
* Permite integrar ERPS
* Puedes crear fichas de productos muy completas y personalizadas, sin tener que tocar el código.
* Tiene una excelente visibilidad en los dispositivos móviles.
* Es muy seguro. Para ser un servidor que se ejecuta en un sistema separado, la información es completamente privada, sin acceso a terceros.

**Desventajas:**

* el panel del administrador es muy complicado.
* No es aconsejable instalarlo en un servidor compartido, necesitas adquirir un dedicado o VPS el cual tiene un coste de más o menos 3,031.21 moneda nacional.
* Algunas traducciones vienen mal hechas, por lo que se debe modificar.
* A pesar de ser un código abierto posee un alto nivel de programación estructurada.
* Es muy difícil aprender Magento, por lo que la gestión y modificación, tendrá que estar a cargo de profesionales.
* Pesa muchísimo y si te equivocas, la tienda tardara mucho en cargarse y perderás clientes.
* Con Magento no es para principiantes, ya que conlleva desarrollar una tienda en esta plataforma lleva tiempo.

**Drupal.**

Link: <https://drupalcommerce.org/>

Es un marco de gestión de contenidos (CMS), libre, modular, multipropósito y muy configurable que permite publicar artículos, imágenes, archivos y ofrece la posibilidad de otros servicios añadidos como foros, encuestas, votaciones, blog y administración de usuarios y permisos.

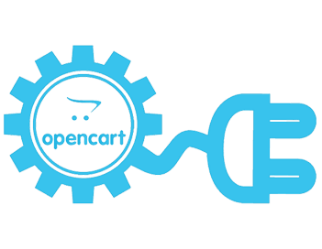
Es un sistema dinámico en ligar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija.

**Ventajas:**

* Es ideal para proyectos medianos.
* Servidores compartidos.
* Puede modificarse el código.
* Pueden construir páginas de comercio electrónico muy profesionales.
* Panel de administración es muy sencillo.

**Desventajas:**

* No es nativo de comercio electrónico, por lo que no tiene muchas funcionalidades de serie y tendrás que instalarlas.
* Algunas veces ocupamos instalar un módulo para que funciones otro.
* La velocidad de carga puede convertirse en un problema debido a su peso.
* Desarrollar una tienda en esta plataforma lleva mucho tiempo.

**Opencart**

Link: <http://www.opencart.com/>

Es un sistema de comercio electrónico de código abierto (open source) basado en PHP para tiendas de venta online. Una solución de comercio electrónico robusta para comerciantes de Internet con la capacidad de crear su propio negocio en línea y obtener un comercio electrónico con un coste mínimo.

Está diseñado con unas características muy completas, fácil de usar, potente, rápida y con un interfaz visualmente atractivo. Su principal creador y actual desarrollador de todo el proyecto es el inglés Daniel Kerr, al cual se le debe agradecer esta magnífica tienda online que ha puesto gentilmente al alcance de todos y hoy en día según muchas valoraciones y estadísticas que hay en Internet es quizás la mejor (o una de las mejores) de entre todas las existentes de código abierto, comparando rapidez, sencillez y eficacia.

**Ventajas:**

* Código abierto.
* Es muy sencillo.
* Ideal para el comercio electrónico medianos.
* Cuenta con muchos módulos para personalizar la tienda.
* Opiniones acerca de los productos y rating, hacen la experiencia de compra más completa.
* Admite muchas formas de pago.
* Fácil de aprender y gestionar.
* Pesa muy poco y por ende puedes escoger el servidor más barato.

**Desventajas:**

* No cuenta con muchas plantillas profesionales.
* Es muy difícil personalizarlo.
* Es malo con los distintos impuestos de los países.
* La traducción es pésima de acuerdo a los usuarios.
* Cualquier modificación en el código o plantilla, lo deberá hacerlo en una profesional.
* Es vulnerabilidad a ataques externos.

**ELECCIÓN DEL SISTEMA DE COMERCIO ELECTRÓNICO.**

La plataforma que consideramos y creemos es la mejor en comercio electrónico es Shopfy. Con la información que pudimos recopilar de sus ventajas y desventajas pudimos hacer varias comparativas con otros sistemas de comercio electrónico, las cuales también eran bastante buenas. Escogimos este sistema porque está orientado para pequeños comercios electrónicos y los cuales desean expandirse.

Unos de los rasgos más importantes de estas plataformas que creemos es la seguridad, ya que cuenta con un certificado nivel 1 de conformidad con las normas PCI el cual es un estándar de seguridad de datos para la industria de tarjetas de pago (tarjetas de crédito y débito, entre otras) y un cifrado de 128 bit, esto quiere decir que tiene 34 seguido de 36 ceros de posibles combinaciones, lo cual es bastante bueno.

Además cuenta con soporte técnico en caso de que surgiera algún problema con el sistema, además cuenta con una tienda de aplicaciones para el sistema las cuales ampliaran y darán nuevos funcionamientos a nuestro sistema.

Otra característica con la que cuenta es una aplicación móvil tanto para iphone como para android, la cual nos permite manipular el sistema, desde cualquier lugar donde tengamos internet, esto serviría de mucho para el manejo de las ventas, dudas con nuestros clientes, etc.

Aunque tengamos que pagar nos permite utilizarlo por 15 días de prueba, en caso de no cumplir con las expectativas simplemente no se compra.

Es por estas razones por lo que el equipo eligió este sistema de comercio electrónico

# LIBRERÍAS DE CODEIGNITER (Technology, 2016)

* **CALENDARING CLASS**

Como la mayoría de las otras clases en CodeIgniter, la clase Calendar es inicializada en su controlador usando la función $ this-> load-> library:

**$ This-> load-> library ( 'calendario');**

Una vez cargada, el objeto Calendario estará disponible usando:

**$this->calendar**

Aquí es un ejemplo muy simple que muestra cómo puede mostrar un calendario:

**$this**->**load**->**library('calendar');**

echo **$this**->**calendar**->**generate();**

El código anterior generará un calendario que muestra el mes de Junio de 2006. El primer parámetro especifica el año, el segundo parámetro especifica el mes.

Al pasar datos a las células del calendario

Para añadir datos a las células del calendario implica la creación de una matriz asociativa en la que las teclas se corresponden con los días en que desea rellenar y el valor de la matriz contiene los datos. La matriz se pasa a la tercera parámetro de la función de generación de calendario. Considere este ejemplo:

**$ This-> load-> library ( 'calendario');**

**$ Datos = array (**

**3 => 'http://example.com/news/article/2006/06/03/',**

**7 => 'http://example.com/news/article/2006/06/07/',**

**13 => 'http://example.com/news/article/2006/06/13/',**

**26 => 'http://example.com/news/article/2006/06/26/'**

**);**

**echo $ this-> Calendario General> generar (2006, 6, $ data);**

* [**Form Validation**](http://www.codeigniter.com/user_guide/libraries/form_validation.html#id6)

Lo que sigue es un "manos en" Tutorial para la implementación de CodeIgniters formulario de validación.

Con el fin de poner en práctica la validación de formularios que necesita tres cosas:

1. Un archivo de vista que contiene un formulario.

2. Presentar una vista que contiene un mensaje de "éxito" que se mostrará tras la presentación exitosa.

3. Un método de controlador para recibir y procesar los datos presentados.

Vamos a crear esas tres cosas, el uso de un miembro de formulario de registro como el ejemplo.

**La forma**

Con un editor de texto, cree un formulario denominado miformulario.php. En él, ubique este código y guardarlo en su application / views / carpeta:

<html>

<head>

<title>My Form</title>

</head>

<body>

<?php echo validation\_errors(); ?>

<?php echo form\_open('form'); ?>

<h5>Username</h5>

<input type="text" name="username" value="" size="50" />

<h5>Password</h5>

<input type="text" name="password" value="" size="50" />

<h5>Password Confirm</h5>

<input type="text" name="passconf" value="" size="50" />

<h5>Email Address</h5>

<input type="text" name="email" value="" size="50" />

<div><input type="submit" value="Submit" /></div>

</form>

</body>

</html>

**La página del éxito**

Con un editor de texto, cree un formulario denominado formsuccess.php. En él, ubique este código y guardarlo en su application / views / carpeta:

<html>

<head>

<title>My Form</title>

</head>

<body>

<h3>Your form was successfully submitted!</h3>

<p><?php echo anchor('form', 'Try it again!'); ?></p>

</body>

</html>

**El controlador**

Con un editor de texto, cree un controlador llamado form.php. En él, ubique este código y guardarlo en su application / controllers / carpeta:

<?php

class Form extends CI\_Controller {

public function index()

{

$this->load->helper(array('form', 'url'));

$this->load->library('form\_validation');

if ($this->form\_validation->run() == FALSE)

{

$this->load->view('myform');

}

else

{

$this->load->view('formsuccess');

}

}

}

* **Pagination Class**

Si no está familiarizado con el término "paginación", se refiere a los enlaces que le permite navegar de una página a otra, de esta manera:

**« First  < 1 2 3 4 5 >  Last »**

He aquí un ejemplo simple que muestra cómo crear la paginación en uno de sus métodos de controlador:

$this->load->library('pagination');

$config['base\_url'] = 'http://example.com/index.php/test/page/';

$config['total\_rows'] = 200;

$config['per\_page'] = 20;

$this->pagination->initialize($config);

echo $this->pagination->create\_links();

**Notas**

La variable $ config contiene sus variables de configuración. Se pasa al método $ this-> pagination-> initialize () como se muestra arriba. Aunque hay una veintena de elementos que se pueden configurar, por lo mínimo que necesita los tres se muestra. He aquí una descripción de lo que representan esos elementos:

base\_url Esta es la URL completa de la clase del controlador / función que contiene su paginación. En el ejemplo anterior, se apunta a un controlador llamado "Test" y una función llamada "página". Tenga en cuenta que usted pueda volver a su URI si necesita una estructura diferente.

TOTAL\_ROWS Este número representa el total de filas en el conjunto de resultados que está creando paginación para. Normalmente, este número será el total de filas que regresó a su base de datos de consulta.

per\_page El número de elementos que la intención de mostrar por página. En el ejemplo anterior, debería estar mostrando 20 elementos por página.

El método create\_links () devuelve una cadena vacía cuando no hay paginación para mostrar.

**Establecimiento de preferencias en un archivo de configuración**

Si usted prefiere no establecer preferencias utilizando el método anterior, en su lugar se les puede poner en un archivo de configuración. Basta con crear un nuevo archivo llamado pagination.php, agregar la variable $ config en ese archivo. A continuación, guarde el archivo en application / config / pagination.php y se utiliza de forma automática. Usted no tendrá que usar $ this-> pagination-> initialize () si guarda sus preferencias en un archivo de configuración.

**Personalización de la paginación**

La siguiente es una lista de todas las preferencias se pueden pasar a la función de inicialización para adaptar la pantalla.

**$config[‘uri\_segment’] = 3;**

La función de paginación determina automáticamente qué segmento de la URI contiene el número de página. Si necesita algo diferente puede especificarlo.

**$config[‘num\_links’] = 2;**

El número de enlaces "dígito" que le gustaría antes y después del número de página seleccionado. Por ejemplo, el número 2 tendrá lugar dos dígitos en ambos lados, como en el ejemplo enlaces en la parte superior de esta página.

**$config[‘use\_page\_numbers’] = TRUE;**

Por defecto, el segmento URI utilizará el índice de partida para las que esté paginación. Si prefiere mostrar el número de la página real, ajuste este parámetro en TRUE.

**$config[‘page\_query\_string’] = TRUE;**

De forma predeterminada, la biblioteca de paginación suponga que está utilizando segmentos URI, y construye sus enlaces algo como:

**http://example.com/index.php/test/page/20**

[**http://www.codeigniter.com/user\_guide/libraries/calendar.html**](http://www.codeigniter.com/user_guide/libraries/calendar.html)

[**http://www.codeigniter.com/user\_guide/libraries/form\_validation.html**](http://www.codeigniter.com/user_guide/libraries/form_validation.html)

[**http://www.codeigniter.com/user\_guide/libraries/pagination.html**](http://www.codeigniter.com/user_guide/libraries/pagination.html)

# CONTINGENCIA Y MITIGACIÓN

Enlaces a los documentos que nos proporcionan la lista de medidas de mitigación y contingencia y el plan de contingencia de riesgos.

**01-RefaccionariaNVW\_BD de Riesgos.xlsx**

**01-RefaccionariaNVW\_Plan de Contingencia de Riesgos.doc**

# ESTUDIO DE SALARIOS

|  |  |  |  |
| --- | --- | --- | --- |
| ESTUDIO DE SALARIOS (México) (Galvan, 2015) | | | |
| ZONA GEOGRÁFICA | | | |
|  |  | **Mediana** | **Salario por hora** |
| Zona Geográfica | Jalisco | $33,500 | $209.37 |
| Distrito Federal | $30,000 | 187.5 |
| Nuevo León | $30,000 | 187.5 |
| Aguascalientes | $28,000 | 175 |
| Sonora | $27,000 | 168.75 |
| Estado de México | $25,000 | 156.25 |
| Quintana Roo | $25,000 | 156.25 |
| Tabasco | $23,500 | 146.875 |
| Morelos | $22,000 | 137.5 |
| Querétaro | $22,000 | 137.5 |
| Colima | $21,500 | 134.375 |
| Guanajuato | $19,100 | 119.375 |
| Chihuahua | $19,000 | 118.75 |
| Baja California | $18,650 | 116.5625 |
| Puebla | $17,800 | 111.25 |
| Michoacán | $16,250 | 101.56 |
| Yucatán | $12,000 | 75 |
| Veracruz | $12,000 | 75 |
| Sinaloa | $10,500 | 65.62 |
| LENGUAJE DE PROGRAMACIÓN | | | |
|  |  | **Mediana** | **Salario por hora** |
| Lenguaje de programación | Scala | $39,000 | 243.75 |
| Elixir | $38,000 | 237.5 |
| Perl | $35,000 | 218.75 |
| Groovy | $32,500 | 203.12 |
| Swift | $31,500 | 196.87 |
| Objetive C | $30,000 | 187.5 |
| Ruby | $29,000 | 181.25 |
| Bash | $28,000 | 175 |
| Go | $28,000 | 175 |
| Cobol | $28,000 | 175 |
| Java | $27,000 | 168.75 |
| Phyton | $26,000 | 162.5 |
| C | $25,750 | 160.93 |
| Javascript | $25,000 | 156.25 |
| Delphi | $24,000 | 150 |
| ABAP | $23,000 | 143.75 |
| PLSQL | $23,000 | 143.75 |
| C# | $23,000 | 143.75 |
| Visual Basic | $20,500 | 128.12 |
| PHP | $19,000 | 118.75 |
| ROL | | | |
|  |  | **Mediana** | **Salario por hora** |
| Rol | Dirección | $45,000 | 281.25 |
| Preventa | $43,000 | 268.75 |
| Venta | $36,500 | 228.12 |
| Mejora de procesos | $35,000 | 218.75 |
| Consultoría de negocios | $35,000 | 218.75 |
| Project Managment | $35,000 | 218.75 |
| Arquitectura de Sistemas | $30,000 | 187.5 |
| Análisis de requerimientos | $27,000 | 168.75 |
| Capacitación | $26,000 | 162.5 |
| Implantación de ERP | $25,000 | 156.25 |
| Adm. Infraestructura | $25,000 | 156.25 |
| Programación back-end | $25,000 | 156.25 |
| Seguridad informática | $25,000 | 156.25 |
| Programación front-end | $25,000 | 156.25 |
| Business Intelligence | $24,000 | 150 |
| Aseguramiento de calidad | $24,000 | 150 |
| User experience design | $24,000 | 150 |
| Administración de datos | $19,000 | 118.75 |
| Docencia | $18,000 | 112.5 |
| Soporte técnico | $16,000 | 100 |
| HABILIDADES/CONOCIMIENTOS | | | |
|  |  | **Mediana** | **Salario por hora** |
| Habilidades/Conocimientos | Maestría | $34,000 | 212.5 |
| Doctorado | $33,000 | 206.25 |
| Posgrado | $30,000 | 187.5 |
| Técnica / Vocacional | $28,000 | 175 |
| Preparatoria | $27,000 | 168.75 |
| Universidad (titulado) | $25,000 | 156.25 |
| Universidad (sin titular) | $21,250 | 132.81 |
| NIVEL DE INGLES | | | |
|  |  | **Mediana** | **Salario por hora** |
| Nivel de ingles | Ninguno (ILR 0) | $17,500 | 109.37 |
| Elemental (ILR 1) | $17,000 | 106.25 |
| Limitado (ILR 2) | $22,000 | 137.5 |
| Profesional (ILR 3) | $30,000 | 187.5 |
| Avanzado (ILR 4) | $36,000 | 225 |
| Nativo (ILR 5) | $39,100 | 244.37 |
| GÉNERO | | | |
| Género | Hombre | $27,000 | 168.75 |
| Mujer | $21,000 | 131.25 |

**Estudio de Salarios en Base al Tope máximo del Estado de Guanajuato.**

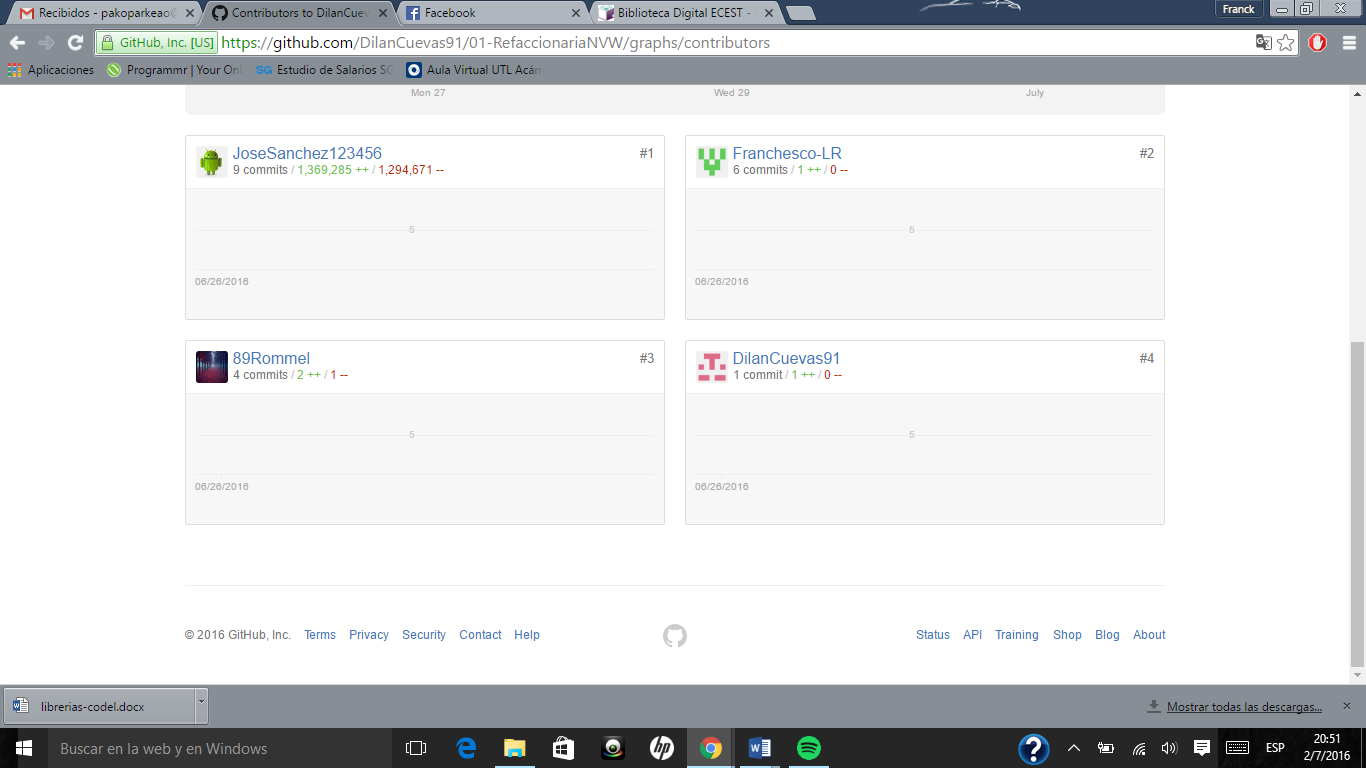
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Salario Tope Edo. Gto.** | **$ 23,352.00** |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Roles** | **Salarios Generales** | **Dias x Mes** | **Horas x Día** | **Horas x Mes** | **Salario x Mes** | **Salario x Hora** |
|  |  |  |  |  |  |  |
| **Administracion de Proyecto** | **$ 37,081.00** | 20 | 8 | 160 | **$ 23,352.00** | **$ 145.95** |
| **Analista** | **$ 28,329.00** | 20 | 8 | 160 | **$ 17,840.37** | **$ 111.50** |
| **Diseñador** | **$ 29,017.00** | 20 | 8 | 160 | **$ 18,273.64** | **$ 114.21** |
| **Programador** | **$ 24,055.67** | 20 | 8 | 160 | **$ 15,149.21** | **$ 94.68** |
| **Tester** | **$ 27,804.56** | 20 | 8 | 160 | **$ 17,510.10** | **$ 109.44** |
| **Admin de la Configuración** | **$ 21,668.00** | 20 | 8 | 160 | **$ 13,645.56** | **$ 85.28** |
| **Asegurador de la Calida** | **$ 26,453.00** | 20 | 8 | 160 | **$ 16,658.95** | **$ 104.12** |

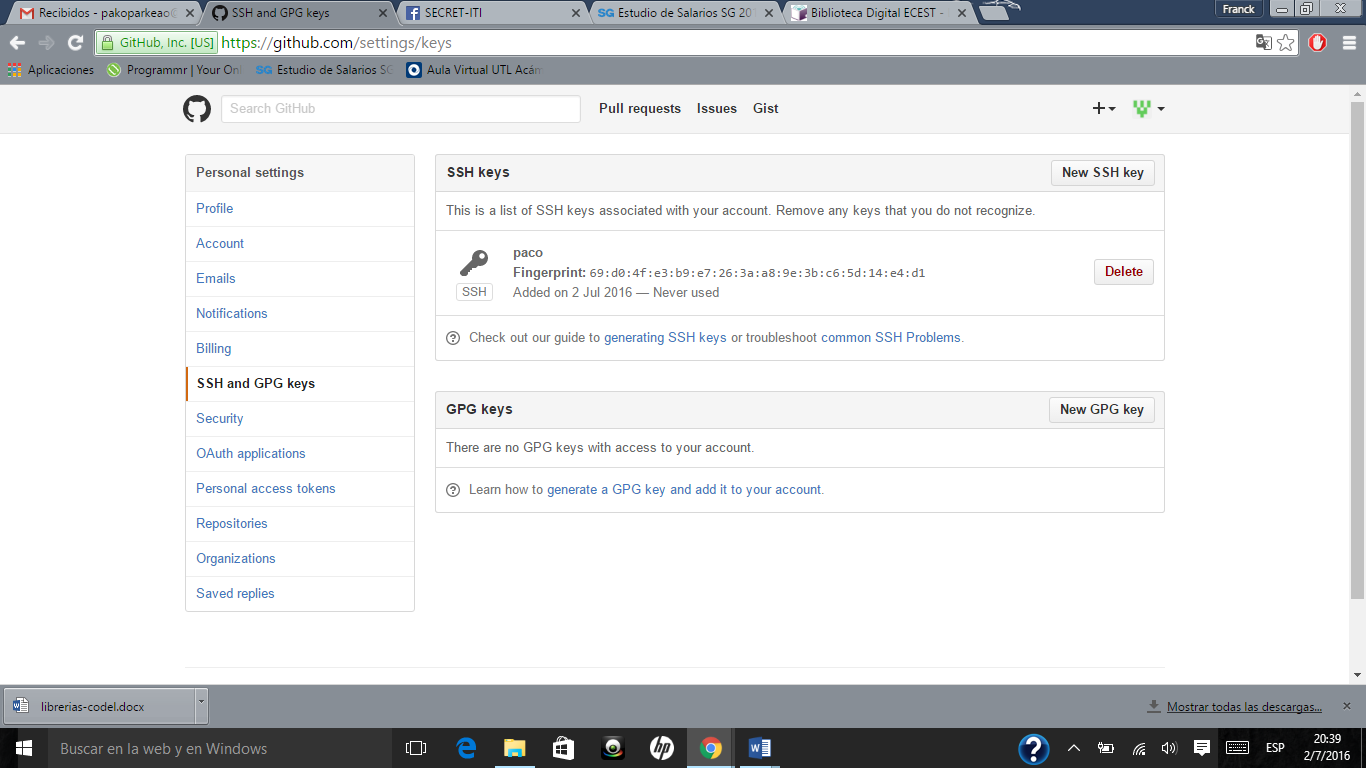
# WBS

Enlace al wbs. Donde se muestran las fechas de inicio fin y duración de proyecto y ruta crítica de las actividades del proyecto.

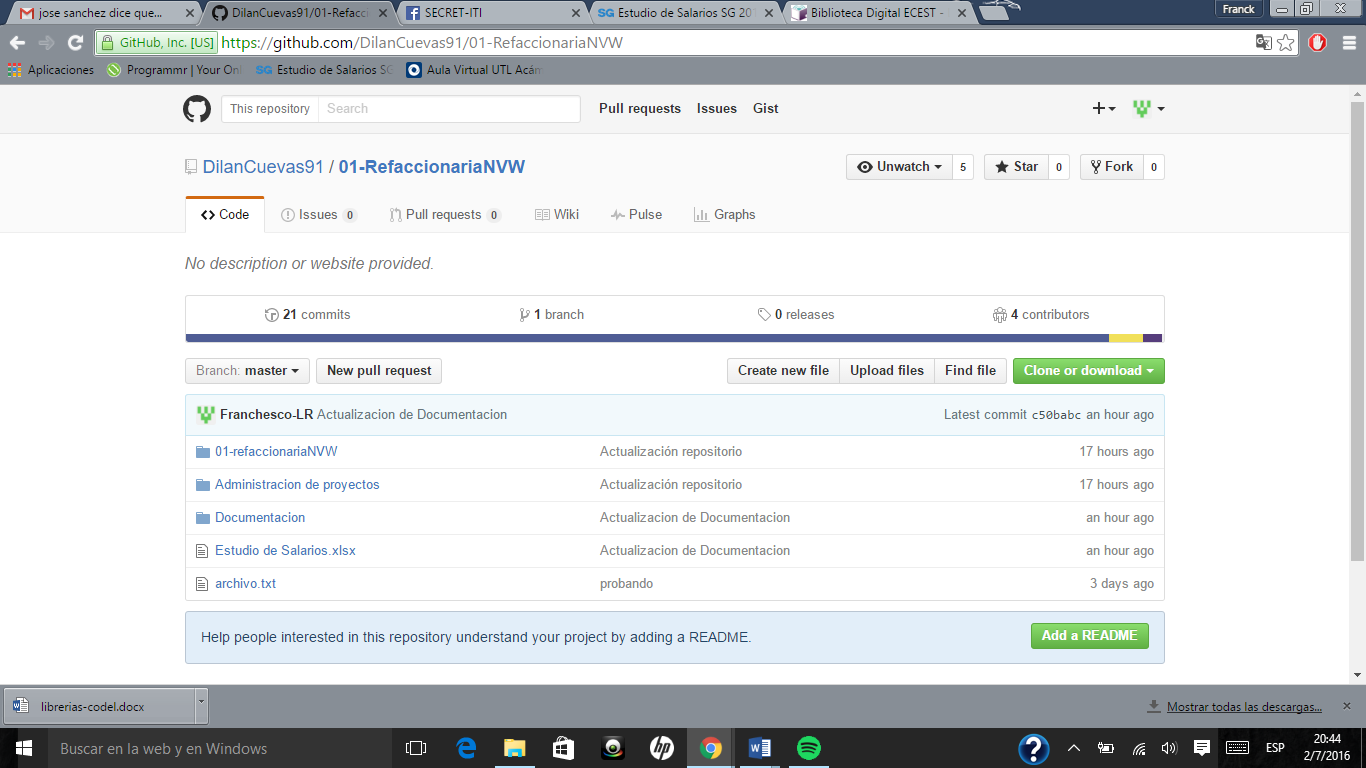
**01-RefaccionariaNVW\_WBS.mpp**

# CONEXIÓN A REPOSITORIOS DEL PROYECTO

En los repositorios del equipo se encuentran agregados todos los colaboradores incluyendo al profesor de la materia de integradora, todos los commits al repositorio (branch master) deben ser ejecutados (push) desde GIT por medio de SSH (Crear las llaves pública/privada).



# FLUJO DE TRABAJO COLABORATIVO

Deben encontrase los repositorios (master) de la materia de DAW y AP, los cuales serán revisados por medio de los logs y la historia del proyecto para comprobar todos los miembros del equipo llevaron un manejo de versiones y trabajaron de forma colaborativa haciendo uso de ramas (branches) y fusiones (merges).

# DOMINIO

**TABLA COMPARATIVA**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *Servidores Web* | *Límite de espacio* | *Ancho de banda* | *Publicidad* | *Soporte FTP* | *Restricciones de subida de archivos* | *Copia de seguridad* | *Panel de Control* |
| 000webhost | 1500 MB | 100 GB | No | Si | 5 MB | Si | Personalizado |
| Hostinger | 2000 MB | 100 GB | No | Si | 8 MB | Si | Personalizado |
| Podserver | 300 MB | 10 GB | Si | Si | 5 MB | Si | Personalizado |
| Zymic | 2000 MB | 50 GB | No | Si | 15 MB | Si | Personalizado |
| Byethost | 5000 MB | 200 GB | No | Si | 4 MB | Si | Personalizado |

**Nuestro dominio:** <http://refaccionarianvw.esy/index.php.Frontend>

# APLICACIÓN WEB

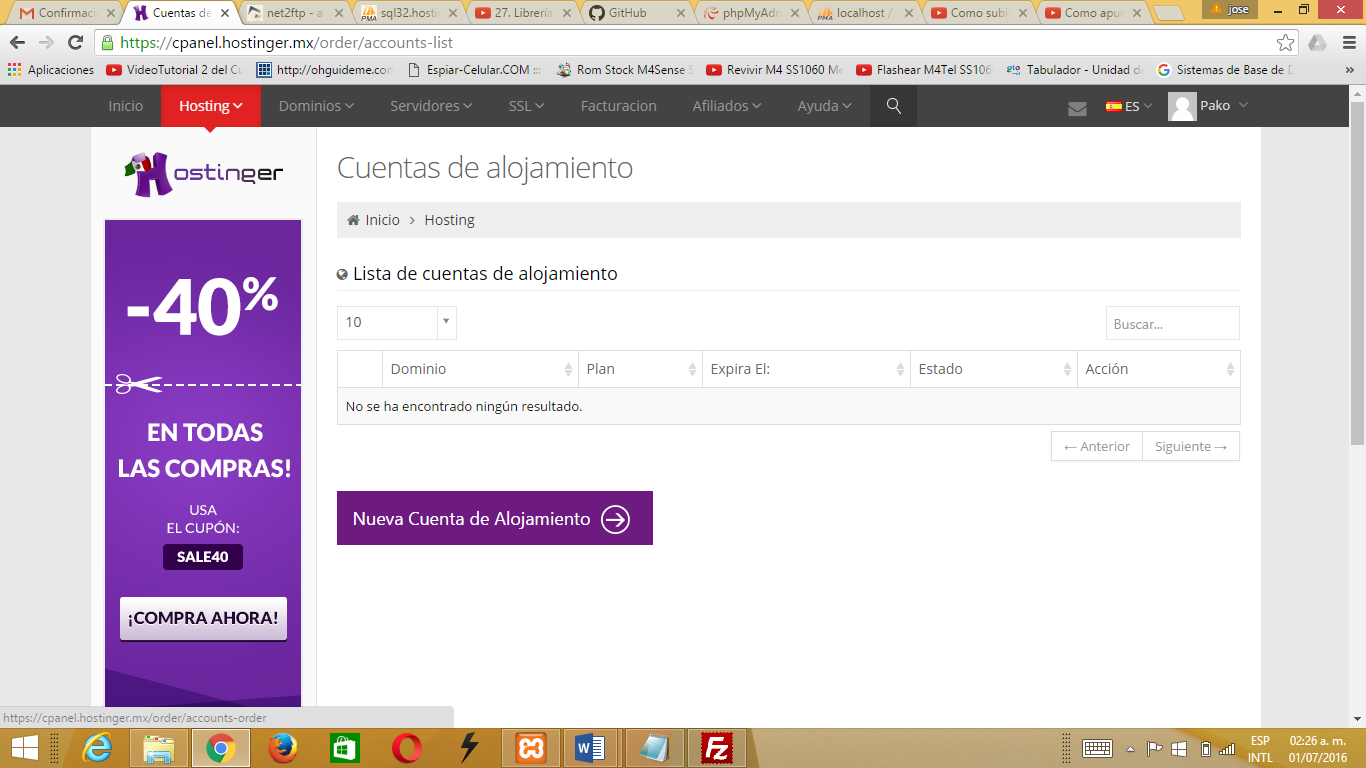
## MANUAL DE COMO CARGAR LOS ARCHIVOS EN EL SERVIDOR

Este apartado servirá como prueba y guía para subir un sitio web a Hostinger, el cual después de haber desarrollado gran parte de nuestro sitio web, el cual contiene PHP, HTML y CSS. El siguiente paso sería que este en internet y es por eso que implementamos un manual el cual los guiara paso a paso para su instalación dentro del servidor web Hostinger.

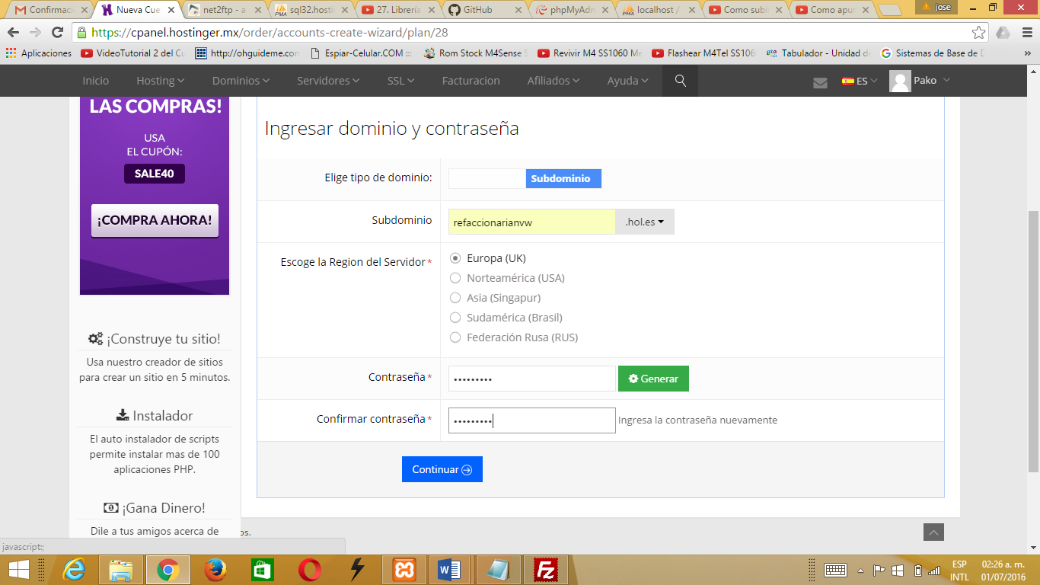
1. Debemos hacer es acceder a la página de Hostinger en México <http://www.hostinger.mx/> en el cual no registraremos con nuestro nombre de usuario, correo y contraseña, o si lo deseamos podeos acceder con una cuenta ya sea de google o Facebook.



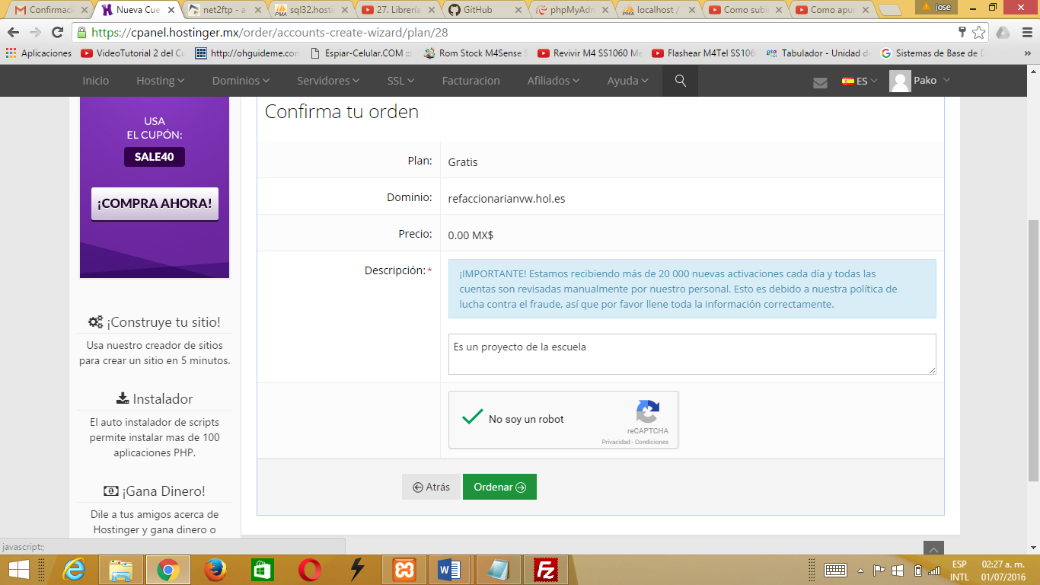
1. A continuación se mostrara una pantalla el cual nos permitirá crear una cuenta de alojamiento dentro del servidor Hostinger.



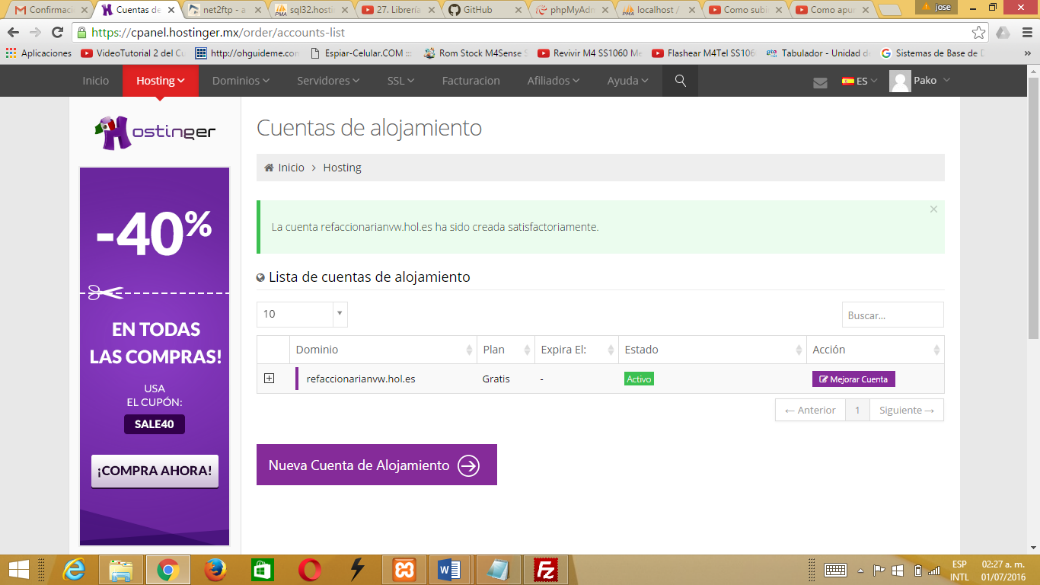
1. El siguiente paso sería elegir el tipo de dominio, el nombre del subdominio, escoger la región del servidor, la contraseña.



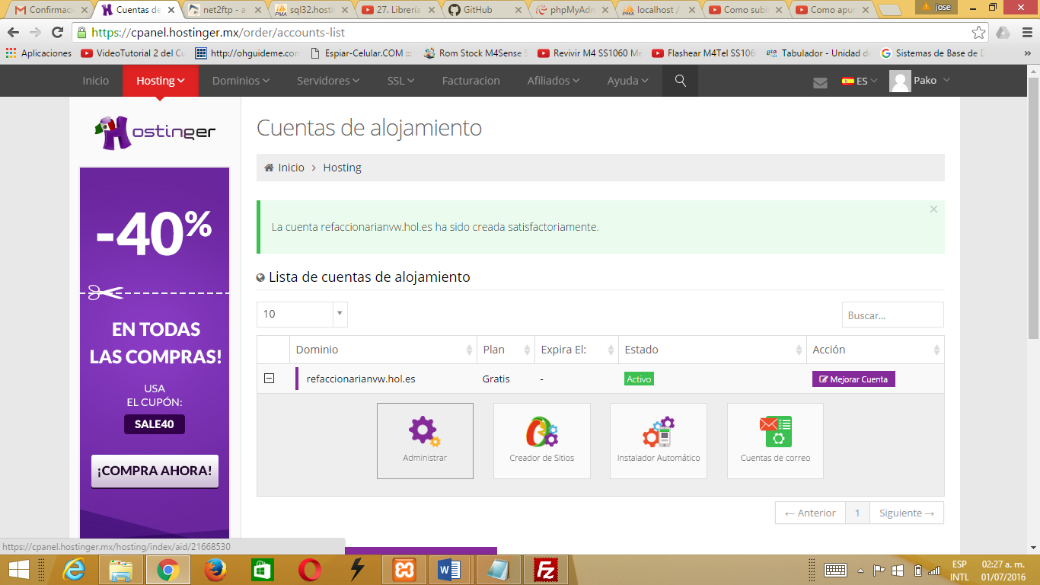
1. Lo siguiente es confirmar el plan, en este caso es gratis, el dominio, además nos mostrara una descripción el cual menciona le revisión del sitio por parte de su personal, mencionamos el propósito y pasar el captcha.



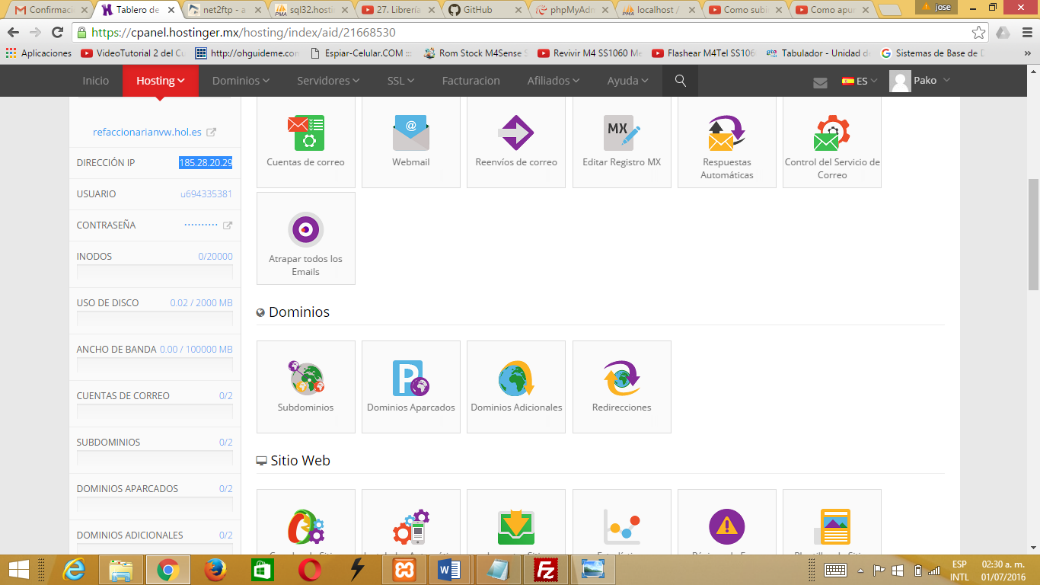
1. Como podemos observar nos muestra una pequeña descripción en donde viene el nombre del dominio, el plan, el estado, el cual creamos hace unos momentos.



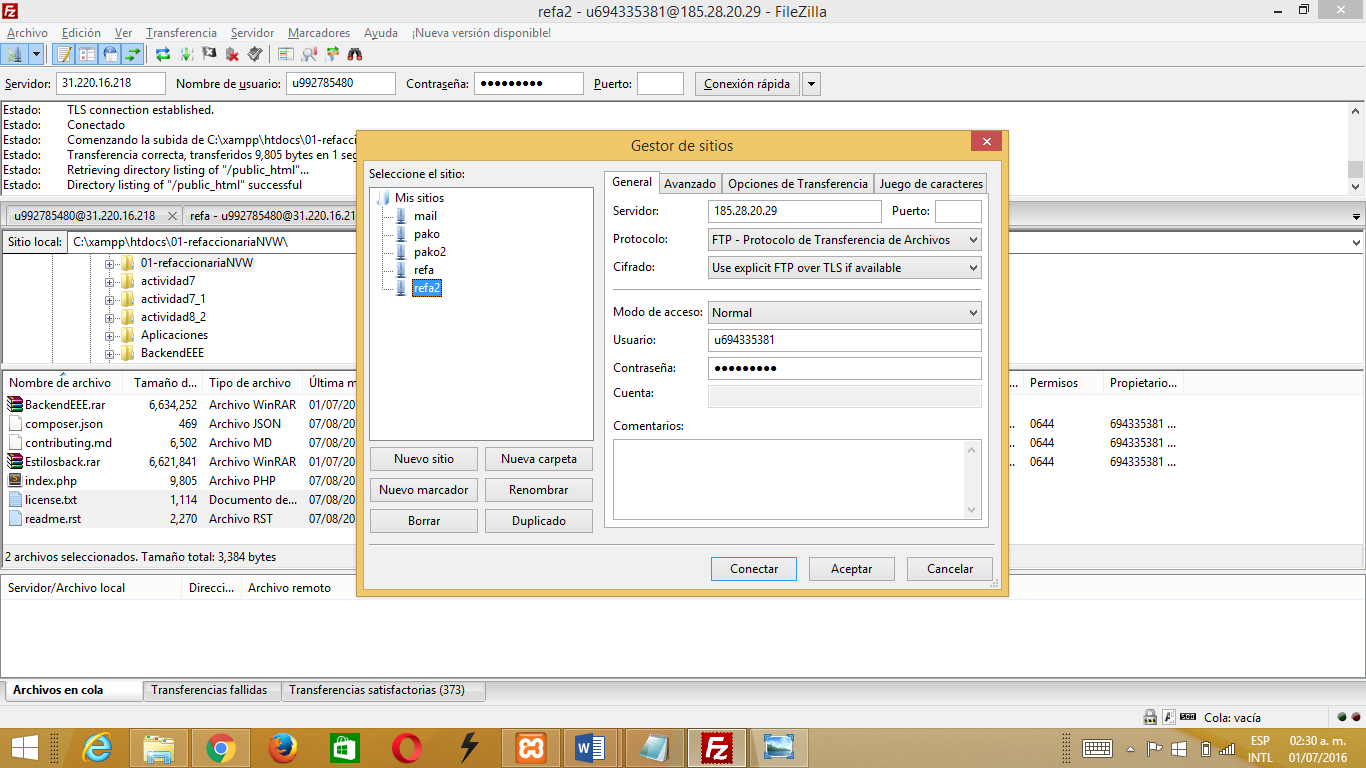
1. Ya contamos con nuestro dominio el cual es el siguiente <http://refaccionarianvw.hol.es/> , el siguiente paso es subir nuestro sitio web



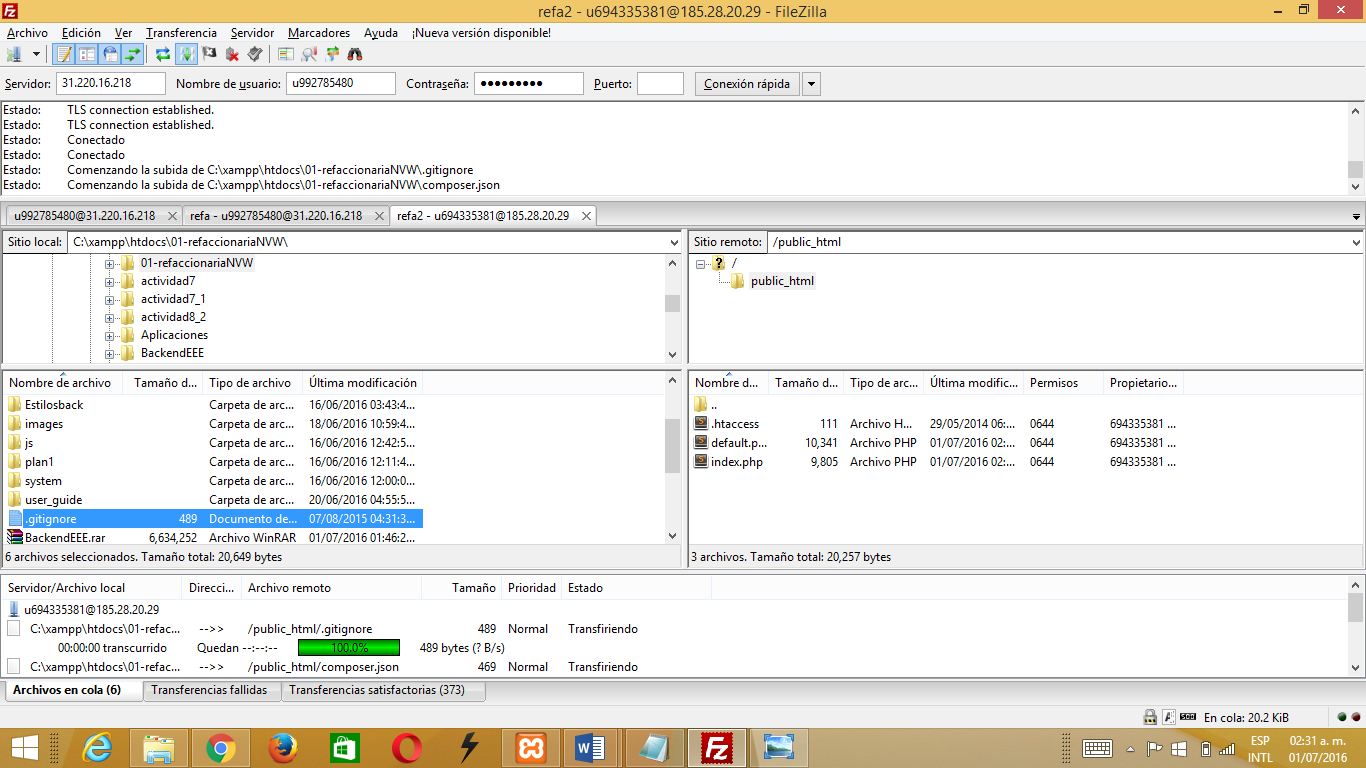
1. A hora entramos en el administrador, para verificar todas las opciones que tenemos a nuestro alcance.



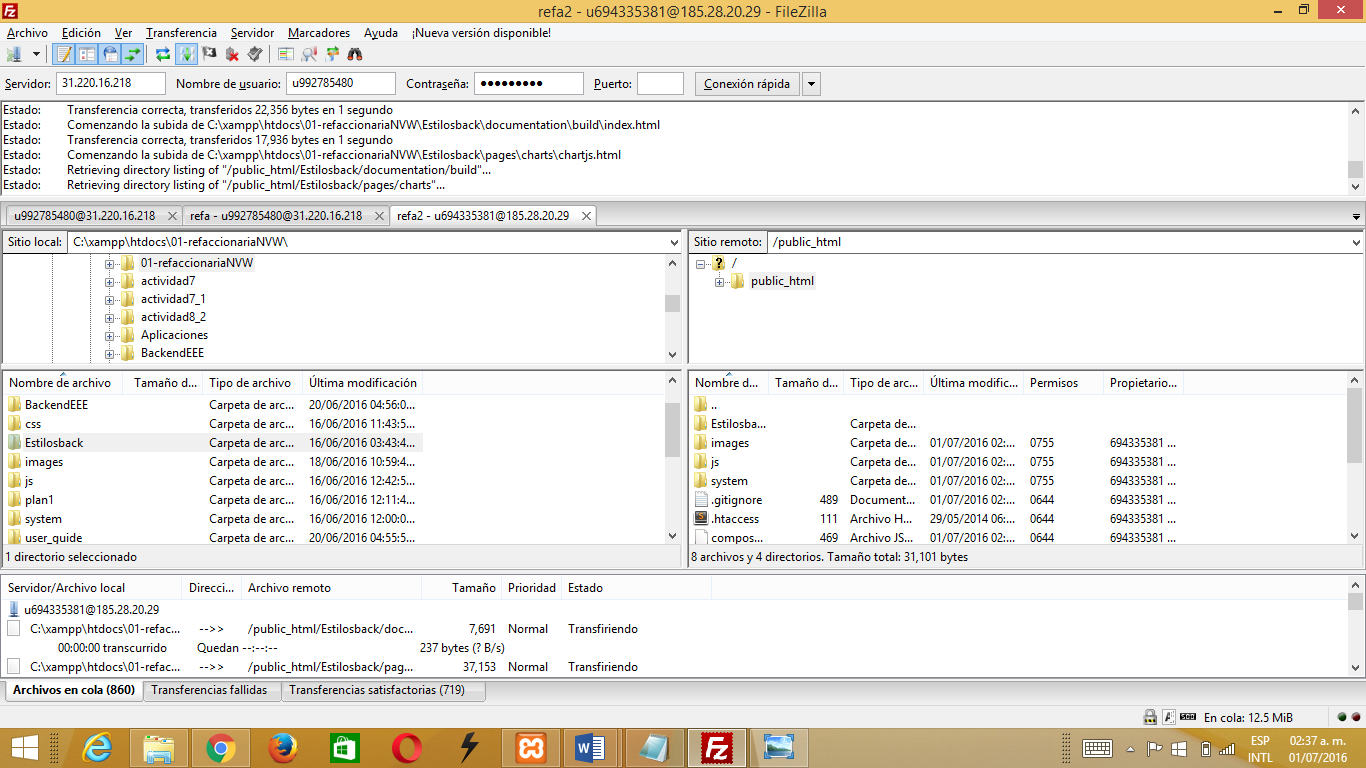
1. Utilizamos FTP, filezzilla el cual permite administrar nuestros archivos y lo utilizamos para subió todos los archivos del sitio al servidor web, podremos observar un cuadro de texto en el cual colocaremos los datos, como la dirección FTP del dominio, nombre del usuario el cual fue asignado desde el panel de configuraciones de Ferozo para la cuenta FTP.

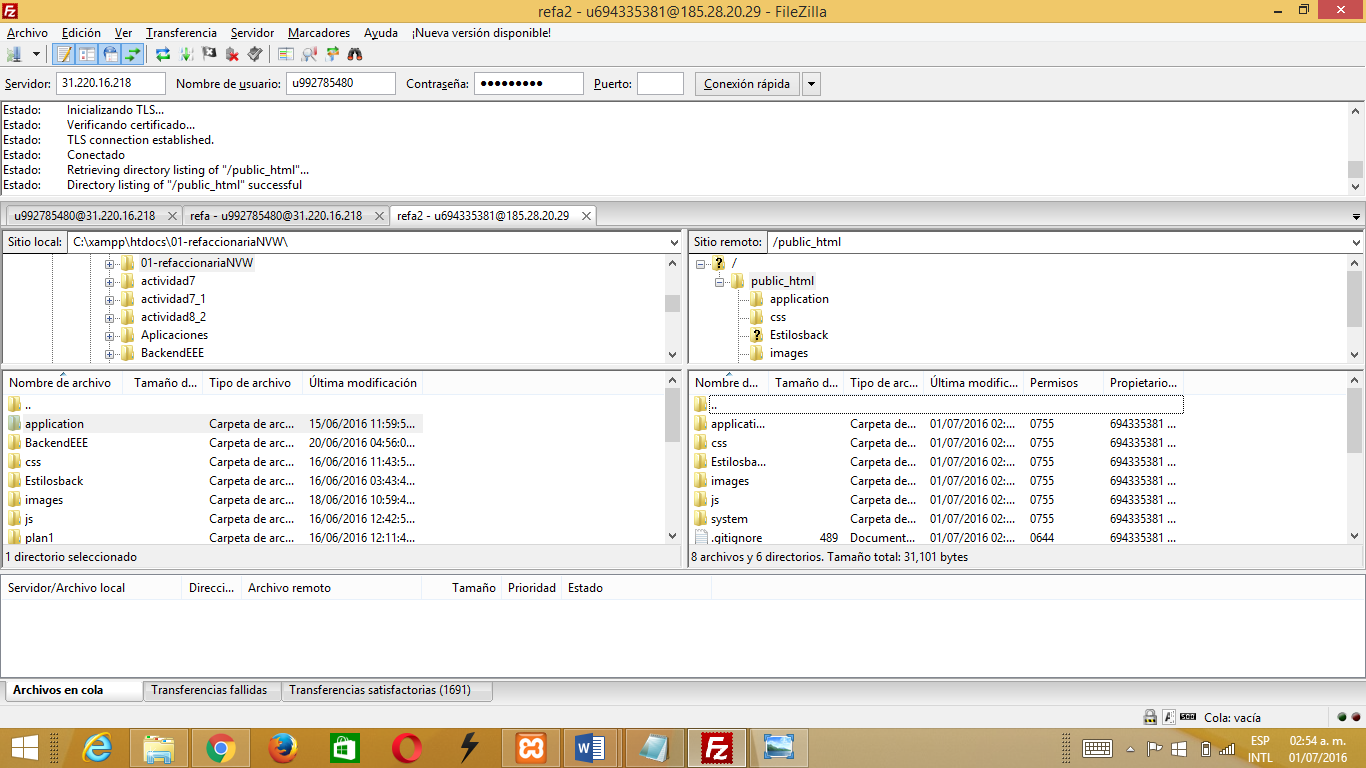


1. Ubicamos en el explorador de archivos del lado derecho la carpeta public\_html y del lado izquierdo de la carpeta se encuentra en tu pc con los archivos del sitio web.

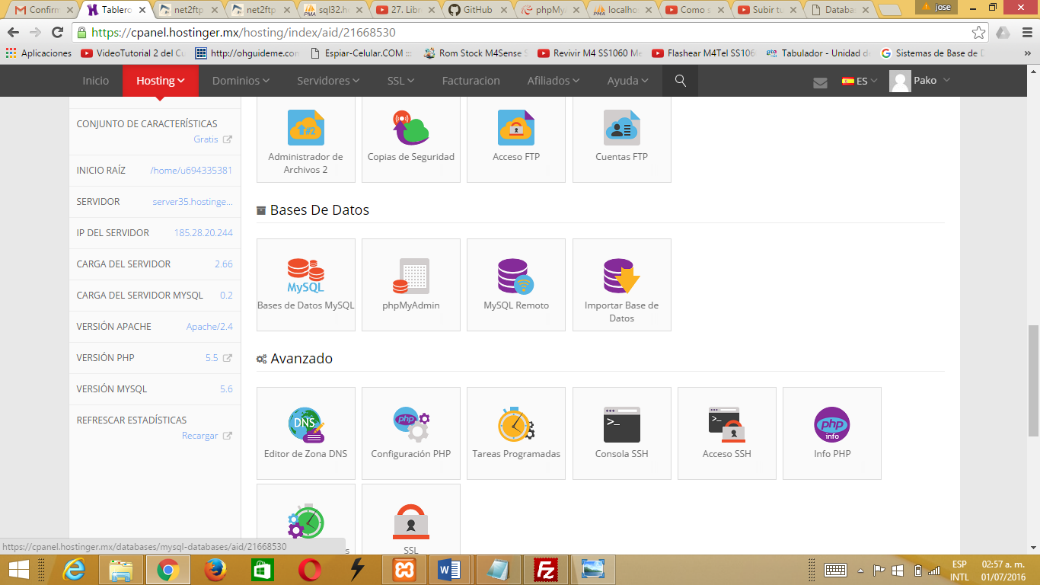


1. Seleccionamos lo archivos que vamos a subir al servidor.

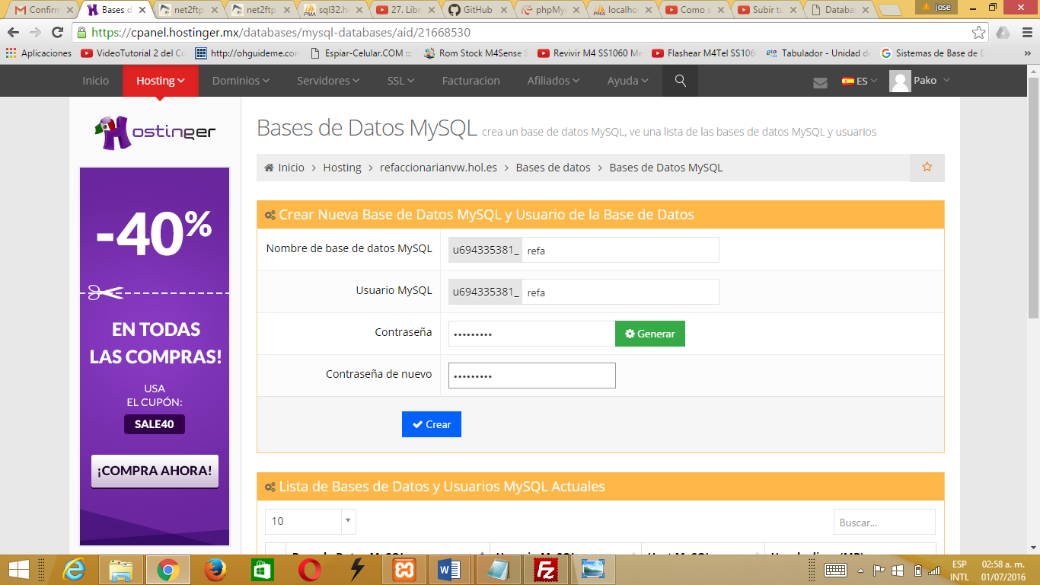




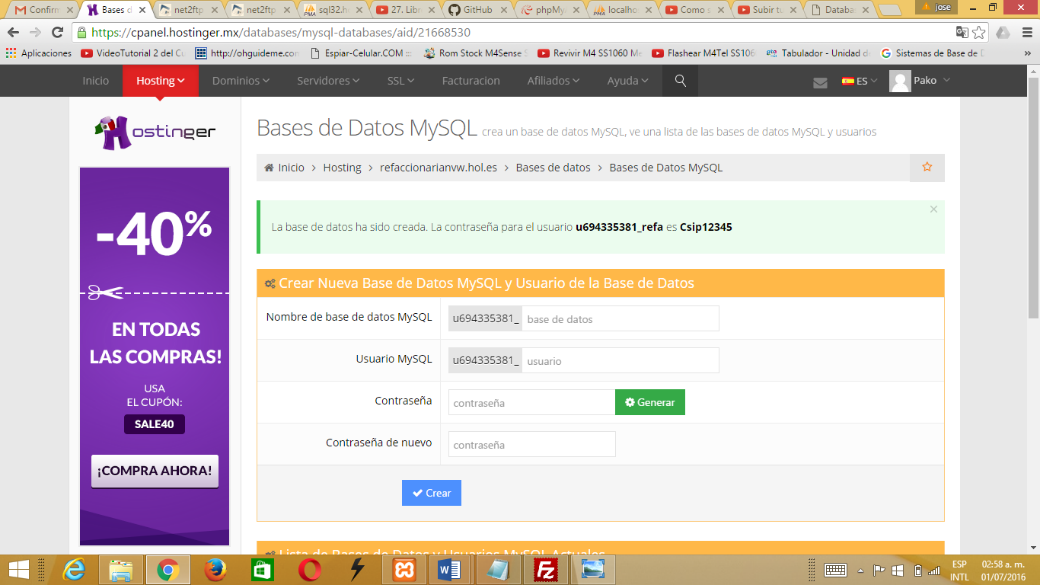
1. La siguiente parte será importar la base de datos al servidor web. Es por eso que ingresamos al panel de control y nos colocamos en base de datos, en mysql, la cual nos abrirá una pestaña nueva.



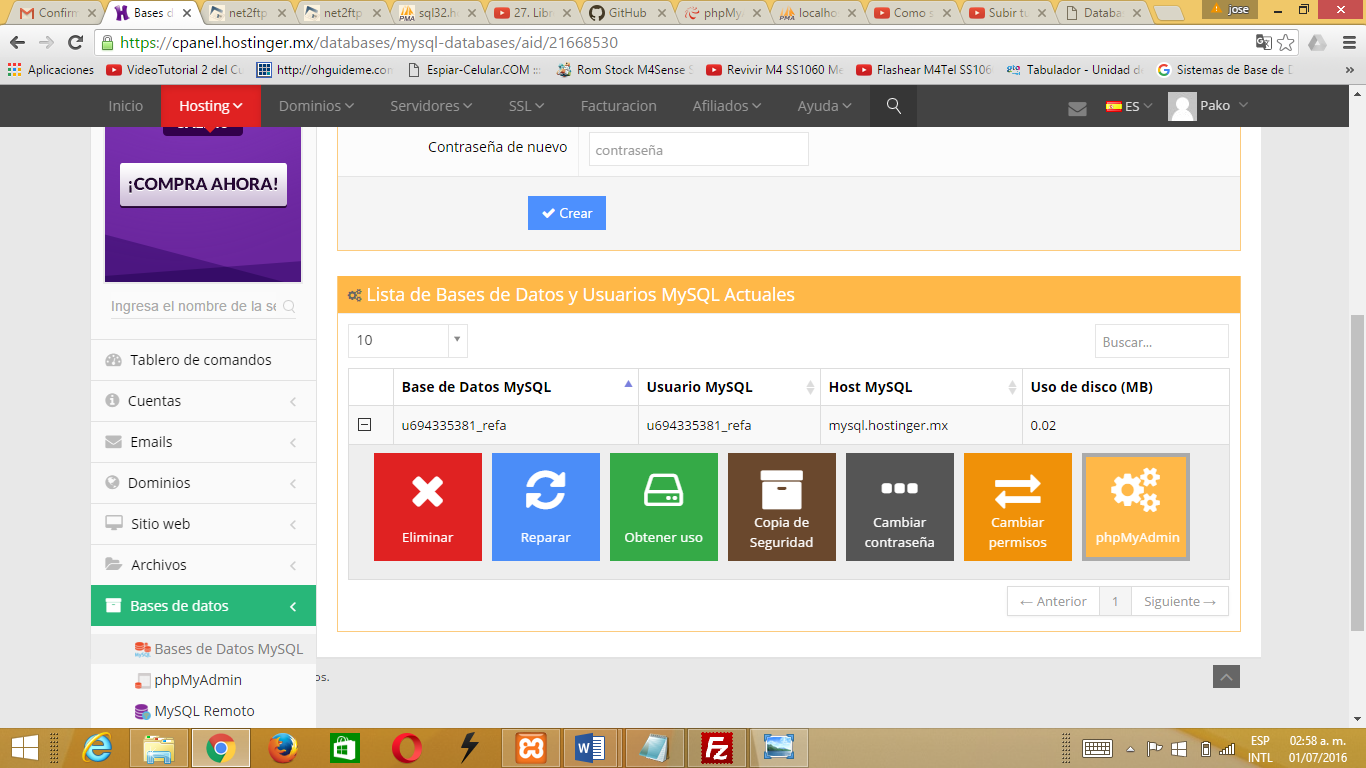
1. En este paso crearemos la base de datos para posteriormente importarla. En este apartado nos pedirá el nombre de la base de datos de MySQL, usuario MySQL y una contraseña, llenado todos estos puntos podemos dar clic en crear.



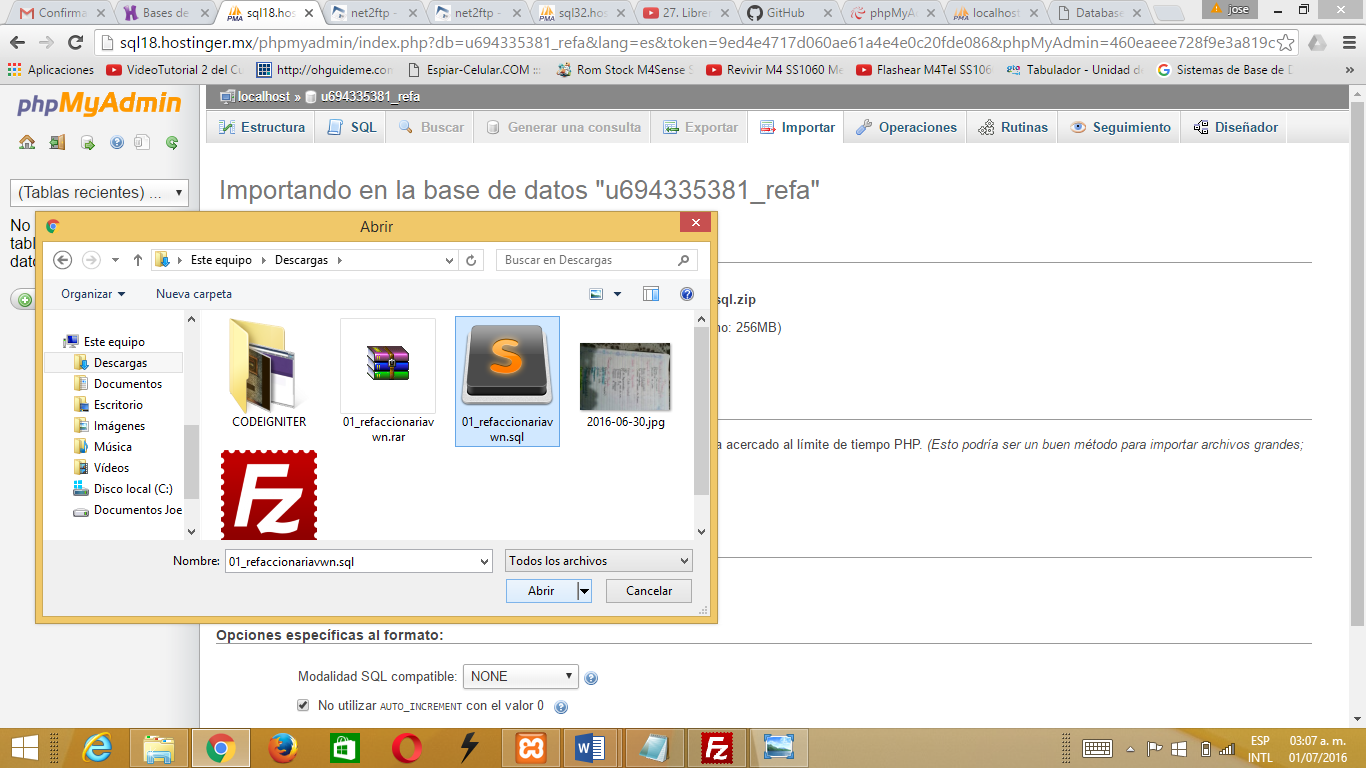
1. La base de datos ha sido creada, así mismo nos mostrara el usuario y contraseña que posteriormente llenamos.



1. Nos colocaremos del lado izquiero de la pantalla, para ubicar la pestaña que dice base de datos la desplegamos y damos clic en phpMyAdmin. Nos aparecera la base de datos que creamos y le damos la opcion de ingresar, el cual nos mandara al phpMyAdmin como aparece en la imagen de abajo.



1. Nos aparecera la base de datos que creamos, le damos clic en la opcion de ingresar, el cual nos mandara al phpMyAdmin como aparece en la imagen de abajo, a hora importaremos la base de datos que ya tengamos creada al servidor web.



1. La base de datos se exportado exitosamente como se muestra en la imagen.



**MANUAL DE COMO SE GENERA LA BASE DE DATOS EN EL SERVIDOR**

# INNOVACIÓN

# ANALÍTICO-SINTÉTICO (CONCLUSIÓN)

# AUTO Y CO-EVALUACIÓN

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Integrantes | 1 | 2 | 3 | 4 | Total |
| 1. José Sánchez Rosales | 9 | 9 | 9 | 9 | 9 |
| 2. José Francisco López Ruiz | 9 | 9 | 9 | 9 | 9 |
| 3. Daniel Ávila Chávez | 9 | 9 | 9 | 9 | 9 |
| 4. Brayan Dilan Cuevas Mora | 9 | 9 | 9 | 9 | 9 |

# BIBLIOGRAFÍA

Castillo Sequera, J. L., & Navarro Huerga, M. Á. (2015). *Comercio electrónico y aspectos prácticos de implementación con Magento.* España: Servicio de Publicaciones. Universidad de Alcalá. Obtenido de http://site.ebrary.com.ezproxy.bibliotecaecest.mx/lib/bidigecestsp/detail.action?docID=11087869&p00=plataformas+comercio+electronico

Galvan, P. (2015). *http://sg.com.mx*. Obtenido de http://sg.com.mx/revista/50/estudio-salarios-sg-2015#.V3hiIWh97IV

Technology, B. C. (21 de march de 2016). *http://www.codeigniter.com*. Obtenido de http://www.codeigniter.com/user\_guide/

# ANEXOS

* KDFHSJKDF

# DIAGRAMA DE ACTIVIDADES.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Número | Actividad | Responsable | Fechas | | | | Áreas de mejora y tentativas de solución | Puntos descontados por penalizaciones |
|  | | | Inicio Propuesto | Fin propuesto | Inicio Real | Fin real |  | |
| 1 | Introducción del documento | José Francisco López Ruiz | 27-junio-2016 | 28-junio-2016 | 27-junio-2016 | 29-junio-2016 | En general para todas las actividades de esta práctica las áreas de mejora serán las mismas ya que se están asignando las tareas correspondientes a cada uno de los integrantes del equipo y se aplicarán las mismas medidas para mejorar nuestra responsabilidad respecto a las fechas que se están estableciendo para entregar las tareas, dichas medidas serán que en caso de que el integrante del equipo no entregue su tarea en la fecha que se establece en este diagrama de actividades se penalizará con 1 punto en el auto y co-evaluación, es decir, en el principio de la actividad cada uno de los integrantes cuenta con los 10 puntos, pero al momento en que se retrase alguno de los integrantes se le quitará 1 punto y en caso de que la tarea sea entregada a tiempo pero no cuente con lo establecido en la rúbrica se le quitará al alumno 1 punto y así respecto a todas y cada una de las tareas hasta llegar a cero puntos. De esta manera se podrá observar que integrante del equipo trabajó como debía trabajar y quien no lo hizo así.  En la siguiente columna se observan los puntos que se descontarán a cada integrante por cada una de las tareas correspondientes.  De esta forma estamos implementando soluciones a los posibles problemas que se nos puedan presentar durante el desarrollo de la actividad y con esto estamos mejorando y formándonos tratando de ser más responsables. | 1 |
| 2 | Estructuración, revisión y corrección del documento | José Francisco López Ruiz | 27-junio-2016 | 02-julio-2016 | 27-junio-2016 |  |  |
| 3 | Elaboración dela plantilla 01-RefaccionariaNVW\_BD de Riesgos | Brayan Dilan Cuevas Mora y apoyo del equipo | 27-junio-2016 | 28-junio-2016 | 27-junio-2016 |  |  |
| 4 | Elaboración de la plantilla 01-RefaccionariaNVW\_Plan de Contingencia de Riesgos | Brayan Dilan Cuevas Mora | 27-junio-2016 | 01-julio-2016 | 27-junio-2016 |  |  |
| 5 | Estudio de salarios | José Sánchez Rosales | 27-junio-2016 | 28-junio-2016 | 27-junio-2016 | 28-junio-2016 | 0 |
| 6 | WBS (Work Breakdown Structure) | José Sánchez Rosales | 27-junio-2016 | 01-julio-2016 | 27-junio-2016 |  |  |
| 7 | Conexión a repositorios del proyecto | Brayan Dylan Cuevas Mora | 27-junio-2016 | 28-junio-2016 | 28-junio-2016 |  |  |
| 8 | Flujo de trabajo colaborativo | Brayan Dilan Cuevas Mora y apoyo del equipo | 27-junio-2016 | 02-julio-2016 | 27-junio-2016 |  |  |
| 9 | Servidores gratuitos de internet | Daniel Ávila Chávez | 27-junio-2016 | 29-junio-2016 | 27-junio-2016 |  |  |
| 10 | Dominio | José Sánchez Rosales | 30-junio-2016 | 02-julio-2016 |  |  |  |
| 11 | Aplicación web módulos | Todos los integrantes del equipo | 28-junio-2016 | 02-julio-2016 | 28-junio-2016 |  |  |
| 12 | Librerías de Codeigniter | Brayan Dilan Cuevas Mora | 27-junio-2016 | 29-junio-2016 | 27-junio-2016 |  |  |
| 13 | Plataformas de comercio electrónico | Daniel Ávila Chávez | 27-junio-2016 | 29-junio-2016 | 27-junio-2016 |  |  |
| 14 | Innovación | Todos los integrantes del equipo | 27-junio-2016 | 30-junio-2016 |  |  |  |
| 15 | Analítico-Sintético | Daniel Ávila Chávez | 27-junio-2016 | 30-junio-2016 |  |  |  |
| 16 | Auto y co-evaluación | Todos los integrantes del equipo | 27-junio-2016 | 01-julio-2016 |  |  |  |
| 17 | Anexos | José Francisco López Ruiz | 27-junio-2016 | 02-julio-2016 |  |  |  |
| 18 | Bibliografía | Daniel Ávila Chávez | 27-junio-2016 | 02-julio-2016 |  |  |  |
| 19 | Diagrama de actividades | José Francisco López Ruiz | 27-junio-2016 | 02-julio-2016 | 27-junio-2016 | 28-junio-2016 | 0 |