



UTPL

La Universidad Católica de Loja

Vicerrectorado de Modalidad Abierta y a Distancia

Desarrollo Web

Guía didáctica





Facultad Ingenierías y Arquitectura

Desarrollo Web

Guía didáctica

Carrera	PAO Nivel
Tecnologías de la información	V

Autor:

Ramiro Leonardo Ramírez Coronel



D S O F _ 3 0 4 2



Universidad Técnica Particular de Loja

Desarrollo Web

Guía didáctica

Ramiro Leonardo Ramírez Coronel

Diagramación y diseño digital

Ediloja Cía. Ltda.

Marcelino Champagnat s/n y París

edilojacialtda@ediloja.com.ec

www.ediloja.com.ec

ISBN digital -978-9942-25-716-1

Año de edición: abril 2020

Edición: primera edición reestructurada en septiembre 2024 (con un cambio del 5%)

Loja-Ecuador



Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional (CC BY-NC-SA 4.0)

Usted acepta y acuerda estar obligado por los términos y condiciones de esta Licencia, por lo que, si existe el incumplimiento de algunas de estas condiciones, no se autoriza el uso de ningún contenido.

Los contenidos de este trabajo están sujetos a una licencia internacional Creative Commons **Reconocimiento-NoComercial-CompartirIgual 4.0** (CC BY-NC-SA 4.0). Usted es libre de **Compartir** — *copiar y redistribuir el material en cualquier medio o formato*. **Adaptar** — *remezclar, transformar y construir a partir del material citando la fuente, bajo los siguientes términos:*
Reconocimiento- *debe dar crédito de manera adecuada, brindar un enlace a la*



licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante. No Comercial-no puede hacer uso del material con propósitos comerciales. Compartir igual-Si remezcla, transforma o crea a partir del material, debe distribuir su contribución bajo la misma licencia del original. No puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

<https://creativecommons.org/licenses/by-nc-sa/4.0/>



Índice

1. Datos de información	10
1.1 Presentación de la asignatura	10
1.2 Competencias genéricas de la UTPL.....	10
1.3 Competencias específicas de la carrera	10
1.4 Problemática que aborda la asignatura	10
2. Metodología de aprendizaje	11
3. Orientaciones didácticas por resultados de aprendizaje	12
Primer bimestre	12
Resultados de aprendizaje 1, 2 y 3:.....	12
Contenidos, recursos y actividades de aprendizaje recomendadas	12
Semana 1	13
Unidad 1. Interfaces web.....	13
1.1 Integración e implementación de sitios web seguros.....	14
1.2 Integración de base de datos	16
1.3 Problemas de accesibilidad	17
Actividades de aprendizaje recomendadas	22
Autoevaluación 1.....	23
Resultados de aprendizaje 1, 4 y 5:.....	26
Contenidos, recursos y actividades de aprendizaje recomendadas	26
Semana 2	26
Unidad 2. Programación en PHP.....	27
2.1 Configuración del entorno de desarrollo	29
2.2 Primeros pasos con PHP.....	31
2.3 Variables	33
2.4 Operadores	35
2.5 Estructuras de control.....	36
Resultados de aprendizaje 1, 4 y 5:.....	37



Contenidos, recursos y actividades de aprendizaje recomendadas	37
Semana 3	37
Unidad 2. Programación en PHP.....	37
2.6 Funciones	37
2.7 Sesiones	39
Actividades de aprendizaje recomendadas	39
Autoevaluación 2.....	40
Resultados de aprendizaje 6 y 7:.....	43
Contenidos, recursos y actividades de aprendizaje recomendadas	43
Semana 4	43
Unidad 3. PHP con MYSQL.....	43
3.1 Gestión de datos	44
3.2 Creación de base de datos	44
3.3 Listar las bases de datos de nuestro servidor	45
3.4 Creación de tablas en la base de datos.....	46
3.5 Manipulación de datos con SQL y phpMyAdmin	47
Resultados de aprendizaje 6 y 7:.....	50
Contenidos, recursos y actividades de aprendizaje recomendadas	50
Semana 5	50
Unidad 3. PHP con MYSQL.....	50
3.6 Conectar una base de datos con PHP	50
3.7 Gestión de datos con PHP y MYSQL CRUD	53
3.8 Formulario de registro	53
Resultados de aprendizaje 6 y 7:.....	56
Contenidos, recursos y actividades de aprendizaje recomendadas	56
Semana 6	56
Unidad 3. PHP con MYSQL.....	56
3.9 Borrar los registros	56
3.10 Actualizar los registros.....	58



Actividad de aprendizaje recomendada	61
Autoevaluación 3.....	61
Resultados de aprendizaje 1 a 7:	64
Contenidos, recursos y actividades de aprendizaje recomendadas	64
Semana 7	64
Actividades finales del bimestre	64
Resultados de aprendizaje 1 a 7:	65
Contenidos, recursos y actividades de aprendizaje recomendadas	65
Semana 8	65
Actividades finales del bimestre	65
Segundo bimestre	66
Resultados de aprendizaje 6, 8, 9 y 10:	66
Contenidos, recursos y actividades de aprendizaje recomendadas	66
Semana 9	67
Unidad 4. Programación orientada a objetos.....	67
4.1 Métodos y funciones para manejar eventos	67
Resultados de aprendizaje 6, 8, 9 y 10:	70
Contenidos, recursos y actividades de aprendizaje recomendadas	70
Semana 10	70
Unidad 4. Programación orientada a objetos.....	70
4.2 Propagación de objetos, eventos.....	70
4.3 Manejo de excepciones	71
Actividad de aprendizaje recomendada	73
Autoevaluación 4.....	73
Resultados de aprendizaje 5, 7, 9 y 11:	76
Contenidos, recursos y actividades de aprendizaje recomendadas	76
Semana 11	76
Unidad 5. Seguridad y vulnerabilidades.....	77
5.1 Ataques XSSCross-Site Scripting.....	77



5.2 Non-persistent XSS	78
Resultado de aprendizaje 3, 5, 9 y 11:	79
Contenidos, recursos y actividades de aprendizaje recomendadas	79
Semana 12	79
Unidad 5. Seguridad y vulnerabilidades.....	80
5.3 Prevención de ataques	80
5.4 Validación de datos	80
Actividad de aprendizaje recomendada	83
Autoevaluación 5.....	84
Resultado de aprendizaje 8, 9, 10 y 11:	87
Contenidos, recursos y actividades de aprendizaje recomendadas	87
Semana 13	87
Unidad 6. Seguridad web.....	88
6.1 Seguridad del servidor	88
6.2 Control del acceso al servidor con el archivo .htaccess	90
Actividad de aprendizaje recomendada	90
Resultado de aprendizaje 3, 5, 9 y 11:	92
Contenidos, recursos y actividades de aprendizaje recomendadas	92
Semana 14	92
Unidad 6. Seguridad web.....	93
6.3 Configuración del servidor ante ataques de servicio	93
6.4 ServerRoot	96
6.5 ServerName	96
6.6 Puesta en marcha lasWebApps	96
Actividades de aprendizaje recomendadas	99
Autoevaluación 6.....	100
Resultados de aprendizaje 3, 5, 6, 7, 8, 9 y 10:	102
Contenidos, recursos y actividades de aprendizaje recomendadas	102
Semana 15	102



Actividades finales del bimestre	102
Resultados de aprendizaje 3, 5, 6, 7, 8, 9 y 10:	104
Contenidos, recursos y actividades de aprendizaje recomendadas	104
Semana 16	104
Actividades finales del bimestre	104
4. Solucionario	105
5. Glosario	111
6. Referencias Bibliográficas	113





1. Datos de información

1.1 Presentación de la asignatura



1.2 Competencias genéricas de la UTPL

Orientación a la investigación e innovación.

1.3 Competencias específicas de la carrera

Diseñar aplicaciones de software que permitan mediante técnicas avanzadas de modelado dar solución a los requerimientos del cliente utilizando estándares de la industria.

1.4 Problemática que aborda la asignatura

Se identifica como tensión la escasez de proyectos para aprovechar la actividad financiera y bancaria que se caracterizan por sus altos niveles de aplicación de tecnología, en esta línea la carrera propone una formación específica en el ámbito de proyectos, arquitectura empresarial y formación avanzada en el uso de tecnologías web, cloud, móviles y desarrollo de sistemas.





2. Metodología de aprendizaje

La metodología de aprendizaje que se va a usar en el desarrollo de las actividades de la asignatura de desarrollo web es una metodología de aprendizaje basado en TIC. Esta permite, a través de la utilización de medios tecnológicos, mejorar el proceso de aprendizaje utilizando el computador, software disponible e internet como herramientas mediadoras del conocimiento.





3. Orientaciones didácticas por resultados de aprendizaje



Primer bimestre

Resultados de aprendizaje 1, 2 y 3:

- Describe los problemas implicados en el desarrollo de una interfaz web.
- Resume la necesidad y las cuestiones implicadas en la ejecución y la integración de una página web.
- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.

Para alcanzar los resultados de aprendizaje planteados, es esencial comprender y abordar los problemas relacionados con el desarrollo y la gestión de interfaces web. Esto implica identificar los desafíos en el diseño de interfaces, resumir la necesidad y las cuestiones clave en la ejecución e integración de páginas web, y explicar la importancia de la interconexión entre sitios web y bases de datos subyacentes. Además, se abordará la integración e implementación de sitios web seguros, la integración de bases de datos y los problemas de accesibilidad, ofreciendo una visión completa para resolver estos aspectos fundamentales

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.





Semana 1

Unidad 1. Interfaces web

Estimado estudiante, iniciamos el estudio de la asignatura, mencionando temas fundamentales que se requiere conocer dentro del maravilloso mundo del desarrollo web para aplicaciones en *Internet*. Comenzamos revisando algunos conceptos básicos de interfaces web, accesibilidad web; esto le ayudará a comprender lo importante de este tema.

La accesibilidad conlleva un trabajo extenso y detallado, ya que abarca una iteración múltiple de elementos como audio, video, imágenes, la estructura lógica de la aplicación. Representando una labor que va más allá de la presentación visual de la información.

Al momento de estar en la fase de diseño es necesario tener en cuenta aspectos como la “usabilidad” y la “accesibilidad”.

- **Usabilidad:** se refiere al tiempo, esfuerzo y capacidad requerido por el usuario para que alcance un determinado nivel de adaptación en el sistema. Esto puede estar vinculado al número de pasos que se tiene que cumplir para lograr un determinado objetivo y el nivel de conocimientos que requiere el usuario para la manipulación de la *WebApp* (Aplicaciones web).
- **Accesibilidad:** una *WebApp* debe ser creada para que sea vista por todo el mundo, pero hay que tener en cuenta que estos usuarios pueden ingresar a la información por diferentes dispositivos, navegadores, idiomas, etc. Estos son factores que pueden impedir que la aplicación se muestre correctamente en su totalidad.



1.1 Integración e implementación de sitios web seguros

La utilización de documentos HTML estáticos puede ser la solución más adecuada cuando la aplicación web se limite a ofrecer siempre la misma información o podamos automatizar la realización de actualizaciones de los documentos HTML que la constituyen; la naturaleza dinámica de la web y las expectativas que ha creado en la actualidad hacen necesaria la implementación de aplicaciones web seguras que generen dinámicamente el contenido que finalmente se les ofrece a los usuarios. De esta forma podemos seleccionar, filtrar, ordenar y presentar la información de la manera más adecuada, en función de las necesidades de cada momento.

Si bien esto se podría conseguir con páginas HTML estáticas, si dispusiésemos de espacio suficiente en disco (y, de hecho, esta es una estrategia que se utiliza para disminuir la carga de la CPU de los servidores), las aplicaciones web nos permiten ofrecer la información actualizada, esto se da al poder acceder directamente a las bases de datos que contienen los datos operativos de una empresa.

Es muy común escuchar a personas pidiendo consejos, tales como, “¿este *website* será seguro para colocar mi información general y bancaria?”. Es por ello que se debe utilizar protocolos o procesos de seguridad como por ejemplo la utilización de certificados de seguridad en dominios con el protocolo HTTPS, denegación de accesos a la información, entre otros.

Control de acceso a la información

Las aplicaciones web tienen acceso abierto a un conjunto de información que explícitamente se hace pública. Sin embargo, en determinadas circunstancias, es interesante poder limitar el acceso a documentos reservados o útiles para un conjunto restringido de personas. Se pueden establecer dos tipos de restricciones:

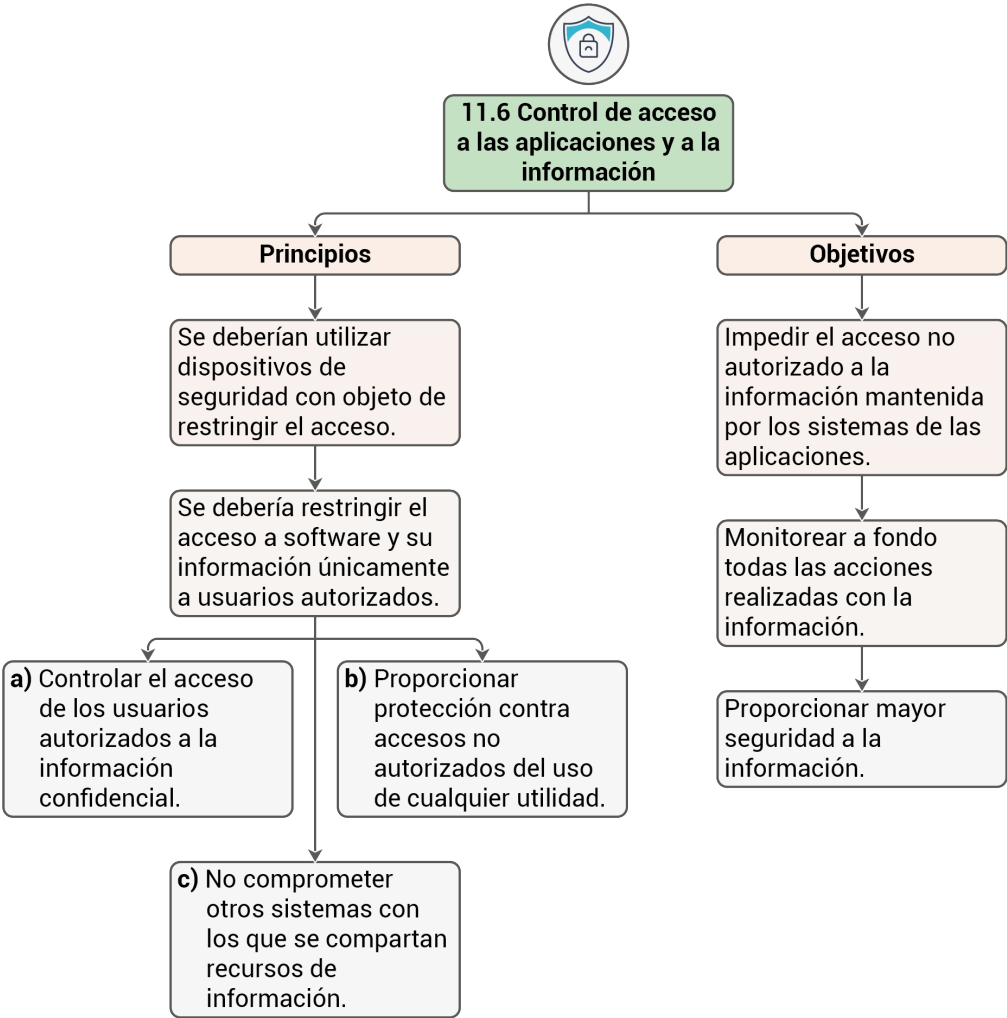
1. **Limitación de acceso en función de direcciones IP o dominio:** solo los usuarios de un dominio u organización tendrán acceso a la información.



2. **Limitación de acceso por nombres de usuario y claves de acceso:** solo los usuarios que conozcan una clave de acceso válida pueden acceder a la información tal como se muestra en la figura 1.

Figura 1

Diagrama del control de acceso a la información, principios y objetivos



Nota. Ramírez, R., 2024.

Otro aspecto que está cobrando especial importancia es la seguridad de la información que se intercambia en la web.

La *Internet* exige disponer de sistemas de comunicación seguros, capaces de adaptarse a las necesidades de los nuevos servicios, como la compra electrónica o la banca a distancia. En estos servicios se manejan dos conceptos fundamentales: la **autenticación** y la **confidencialidad**.

Control de acceso a la información se utiliza para limitar el acceso a determinados documentos de un servidor web, en función del origen y tipo de petición. La forma de hacerlo varía con el entorno en el que se publican las páginas (sistema operativo y servidor HTTP, principalmente); en general, todas las soluciones pasan por definir un fichero que contiene las diferentes limitaciones de acceso, en un formato característico del servidor HTTP. En algunos casos se utiliza un fichero global con las restricciones de acceso o bien un fichero por cada directorio al que se quiere limitar el acceso.

Cuando un cliente web accede a un fichero protegido, el servidor devuelve un **código de error** asociado a la falta de permisos para realizar la operación (código 401).



No olvide que debe acceder al Entorno Virtual de Aprendizaje para interactuar con el tutor y sus compañeros, por si le quedó alguna duda de cómo implementar seguridad en las WebsApps.

1.2 Integración de base de datos

La integración de datos es el proceso que permite combinar datos heterogéneos de muchas fuentes diferentes en la forma y estructura de una única aplicación. Este proceso de integración de datos facilita que diferentes tipos de datos, tales como matrices de datos, documentos y tablas, sean fusionados por usuarios, organizaciones y aplicaciones para un uso personal, de procesos de negocio o de funciones.



La integración de datos soporta el procesamiento analítico de grandes conjuntos de datos, alineando, combinando y presentando cada conjunto de datos de departamentos organizacionales y fuentes de datos remotas y externas, para cumplir con los objetivos del integrador.

La integración de datos se implementa generalmente en una “**data warehouse**” mediante *software* especializado que aloja grandes repositorios de datos de recursos internos y externos. Los datos se extraen, se mezclan y se presentan de forma unificada. Por ejemplo, el conjunto completo de datos de un usuario puede incluir datos extraídos y combinados de *marketing*, ventas y operaciones, que se combinan para formar un informe completo (base de datos).

Un proyecto de integración de datos generalmente implica los siguientes pasos:

- **Acceso a los datos** desde todas las fuentes y localizaciones, tanto si se trata de locales, en la nube o de una combinación de ambos.
- **Integración de datos** de modo que los registros de una fuente de datos mapean registros en otra. Por ejemplo, incluso si un conjunto de datos utilizara “nombre, apellidos” y otro “nom, ape”, el conjunto integrado se asegurará de que en ambos casos los datos se almacenen en el lugar correcto. Se trata, pues, de un tipo de preparación de datos esencial para que las analíticas y otras aplicaciones sean capaces de utilizar los datos con éxito.
- **Entrega de datos integrados** al negocio justo en el momento en que la empresa los necesita, ya sea por lotes, casi en tiempo real o en tiempo real (base de datos).

1.3 Problemas de accesibilidad

Estimado estudiante, empezamos el apartado de la asignatura tomando temas fundamentales que se requiere entender dentro de la accesibilidad.



¿Qué es accesibilidad web?

La accesibilidad tiene como principal objetivo que personas con algún tipo de discapacidad puedan hacer uso de la web. Al hablar de accesibilidad web como se lo representa en la figura 2, se está haciendo referencia a un diseño web que va a permitir que estas personas puedan percibir, entender, navegar e interactuar con la web, aportando a su vez contenidos.

Figura 2
Accesibilidad web



Nota. Tomado de Accesibilidad Web - Definición, características y ejemplos [Ilustración], por Delgado, H., 2022, [disenowebakus](#). CC BY.4.0.

Esta engloba muchos tipos de discapacidades, incluyendo problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla. Pero la accesibilidad web beneficia también a organizaciones y a personas sin discapacidad. Por ejemplo, un principio básico de la accesibilidad web es la flexibilidad, con el objetivo de satisfacer diferentes necesidades, situaciones

y preferencias. Esta flexibilidad va a beneficiar a todas aquellas personas que utilizan la web, incluyendo personas que no tienen ninguna discapacidad, pero que, debido a determinadas situaciones, tienen dificultades para acceder a la web (por ejemplo, una conexión lenta), también estaríamos hablando de aquellas personas que sufren una incapacidad transitoria (por ejemplo, un brazo roto), y de personas de edad avanzada.



Espero que la lectura que realizó haya sido entendida y comprendida, para ello acuda a su cuaderno de trabajo y escriba cuál es el concepto de accesibilidad web y responda las siguientes preguntas:

- ¿A quiénes beneficia la accesibilidad web?
- ¿Ayudará a personas sin discapacidad y por qué?

Para entender un poco más en lo que es accesibilidad web, tenemos que ver la importancia de incorporar estas técnicas de desarrollo en nuestras aplicaciones web.

¿Por qué es importante la accesibilidad web?:

Actualmente, la web ha cobrado mucha importancia, especialmente en áreas como educación, empleo, gobierno, comunicación, economía, entretenimiento, etc., por ello es necesario que sea accesible para así proporcionar un acceso equitativo e igualdad de oportunidades a las personas con discapacidad y público en general.

Muy sencillo de entender el concepto. ¿Verdad?

¿Qué opina usted sobre el siguiente enunciado o aseveración?:

¿La accesibilidad web es importante porque permite el acceso equilibrado e igual oportunidad a las personas con discapacidad?



Con base en la pregunta propuesta se puede concluir que es muy relevante e importante la utilización de la accesibilidad en aplicaciones web, ya que no debe existir ninguna restricción de acceso de información a ninguna persona que acceda a la información. Para ello es importante cumplir las diferentes reglas de accesibilidad.

Cómo hacer un sitio accesible

Una de las funciones de la Iniciativa de Accesibilidad Web (WAI: *Web Accessibility Initiative*) es desarrollar pautas y técnicas que proporcionen soluciones accesibles para el *software web* y para los desarrolladores web. Las pautas de WAI son consideradas como estándares internacionales de accesibilidad web.

Hacer un sitio web accesible puede ser algo sencillo o complejo, depende de muchos factores como el tipo de contenido, el tamaño y la complejidad del sitio, así como de las herramientas de desarrollo y el entorno.

Muchas de las características accesibles de un sitio se implementan de forma sencilla si se planean desde el principio del desarrollo del sitio web, o al comienzo de su rediseño. La modificación de sitios web inaccesibles puede requerir un gran esfuerzo, sobre todo aquellos que no se “etiquetaron” correctamente con etiquetas estándares de HTML5 (*HyperText Markup Language 5*), y sitios con cierto tipo de contenido, como multimedia.

Las nuevas etiquetas que se crearon en el lenguaje de etiquetado HTML5 ayudan a reemplazar el abuso excesivo de etiquetas <div> que habitualmente se utilizan con el fin de delimitar o emular las partes de un documento como cabecera, pie de página, contenedores, etc. Con el desarrollo y la utilización de las etiquetas HTML5 se pretende crear una estructura de vistas más homogeneizada y que el manejo de meta data sea más adecuado. Se trata de llegar a lo que se ha dado en denominar la “web semántica”, siendo la presentación de documentos mucho más coherente y fácil de entender por



otras personas y lo que es más importante, siendo trivial de entender para una máquina, concretamente hacer que el buscador realice las búsquedas mucho más fáciles hacia el contenido de nuestras aplicaciones.

Continuemos con el aprendizaje mediante la revisión del siguiente video donde se explica a detalle los objetivos de la [accesibilidad web](#).

1.3.1 Iniciativa de accesibilidad web

A continuación, en la siguiente infografía se muestra una lista de consideraciones a tomar en cuenta al momento del desarrollo web, ya que son puntos muy importantes a aplicar para que los resultados accesibles sean los correctos.

[Estructura y presentación en el desarrollo web](#)

Con la lectura y el análisis que acabó de realizar, acuda a su cuaderno de trabajo y mencione dos factores que intervienen en la accesibilidad de un sitio web



Ingresa al EVA (foro) y mencione las formas de hacer un sitio web accesible que usted encontró y que haya investigado.

Terminada esta tarea podemos pasar al siguiente tema que es:

Evaluación de la accesibilidad

La evaluación de la accesibilidad de forma temprana, y a lo largo del desarrollo, permite encontrar al principio problemas de accesibilidad, cuando es más fácil resolverlos. Técnicas sencillas, como cambiar la configuración en un buscador, pueden determinar si una página web cumple algunas de las pautas de accesibilidad.



Hay herramientas de evaluación que ayudan a realizar evaluaciones de accesibilidad. No obstante, ninguna herramienta en sí misma puede determinar si un sitio cumple o no las pautas de accesibilidad. Para determinar si un sitio web es accesible, es necesaria la evaluación humana.

A continuación, una lista de herramientas en línea que ayudará a medir la accesibilidad de una WebApp:

- [Tawdis](#).
- [Sidar](#).
- [Validator](#).
- [Usableyaccesible](#).



Actividades de aprendizaje recomendadas

Es hora de reforzar los conocimientos adquiridos resolviendo las siguientes actividades:

1. Con la lectura rápida que haya realizado sobre accesibilidad está en condiciones de completar la siguiente frase:

La evaluación temprana de la accesibilidad permite ...

Para dar respuesta a esta pregunta puede utilizar *Internet*.

2. Continuemos con el aprendizaje mediante la revisión de la siguiente página denominada "[Validadores y herramientas para consultorías de accesibilidad y usabilidad](#)" y, mencione las diferentes herramientas que encontró.

Nota: por favor, complete las actividades en un cuaderno o documento Word.

3. Ahora, acuda a realizar la validación del portal de la UTP con mínimo 2 herramientas encontradas, una vez obtenido los



resultados menciónelos en el foro de accesibilidad del curso y emita un criterio técnico para mejorar dichos resultados.

4. Estimado estudiante, le invito a medir sus conocimientos de esta unidad con la siguiente autoevaluación, esta le ayudará a valorar cuál es su nivel de conocimientos obtenidos durante el transcurso de este período transcurrido, en caso de no obtener resultados satisfactorios le sugiero revisar nuevamente los contenidos en el cual tiene problemas.



Autoevaluación 1

Conteste correctamente las preguntas según sea el caso:

1. Seleccione de la siguiente lista cuál es el objetivo de la accesibilidad web.
 - a. Acceso a aplicaciones restringidas para cualquier usuario.
 - b. Acceso a aplicaciones para cualquier usuario normal o con discapacidad.
 - c. Acceso a recursos ricos como PDF, doc.
2. La accesibilidad ha cobrado importancia en áreas como, seleccione más de una respuesta:
 - a. Educación.
 - b. Industria automotriz.
 - c. Comunicación digital.
 - d. Empleo.
3. Las siglas WAI en español significa, seleccione la respuesta correcta.
 - a. Accesibilidad Web Incorporada.
 - b. Iniciativa de Accesibilidad Web.
 - c. Importancia de la Accesibilidad Web.
4. Localice las técnicas de accesibilidad de estructura y presentación.
 - a. Sitio marcado semánticamente.
 - b. Proporcionan resúmenes a las tablas.
 - c. Tamaño de imágenes proporcionales al contenido.



- d. Contenido organizado lógicamente.
5. De la siguiente lista marque cuáles son los validadores de accesibilidad.
- a. [Tawdis](#).
 - b. www.accessvalidatorwai.com.
 - c. <http://www.sidar.org/hera/>.
 - d. <http://walidator.net/uwem/index.es.php>.
6. La característica principal de las etiquetas HTML5 es:
- a. Etiquetado semántico para las partes importantes de los elementos que componen la página web.
 - b. Etiquetas que ayudan a mejorar la programación de la página web.
 - c. Etiquetas, mejorar la representación semántica y ayuda a la accesibilidad web.
7. La etiqueta NAV sirve para representar:
- a. La cabecera de la página web.
 - b. Los módulos internos de la página web.
 - c. Los menús de navegación.
8. La etiqueta section sirve para:
- a. Colocar la cabecera de la página web.
 - b. Maquetar las secciones de contenido de la página web.
 - c. Colocar los pies de página de las páginas web.
9. La etiqueta footer sirve para:
- a. Colocar la cabecera de la página web.
 - b. Maquetar las secciones de contenido de la página web.
 - c. Colocar los pies de página de las páginas web.
10. La etiqueta main sirve para:
- a. Colocar la cabecera de la página web.
 - b. Colocar el contenido principal de la página web.
 - c. Colocar los pies de página de la página web.

[Ir al solucionario](#)





Muy bien, lo invito a continuar con las mismas energías con el cual inició la materia. Seguidamente, vienen temas muy interesantes que le ayudarán a enriquecer sus conocimientos en el mundo del desarrollo web.

¡Hemos terminado la primera unidad!
Felicitaciones.



Resultados de aprendizaje 1, 4 y 5:

- Describe los problemas implicados en el desarrollo de una interfaz web.
- Explica por qué los problemas de accesibilidad son una consideración importante en el desarrollo de páginas Web.
- Construye aplicaciones web accesibles y seguras.

Para alcanzar los resultados de aprendizaje planteados, es esencial comprender y abordar tanto los problemas en el desarrollo de interfaces web como la importancia de la accesibilidad. Esto incluye describir los desafíos en el diseño de interfaces, explicar por qué los problemas de accesibilidad deben ser una prioridad en el desarrollo de páginas web, y construir aplicaciones web que sean accesibles y seguras. Se estudiará la programación en PHP, desde la configuración del entorno de desarrollo hasta el manejo de variables, constantes, estructuras de control, fusiones y sesiones, ofreciendo una base integral para desarrollar aplicaciones web efectivas.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 2



Unidad 2. Programación en PHP

Estimado alumno, continuamos con la segunda unidad, para adentrarnos en el desarrollo de las aplicaciones web, este es un tema fundamental para comprender las siguientes unidades. Le recomendamos tomarse un tiempo prudencial para revisarlo porque en esta unidad se explica todo el proceso del diseño web, pero tranquilo vamos paso a paso.

En la actualidad, la web está sufriendo grandes cambios, que han obligado a expertos en el tema a utilizar herramientas y técnicas basadas en la ingeniería del software, para poder garantizar el buen funcionamiento y administración de los sitios web.

Es por ello que se debe contar con ciertos atributos y características que, en conjunto, forman un concepto muy importante para alcanzar el éxito en cualquier organización, herramienta, y todo aquello que se pueda considerar como servicio. La calidad de las aplicaciones web que cuenten con atributos como, usabilidad, navegabilidad, seguridad, mantenimiento, entre otros, hace posible, por un lado, la eficiencia de la WebApp (Aplicaciones Webs) y por ende la satisfacción del usuario final.

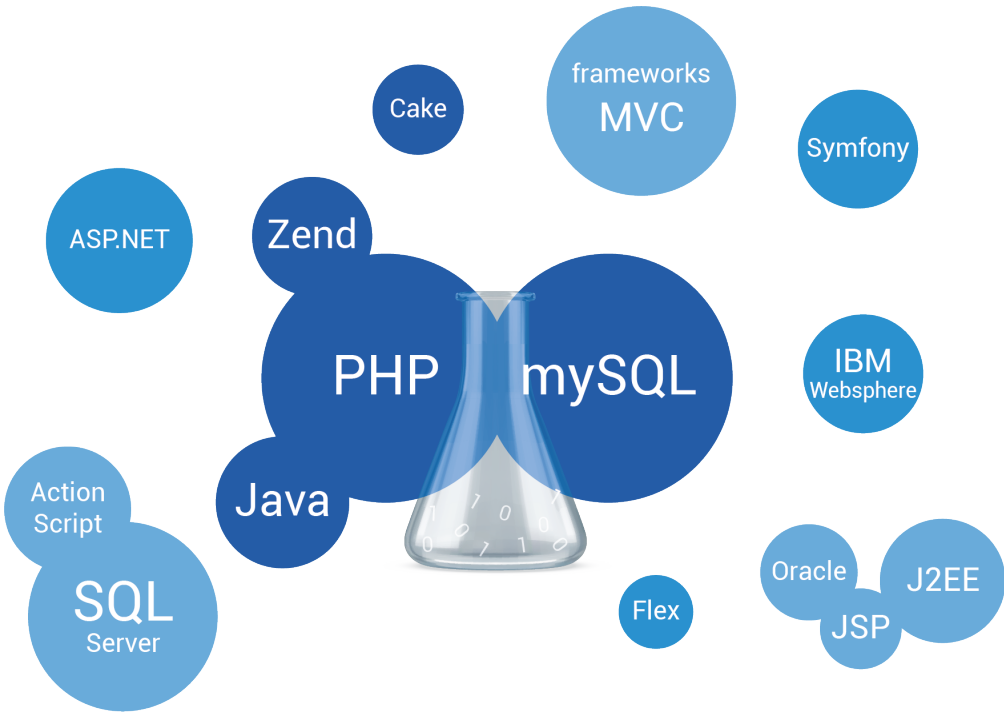
Para el desarrollo de una WebApp se puede seguir un proceso sistematizado en donde se debe planificar, programar, velar por la seguridad, controlar, es decir, la calidad no podrá ser agregada sin seguir un correcto proceso. Para finalizar el resultado de un proceso de calidad, podría arrojar recomendaciones para introducir mejoras, y la decisión final podría consistir en lanzar una nueva versión del sitio web o en modificar algunos atributos ausentes o pobremente diseñados. Cabe destacar que el **Desarrollo Web** hace una diferencia entre un *website* y una aplicación tradicional, ya que ella se dedica a la construcción de aplicaciones web. La principal característica que los distingue, entre las aplicaciones de sitios web, es que los sitios web son sitios en la web en donde se publica contenido generalmente estático, con un bajo nivel de interactividad con el usuario, mientras que las aplicaciones son lugares con alto contenido de interactividad y funcionalidades que bien podrían ser de un software convencional. Entonces,



la aplicación web más sencilla sería una que contenga formularios; subiendo de nivel de complejidad, encontramos los que realizan conexiones con bases de datos remotas, y administradores de contenidos, entre otras.

Figura 3

Elementos de desarrollo web



Nota. Tomado de Ingeniería Web [Ilustración], por idearí, 2013, [idearí](#). CC BY 4.0.

Entonces el Desarrollo Web son procesos, disciplinados y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la World Wide Web. En este sentido, se hace referencia a las técnicas y herramientas que se utilizan en el desarrollo de aplicaciones web complejas. Como se aprecia en la figura 3, los elementos del desarrollo web se pueden organizar en lenguajes de programación, bases de datos, frameworks (conjunto estandarizado de procesos), tecnologías, etc., estas son de gran dimensión, creando así WebApps de alta calidad, usabilidad, accesibilidad, etc.

2.1 Configuración del entorno de desarrollo

Ahora que ya tiene idea sobre el desarrollo de aplicaciones web en PHP, es necesario que acuda al **texto básico** en el capítulo 1 y 2, realice la lectura de los conceptos iniciales de lo que es el lenguaje, que es un servidor web, ventajas del servidor web y de PHP, base de datos MySQL.

¿Qué opina de la lectura?, ¿tiene dudas o inquietudes? ¡A *continuación las iremos resolviendo!*

Con la lectura realizada usted está en la capacidad de argumentar el proceso que se realiza al visitar una página web. Por favor, vaya al foro de actividades y realice esta actividad.

Configuración de un servidor web

Para poder construir sitios web, ya sean estos realizados con *software* libre o desarrollados manualmente, es pertinente levantar una plataforma sobre la que podamos trabajar; en nuestro caso puntual instale un servidor web con las aplicaciones que necesitamos. Empecemos esta unidad explicando cómo instalarlo:



Figura 4

Herramientas XAMPP



Nota. Tomado de ¿Qué es XAMPP? [Ilustración], por Apachefriends, s.f., [Apachefriends](#). CC BY 4.0.

La aplicación con la que vamos a trabajar es XAMPP. XAMPP es un auténtico software de instalación de Windows que nos permite instalar de manera fácil: **APACHE** (Servidor Web), **PHP** (Plataforma y Lenguaje de Programación Web Dinámica por Excelencia), y **MySQL** (Gestor de Base de Datos). La instalación es tan sencilla como descargar y descomprimir, y mejor aún es multiplataforma. En la figura 4 se muestra el logotipo de XAMPP y sus componentes. Para encontrar más referencias, le invito a visitar la página de [XAMPP](#).

La instalación la realizaremos en cuatro etapas:

1. Descargar el XAMPP.
2. Ejecutar el XAMPP para su instalación.
3. Levantar los servicios del Apache y MySQL.
4. Pruebas de los servicios.

Podemos concluir que el proceso de transformar nuestro computador en un servidor web es muy sencillo, siempre y cuando se utilice correctamente el *software* adecuado para el sistema operativo que utiliza.



Es tiempo de acudir al **texto básico**, a la sección 2.1 Instalación y configuración de Apache, y realizar la lectura y la práctica de la instalación del *software* necesario para transformar nuestro equipo local en un servidor web. No olvide revisar las directivas o parámetros de configuración del servidor Apache.

2.2 Primeros pasos con PHP

Para avanzar con el desarrollo web empezaremos viendo el lenguaje de programación PHP, el cual nos ayudará a crear las WebApps.

Para empezar este apartado es necesario saber las configuraciones necesarias que debe tener el lenguaje de programación PHP. Esto se lo realiza en el archivo `php.ini`, para ello y más detalle vaya al **texto básico** y realice una lectura minuciosa y práctica de lo que se puede configurar en este archivo.

Estimado alumno, espero que todo el proceso de instalación y configuración se haya hecho sin mayor problema y tengamos lista la plataforma sobre la cual vamos a trabajar. Si usted tiene dudas de si la aplicación está funcionando correctamente, le recomiendo que realice la siguiente actividad, esta nos ayudará a verificar correctamente si todo lo instalado está correcto y funcionando a la perfección.



Levante el servidor web en su máquina local y cree un archivo con extensión PHP y coloque el siguiente código:

```
<?php  
phpinfo();  
?>
```

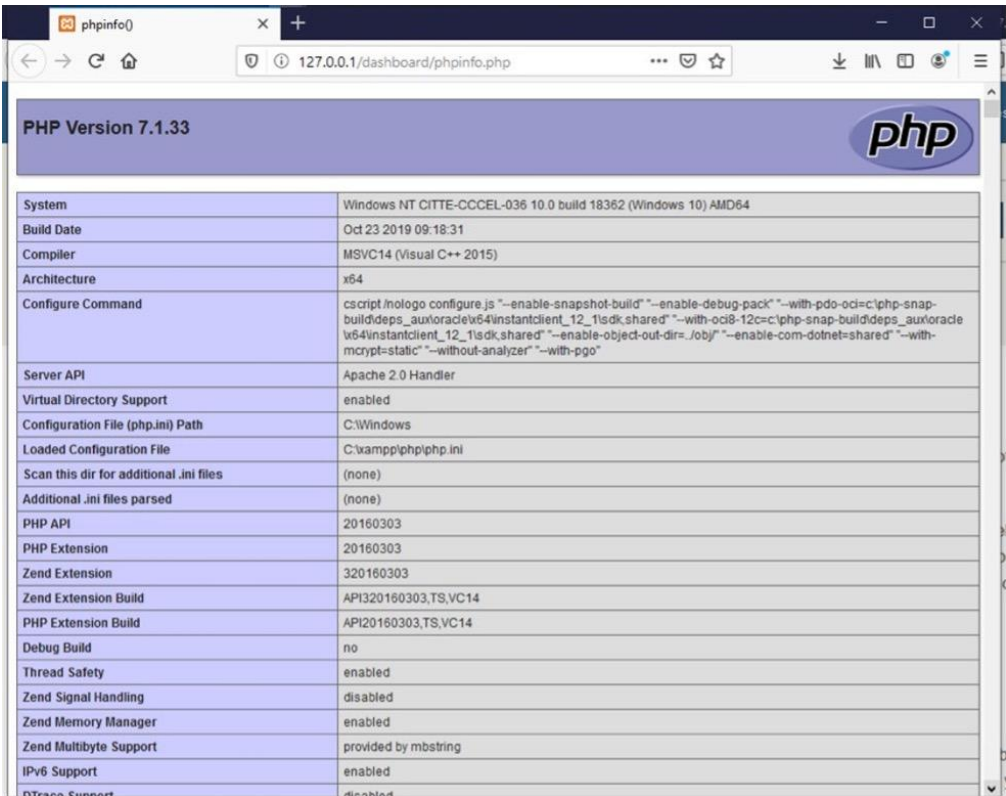
Y hágalo correr en su servidor



¿Qué resultados obtuvo? Si todo está bien, usted debe obtener un archivo similar al de la siguiente figura 5 donde se puede ver las versiones de las aplicaciones instaladas que tenemos en el servidor web y las configuraciones de estas:

Figura 5

Datos de configuración, módulos y componentes de PHP



System	Windows NT CITTE-CCCEL-036 10.0 build 18362 (Windows 10) AMD64
Build Date	Oct 23 2019 09:18:31
Compiler	MSVC14 (Visual C++ 2015)
Architecture	x64
Configure Command	cmd /c "cd /d %~dp0 & phpize --enable-snapshot-build --enable-debug-pack --with-pdo-oci=c:\php-snap-build\deps_aux\oracle\w64\instantclient_12_1\sdk\shared --with-oci8-12c=c:\php-snap-build\deps_aux\oracle\w64\instantclient_12_1\sdk\shared --enable-object-out-dir=.obj --enable-com-dotnet=shared --with-mcrypt=static --without-analyzer --with-pgsql"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20160303
PHP Extension	20160303
Zend Extension	320160303
Zend Extension Build	API320160303,TS,VC14
PHP Extension Build	API20160303,TS,VC14
Debug Build	no
Thread Safety	enabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
Dynamic Shared Object (DSO) Support	disabled

Nota. Ramírez, R., 2024.

Ahora sí, con la plataforma totalmente lista podemos empezar a desarrollar sitios web de una manera bastante sencilla, sigamos adelante con el siguiente tema, vamos a ver lo relacionado con la base de datos, cómo gestionar la información que se manipulará mediante nuestra aplicación web.

Conociendo el servidor MYSQL

Es tiempo de acudir nuevamente al **texto básico** y revisar el tema relacionado con la instalación y configuración de MYSQL, ubicado en el capítulo 2, en el apartado de “Instalación y configuración de MYSQL” (2.3).

¿Cómo le fue en la lectura? Muy interesante el artículo, ¿verdad? Ahora usted ya tiene conocimiento de bases de datos y de SQL.

Para reforzar la lectura, ingrese en su computador local al editor PHPMYADMIN (entorno de administración de la base de datos MYSQL), configure los usuarios de acceso y cree una base de datos de prueba. Si tiene algún problema en el desarrollo del ejercicio, no dude en realizar preguntas en el foro de la materia.

¡Continuemos optimistas con la siguiente temática!

2.3 Variables

Pasamos a ver un nuevo tema en el desarrollo web, dentro del lenguaje PHP, lo que corresponde a variables y constantes.

Para iniciar el estudio de este tema le recomendamos acudir al **texto básico** en el capítulo 4 “Variables y constantes” y realizar una lectura introductoria de lo que son las **variables**, acá encontraremos una explicación detallada de cómo se crean las variables, formas y presentación.

De acuerdo con lo expuesto anteriormente, le recomiendo desarrollar la siguiente actividad que le permitirá reforzar los conocimientos obtenidos en cuanto a la creación de variables en PHP.



Marque con un visto las variables que están creadas correctamente:



- \$var1=1; _____
- #var1=1 _____
- Var1=1; _____
- \$var="2"; _____

Nota: por favor, complete la actividad en un cuaderno o documento Word.

¿Cómo le pareció el tema de creación de las variables? ¡Muy sencillo, verdad! Es importante destacar que en PHP no se necesita convertir variables de enteros a caracteres, toda esta operación la realiza el compilador. Ahora que ya sabe cómo declarar variables en PHP, pasemos a ver:

Constantes

La primera y gran diferencia que existe entre las variables y las constantes es que las últimas van a tener un valor fijo, es decir, su valor no se va a poder modificar durante la ejecución de una página. Por el contrario, una misma variable puede tomar varios valores en una misma ejecución.



Ahora, como práctica, le invito a declarar 4 constantes entre valores numéricos y valores de texto.

Nota: por favor, complete la actividad en un cuaderno o documento Word.

Ahora que ya sabe cómo declarar variables y constantes en PHP, es hora de ver un nuevo tema:



2.4 Operadores

Para iniciar el estudio de este tema, le recomendamos acudir al **texto básico**, capítulo 5 “Operadores”, y realizar una lectura de los operadores: qué tipo de operadores existen y sus características.

¿Cómo le pareció la lectura?, espero que sea clara y entendible. Ahora ya sabe qué es un operador aritmético, operadores de comparación, operadores lógicos y operadores de unión de cadenas.

Está en capacidad de ejemplificar cada uno de ellos. Así pues, le recomiendo realizar la siguiente actividad:

Llene la siguiente tabla en su cuaderno o documento Word con un resumen de las características de cada uno de los operadores revisados.

Tabla 1.

Ejercicio de las características de los operadores

Operadores aritméticos	Operadores de comparación	Operadores lógicos	Operadores unión
+, -, *, /, %, ++, --, **			
Operaciones matemáticas			
Incrementos			

Nota. Ramírez, R., 2024.

Esta tabla le ayudará a resumir todos los tipos de operadores que existen y su forma de utilización y funcionamiento. Ahora que ya se tiene conocimiento de este tema, pasamos a un nuevo e interesante apartado: las estructuras de control.



2.5 Estructuras de control

Las estructuras de control son instrucciones utilizadas en programación para ejecutar una serie de acciones en las aplicaciones que creamos, en programación web nos ayudará de mucho para realizar algoritmos que impliquen condiciones de ejecución.

Es tiempo de acudir a nuestro **texto básico** y revisar las estructuras de control en el capítulo 6 “Estructuras de control”, mire detalladamente la estructura de los ejemplos para posterior implementación.

Como ya habrá revisado, existen diferentes tipos de estructuras de control, entre ellas:

- Instrucciones condicionales.
- Instrucciones de bucle.
- Otras instrucciones.



Una vez detectados los tipos de estructuras de control, realice un resumen de las características de cada uno de ellos, ejemplifíquelos y publíquelos en el foro de la clase. No olvide incluir ejemplos diferentes a los que nos plantea el **texto básico**, esto nos ayudará a reforzar los conocimientos obtenidos mediante la lectura.



Resultados de aprendizaje 1, 4 y 5:

- Describe los problemas implicados en el desarrollo de una interfaz web.
- Explica por qué los problemas de accesibilidad son una consideración importante en el desarrollo de páginas Web.
- Construye aplicaciones web accesibles y seguras.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 3

Unidad 2. Programación en PHP

2.6 Funciones

Estamos avanzando en los interesantes contenidos de nuestra materia de Desarrollo web, le invito a seguir con ese mismo ánimo y cumplir los objetivos trazados al inicio del periodo de estudio.

Una función es un bloque de código que introducimos en nuestra página y que puede ser utilizado a lo largo de todo nuestro código PHP. La principal ventaja de las funciones es que nos permiten ahorrar código.

Para continuar con los conocimientos de las funciones en PHP le sugiero acudir al **texto básico**, realizar una lectura atenta y la práctica de los ejemplos en el capítulo 7 sobre Funciones.

Es hora de poner en práctica los conocimientos obtenidos en este capítulo, para ello observe detenidamente el ejemplo planteado a continuación, en la figura 6, e identifique los errores que existen en el ejemplo de funciones.



Figura 6

Ejemplo de función

```
1  <?php
2      function calculoiva($monto){
3          $calculo=$monto * 0.12;
4          $total = $monto + $calculo;
5          print "El monto total a pagar es: ".$total;
6      }
7  ?>
8  <!DOCTYPE html>
9  <html>
10 <head>
11     <title>Calculo de Iva</title>
12 </head>
13 <body>
14 <?php
15     print calculoiva(4000),
16 ?>
17 </body>
18 </html>
```

Nota. Ramírez, R., 2020

Con base en el ejemplo de la figura anterior, liste los errores encontrados de funciones y explique el problema y su solución.

Nota. Por favor, completar la actividad en su cuaderno de trabajo o en un documento de Word.

Ahora ya tiene conocimiento de los errores relacionados con las funciones y la forma de resolverlos y el porqué del problema. Esto le ayudará, con futuros trabajos, a implementar la mejor solución ante un problema de esta magnitud.

Continuando con el contenido de las funciones, el lenguaje PHP tiene unas funciones predeterminadas, para el manejo de cadenas de caracteres, las cuales se detallan en el capítulo 8 del **texto básico**. En ellas se explica la forma de uso y los resultados que se obtienen de cada una de ellas.



En su computador implemente los ejemplos planteados en el **texto básico** del capítulo 8 y obtenga los resultados, compare con los del **texto básico**, si posee problemas en la ejecución de estos acuda al foro de la clase y expóngalos.

Ahora, es tiempo de ver un nuevo tema:

2.7 Sesiones

Las “cookies” son pequeños ficheros de texto que maneja nuestro servidor para conocer datos de los usuarios y emplearlos, si es necesario, en cada una de sus visitas. Las cookies se pueden usar para, entre otras cosas, recordar el nombre del usuario y sus preferencias.

También podemos definir a las sesiones como una serie de variables almacenadas en nuestro servidor que ofrecen información acerca de nuestros usuarios y que son diferentes para cada uno.

Para ver detalladamente el desarrollo de las sesiones, vaya al **texto básico** en la sección 10.2 y realice una lectura comprensiva del concepto de sesiones y las configuraciones que se tiene que realizar para su implementación.

Si ya termino la lectura Ud. ya está en capacidad de diferenciar lo que son las cookies y las sesiones, ambas nos ayudan para aplicar seguridad en nuestra aplicación web, ¿verdad?



Actividades de aprendizaje recomendadas

Reforcemos el aprendizaje resolviendo las siguientes actividades:

1. Argumente lo que realiza la función `session_start()`, y cree un ejemplo completo de sesiones y compártalo en la *wiki* de la materia.

Nota. Por favor, conteste la actividad en un cuaderno de apuntes o documento Word.



2. Antes de pasar al siguiente capítulo, le invito a desarrollar la siguiente autoevaluación para determinar el nivel de aprovechamiento del tema estudiado.



Autoevaluación 2

Conteste correctamente las siguientes preguntas:

1. Para el desarrollo web se utilizan metodologías:
 - a. Sistemáticas.
 - b. Multidisciplinarias.
 - c. Disciplinadas.
 - d. Cuantificables.
2. Para configurar nuestro ordenador en un servidor web necesita instalar:
 - a. Windows 10.
 - b. Apache.
 - c. PHP.
 - d. Photoshop.
3. Seleccione cuál de las alternativas es el gestor de base de datos que se instala con XAMPP:
 - a. Oracle.
 - b. SQL Server.
 - c. MYSQL.
4. Seleccione las etiquetas para ingresar código PHP:
 - a. `<php> ... </php>`.
 - b. `<?php ... ?>`.
 - c. `<php? ... ?>`.
5. Seleccione la o las formas de declarar una variable en PHP:
 - a. `$var1=1;`
 - b. `&var1=1;`
 - c. `%var1=1;`



6. Las constantes son:
- a. Variables aleatorias que cambian con el compilador del programa.
 - b. Variables que son generadas por el gestor de base de datos y no cambian.
 - c. Son variables que tienen un valor fijo en toda la ejecución del programa.
7. Seleccione el o los operadores de comparación:
- a. =.
 - b. -.
 - c. ==.
 - d. !=.
8. Seleccione el o los operadores de incremento:
- a. &&.
 - b. ++.
 - c. j-.
 - d. %.
9. Seleccione la estructura correcta de una instrucción condicional:
- a. For(\$i=0;\$i>10;\$i++).
 - b. If(\$i==10){ .. }.
 - c. If(\$i=0;\$i>10;\$i++){ .. }.
10. Una función es:
- a. Un ciclo repetitivo de código que se ejecuta al inicio de cada compilación del programa.
 - b. Es un bloque de código que se ejecuta cuando es invocada en cualquier parte del programa.
 - c. Una estructura de control para verificar una condición de una variable.

[Ir al solucionario](#)



Si no logró un buen resultado en la autoevaluación, no se preocupe, le recomiendo leer nuevamente el/los capítulos confusos y reforzar sus conocimientos. Y si aún tiene inquietudes, no dude en preguntar al profesor.



¡Suerte! Siga adelante.

¡Hemos terminado la segunda unidad! **Felicitaciones**



Resultados de aprendizaje 6 y 7:

- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.

Para alcanzar los resultados de aprendizaje planteados, es fundamental entender las diferencias entre la programación orientada a eventos y la programación de línea de comandos, destacando cómo cada enfoque gestiona la interacción y el flujo de datos. Además, es esencial diseñar, codificar, probar y depurar programas simples que reaccionen a eventos de usuario. Utilizar PHP con MySQL permitirá crear aplicaciones interactivas que respondan eficientemente a estas acciones, integrando el manejo de eventos con la gestión de datos.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 4

Unidad 3. PHP con MYSQL

Empezamos un nuevo capítulo en el mundo del desarrollo web. Este se refiere al manejo de los datos con MYSQL. De entre las muchas ventajas que se obtienen al crear una base de datos en una página web, una de las principales es que se consiguen páginas dinámicas.



Para usar y manejar la base de datos MYSQL, vamos a emplear una aplicación muy usual y extendida entre los usuarios. El **texto básico** usará el editor: **phpMyAdmin** y nosotros en la guía trabajaremos con las sentencias SQL.

3.1 Gestión de datos



Para entender claramente lo que es el editor *phpMyAdmin* es preciso acudir al **texto básico** y realizar una lectura comprensiva de la unidad 13, especialmente lo que es *phpMyAdmin*. Después crearemos una base de datos y una tabla. A continuación, iremos viendo cuáles son las sentencias SQL equivalentes para estos temas.

Para el desarrollo de esta unidad, usted ya revisó la unidad 13 y sabe cuál es el proceso de creación de las bases de datos.

Para la realización de este ejercicio tiene que levantar la consola de comando de MYSQL que se encuentra en el panel de control de XAMPP, o ejecutando el CMD de Windows, dentro de ella ejecutaremos la sentencia de creación de la base de datos.

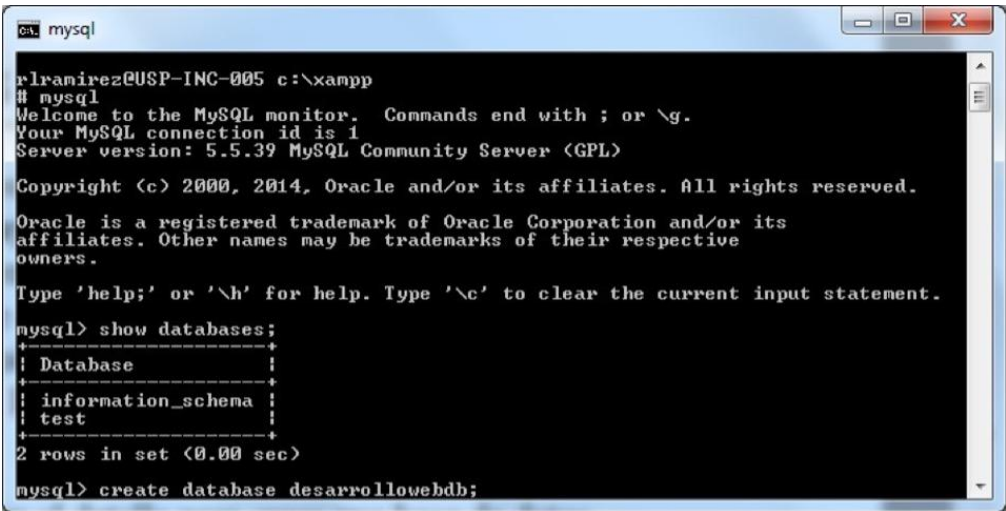
3.2 Creación de base de datos

La sentencia SQL es: **create database desarrollowebdb;** tal como lo muestra la figura 7.



Figura 7

Consola mysql, creación de base de datos



```
r1ramirez@USP-INC-005 c:\xampp
# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.5.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| test      |
+-----+
2 rows in set (0.00 sec)

mysql> create database desarrollowebdb;
```

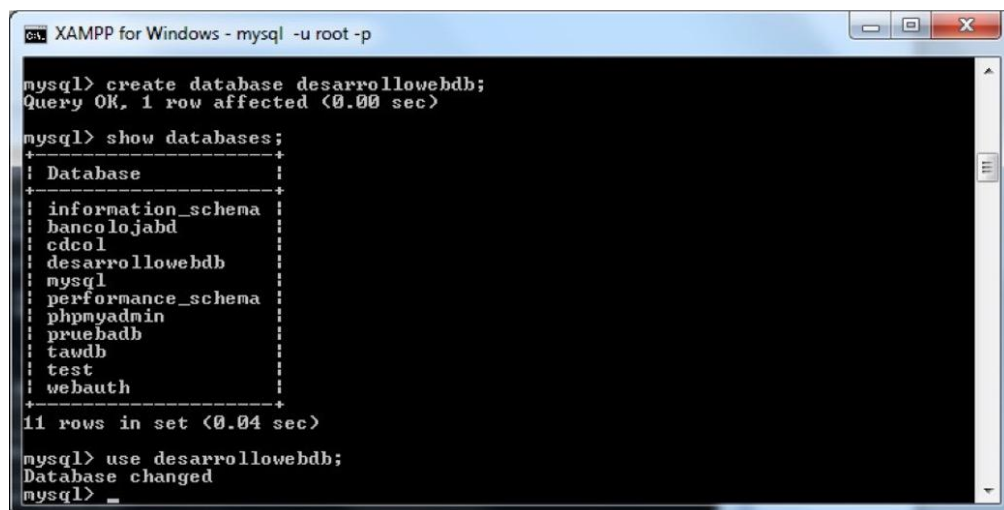
Nota. Ramírez, R., 2020

3.3 Listar las bases de datos de nuestro servidor

El comando para listar las bases de datos es: **show databases;** y para ingresar a la edición de la base de datos es: **use desarrollowebdb;** tal como se muestra en la siguiente figura 8.

Figura 8

Consola mysql, listado y uso de base de datos



```
mysql> create database desarrollowebdb;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bancojajab |
| cdcol |
| desarrollowebdb |
| mysql |
| performance_schema |
| phpmyadmin |
| pruebadb |
| tawdb |
| test |
| webauth |
+-----+
11 rows in set (0.04 sec)

mysql> use desarrollowebdb;
Database changed
mysql>
```

Nota. Ramírez, R., 2020

3.4 Creación de tablas en la base de datos

Para crear una base de datos en consola necesitamos estructurar la sentencia SQL de la siguiente forma:

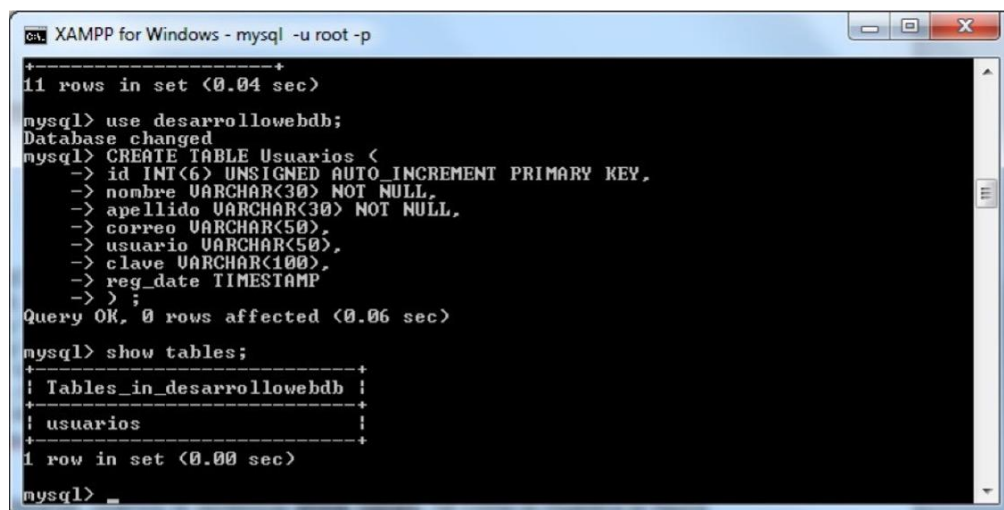
```
CREATE TABLE Usuarios(
  id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(30) NOT NULL,
  apellido VARCHAR(30) NOT NULL,
  correo VARCHAR(50),
  usuario VARCHAR(50),
  clave VARCHAR(100),
  reg_date TIMESTAMP
);
```

Y para poder listar las tablas creadas, usamos la sentencia **show tables**; tal como lo muestra la figura 9:



Figura 9

Consola mysql, creación de tablas



```
XAMPP for Windows - mysql -u root -p
+-----+
11 rows in set (0.04 sec)

mysql> use desarrollowebdb;
Database changed
mysql> CREATE TABLE Usuarios (
  -> id INT(6) UNSIGNED AUTO INCREMENT PRIMARY KEY,
  -> nombre VARCHAR(30) NOT NULL,
  -> apellido VARCHAR(30) NOT NULL,
  -> correo VARCHAR(50),
  -> usuario VARCHAR(50),
  -> clave VARCHAR(100),
  -> reg_date TIMESTAMP
  -> );
Query OK, 0 rows affected (0.06 sec)

mysql> show tables;
+-----+
| Tables_in_desarrollowebdb |
+-----+
| usuarios                  |
+-----+
1 row in set (0.00 sec)

mysql>
```

Nota. Ramírez, R., 2020

3.5 Manipulación de datos con SQL y phpMyAdmin

- *Insertar y listar datos en una tabla*

Otra forma de insertar los datos en la base de datos MySQL diferente a la que nos indica el **texto básico** es la siguiente sentencia SQL:

Insert into Usuarios values

(*'Ramiro','Ramirez','rlramirez@utpl.edu.ec','rlramirez','12345'*);

De la misma forma, la sentencia SQL para listar los registros de la base de datos es:

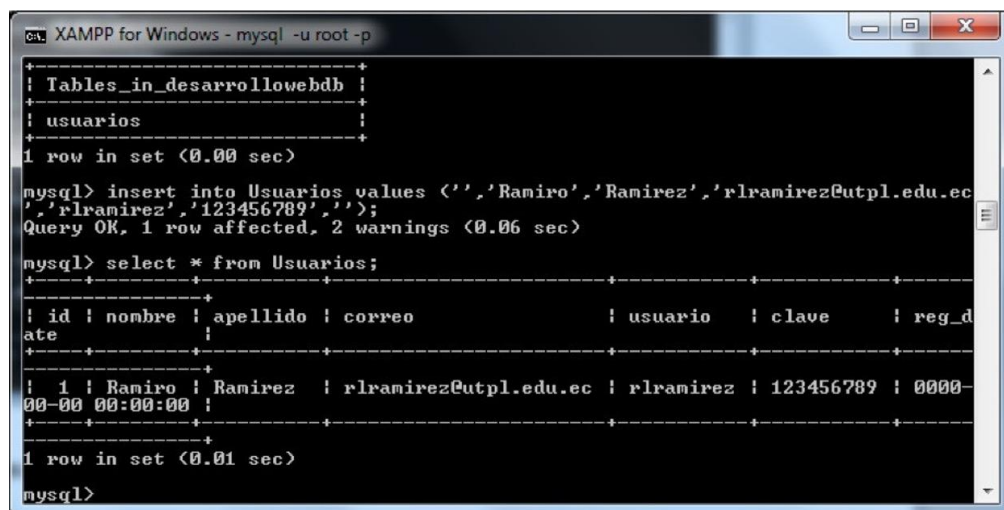
Select * from Usuarios;

Los resultados los tenemos en la siguiente figura 10.



Figura 10

Consola mysql, insertar datos en una tabla



```
XAMPP for Windows - mysql -u root -p
+-----+
| Tables_in_desarrollowebdb |
+-----+
| usuarios                   |
+-----+
1 row in set <0.00 sec>

mysql> insert into Usuarios values ('','Ramiro','Ramirez','rlramirez@utpl.edu.ec',
'rlramirez','123456789','');
Query OK, 1 row affected, 2 warnings (0.06 sec)

mysql> select * from Usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id | nombre | apellido | correo | usuario | clave | reg_date |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Raniro | Ramirez | rlramirez@utpl.edu.ec | rlramirez | 123456789 | 0000-00-00:00:00 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set <0.01 sec>

mysql>
```

Nota. Ramírez, R., 2020

- *Borrar datos en una tabla*

Otra forma de borrar los datos o registros en la base de datos MYSQL, aparte de la que nos indica el **texto básico**, es la siguiente sentencia SQL:

Delete from Usuarios where id=1;

Es importante colocar el condicional, ya que se corre el riesgo de eliminar todos los registros en la tabla.

- *Actualizar datos en una tabla*

Otra forma de actualizar los datos o registros en la base de datos MYSQL, aparte de la que nos indica el **texto básico**, es la siguiente sentencia SQL:

update Usuarios set nombre="Carlos", apellido="Jaramillo" where id=1;



Es importante colocar el condicional, ya que se corre el riesgo de actualizar todos los registros de la tabla.



Resultados de aprendizaje 6 y 7:

- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 5

Unidad 3. PHP con MYSQL

3.6 Conectar una base de datos con PHP



Para entender un poco más la conexión entre MySQL y PHP, le sugiero ir al **texto básico**, capítulo 14, y realizar una lectura comprensiva y práctica de los temas y ejemplos que el libro nos brinda, esto le ayudará a entender mejor el funcionamiento.

Es importante saber los parámetros requeridos para poder conectarse a la base de datos MYSQL con PHP, tal como lo muestra el ejemplo 14.1.1, son 4, estos parámetros dependerán de la configuración de su ordenador.

Tal como se ve en la figura 11, se muestra el resultado de implementar el ejercicio en su ordenador, ¿verdad?, el resultado debe ser idéntico a la figura 11.



Figura 11

Listado de datos en el navegador web



The screenshot shows a web browser window with the title 'Desarrollo Web | UTPL'. The address bar displays '127.0.0.1/desarrolloweb/ejemplo1.php'. Below the address bar, a message states: 'A continuación se muestra el resultado de seleccionar todos los registros de las tablas'. This is followed by a table with three rows of user data. The table has six columns: an index, a first name, a last name, an email address, a username, and a timestamp.

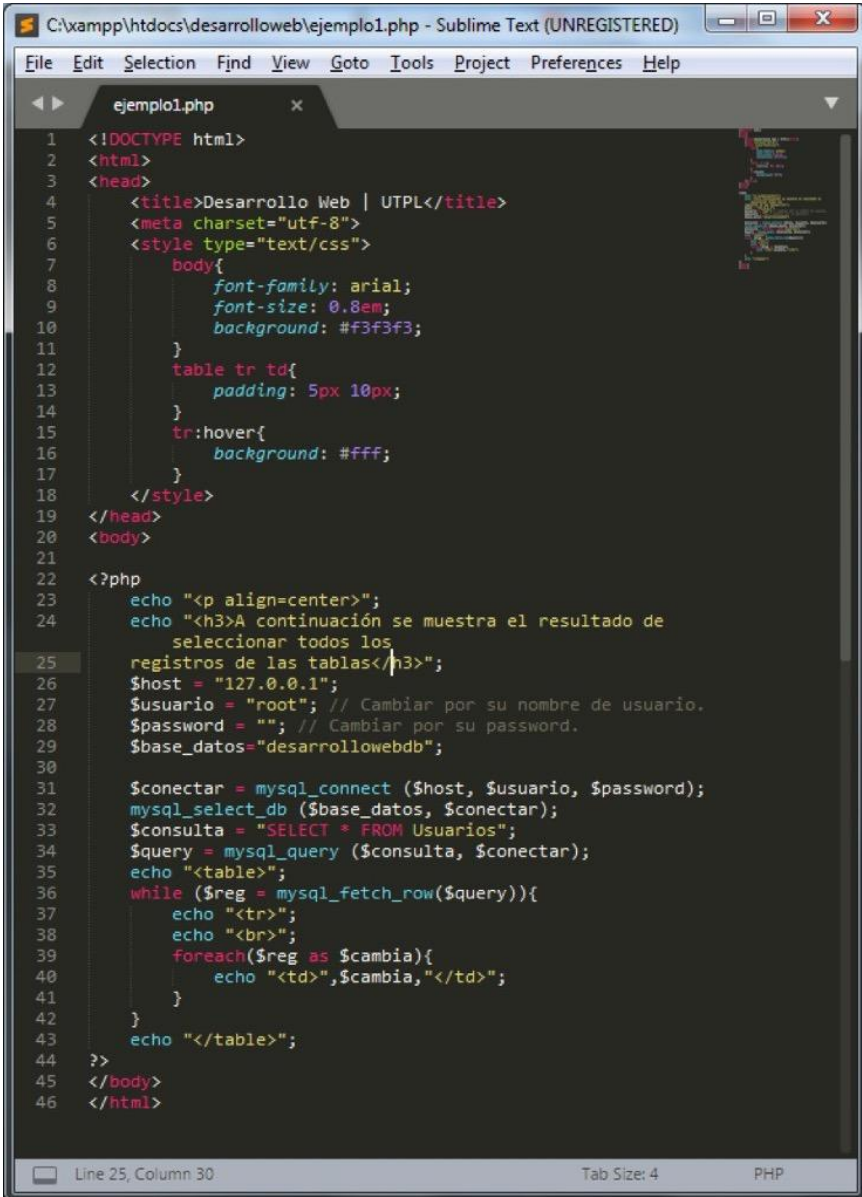
	Nombre	Apellido	Correo	Usuario	Fecha
1	Ramiro	Ramirez	r Ramirez@utpl.edu.ec	r Ramirez	123456789 0000-00-00 00:00:00
2	Carlos	Jaramillo	c Jaramillo@utpl.edu.ec	c Jaramillo	123121312 2017-10-11 00:00:00
3	Karina	Estrada	k Estrada@utpl.edu.ec	k Estrada	44223121312 2017-10-10 05:11:12

Nota. Ramírez, R., 2020

Para obtener este resultado del listado de los usuarios de la base de datos, le invito a codificarlo, tal como lo muestra la siguiente figura 12, donde se detalla la cadena de conexión, la ejecución de la consulta y la abstracción de los resultados mediante un ciclo repetitivo. Le deseo mucha suerte en la realización de este ejercicio. Estoy seguro de que le va a ir muy bien.

Figura 12

Código del ejemplo de la figura 11



```
C:\xampp\htdocs\desarrolloweb\ejemplo1.php - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

ejemplo1.php x
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Desarrollo Web | UTPL</title>
5     <meta charset="utf-8">
6     <style type="text/css">
7         body{
8             font-family: arial;
9             font-size: 0.8em;
10            background: #f3f3f3;
11        }
12        table tr td{
13            padding: 5px 10px;
14        }
15        tr:hover{
16            background: #fff;
17        }
18    </style>
19 </head>
20 <body>
21
22 <?php
23     echo "<p align=center>";
24     echo "<h3>A continuación se muestra el resultado de
25     seleccionar todos los
26     registros de las tablas</h3>";
27     $host = "127.0.0.1";
28     $usuario = "root"; // Cambiar por su nombre de usuario.
29     $password = ""; // Cambiar por su password.
30     $base_datos="desarrollowebdb";
31
32     $conectar = mysql_connect ($host, $usuario, $password);
33     mysql_select_db ($base_datos, $conectar);
34     $consulta = "SELECT * FROM Usuarios";
35     $query = mysql_query ($consulta, $conectar);
36     echo "<table>";
37     while ($reg = mysql_fetch_row($query)){
38         echo "<tr>";
39         echo "<br>";
40         foreach($reg as $cambia){
41             echo "<td>",$cambia,"</td>";
42         }
43         echo "</table>";
44     }
45 </body>
46 </html>
```

Line 25, Column 30 Tab Size: 4 PHP

Nota. Ramírez, R., 2020

3.7 Gestión de datos con PHP y MYSQL CRUD

Estamos avanzando en el estudio del lenguaje de desarrollo PHP, ahora gestionando los datos con la base de datos MYSQL, le invito a seguir trabajando en este emocionante mundo del desarrollo de WebApps. Pasamos ahora al desarrollo de un CRUD, con estas tecnologías, para ello acuda al **texto básico** y continuemos revisando el capítulo 14.

En la sección anterior ya vimos cómo conectarnos a una base de datos y cómo listar los registros. Ahora trabajaremos en el **desarrollo de un formulario en HTML** para registrar los usuarios de la tabla.

3.8 Formulario de registro

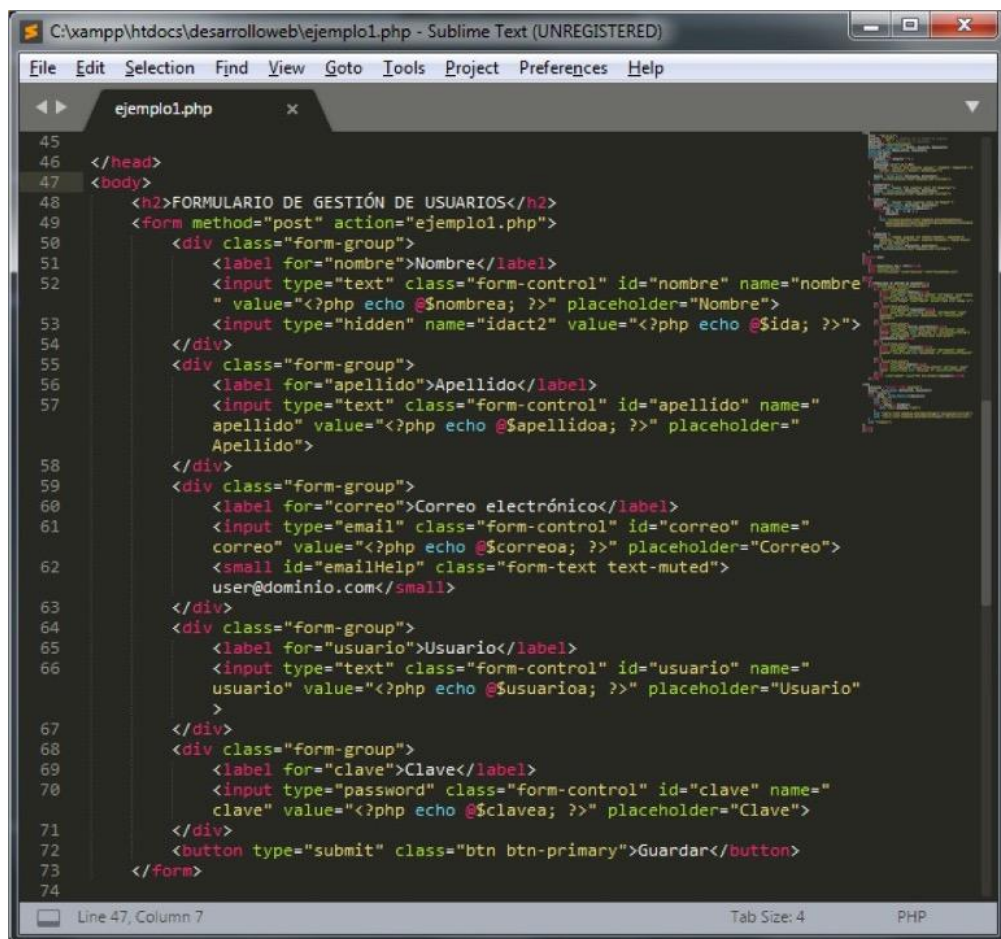
En la siguiente figura se muestra el código en HTML de un formulario con los campos que fueron creados en la tabla de usuarios. Estimado estudiante, acuda a su editor de código y ejemplifique esta actividad.

Iremos indicando paso a paso el flujo de este código de la figura 13.



Figura 13

Código de un formulario en HTML



```
45 </head>
46 <body>
47 <h2>FORMULARIO DE GESTIÓN DE USUARIOS</h2>
48 <form method="post" action="ejemplo1.php">
49 <div class="form-group">
50 <label for="nombre">Nombre</label>
51 <input type="text" class="form-control" id="nombre" name="nombre"
52   value="<?php echo @$nombrea; ?>" placeholder="Nombre">
53 <input type="hidden" name="idact2" value="<?php echo @$ida; ?>"
54 </div>
55 <div class="form-group">
56 <label for="apellido">Apellido</label>
57 <input type="text" class="form-control" id="apellido" name="
  apellido" value="<?php echo @$apellidoa; ?>" placeholder="
  Apellido">
58 </div>
59 <div class="form-group">
60 <label for="correo">Correo electrónico</label>
61 <input type="email" class="form-control" id="correo" name="
  correo" value="<?php echo @$correoa; ?>" placeholder="Correo">
62 <small id="emailHelp" class="form-text text-muted">
  user@dominio.com</small>
63 </div>
64 <div class="form-group">
65 <label for="usuario">Usuario</label>
66 <input type="text" class="form-control" id="usuario" name="
  usuario" value="<?php echo @$usuarioa; ?>" placeholder="Usuario"
67 >
68 </div>
69 <div class="form-group">
70 <label for="clave">Clave</label>
71 <input type="password" class="form-control" id="clave" name="
  clave" value="<?php echo @$clavea; ?>" placeholder="Clave">
72 </div>
73 <button type="submit" class="btn btn-primary">Guardar</button>
74 </form>
```

Nota. Ramírez, R., 2020

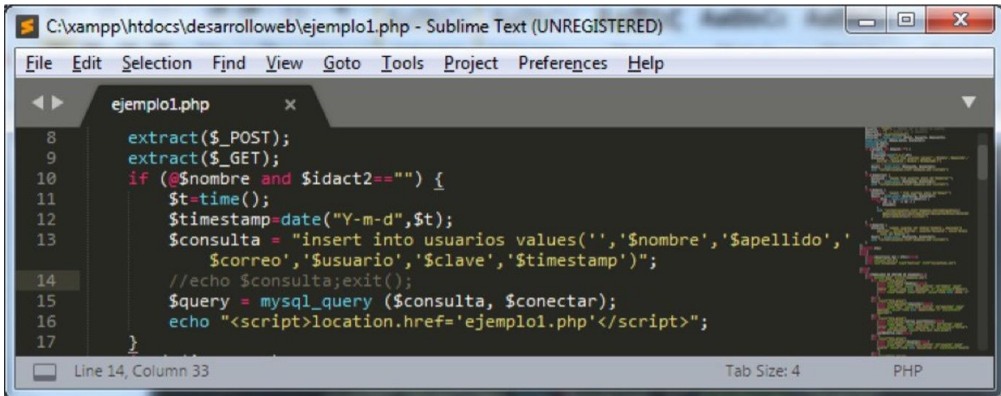
El formulario posee una propiedad que es el *action*, esta apunta a un archivo a donde van a ir los datos que ingresa el usuario en el formulario.

Recuerde que cada caja de texto tiene un nombre, este será el nombre de la variable en el cual nosotros haremos referencia para obtener el valor de cada variable. Por esta ocasión, el *value* de las cajas de texto no lo vamos a tomar en cuenta, pues esta nos servirá para las actualizaciones de los registros.

En la siguiente figura 14, se muestra el código en PHP, el cual recepta las variables y las guarda en la base de datos.

Figura 14

Código de insertar datos en la tabla



```
C:\xampp\htdocs\desarrolloweb\ejemplo1.php - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

ejemplo1.php
8  extract($_POST);
9  extract($_GET);
10 if (@$nombre and $idact2=="") {
11     $t=time();
12     $timestamp=date("Y-m-d",$t);
13     $consulta = "insert into usuarios values('','$nombre','$apellido','$
        $correo','$usuario','$clave','$timestamp')";
14     //echo $consulta;exit();
15     $query = mysql_query ($consulta, $conectar);
16     echo "<script>location.href='ejemplo1.php'</script>";
17 }
```

Nota. Ramírez, R., 2020

La línea 8, **extract(\$_POST)**, es el comando que obtiene los datos del formulario, ya que fueron enviados por el método POST en el navegador. Si no se coloca esta línea de código, no podremos ver el contenido de las variables del formulario. Luego armamos el comando SQL para finalmente, en la línea 15, ejecutar este comando e insertar los datos.

Estimados alumnos, espero que sea de comprensión todo lo expuesto hasta ahora. Si existe alguna duda, pueden acudir al foro de actividades y plantear la inquietud. Estaré gustoso de responder sus preguntas.

Resultados de aprendizaje 6 y 7:

- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 6

Unidad 3. PHP con MYSQL

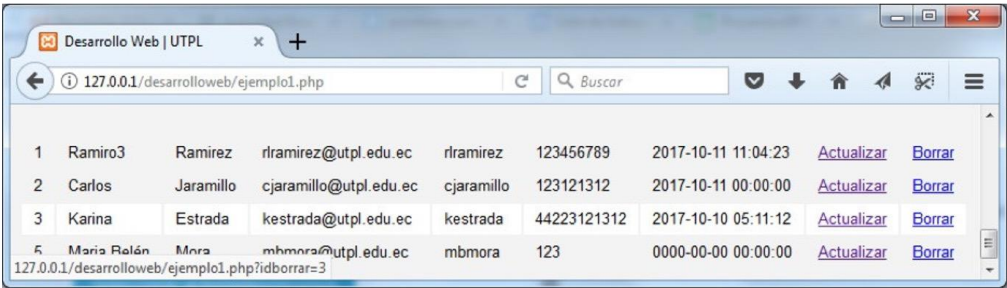
3.9 Borrar los registros

Para realizar la eliminación de los registros de la base de datos acudimos y reutilizamos la parte de listar los usuarios, los cuales lo vimos en la sección anterior, en ello realizamos la modificación al código y el resultado lo tenemos en la figura 15 en ella listamos los registros y aumentamos dos *links* más que es de borrar y de actualizar.



Figura 15

Listado de datos de la tabla usuario con opciones de CRUD



A screenshot of a web browser window titled 'Desarrollo Web | UTPL'. The address bar shows '127.0.0.1/desarrolloweb/ejemplo1.php'. The page displays a table with 5 rows of user data. Each row has two links: 'Actualizar' and 'Borrar'.

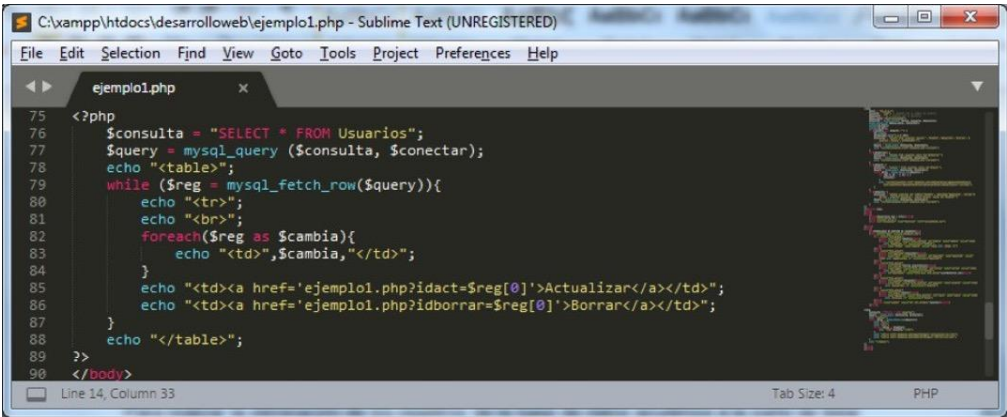
1	Ramiro3	Ramirez	rlramirez@utpl.edu.ec	rlramirez	123456789	2017-10-11 11:04:23	Actualizar	Borrar
2	Carlos	Jaramillo	cjaramillo@utpl.edu.ec	cjaramillo	123121312	2017-10-11 00:00:00	Actualizar	Borrar
3	Karina	Estrada	kestrada@utpl.edu.ec	kestrada	44223121312	2017-10-10 05:11:12	Actualizar	Borrar
5	Maria Belán	Mora	mhmora@utpl.edu.ec	mhmora	123	0000-00-00 00:00:00	Actualizar	Borrar

Nota. Ramírez, R., 2020

Para realizar esta actividad creamos una consulta **select** conjuntamente con los dos enlaces para borrar y actualizar tal como lo muestra la figura 16.

Figura 16

Código del listado de los datos y CRUD



A screenshot of a code editor window titled 'C:\xampp\htdocs\desarrolloweb\ejemplo1.php - Sublime Text (UNREGISTERED)'. The code is in PHP and shows a query to fetch all users from a table named 'Usuarios'. It then iterates through the results, displaying each user's information and two links: 'Actualizar' and 'Borrar'.

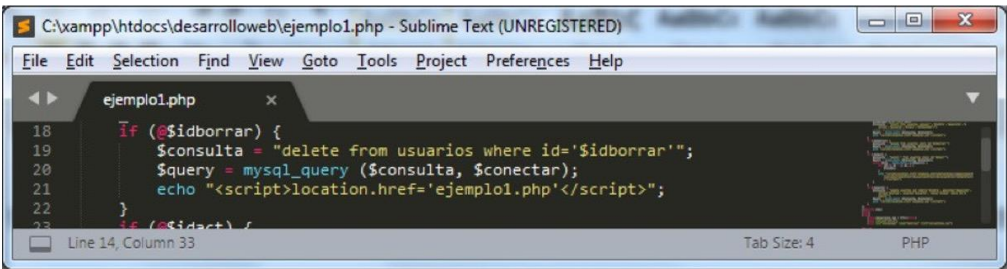
```
<?php
75 $consulta = "SELECT * FROM Usuarios";
76 $query = mysql_query ($consulta, $conectar);
77 echo "<table>";
78 while ($reg = mysql_fetch_row($query)){
79     echo "<tr>";
80     echo "<br>";
81     foreach($reg as $cambia){
82         echo "<td>",$cambia,"</td>";
83     }
84     echo "<td><a href='ejemplo1.php?idact=$reg[0]'">Actualizar</a></td>";
85     echo "<td><a href='ejemplo1.php?idborrar=$reg[0]'">Borrar</a></td>";
86 }
87 echo "</table>";
88 ?>
89 </body>
```

Nota. Ramírez, R., 2020

Algo que destacar en el código es que en la línea 85 y 86 se crean *links* y se envía el, id de cada registro para poder eliminar y actualizar el registro correcto, esto con la línea de código **\$reg[0]**, que es asignado a la variable **idborrar**, el cual nos servirá para obtenerlo por el método GET y borrar el registro, tal como se ejemplifica en la figura 17.

Figura 17

Código para eliminar datos de la tabla



```
18 if (@$idborrar) {
19     $consulta = "delete from usuarios where id='$idborrar'";
20     $query = mysql_query ($consulta, $conectar);
21     echo "<script>location.href='ejemplo1.php'</script>";
22 }
23 if (@$idact) {
```

Nota. Ramírez, R., 2020

Esta es la sentencia SQL que, con ayuda del lenguaje PHP, elimina el registro que se selecciona.

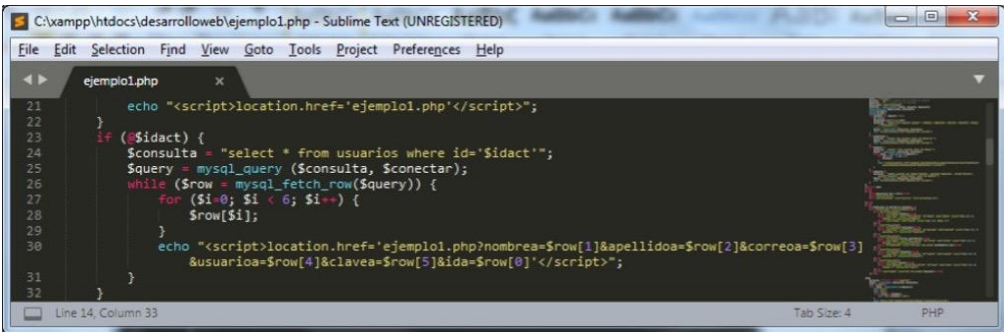
3.10 Actualizar los registros

Continuamos revisando cómo actualizar los registros de nuestra base de datos, para ello siguiendo el mismo proceso de listado de los usuarios en la figura 18 activamos el *link* de actualizar, este enviará el, id o llave primaria para actualizar el registro, pero antes de actualizar el registro primero se tiene que obtener todos los datos del usuario seleccionado para mostrarlos en el formulario, tal como lo muestra la figura con el código, primero obtenemos los datos y lo enviamos por el **método GET** al formulario, esto lo puede ver en la línea 30.



Figura 18

Obtener los datos de un registro



```
C:\xampp\htdocs\desarrollob\ejemplo1.php - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

ejemplo1.php
21 echo "<script>location.href='ejemplo1.php'</script>";
22
23 if (@$idact) {
24     $consulta = "select * from usuarios where id='$idact'";
25     $query = mysql_query($consulta, $conectar);
26     while ($row = mysql_fetch_row($query)) {
27         for ($i=0; $i < 6; $i++) {
28             $row[$i];
29         }
30         echo "<script>location.href='ejemplo1.php?nombrea=$row[1]&apellidoa=$row[2]&correoa=$row[3]
31             &usuarioa=$row[4]&clavea=$row[5]&ida=$row[0]'</script>";
32     }
33 }
```

Nota. Ramírez, R., 2020

Posteriormente en el formulario llenamos estas variables haciendo uso del **value="<?php echo @\$ida; ?>"** de las cajas de texto.



Para complementar esta temática le invito a investigar cuál es el objetivo de colocar la @ al inicio de la variable, comparta esta investigación en la *wiki* de la materia.

Nota: por favor, complete la actividad en un documento de Word o cuaderno de trabajo.

Como resultado tenemos el formulario actualizado tal como lo muestra la figura 19.



Figura 19

Formulario con los datos a modificar de un registro

Desarrollo Web | UTPL

127.0.0.1/desarrolloweb/ejemplo1.php?nombrea=Carlos&apellid

FORMULARIO DE GESTIÓN DE USUARIOS

Nombre
Carlos

Apellido
Jaramillo

Correo electrónico
cjaramillo@utpl.edu.ec user@dominio.com

Usuario
cjaramillo

Clave

Guardar

1	Ramiro3	Ramirez	rlramirez@utpl.edu.ec	rlramirez	123456789	2017-10-11 11:04:23	Actualizar	Borrar
2	Carlos	Jaramillo	cjaramillo@utpl.edu.ec	cjaramillo	123121312	2017-10-11 00:00:00	Actualizar	Borrar
3	Karina	Estrada	kestrada@utpl.edu.ec	kestrada	44223121312	2017-10-10 05:11:12	Actualizar	Borrar
5	Maria Belén	Mora	mbmora@utpl.edu.ec	mbmora	123	0000-00-00 00:00:00	Actualizar	Borrar

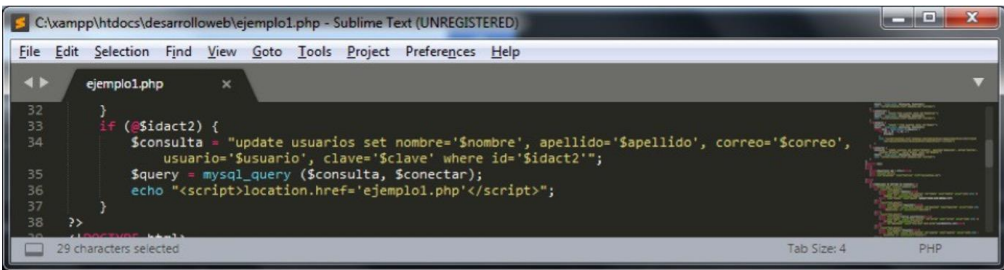
127.0.0.1/desarrolloweb/ejemplo1.php?idact=2

Nota. Ramírez, R., 2020

Finalmente, cuando el cliente presiona el botón, guardar el código PHP evalúa el contenido de las variables y si la variable **\$idact2** tiene algún valor (el, id del registro a actualizar) este ingresa al condicional y ejecuta la sentencia SQL Update tal como lo muestra la siguiente figura 20.

Figura 20

Código para actualizar los datos de una tabla



```
C:\xampp\htdocs\desarrolloweb\ejemplo1.php - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

ejemplo1.php
32 }
33 if (@$idact2) {
34     $consulta = "update usuarios set nombre='$nombre', apellido='$apellido', correo='$correo',
35     usuario='$usuario', clave='$clave' where id='$idact2'";
36     $query = mysql_query($consulta, $conectar);
37     echo "<script>location.href='ejemplo1.php'</script>";
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }
```

Nota. Ramírez, R., 2020


Todo el código completo se encuentra en el repositorio de GitHub. Diríjase al archivo denominado [CRUD en PHP](#) donde usted podrá tener acceso al mismo para que pueda editarlo.



Actividad de aprendizaje recomendada

Continuemos con el aprendizaje mediante su participación en la actividad que se describe a continuación:

Estimado estudiante, le invito a medir sus conocimientos de esta unidad con la siguiente autoevaluación. Esta le ayudará a valorar cuál es su nivel de conocimientos obtenidos durante el transcurso de este periodo transcurrido. En caso de no obtener resultados satisfactorios, le sugiero revisar nuevamente los contenidos en el cual tiene problemas.



Autoevaluación 3

Conteste correctamente las siguientes preguntas de la unidad 3:

1. El editor de base de datos MYSQL que se instala con XAMPP es:
 - a. xamppMyAdmin.
 - b. mysqlMyAdmin.



- c. phpMyAdmin.
 - d. mysqlAdmin.
2. Seleccione la sentencia correcta para crear una base de datos en MYSQL:
- a. Create table proyecto1db;
 - b. Create database proyecto1db;
 - c. Create databases proyecto1db;
3. Seleccione la sentencia correcta para listar las bases de datos de nuestro servidor:
- a. Select * from databases;
 - b. Select show databases;
 - c. Show databases;
4. La propiedad para crear una llave primaria de una tabla es:
- a. Auto_incremente.
 - b. Primary key.
 - c. Unsigned auto_increment.
5. La estructura correcta para insertar en una base de datos es:
- a. Insert into tabla values(",");
 - b. Insert into tabla (campos) values(",");
 - c. Insert tabla values(",");
6. La estructura correcta para listar los datos de una tabla es:
- a. Select * from tabla;
 - b. Select * tabla;
 - c. Select nombre, apellido from tabla;
7. La estructura correcta para eliminar los datos de una tabla es:
- a. Delete * from usuarios;
 - b. Delete from usuarios where id=1;
 - c. Delete usuarios where id=1;
8. La estructura correcta para actualizar los datos de un registro es:
- a. Update usuarios values (campos);
 - b. Update usuarios set (capos) where id=2;
 - c. Update usuarios set nombre='nuevo nombre' where id=1;
9. La sentencia para seleccionar una base de datos en PHP es:
- a. Mysqlconnect();



- b. Mysqlquery();
- c. Mysqlselectdb();

10. Para ejecutar una consulta en PHP de MYSQL es:

- a. Mysqlconect ();
- b. Mysqlquery();
- c. Mysqlfield();

[Ir al solucionario](#)

Si no logró un buen resultado en la autoevaluación, no se preocupe, le recomiendo leer nuevamente el/los capítulos confusos y reforzar sus conocimientos. Y si aún tiene inquietudes, no dude en preguntar al profesor.



Le sigo invitando a que continuemos con el mismo ritmo de trabajo con que empezamos, ya que vamos a comenzar un nuevo bimestre.



Resultados de aprendizaje 1 a 7:

- Describe los problemas implicados en el desarrollo de una interfaz web.
- Resume la necesidad y las cuestiones implicadas en la ejecución y la integración de una página web.
- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Explica por qué los problemas de accesibilidad son una consideración importante en el desarrollo de páginas Web.
- Construye aplicaciones web accesibles y seguras.
- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 7

Actividades finales del bimestre

En esta semana nos vamos a dedicar a revisar las temáticas estudiadas en las semanas anteriores para que se puedan ir afianzando los temas previos a la evaluación presencial.

Además de eso, tenemos algunas actividades calificadas que es importante que usted realice como parte del proceso de preparación, como es el *chat* académico que se ha programado para esta semana, además de la actividad suplementaria para el caso de los alumnos que por algún motivo no puedan realizar la actividad síncrona.



Resultados de aprendizaje 1 a 7:

- Describe los problemas implicados en el desarrollo de una interfaz web.
- Resume la necesidad y las cuestiones implicadas en la ejecución y la integración de una página web.
- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Explica por qué los problemas de accesibilidad son una consideración importante en el desarrollo de páginas Web.
- Construye aplicaciones web accesibles y seguras.
- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 8

Actividades finales del bimestre

En la última semana del primer bimestre. Así que nos vamos a enfocar en la revisión del contenido, previo a la evaluación presencial. Se recomienda aprovechar esta semana revisando nuevamente, cada uno de los temas y si surgieran dudas, por favor, contactarse con el docente-tutor para resolverlas.





Segundo bimestre

Resultados de aprendizaje 6, 8, 9 y 10:

- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Desarrolla código que responde a las condiciones de excepción planteadas durante la ejecución.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Identifica los sitios web que están utilizando gráficos de páginas web como iconos web.

Para alcanzar los resultados de aprendizaje planteados, es crucial diferenciar entre la programación orientada a eventos y la programación de línea de comandos, y desarrollar código que maneje de manera efectiva las excepciones que surgen durante la ejecución. Además, es importante entender cómo asegurar cookies y los riesgos para la privacidad que pueden derivarse de su mal uso. También se debe identificar los sitios web que usan gráficos de páginas web como iconos. La programación orientada a objetos ofrece herramientas clave para gestionar eventos y excepciones, incluyendo métodos para manejar eventos, decidir cuándo lanzar excepciones frente a errores, y gestionar adecuadamente las excepciones.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.





Semana 9

Unidad 4. Programación orientada a objetos

Estimado alumno, empezamos el cuarto capítulo de la asignatura abordando temas fundamentales que se requiere entender en programación orientada a objetos en Desarrollo Web.



Para iniciar el estudio del tema de POO le recomendamos leer el siguiente [recurso digital](#).

¿Qué le pareció la lectura?, ¿tiene algunas dudas? No se preocupe, las vamos a ir resolviendo.

4.1 Métodos y funciones para manejar eventos

Para el manejo de métodos y funciones trabajaremos en el desarrollo de una clase que realice toda la gestión de la base de datos con MySQL, ejecute las consultas SQL para ser llamadas desde el objeto y así consumir sus recursos.

La Programación Orientada a Objetos es la creación de clases y definición de objetos, esto con el fin de reutilizar código en toda nuestra WebApp.

• Objeto

Es una entidad independiente con sus propios datos y programación. Las ventanas, menús, carpetas de archivos pueden ser identificados como objetos; el motor de un auto también es considerado un objeto, en este caso, sus datos (atributos) describen sus características físicas y su programación (métodos) describen el funcionamiento interno y su interrelación con otras partes del automóvil (también objetos).



Con los objetos y las clases ahora es posible agrupar información y acciones de una forma muy fácil. Lo único que tenemos que pensar es en un nombre genérico para nuestra clase, puesto que las clases y los objetos son conceptos diferentes:

- **Clase**

Es un término genérico que, como su nombre lo indica, representa una clase, tipo o conjunto.

Tal como se ilustra en la figura 21, en la línea 2 se muestra la creación de una clase *DB_mysql*, seguidamente con las variables o propiedades de las clases.

Las propiedades pueden ser públicas o privadas:

- a. **public** quiere decir que la propiedad se puede acceder y modificar fuera de la clase.
- b. **private** indica que no es posible ni leer ni reescribir la propiedad fuera de la clase y esto nos trae un nuevo concepto muy importante llamado encapsulamiento.

Por otro lado, todas las **funciones** o acciones que van a consumir o interactuar con dichos datos, tan solo colocando dichas funciones dentro de la clase como se listan a continuación:

- Conectar.
- Consulta.
- Numcampos.
- Verconsulta.
- Etc.



Figura 21

Código de una clase en php

```
1 <?php
2 class DB_mysql
3 {
4     /*Variables de conexión*/
5     var $BaseDatos;
6     var $Servidor;
7     var $Usuario;
8     var $Clave;
9
10    /*Variable de control de la conexión y consulta*/
11    var $Conexion_ID=0;
12    var $Consulta_ID=0;
13
14    var $Errno = 0;
15    var $Error = 0;
16
17    function conectar($db, $host, $user, $pass){
18        if($db != "") $this->BaseDatos=$db;
19        if($host != "") $this->Servidor=$host;
20        if($user != "") $this->Usuario=$user;
21        //if($pass != "") $this->Clave=$pass;
22
23        $this->Conexion_ID = mysql_connect($this->Servidor,$this->Usuario,$this->Clave);
24        if(!$this->Conexion_ID){
25            $this->Error = "La conexión con el servidor a fallado..";
26            return 0;
27        }
28
29        if(!mysql_select_db($this->BaseDatos,$this->Conexion_ID)){
30            $this->Error = "No se pudo conectar a la db" . $this->BaseDatos;
31            return 0;
32        }
33
34        return $this->Conexion_ID;
35    }
36
37    /*Funcion de ejecución de la sentencia sql*/
38    function consulta($sql){
39        if($sql==""){
40            $this->Error="No hay ninguna sentencia sql";
41            return 0;
42        }
43        //ejecutamos la consulta
44        $this->Consulta_ID=mysql_query($sql,$this->Conexion_ID);
45        return $this->Consulta_ID;
46    }
47
48    /*Devuelve el numero de campos de la consulta*/
49    function numcampos(){
50        return mysql_num_fields($this->Consulta_ID);
51    }
52 }
```



Nota. Ramírez, R., 2020

Para trabajar con una variable en PHP utilizaremos el signo de dólar \$, para interactuar con una propiedad dentro de la clase usamos el signo \$, seguido de la palabra ***“this”*** y una flecha: ***\$this->***

Resultados de aprendizaje 6, 8, 9 y 10:

- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Desarrolla código que responde a las condiciones de excepción planteadas durante la ejecución.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Identifica los sitios web que están utilizando gráficos de páginas web como iconos web.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 10

Unidad 4. Programación orientada a objetos

4.2 Propagación de objetos, eventos

En la **clase** representa y engloba los métodos o funciones que realizan operaciones de gestión de la base de datos. Un **objeto** es la representación específica de una clase y la forma de cómo accedemos a los métodos. Cuando tenemos una clase como la que tenemos en la figura 22, podemos declarar uno o más objetos de dicha clase de esta forma:



Figura 22

Consumir las funciones de la clase

```
1  <?php
2  /*Llamamos a la clase y al archivo de configuración*/
3  include("dll/config.php");
4  include("dll/clase_mysql.php");
5
6  //creación de un objeto
7  $miconexion = new DB_mysql;
8  $miconexion->conectar($dbname,$dbhost,$dbuser,$dbpass);
9
10 //consumo de las duncionesde las clases con el objeto
11 $sql="select * from usuarios";
12 $miconexion->consulta($sql);
13 $miconexion->verconsulta();
14
15  >>
```

Nota. Ramírez, R., 2020

Para esto se utiliza el operador “new”, seguido del nombre de la clase y luego los argumentos que acepta el *constructor de la clase*.

4.3 Manejo de excepciones

Excepción en PHP

Una excepción es un evento que ocurre durante la ejecución de un programa y requiere de la ejecución controlada de un bloque de código fuera del flujo normal de ejecución.

Esta técnica es una herramienta muy potente a la hora de realizar una manipulación de una situación ante cualquier posible imprevisto. Lo primero que hay que comprender es que una excepción no es un error. Es una situación fuera de lo común que se experimenta en un bloque de código y que este no es capaz de manejar.

¿Cuándo lanzar una excepción y cuándo un error?

En general, si estamos trabajando en una clase de alto nivel de una aplicación y muy acoplada con la misma, la mejor opción puede ser informar del error, mediante la gestión de errores de la propia aplicación, y tratar de controlarlo en este mismo lugar. Si, por el contrario, estamos trabajando en una clase de bajo nivel, desacoplada de la aplicación en la que estás trabajando y susceptible de ser reutilizada, lo mejor es lanzar una excepción y dejar que las clases de arriba tratan de capturarlo, para informar al usuario.

Cuando una excepción es lanzada, PHP tratará de encontrar un bloque **catch{}**, capaz de capturarla, si no encuentra ninguno, entonces, se interrumpirá la ejecución y se emitirá un error fatal de PHP con el mensaje *"Uncaught Exception"*. Para capturar una excepción, el código debe encontrarse dentro de un bloque **try{}** y debe tener al menos un bloque **catch{}** . Veamos un ejemplo.



```
<?php
function foo(){
    try {
        throw new Exception(' Upps !!');
    } catch (Exception $e) {
        echo 'ErrorException' . $e->getMessage();
    }
}
foo();
```

Métodos de una excepción

Una excepción es un objeto, al fin y al cabo, a continuación, listamos los métodos que tiene disponibles en el lenguaje PHP:

- `getMessage()` devuelve el mensaje de la excepción.
- `getCode()` devuelve el código del error.
- `getFile()` devuelve el nombre del fichero que lanzó la excepción.



- `getLine()` devuelve el la línea del fichero que lanzó la excepción.
- `getTrace()` devuelve la pila de ejecución en formato de *array*.
- `getTraceAsString()` devuelve la pila de ejecución en formato de *string*.



Actividad de aprendizaje recomendada

Es hora de reforzar los conocimientos adquiridos resolviendo la siguiente actividad:

Estimado estudiante, le invito a medir sus conocimientos de esta unidad con la siguiente autoevaluación. Esta le ayudará a valorar cuál es el nivel de conocimientos obtenidos durante el transcurso de este período transcurrido. En caso de no obtener resultados satisfactorios, le sugiero revisar nuevamente los contenidos en el cual tiene problemas.



Autoevaluación 4

Conteste las siguientes preguntas de la unidad 4:

1. La Programación Orientada a Objetos es:
 - a. La Programación Orientada a Objetos es la creación de clases y definimos objetos, esto con el fin de reutilizar código en toda nuestra WebApp.
 - b. La Programación Orientada a Objetos con funciones que se crean para ser invocadas desde nuestra WebApp.
 - c. La Programación Orientada a Objetos son instrucciones que crean para la gestión de la base de datos.
2. Un objeto es:
 - a. Es un estilo de crear una función.
 - b. Es una variable con características de valores numéricos.
 - c. Es una entidad independiente con sus propios datos y programación.



3. Una clase es:
 - a. Es un conjunto de objetos.
 - b. Es un término genérico que, como su nombre lo indica, representa una clase, tipo o conjunto.
 - c. Es una forma de gestión de base de datos.
4. En una clase las propiedades pueden ser:
 - a. Public.
 - b. Unsigned.
 - c. Private.
5. Para acceder a las propiedades internas de las clases usamos:
 - a. This->.
 - b. \$this->.
 - c. \$\$this->.
6. Para utilizar los métodos de una clase utilizamos el siguiente código:
 - a. \$miconexion = new clase_mysql;
 - b. \$miconexion->conectar
 - c. \$miconexion = clase_mysql;
 - d. \$miconexion->conectar
 - e. \$miconexion = new clase_mysql;
 - f. \$ clase_mysql ->conectar
7. Instanciar un objeto en PHP de la forma correcta es:
 - a. \$miconexion->consulta();
 - b. \$miconexion->this->consulta();
8. Para consumir los métodos internos de una misma clase se realiza de la siguiente manera:
 - a. Consulta();
 - b. This->consulta();
 - c. \$this->consulta();
9. Una excepción es:
 - a. Es un evento que ocurre durante la ejecución de un programa y requiere de la ejecución controlada de un bloque de código.



- b. Es un incidente que ocurre durante la ejecución de un programa y requiere de la ejecución controlada de un bloque de código.
 - c. Es un evento que ocurre durante la ejecución de un programa.
10. Existen varios métodos de una excepción, seleccione los que sean correctos:
- a. getMessage().
 - b. get\$Code().
 - c. getFile().
 - d. getLine().

[Ir al solucionario](#)

Si no logró un buen resultado en la autoevaluación, no se preocupe, le recomiendo leer nuevamente el/los capítulos confusos y reforzar sus conocimientos. Y si aún tiene inquietudes, no dude en preguntar al profesor.



Resultados de aprendizaje 5, 7, 9 y 11:

- Construye aplicaciones web accesibles y seguras.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Para alcanzar los resultados de aprendizaje planteados, es fundamental desarrollar aplicaciones web que sean accesibles y seguras. Esto implica diseñar, codificar, probar y depurar programas simples que respondan a eventos del usuario de manera efectiva. Además, es importante conocer los métodos de seguridad para cookies y cómo pueden comprometer la privacidad del usuario si no se manejan adecuadamente. También es necesario entender cómo aumentar la fiabilidad de un sitio web, utilizando medidas como certificados de seguridad. La comprensión de aspectos relacionados con la seguridad y vulnerabilidades, incluyendo ataques XSS y sus variantes no persistentes, es clave para proteger y fortalecer las aplicaciones web.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 11



Unidad 5. Seguridad y vulnerabilidades

5.1 Ataques XSS Cross- Site Scripting

¿Qué es Cross-Site Scripting?

XSS ocurre cuando un atacante es capaz de inyectar un *script*, normalmente *Javascript*, en el *output* de una aplicación web, de forma que se ejecuta en el navegador del cliente. Los ataques se producen principalmente por validar incorrectamente datos de usuario, y se suelen inyectar mediante un formulario web, o mediante un enlace alterado.

Los desarrolladores muchas veces subestiman el potencial peligro que conlleva este tipo de ataques. Si un atacante puede inyectar *Javascript* en el *output* de una aplicación web y ejecutarlo, podrá ejecutar cualquier código *Javascript* en el navegador de un usuario.

Algunos de los objetivos que se quieren conseguir con ataques XSS son:



- Robar *cookies* y sesiones de usuarios.
- Modificar el sitio web.
- Realizar HTTP Requests con la sesión del usuario.
- Redireccionar a usuarios a sitios dañinos.
- Atacar al navegador o instalar *malware*.
- Reescribir o manipular extensiones de navegador, etc.

Desde la perspectiva del navegador, el **script** es originado por la aplicación web, por lo que se asume como fuente fiable. Esta es una de las causas que permiten que estos ataques se puedan llevar a cabo.

Para ello detallamos los tipos de ataques XSS que existen:



5.2 Non-persistent XSS

Los ataques **non-persistent XSS** o **reflected XSS** no almacenan el código malicioso en el servidor, sino que lo ejecutan y muestran directamente a la víctima. Es el método más popular de ataque XSS. El ataque se lanza desde una fuente externa, mediante *email* o un sitio de terceros.

Hecho «Has buscado la palabra: « . \$_GET[«query»]; http://example.com/search.php?query=<script>alert(«hackeado»)</ script>

• Persistent XSS

El código malicioso ya ha superado la barrera del proceso de validación y está albergado en un almacén de datos. Puede ser un comentario, un archivo log, un mensaje de notificación, o cualquier otro tipo de sección del sitio web que solicita algún *input* al usuario. Cuando esta información en particular se presenta en el sitio web, el código malicioso se ejecuta.

Si suponemos un sistema de comentarios en el que los comentarios se almacenan en un archivo llamado **comentarios.txt**:

file_put_contents(«comentarios.txt», \$_POST[«comentario»], FILE_APPEND);

Cuando el atacante envía un comentario, inserta, código malicioso en comentarios.txt. Cuando los comentarios se muestran, este código se ejecutará sin ser validado o escapado de ninguna forma.



Resultado de aprendizaje 3, 5, 9 y 11:

- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Construye aplicaciones web accesibles y seguras.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Para alcanzar los resultados de aprendizaje planteados, es esencial entender la importancia de la interconexión entre sitios web y sus bases de datos subyacentes. Además, es fundamental desarrollar aplicaciones web que sean accesibles y seguras, abordando la seguridad de cookies y los riesgos que pueden comprometer la privacidad del usuario. También es necesario conocer estrategias para aumentar la fiabilidad de un sitio web, como el uso de certificados de seguridad. La gestión de seguridad y vulnerabilidades, incluyendo la prevención de ataques y la validación de datos, es clave para proteger y fortalecer las aplicaciones web.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 12



Unidad 5. Seguridad y vulnerabilidades

5.3 Prevención de ataques

Tan fácil como un atacante puede afectar a un sitio web no protegido contra ataques *Cross-Site Scripting*, un desarrollador puede defenderse de estos. La prevención ha de tenerse siempre en cuenta, incluso antes de escribir el propio código.

La regla o política más básica que ha de tenerse siempre en cuenta es simple: nunca confíes en datos que vienen de usuarios o de cualquier otra fuente externa. Cualquier dato debe ser validado o escapado para su *output*.

Las medidas a tomar se pueden dividir en tres: *data validation*, *data sanitization* y *output escaping*.

5.4 Validación de datos

La validación de datos es el proceso de asegurarse que tu aplicación analiza el tipo de datos correctos. Si tu *script* PHP espera un *integer* de un *input*, cualquier otro tipo de dato debe de rechazarse. Cada dato debe ser validado cuando se recibe para asegurarse que es del tipo correcto, y rechazado, si no pasa ese proceso de validación.

Si quieres validar un número de teléfono, por ejemplo, deberás rechazar cualquier *string* que contenga letras, porque solo consistirá en dígitos. También puedes tener en cuenta la longitud que deberán tener estos dígitos. Siendo más permisivo, se pueden aceptar algunos otros símbolos como +, (), y - que a veces se utilizan al indicar números de teléfono:





```
// Comprobar un número de teléfono en Estados Unidos:
$telefono = '1-909-466-4344';
if (preg_match('/^((1-)?\d{3})-\d{3}-\d{4}/', $telefono)){ echo "El
teléfono $telefono es válido";
} else {
echo «El teléfono $telefono NO es válido»;
}
```

Data sanitization

La **sanitización de datos** se centra en manipular los datos para asegurarse que son confiables, eliminando cualquier parte indeseable y normalizándolos en la forma correcta.

Por ejemplo, si se espera un texto *string* de los usuarios, pueden querer evitar cualquier tipo de *markup* HTML:



```
// Sanitizar comentario de usuario.
$comentario = strip_tags($_POST[«comentario»]);
$telefono = «1234567»;
$telefono = preg_replace(</[^\d]/>, «», $telefono);
$length = strlen($telefono);
if ($length = 7 || $length = 10 || $length = 11){ echo «$telefono
es un formato válido»;
}
```

Output escaping

Para proteger la integridad de los **datos que se devuelven**, el *output data*, se debe escapar cualquier dato que se devuelve al usuario.

Esto evita que el navegador malinterprete alguna **secuencia especial de caracteres**:

```
echo «Has buscado la palabra: « . htmlspecialchars($_GET[«query»]);
```



Puede emplearse también la función *htmlentities()*. La diferencia entre ambas es que *htmlspecialchars()* **solo** traduce los símbolos **&**, **«»**, **«»**, **<** y **>** en **entidades HTML**, en cambio, *htmlentities()* traduce todos los caracteres posibles que tengan su equivalencia en HTML. Normalmente, vale con *htmlspecialchars()* a no ser que uses algún tipo de codificación diferente a **ISO-8859-1** o **UTF-8**.

Ejemplo de prevención contra ataques XSS

Mezclando un poco las tres **formas de prevenir ataques XSS**, vamos a ver un sencillo sistema de comentarios:



```
// Validar el comentario.
$comentario = trim($_POST["comentario"]);
if(empty($comentario)){
    exit("Debes proporcionar un comentario");
}

// Sanitizar comentario.
$comentario = strip_tags($comentario);
// El comentario ya se puede guardar de forma segura
file_put_contents("comentarios.txt", $comentario,
FILE_APPEND);
// Escapar comentarios antes de mostrarlos
$comentarios = file_get_contents("comentarios.txt"); echo
htmlspecialchars($comentarios);
```

Primero, nos aseguramos de que no se guardan **comentarios vacíos**. Después se sanitizan los datos, eliminando cualquier posible **etiqueta HTML** que pudiera contener. Finalmente, los comentarios se devuelven **filtrados**. La función *_strip_tags* hace que no sea posible insertar enlaces en los comentarios, ya que estos utilizan una etiqueta que será eliminada. Para que puedan insertarse se puede utilizar *htmlentities* o *htmlspecialchars* en su lugar.



Hay que tener en cuenta que ninguna solución es fiable al 100 %, y que es conveniente estar al tanto de novedades respecto a los **ataques Cross-Site Scripting** ya que van evolucionando a medida que lo hacen las plataformas que los facilitan (navegadores, HTML...).

Ataques CSRF *Cross-Site Request Forgeries*

Los ataques *Cross-Site Request Forgeries*, más comúnmente llamados ataques CSRF, se producen cuando el atacante provoca que el usuario ejecute una acción de forma no intencionada en una aplicación en la que había iniciado sesión.

Por ejemplo, cuando, estando el usuario “logueado” en su sitio favorito, procede a hacer clic en un enlace que parece inofensivo. En el fondo, su información de perfil está siendo actualizada con la dirección *email* del atacante. El atacante puede, ahora, usar la opción de **recuperar la contraseña** en el sitio web para cambiar la contraseña de la cuenta mediante *email*.

Cualquier acción que pueda realizar un usuario cuando está logeado en un sitio web la puede realizar también el atacante, ya sea.

- Actualizar su perfil.
- Añadir objetos a la cesta de la compra.
- Postear mensajes en un foro o cualquier otra cosa.



Actividad de aprendizaje recomendada

Es momento de aplicar su conocimiento a través de la actividad que se ha planteado a continuación:



Estimado estudiante, le invito a medir sus conocimientos de esta unidad con la siguiente autoevaluación. Esta le ayudará a valorar cuál es el nivel de conocimientos obtenidos durante el transcurso de este período transcurrido. En caso de no obtener resultados satisfactorios, le sugiero revisar nuevamente los contenidos en el cual tiene problemas.



Autoevaluación 5

Conteste la siguiente autoevaluación de la unidad 5:

1. El CrossSite Scripting es:
 - a. Una medida de seguridad en el momento del desarrollo de las WebApps.
 - b. Un ataque de concurrencia a nuestra aplicación web.
 - c. Un ataque capaz de inyectar un script en las salidas de nuestra WebApp.
2. Los ataques se producen principalmente por:
 - a. Uso incorrecto de la metodología de desarrollo.
 - b. Inyectar código malicioso mediante un formulario web o mediante un enlace alterado.
 - c. Una mala compra de los servidores de alojamiento.
3. Los objetivos de los ataques XSS son:
 - a. Robar cookies o sesiones de los usuarios.
 - b. Obtener datos como números de tarjetas de créditos.
 - c. Modificar el sitio web.
4. Los tipos de ataques XSS pueden ser:
 - a. Auto-persistent.
 - b. Non-persistent XSS.
 - c. Reflected XSS.
5. El ataque persistent XSS es:
 - a. El código malicioso ya ingresó por el proceso de validación y está almacenado en los datos.
 - b. Es el código malicioso que está buscando ingresar por algún puerto inseguro.



- c. Un código dañino en la configuración del servidor de base de datos.
- 6. La regla o política básica que se debe tener en cuenta ante estos ataques es:
 - a. Dejar la seguridad en manos del servidor web.
 - b. Nunca confiar en la configuración del servidor.
 - c. Nunca confiar en los datos que vienen del usuario o fuente externa.
- 7. Las medidas a tomar en la prevención de los ataques pueden ser:
 - a. Data validation.
 - b. Data sharing.
 - c. Output escaping.
- 8. La validación de datos es:
 - a. La validación de datos es el proceso de asegurarse que tu aplicación analiza el tipo de datos correcto.
 - b. La validación de datos es el proceso de analizar las propiedades de los formularios, ya que son estos por donde ingresa el usuario los datos.
 - c. Ninguna de las anteriores.
- 9. En PHP la función para sanitizar los datos es:
 - a. `Decode_utf8()`;
 - b. `Strleng()`;
 - c. `Strip_tags()`;
- 10. Los ataques CSRF tienen como objetivo:
 - a. Enviar conexiones al servidor haciéndolo colapsar.
 - b. El usuario, ejecute una acción de forma no intencionada.
 - c. Eliminar archivos de configuración del servidor.

[Ir al solucionario](#)

Si no logró un buen resultado en la autoevaluación, no se preocupe, le recomiendo leer nuevamente el/los capítulos confusos y reforzar sus conocimientos. Y si aún tiene inquietudes, no dude en preguntar al profesor.





¡Hemos terminado la quinta unidad! Felicitaciones.



Resultado de aprendizaje 8, 9, 10 y 11:

- Desarrolla código que responde a las condiciones de excepción planteadas durante la ejecución.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Identifica los sitios web que están utilizando gráficos de páginas web como iconos web.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Para alcanzar los resultados de aprendizaje planteados, es esencial desarrollar código que maneje adecuadamente las excepciones durante la ejecución. Además, es importante entender los métodos de seguridad para cookies y cómo pueden afectar la privacidad del usuario si no se utilizan correctamente. Identificar los sitios web que emplean gráficos como iconos también es crucial, así como describir las estrategias para aumentar la fiabilidad de un sitio web, como la implementación de certificados de seguridad. La seguridad web, que incluye la protección del servidor y el control de acceso mediante el archivo .htaccess, es fundamental para mantener la integridad y la seguridad de las aplicaciones.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 13



Unidad 6. Seguridad web

Estimado alumno, empezamos el sexto capítulo de la asignatura, abordando temas fundamentales que se requiere entender dentro de la seguridad web.



Para iniciar el estudio de este tema, le invito a revisar el artículo titulado [Evaluación multicriterio sobre herramientas de análisis de seguridad en aplicaciones web](#) y *blog* [servidor apache](#).

Posteriormente, realice una lectura comprensiva del tema de seguridad de aplicaciones web.

¿Qué opina de lo aprendido?, ¿tiene inquietudes o dudas? ¡A continuación las iremos resolviendo!

6.1 Seguridad del servidor

Las aplicaciones web tienen acceso abierto a un conjunto de información que explícitamente se hace pública. Sin embargo, en determinadas circunstancias, es interesante poder limitar el acceso a documentos reservados o útiles para un conjunto restringido de personas. Se pueden establecer dos tipos de restricciones:

1. Limitación de acceso **en función de direcciones IP o dominio**. Solo los usuarios de un dominio u organización tendrán acceso a la información.
2. Limitación de acceso **por nombres de usuario y claves de acceso**. Solo los usuarios que conozcan una clave de acceso válida pueden acceder a la información.

Otro aspecto que está cobrando especial importancia es la seguridad de la información que se intercambia en la web.

La *Internet* exige disponer de sistemas de comunicación seguros, capaces de adaptarse a las necesidades de los nuevos servicios, como la compra electrónica o la banca a distancia. En estos servicios, se manejan dos conceptos fundamentales, la autenticación y la confidencialidad.



Con los sistemas de comunicación, actualmente en uso, es técnicamente posible 'pinchar' un enlace de comunicaciones e interceptar el contenido de las comunicaciones TCP/IP que por él se transmiten. Cuando se envía información privada, por ejemplo, un número de tarjeta de crédito en un formulario de compra, es vital garantizar que la información sea recibida exclusivamente por su destinatario, y que la identidad es la esperada.

Control de acceso a la información

Se utiliza para limitar el acceso a determinados documentos de un servidor web, en función del origen y tipo de petición. La forma de hacerlo varía con el entorno en el que se publican las páginas (sistema operativo y servidor HTTP, principalmente). En general, todas las soluciones pasan por definir un fichero que contiene las diferentes limitaciones de acceso, en un formato característico del servidor HTTP. En algunos casos, se utiliza un fichero global con las restricciones de acceso o bien un fichero por cada directorio al que se quiere limitar el acceso.

Cuando un cliente web accede a un fichero protegido, el servidor devuelve un código de error asociado a la falta de permisos para realizar la operación (código 401).



Para reforzar esta temática, realice una lectura del artículo [Consejos de seguridad](#) y comente los temas principales en el foro del EVA.

Así mismo, en el foro de trabajo, comente cuál es el objetivo de estas líneas de código.

```
<Directory"/">
```

```
Require all denied
```

```
</Directory>
```



Con la lectura que hizo, ahora ya está en capacidad de asignar permisos en directorios de un servidor web, así como proteger las carpetas sensibles del servidor. Todos estos temas le ayudarán a implementar un nivel aceptable de seguridad al servidor web.

Muy bien, si está todo claro, podemos pasar al siguiente tema: Control de acceso al servidor apache.

6.2 Control del acceso al servidor con el archivo .htaccess

.htaccess (Acceso de hipertexto) es el nombre por defecto del archivo de configuración de directorios de Apache. Provee la habilidad para **personalizar la configuración de las directivas** definidas en el archivo de configuración principal. Las directivas de configuración necesitan estar en el contexto de .htaccess y el usuario necesita los permisos apropiados.

A continuación, en la siguiente infografía listaremos las configuraciones generales que suelen hacer con el archivo.

[Configuraciones esenciales para controlar el acceso con el archivo .htaccess](#)

Le resultó muy sencilla la lectura, ¿verdad? Ahora pasemos a realizar la siguiente actividad.



Actividad de aprendizaje recomendada

Implemente este siguiente código .htaccess en una aplicación de su servidor apache, para ello:

- Cree una carpeta llamada aplicacionweb1.
- Coloque algunos archivos como imágenes, PDF, etc.
- Ejecute la aplicación son 127.0.0.1/ aplicaiconweb1.
- Implemente el siguiente código en el .htaccess.
 - **RewriteEngine on.**
 - **RewriteRule ^\$ webroot/ [L].**



- **RewriteRule** (.*) webroot/\$1 [L].

- **Options -Indexes.**

- Socialice el resultado en el foro del EVA y emita comentarios.

Continuemos con algunos temas más de seguridad



Resultado de aprendizaje 3, 5, 9 y 11:

- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Construye aplicaciones web accesibles y seguras.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Para alcanzar los resultados de aprendizaje planteados, es fundamental comprender la importancia de la interconexión entre sitios web y sus bases de datos subyacentes. También es crucial desarrollar aplicaciones web que sean accesibles y seguras, gestionando adecuadamente la seguridad de las cookies para proteger la privacidad del usuario. Además, es necesario identificar estrategias para mejorar la fiabilidad de un sitio web, como la implementación de certificados de seguridad. La seguridad web abarca la configuración del servidor para defenderlo contra ataques de servicio y la correcta puesta en marcha de aplicaciones web para garantizar su funcionamiento seguro y eficiente.

Contenidos, recursos y actividades de aprendizaje recomendadas

Recuerde revisar de manera paralela los contenidos con las actividades de aprendizaje recomendadas y actividades de aprendizaje evaluadas.



Semana 14



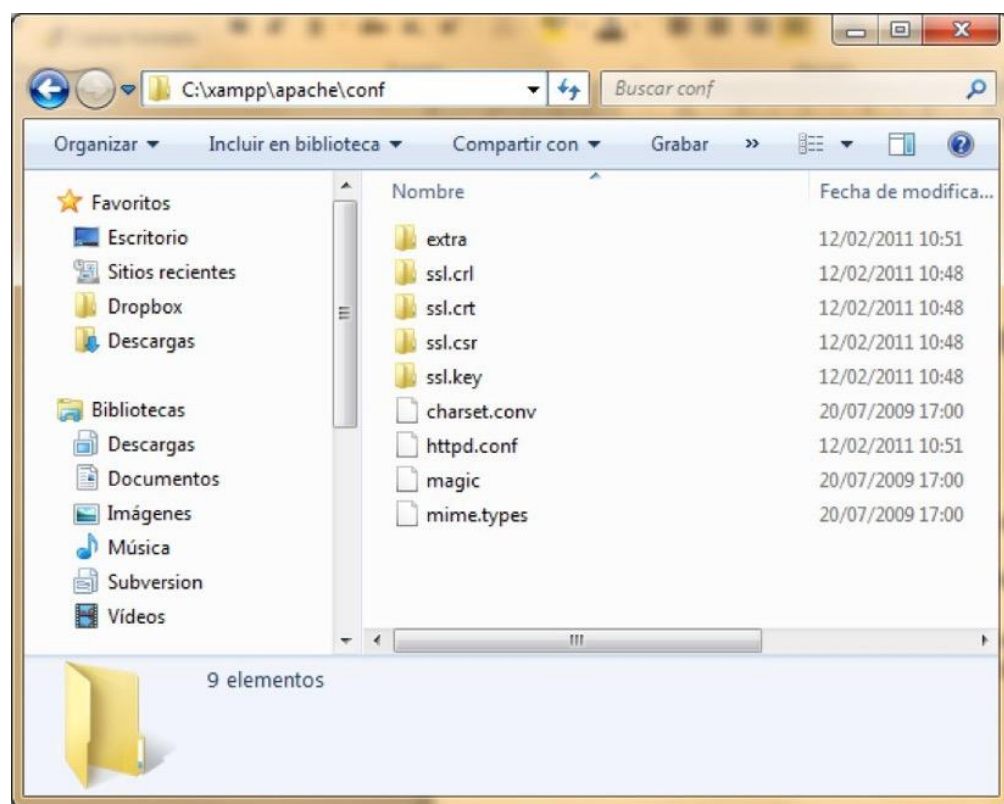
Unidad 6. Seguridad web

6.3 Configuración del servidor ante ataques de servicio

El servidor Apache, el cual es el que se instala con el XAMPP, contiene su propio archivo de configuración que se llama **httpd.conf**. Este se encuentra ubicado en la figura 23:

Figura 23

Directorio del archivo httpd.conf



Nota. Ramírez, R., 2020

El archivo httpd.conf está dividido en 3 partes:

- Configuración de parámetros globales.

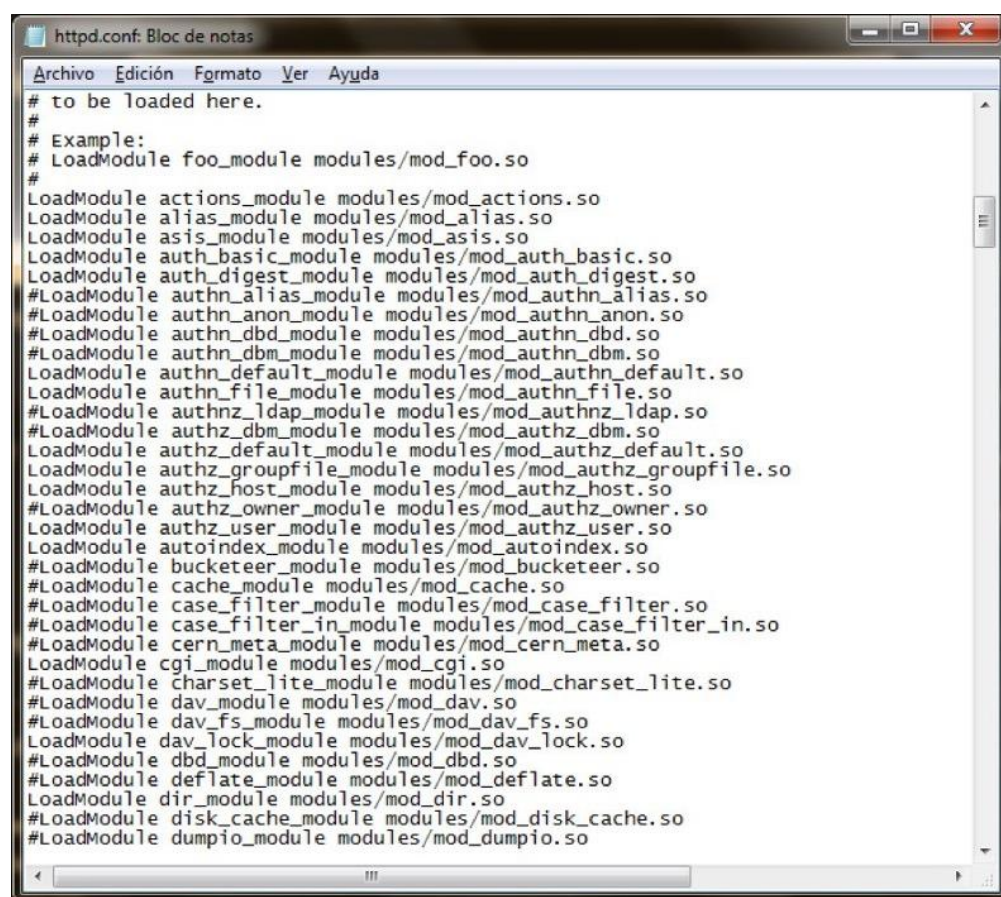
- Directivas de funcionamiento.
- Configuración de los *hosts* virtuales.

Por lo regular, el archivo contiene parámetros de configuración que vienen por defecto en la instalación. Pero también existen parámetros que se los pueden configurar a conveniencia del usuario.

Dentro de este archivo podemos activar o desactivar librerías tal como lo muestra la figura 24:

Figura 24

Contenido del archivo httpd.conf



```

httpd.conf: Bloc de notas
Archivo Edición Formato Ver Ayuda
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule authn_alias_module modules/mod_authn_alias.so
LoadModule authn_anon_module modules/mod_authn_anon.so
LoadModule authn_dbd_module modules/mod_authn_dbd.so
LoadModule authn_dbm_module modules/mod_authn_dbm.so
LoadModule authn_default_module modules/mod_authn_default.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authnz_ldap_module modules/mod_authnz_ldap.so
LoadModule authz_dbm_module modules/mod_authz_dbm.so
LoadModule authz_default_module modules/mod_authz_default.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule authz_owner_module modules/mod_authz_owner.so
LoadModule authz_user_module modules/mod_authz_user.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule bucketeer_module modules/mod_bucketeer.so
LoadModule cache_module modules/mod_cache.so
LoadModule case_filter_module modules/mod_case_filter.so
LoadModule case_filter_in_module modules/mod_case_filter_in.so
LoadModule cern_meta_module modules/mod_cern_meta.so
LoadModule cgi_module modules/mod_cgi.so
LoadModule charset_lite_module modules/mod_charset_lite.so
LoadModule dav_module modules/mod_dav.so
LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule dav_lock_module modules/mod_dav_lock.so
LoadModule dbd_module modules/mod_dbd.so
LoadModule deflate_module modules/mod_deflate.so
LoadModule dir_module modules/mod_dir.so
LoadModule disk_cache_module modules/mod_disk_cache.so
LoadModule dumpio_module modules/mod_dumpio.so

```

Nota. Ramírez, R., 2020

Eliminando el signo # desconectamos la librería para luego reiniciar los servicios y se active con dicha opción.

Otros parámetros pueden ser:

<Directory>: los parámetros que se encuentran dentro de esta sección, solo se aplicarán al directorio especificado y a sus subdirectorios.

<DirectoryMatch>: igual que *Directory*, pero acepta en el nombre del directorio expresiones regulares.

<Files>: los parámetros de configuración proporcionan.

<FilesMatch>: igual que *Files*, pero acepta expresiones regulares en el nombre del fichero.

<Location>: proporciona un control de acceso de los ficheros por medio de la URL.

<LocationMatch>: igual que *Location*, pero acepta expresiones regulares en el nombre del fichero.

Algunas veces, las directivas de funcionamiento de las secciones anteriores se pueden cruzar, en cuyo caso tienen el siguiente orden de preferencia:

1. <Directory> y .htaccess (.htaccess prevalece frente a <Directory>)
2. <DirectoryMatch> y <Directory>
3. <Files> y <FilesMatch>
4. <Location> y <LocationMatch>

Timeout

Timeout define, en segundos, el tiempo que el servidor esperará por recibir y transmitir durante la comunicación. *Timeout* está configurado por defecto a **300 segundos**, lo cual es apropiado para la mayoría de las situaciones.



6.4 ServerRoot

La directriz **ServerRoot** especifica el directorio de nivel superior que tiene el contenido web. Por defecto, ServerRoot está configurado a “/etc/httpd” para servidores seguros y no seguros.

6.5 ServerName

Use la directriz ServerName para configurar un nombre de servidor y un número de puerto (que coincida con la *directriz Listen*) para el servidor. El ServerName no necesita coincidir con el nombre real de la máquina. Por ejemplo, el servidor web puede ser `www.example.com`, pero el nombre del servidor es en realidad `foo.example.com`.

El valor especificado en ServerName debe ser un nombre del Servicio de Nombres de Dominio (*Domain Name Service*, DNS) válido que pueda ser resuelto por el sistema.

Lo siguiente es una directriz ServerName de ejemplo: `www.servidorweb.com:80`.

Para reforzar aún más este tema le invito a realizar la siguiente actividad.



Desarrolle una práctica de configuración del servidor local, active la librería para la base de datos *postgres* y cambie su *servername* a `www.ejemploweb.com`. Capture la pantalla de la práctica y súbalas al EVA y emita criterios de la misma.

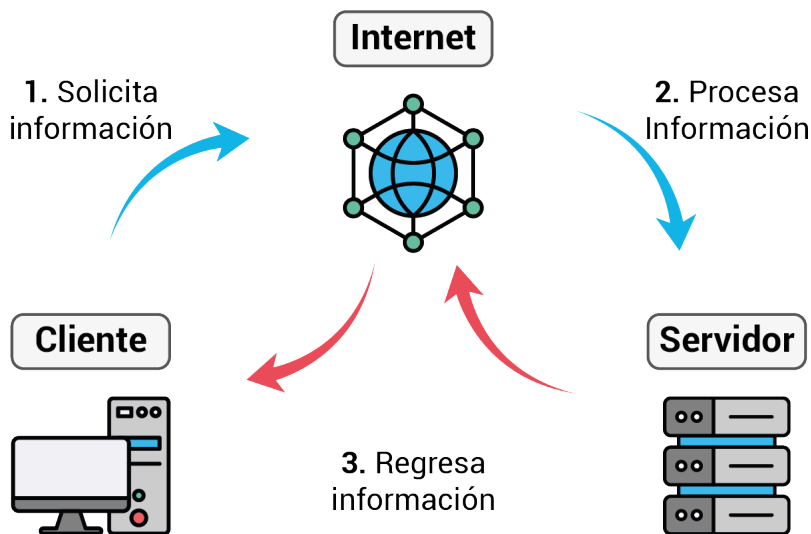
6.6 Puesta en marcha las WebApps

Es importante conocer cuál es el proceso de implementación de nuestra *WebApp*, que se debe comprar y donde se lo puede realizar, ventajas y desventajas en la implementación. A continuación, se describe la forma de cumplir este objetivo.



Figura 25

Proceso de la petición de una WebApp con el servidor



Nota. Ramírez, R., 2024.

El Web Hosting es una computadora con características especiales donde se guarda toda la información de la WebApp, correos, bases de datos, etc.

En la figura 25 se describe el proceso que se realiza cuando el cliente solicita una aplicación web por Internet, ejemplo www.utpl.edu.ec, la computadora del usuario se conecta con otra computadora en internet para descargar la información que va a mostrar, a esta otra computadora se le llama **Servidor**, porque es la encargada de brindar la información, y a tu computadora se le llama **Cliente**, porque es quien recibe la información.

El servidor que brinda el *Hosting*, que es todo lo necesario para que una página pueda ser utilizada desde *Internet*, tiene varios servicios, como son:

- Espacio en disco para almacenar la página.
- Espacio para guardar los correos electrónicos.
- Recibir y enviar correos electrónicos.
- Guardar y administrar bases de datos.

- Guardar estadísticas sobre las visitas a la página de *Internet*.

Dominios

Los **dominios** son «nombres» de las páginas en *Internet*. Cuando se quiere entrar a una WebApp, normalmente se escribe el nombre, por ejemplo:

www.utpl.edu.ec.

El nombre de dominio está compuesto por dos partes:

1. El nombre: utpl.
2. TLD: .edu.ec.

La terminación. Que puede ser **.com**, **.com.ec**, **.org**, **.ec**, **.edu**, **.edu. ec**, etc. Esta terminación ayuda a indicar el motivo o la razón de tu empresa y ubicación:

- **.com**
- **.com.ec** -> comercial en Ecuador.
- **.org** organización sin fines de lucro.
- **.tv** canal de televisión o televisora.
- **.edu**
- **.edu.ec** educativas de Ecuador.

Hay reglas para escoger un nombre de dominio:

- **Tiene que ser único:** si alguien ya registró un dominio, no se puede repetir.
- **Pueden tener los siguientes caracteres:** letras de la A a la Z (con la excepción de la ñ) o Números del 1 al 0 o guion medio "-".
- Cualquier otro carácter, como espacios en blanco, acentos, signos de puntuación, paréntesis, guion bajo "_", etc., no pueden ser parte de un dominio.
- No pueden empezar con un guion medio.

Para poder ver la disponibilidad del dominio, le invito a ingresar al sitio web networksolutions y buscar si está disponible el dominio.



Luego de realizar la lectura sugerida, ¿le parece aplicable? Reflexione sobre lo que tiene en su entorno y aplique lo indicado. Si le quedaron dudas vuelva a leer detenidamente el tema, formule preguntas y consúltelas con su tutor.



Actividades de aprendizaje recomendadas

Es momento de aplicar sus conocimientos a través de las actividades que se han planteado a continuación:

1. Con base en el conocimiento adquirido, conste las siguientes preguntas:
 - Liste que es un servidor y un cliente.
 - Qué es el *hosting*.
 - Qué es un dominio.
 - Cuáles son las terminologías que pueden tener.
 - Con ayuda del foro del EVA comente qué empresa vende dominios con la terminología .ec.

Nota: por favor, copie las preguntas y de respuesta en un documento Word o cuaderno.

2. Realice un presupuesto para la implementación del cms instalado anteriormente en la *Internet*, que abarque lo que es el dominio .ec y el *hosting*. Puede comentarlo en el foro del EVA.
3. Estamos avanzando en los contenidos, es por ello que lo invito a medir sus conocimientos, desarrollando la autoevaluación. Esta le ayudará a autoevaluarse y ver su estado en la materia.

¡Suerte en la autoevaluación! Siga adelante.





Autoevaluación 6

Seleccione la opción correcta:

1. Dentro de la seguridad, se encuentra el control de acceso de información. Localice el tipo de restricciones:
 - a. Limitación de acceso en función de direcciones IP o dominio.
 - b. Solo los usuarios de un dominio u organización tendrán acceso a la información.
 - c. Limitación de acceso por sistema operativo.
2. El significado del archivo .htaccess en español es:
 - a. Acceso de protocolo de hipertexto.
 - b. Acceso de hipertexto.
 - c. Acceso de hipermedia.
3. Para la restricción de una carpeta en el archivo .htaccess es:
 - a. Allow from all.
 - b. Deny from all.
 - c. Allow dem /carpeta.
4. Para permitir el acceso desde un rango de IP es:
 - a. Deny from 10.2.2.2.
 - b. Allow from 192.167.2.3/34.-.
 - c. Deny from all.
5. El archivo de configuración del servidor Apache se denomina:
 - a. Htpd.conf.
 - b. Httpd.ini.
 - c. Httpd.conf.
6. De la siguiente lista marque las partes en que se divide el archivo httpd:
 - a. Configuración de parámetros internos.
 - b. Configuración de parámetros globales.
 - c. Directivas de funcionamiento.
 - d. Restricciones de dominios.
7. Localice el proceso de una petición web de la siguiente lista:
 - a. Servidor – cliente- Internet – servidor- cliente.



- b. Cliente – servidor – Internet – servidor - cliente.
- c. Petición del cliente – Internet – servidor – Internet – cliente.
8. () Coloque una V si es verdadero o una F si es falso, lo siguiente: el web hosting es el alojamiento de mi aplicación web en otra máquina local.
9. () Coloque una V si es verdadero o una F si es falso a lo siguiente: la terminología .ec significa que son de Ecuador.
10. () Coloque una V si es verdadero o una F si es falso, a lo siguiente: el dominio son los nombres de mi página web.

[Ir al solucionario](#)

Si no logró un buen resultado en la autoevaluación, no se preocupe, le recomiendo leer nuevamente el/los capítulos confusos y reforzar sus conocimientos. Y si aún tiene inquietudes, no dude en preguntar al profesor.



Resultados de aprendizaje 3, 5, 6, 7, 8, 9 y 10:

- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Construye aplicaciones web accesibles y seguras.
- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.
- Desarrolla código que responde a las condiciones de excepción planteadas durante la ejecución.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Identifica los sitios web que están utilizando gráficos de páginas web como iconos web.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 15

Actividades finales del bimestre

En esta semana nos vamos a dedicar a revisar las temáticas estudiadas en las semanas anteriores para que se puedan ir afianzando los temas previos a la evaluación presencial del segundo bimestre.



Además de eso, se tiene algunas actividades calificadas que es importante que usted realice como parte del proceso de preparación, como el *chat* académico que se ha programado para esta semana.

Asimismo, el desarrollo de la actividad suplementaria, para el caso de los alumnos que por algún motivo no puedan realizar la actividad síncrona.



Resultados de aprendizaje 3, 5, 6, 7, 8, 9 y 10:

- Explica la importancia de la interconexión entre sitios web para bases de datos subyacentes.
- Construye aplicaciones web accesibles y seguras.
- Explica la diferencia entre la programación orientada a eventos y programación de línea de comandos.
- Diseña, codifica, prueba y depura programas simples basados en eventos que respondan a eventos de usuario.
- Desarrolla código que responde a las condiciones de excepción planteadas durante la ejecución.
- Describe los métodos de seguridad para cookies y las formas en las que pueden ser usadas para poner en peligro la privacidad del usuario.
- Identifica los sitios web que están utilizando gráficos de páginas web como iconos web.
- Describe formas de aumentar la fiabilidad de un sitio web, tales como certificados de seguridad.

Contenidos, recursos y actividades de aprendizaje recomendadas



Semana 16

Actividades finales del bimestre

En la última semana del segundo bimestre nos vamos a enfocar en la revisión del contenido, previo a la evaluación presencial. Se recomienda aprovechar esta semana revisando nuevamente cada uno de los temas, y si surgieran dudas, por favor contactarse con el docente-tutor para resolverlas.





4. Solucionario

Autoevaluación 1

Pregunta	Respuesta	Retroalimentación
1	b	Porque la accesibilidad web pretende llegar a todo tipo de personas sin discriminación de ningún tipo.
2	a, c	Porque en educación en línea es importante que puedan llegar a todos los usuarios de la misma manera que en la comunicación digital.
3	b	Las siglas de WAI significan Iniciativa de Accesibilidad Web.
4	a, b, d	Las técnicas estudiadas en la unidad señalan que un sitio sea estructurado semánticamente en su contenido, proporcionan resúmenes a las tablas de información y las imágenes deben tener una proporción en cuanto a tamaño versus al contenido.
5	a, c	Según las herramientas de validación, la plataforma tawdis y sidar/hera son algunas de las herramientas que nos permiten la validación de la accesibilidad en aplicaciones web.
6	a, c	Una de las características mencionadas de las etiquetas en la unidad sobre Html5 es que se ha creado etiquetas relevantes para cada apartado de los elementos de una página web como los títulos, módulos, contenidos, etc., esto ayuda notablemente a la presentación semántica y por ende a la accesibilidad web.
7	c	La etiqueta nav fue creada para representar los menús de una aplicación web.
8	b	Porque representa las secciones de contenido de una aplicación web y su mejor forma de representación es con la etiqueta <section>.
9	c	Porque en html5 la etiqueta footer fue creada para representar los pies de página.
10	b	Porque con esta etiqueta html se representa el contenido principal de una página y es lo que los buscadores analizarán en primera instancia para el posicionamiento.



[Ir a la autoevaluación](#)

Autoevaluación 2

Pregunta	Respuesta	Retroalimentación
1	b	Porque intervienen diferentes áreas como diseño, desarrollo, control de calidad, etc.
2	b, c	Entre los programas que tiene XAMPP es PHP como lenguaje de programación y apache como servidor web.
3	c	La base de datos que se instala con el programa XAMPP es MYSQL.
4	b	La etiqueta que el compilador de PHP entiende para leer su código es <?php ... ?>
5	a	Siempre se utiliza el signo de dólar para la creación de variables en PHP ejemplo \$var=0;
6	c	Las constantes son variables que poseen una asignación determinada y dura toda la ejecución del programa.
7	c, d	Dentro de los operadores de comparación se encuentran == es igualdad y != para representar la diferencia.
8	b	Dentro de los operadores del incremento se encuentran los signos ++ que indican que la variable aumentará de 1 en 1.
9	b	Las condiciones son representadas en programación con el IF y la estructura correcta de esta es como lo indica la opción b.
10	b	Una función son bloques de códigos que son ejecutadas siempre que sean llamadas o invocadas, estas pueden enviar y retornar parámetros.

[Ir a la autoevaluación](#)



Autoevaluación 3

Pregunta	Respuesta	Retroalimentación
1	c	Porque el editor web para la gestión de bases de datos que se instala con XAMPP es phpMyAdmin.
2	b	La forma correcta para crear una base de datos a nivel de consola es <code>create database nombre;</code> ; esta debe ser ejecutada en el editor o en la consola para su ejecución.
3	c	Una vez creada la base de datos y poder visualizar las bases de datos que se tiene en el servidor, la sentencia correcta es <code>show databases.</code>
4	b	En base de datos las tablas deben tener una llave primaria para poder identificar sus registros, para ello se debe colocar al campo que cumpla como valor único e irrepetible con la propiedad de <code>primary key</code> .
5	a, b	Existen diferentes formas para construir una sentencia SQL para insertar datos en la base de datos, entre ellas pueden ser las opciones a y b, ya que no poseen errores de estructura.
6	a, c	La forma correcta de obtener los datos de las tablas de una base de datos son las opciones a y c, ya que no poseen problemas en la estructura de la sentencia SQL.
7	b	La forma correcta para eliminar registros de una tabla es con la opción b, ya que posee una cláusula <code>where</code> que identifica qué registro borrar, caso contrario se eliminaría todos los valores de la tabla.
8	c	Dentro de la gestión de los datos en una base de datos está la opción <code>update</code> la cual permite la actualización de los registros, en la opción c, se muestra la estructura correcta mencionando que se debe tener una cláusula <code>where</code> para identificar el registro a modificar.
9	c	Dentro de las funciones reservadas en PHP para la selección de la base de datos está <code>mysql_select_db</code> que nos permite escoger la base de datos a utilizar en nuestra WebApp.
10	b	<code>mysql_query</code> es una función reservada dentro de PHP para la ejecución de una sentencia SQL.

[Ir a la autoevaluación](#)



Autoevaluación 4

Pregunta	Respuesta	Retroalimentación
1	a	La POO tiene como objetivo la realización de código mediante el uso de clases y funciones para ser utilizados en cualquier parte de la aplicación.
2	c	Se trata de un ente abstracto usado en programación que permite separar los diferentes componentes de un programa, simplificando así su elaboración, depuración y posteriores mejoras. Los objetos integran, a diferencia de los métodos procedurales, tanto los procedimientos como las variables y datos referentes al objeto.
3	b	Una clase es la descripción de un conjunto de objetos similares; consta de métodos y de datos que resumen las características comunes de dicho conjunto.
4	a, c	Dentro de las propiedades de las clases para su invocación pueden ser públicas o privadas.
5	b	Para acceder a las propiedades internas como los métodos o funciones de la clase se utiliza \$this-> con ello podemos reutilizar propiedades internas de la clase.
6	a	La forma correcta para crear una instancia de la clase MYSQL es la opción a, ya que esta no tiene ningún problema de estructura.
7	b	La forma correcta para crear una instancia de un objeto de una clase MYSQL es la opción b, ya que esta no tiene ningún problema de estructura.
8	c	La forma correcta de reutilizar un método de la clase es con la propiedad \$this-> más el método o la función a utilizar.
9	a	Una excepción se refiere a un evento que sucede mientras un programa está en ejecución y que interrumpe el flujo normal del programa.
10	a, c, d	En PHP, los métodos más comunes para manejar excepciones incluyen getMessage(), que devuelve el mensaje de la excepción; getFile(), que proporciona el nombre del archivo donde ocurrió el error, y getLine(), que indica la línea del archivo en la que se produjo la excepción.

[Ir a la autoevaluación](#)



Autoevaluación 5

Pregunta	Respuesta	Retroalimentación
1	c	Un Cross Site son ataques que sufren las aplicaciones web al momento de presentar los valores solicitados al servidor.
2	b	Los ataques por lo regular suelen ocurrir cuando existe código malicioso en el servidor web o es incrustado por vías como los formularios o el navegador.
3	a, c	Los ataques XSS utilizan las sesiones creadas por las aplicaciones con el objetivo de modificar la información de la aplicación web.
4	b, c	Dentro de las vulnerabilidades a las aplicaciones web están las opciones b y c que son ataques de terceros y están dentro de la XSS para la modificación del sitio web.
5	a	El ataque persistente XSS ejecuta código malicioso en el proceso de validación del usuario para poder suplantar la identidad y poder tener acceso a la información restringida.
6	c	Existen muchas reglas a tomar en cuenta en el momento de aplicar seguridad, pero una importante es la opción c, ya que se debe estar preparado ante cualquier dato que ingrese el usuario final.
7	a, c	Dentro de las estrategias para ver la información que ingresa el usuario y la validación de los datos es la opción a y c, ya que garantiza un nivel de seguridad mayor.
8	a	La validación de los datos es la utilización de estrategias como las del navegador en el lado del cliente como el análisis del dato en el lado del back-end para garantizar la integridad de la información que llega al servidor.
9	c	En el lenguaje PHP nos ayuda a garantizar la integridad de los datos que ingresan del lado del cliente mediante la función strip_tags().
10	b	Dentro de las vulnerabilidades de una aplicación web se encuentran las que el usuario ejecuta de una forma no intencionada, para ello se debe restringir al máximo las opciones al usuario.

[Ir a la autoevaluación](#)



Autoevaluación 6

Pregunta	Respuesta	Retroalimentación
1	a	Otra forma de brindar seguridad a la aplicación web es garantizar seguridad al servidor mediante la restricción a los archivos de la app.
2	b	El significado del archivo .htaccess es acceso de hipertexto que es un archivo de configuración del servidor.
3	b	Dentro de los parámetros que se pueden configurar en el archivo .htaccess está la restricción de los directorios que se configura con la opción b.
4	b	Dentro de los parámetros que se pueden configurar en el archivo .htaccess es la restricción de las direcciones IPS que se configura con la opción b.
5	a	El archivo general del servidor apache se denomina httpd.conf. Este archivo es crucial para la configuración del servidor, permitiendo ajustar parámetros y directivas que controlan el funcionamiento de Apache.
6	b, c	El archivo httpd.conf se compone de dos partes que son la configuración de parámetros globales del servidor y las directivas de funcionamiento. Ambas partes son esenciales para una configuración adecuada y segura del servidor Apache, permitiendo una administración flexible y eficiente del servidor web.
7	c	El proceso de petición desde el cliente hasta el servidor sigue el patrón cliente – Internet – servidor – Internet – cliente, con el fin de obtener los datos solicitados por el cliente y consumidos en el servidor.
8	f	Es falsa ya que el hosting debe ser público para el acceso de todo el público de la app.
9	v	Todos los dominios que terminan en .ec, ejemplo utpl.edu.ec, indican que son de procedencia de Ecuador.
10	v	Es correcto porque es el nombre con el cual el usuario podrá reconocer y acceder a la aplicación web para poder consumir la información que brinda la app.

[Ir a la autoevaluación](#)





5. Glosario

Código fuente: conjunto de instrucciones que forman un programa o subprograma informático.

CSS: hojas de estilo en cascada (*Cascading Style Sheets*) es el lenguaje utilizado para describir la presentación de documentos HTML o XML.

PHP: lenguaje de desarrollo para la construcción de aplicaciones *web*.

Hipervínculo: es un enlace de una página *web* o un archivo a otra página *web* u otro archivo.

HTML: Lenguaje de Marcado para Hipertextos (*HyperText Markup Language*) es el elemento de construcción más básico de una página *web*.

HTML5: es la última versión de HTML, con nuevos elementos, atributos y comportamientos.

HTTPS: protocolo de transferencia de hipertexto (en inglés: *Hypertext Transfer Protocol* o HTTP) es el protocolo de seguridad.

JS: es un archivo de texto plano que contiene *scripts* de Javascript, y que puede, por tanto, ser modificado con cualquier editor de textos.

Link: apuntador de hipertexto que tiene como misión llevarnos de una información a otra, de una página a otra, o de un servidor a otro.

Navegador web: es un *software*, aplicación o programa que permite el acceso a la *web*, que se instala en los clientes.

CMS: Sistemas Manejadores de archivos que ayudan a la creación de aplicaciones *web*.

Regla: es cada uno de los estilos que componen una hoja de estilos CSS.



Selecto: indica el elemento o elementos HTML a los que se aplica la regla CSS.

Servidor web: es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente.

URL: es una sigla del idioma inglés correspondiente a *Uniform Resource Locator* (Localizador Uniforme de Recursos); es una secuencia de caracteres que sigue un estándar y que permite denominar recursos dentro de la web.

Valor: establece el nuevo valor de la característica modificada en el elemento.

W3C: el Consorcio *World Wide Web* es una comunidad internacional donde las organizaciones Miembro [ing, personal [inglés] a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares web [inglés].





6. Referencias Bibliográficas

Básica

Pavón, J. Llanera, E. (2016). Creación de un sitio web con php y msyql. Madrid. España: RA-MA Editorial. ISBN: 978-84-9964- 567-4

Este libro posee un importante contenido ilustrativo e intuitivo que nos ayudará a cumplir nuestros objetivos planteados a lo largo de nuestra materia, como es el desarrollo de aplicaciones en PHP, así como también la gestión de los datos en una base de datos relacional como lo es mysql.

Ramírez, R. (2017): Guía didáctica de Desarrollo Web, Loja-Ecuador, Editorial Universidad Técnica Particular de Loja.

En la guía didáctica encontrará los lineamientos necesarios para que se pueda guiar a través del **texto básico**, además encontrará explicaciones adicionales sobre algunos temas que le permitirán entenderlos mejor.

Complementaria

PRESSMAN R.(2010), Ingeniería de Software, un Enfoque Práctico.

México 7.^a edición. McGraw-Hill. ISBN 970-10-5473-3. **Texto básico** de estudio de la asignatura de Ingeniería de software, en esta obra nos acerca y profundiza a los contenidos de Ingeniería de software. Este texto es de lectura y le servirá de mucho durante el transcurso de este período de estudios.

SOFTWARE ENGINEERING FOR INTERNET APPLICACIONES. Eve Andersson, Philip Greenspun, and Andrew Grumet. 6.^a edición. [en línea] Disponible en: <http://ndl.ether.net.edu.et/bitstream/123456789/4863/1/18pdf.pdf>, [consultado a: 11/12/2017].



Recurso digital donde encontrará los contenidos relacionados con el proceso de desarrollo de aplicaciones web.

SOFTWARE ENGINEERING FOR INTERNET APPLICACIONES. Eve Andersson, Philip Greenspun, and Andrew Grumet. 6.ª edición. [en línea] Disponible en: <http://www.kiarash.net/Teaching/22-24-118/Resources/Software%20Engineering%20for%20Internet%20Applications.pdf> [consultado a: 2 de junio del 2017].

Recurso digital donde encontrará los contenidos de la metodología de desarrollo de aplicaciones web, fases o procesos a seguir para el desarrollo de las apps.

Accesibilidad Web [en línea] Disponible en: <https://blog.hubspot.es/website/que-es-accesibilidad-web> [consultado a: 12 de diciembre del 2017].

Recurso digital donde elementos básicos para el desarrollo web accesible, con estándares.

Codigofacilito (2017). Curso de Introducción a HTML 5. Disponible en: <https://codigofacilito.com/cursos/HTML5>

En este sitio web hay disponibles varios videos cortos pero explicativos sobre los nuevos elementos HTML 5, con ejemplos en código y explicaciones detalladas.

Gómez, M. R. (2013). HTML5, CSS3 y Javascript. Anaya Multimedia- Anaya Interactiva.

En este libro se explica a detalle los elementos y estructura del Lenguaje de etiquetado HTML5 con sus componentes en CSS3 y Javascript.

Darío Andrés Silva, Construyendo aplicaciones web con una metodología de diseño orientado a objetos. [en línea] Disponible en: <http://revistas.unab.edu.co/index.php?journal=rcc&page=article&op=viewArticle&path%5B%5D=1116> [consultado a: 12 de diciembre del 2017].



Recurso digital donde se explica la metodología de diseño orientado a objetos orientado a aplicaciones web.

Apache, [en línea] Disponible en: https://httpd.apache.org/docs/trunk/es/misc/security_tips.html [consultado a: 12 de diciembre del 2017].

Recurso digital donde se explica lo que se debe tener en cuenta para contar con una buena seguridad en nuestro servidor y algunos consejos prácticos en la configuración del servidor Apache.

Base de Datos: Guía de base de datos. Recuperado de <https://www.powerdata.es/integracion-de-datos>

Guía Digital: Usabilidad web. Recuperado de <https://rockcontent.com/es/blog/usabilidad/>

Guía y configuración del servidor web apache. Recuperado de https://httpd.apache.org/docs/trunk/es/misc/security_tips.html

Guía de accesibilidad web. Recuperado de <https://userway.org/es/blog/accesibilidad-web/>

Validadores de accesibilidad web. Recuperado de https://www.usableyaccessible.com/recurso_misvalidadores.php

Desarrollo de aplicaciones con en HTML5, maquetado semántico.
Recuperado de http://www.aulaclie.es/articulos/html5-semantic_1.html

Seguridad Web: PHP con Mysql. Recuperado de <https://diego.com.es/seguridad-web-en-php>

Desarrollo de aplicaciones web. Recuperado de https://rua.ua.es/dspace/bitstream/10045/90894/1/daw-18_19-prac8-PHPbd1.pdf

PHP con base de datos Mysql. Recuperado de https://www.w3schools.com/php/php_mysql_intro.asp

