

CONTROL DE CAMBIOS:

Versión	Fecha	Aprobación	Observación
001	Enero 2019	Comité Nacional de Desarrollo y Calidad Académica	

**PROGRAMA DE FORMACIÓN DE ASIGNATURA**

I. DATOS GENERALES DEL MÓDULO/IDENTIFICACIÓN				
Asignatura: Tecnología Web II Semestre: Séptimo Modalidad: Presencial Ciclo: Tercero Carrera o Programa de Formación: Ingeniería de Sistemas Facultad: Ingeniería	ASIGNATURA	Nodo problematizador: La falta de aplicaciones de software y sistemas de información que implementen bases de datos en diferentes plataformas para sistematizar la información limita el desarrollo de aplicaciones de software ágil que respondan a las necesidades del entorno.		
		Sigla: SIS-0151		
		Créditos: 6		
		Horas	HT	HP
			40	80
		Total		
		120		
		Pre- requisitos Tecnología Web I		
		Co-requisito Aplicaciones Móviles I, Programación IV		

II. COMPETENCIAS:

Competencias genéricas: Investigación, Ética y Prosocialidad, Medioambiente, Emprendedurismo y NTICs.
Micro Competencias Específicas: Escribe programas de aplicaciones web con acceso a bases de datos para plantear soluciones a problemas del contexto en base a lenguajes de programación del lado del servidor considerando técnicas de optimización de algoritmos y paradigmas de programación.
Criterios de verificación: 1. Aplica la POO en la creación de una pequeña aplicación en base a el lenguaje de programación PHP. 2. Crea soluciones webs conectadas a gestores función a base de datos mostrando información veraz y concisa. 3. Implementa servicios web en concordancia a la tecnología AJAX para descargar información.
Evidencia de Producto 1. Portafolio de evidencias 2. Trabajos de laboratorio 3. Implementación de programas de aplicación 4. Informe final de proyecto de aplicación

III. SABERES ESENCIALES:

SABERES ESENCIALES Propios de cada bloque	BLOQUES DE CONTENIDOS	SABERES ESENCIALES Comunes a todos los bloques		
	1. Introducción a la programación del lado del servidor	SER	CONVIVIR	
CONOCER	1.1. El servicio web. 1.2. Arquitectura cliente/servidor. 1.3. Tipos de lenguajes de programación. 1.4. Programación del lado del cliente vs programación del lado del servidor 1.5. Tecnologías de programación del lado del cliente 1.6. Tecnologías de programación del lado del servidor	<ul style="list-style-type: none">• Muestra actitud crítica, analítica y responsable.• Muestra Interés en la adquisición de conocimientos.	<ul style="list-style-type: none">• Valora la importancia del trabajo en equipo.	
HACER	<ul style="list-style-type: none">• Identifica las diferencias y ámbitos de aplicación de la programación en el lado del cliente y el lado del servidor.	<ul style="list-style-type: none">• Valorar el aprendizaje autónomo.	<ul style="list-style-type: none">• Respeta las decisiones y opiniones ajenas.	
	2. Lenguaje de programación PHP	Trabaja en situaciones de falta de información y bajo presión, teniendo nuevas ideas, siendo creativo. <ul style="list-style-type: none">• Capacidad de comunicarse de forma efectiva y ética con los compañeros, usuarios (potenciales) y el público en general. <ul style="list-style-type: none">• Creatividad para plantear soluciones.	<ul style="list-style-type: none">• Se relaciona y comunica de forma asertiva en su entorno social.• Compromiso con la sociedad para describir problemas del contexto.• Valora el rol de los integrantes de un equipo de investigación.	
CONOCER	2.1. Plataforma PHP 2.2. Aplicaciones PHP 2.3. Lenguaje básico 2.4. Variables 2.5. Operadores 2.6. Expresiones, sentencias y bloques 2.7. Sentencias de control de flujo 2.8. Arreglos 2.9. Cadenas			
HACER	<ul style="list-style-type: none">• Reconoce las estructuras de control del lenguaje.• Implementa programas sencillos utilizando el lenguaje de programación PHP			
	3. Funciones y formularios			
CONOCER	3.1. Creación de funciones. 3.2. Funciones con parámetros. 3.3. Recepción de datos de formulario. 3.3.1. Método GET. 3.3.2. Método POST. 3.4. Recepción de archivos. 3.5. Inclusión de código 3.5.1. Include. 3.5.2. Require.			

HACER	1. Crea aplicaciones pequeñas con la utilización de funciones. 2. Crea formularios manejando los métodos Get y Post		
	4. Programación Orientada a Objetos (POO)		
CONOCER	4.1. Conceptos básicos. Creación de clases, métodos, propiedades y objetos. 4.2. Excepciones y manejo de errores. 4.3. Concepto de template.		
HACER	<ul style="list-style-type: none"> Reconoce la POO utilizada en el lenguaje PHP Crea aplicaciones utilizando la POO 		
	5. Conexión a Base de Datos MySql		
CONOCER	5.1. Definición – ODBC 5.2. Creación de tablas 5.3. Consultas 5.4. Funciones de php para interactuar con mysql 5.5. Identificación de usuarios - Seguridad		
HACER	<ul style="list-style-type: none"> Realizar consultas a la base de datos Conectar a la base de datos desde la aplicación Definir niveles de seguridad Crea de aplicaciones web con acceso a base de datos 		
	6. PHP avanzado		
	6.1. Web Service 6.1.1. Fundamentos de sintaxis JSON 6.1.2. Fundamentos de sintaxis XML 6.2. Servicios SOAP Y REST 6.3. Consumir Web Service 6.4. PHP y AJAX 6.4.1. Validación de formularios con AJAX. 6.4.2. Sugerir y autocompletar. 6.4.3. Grid 6.4.4. Lector Rss		
	<ul style="list-style-type: none"> Comprender la lógica de las aplicaciones Cliente para consumo de datos. Comprender la lógica de las interfaces Servidor para proveer datos. Creación de un web service básico Crea un cliente para un web Service 		

	<ul style="list-style-type: none"> • Construye una aplicación sencilla con Ajax y PHP 		
--	--	--	--

IV. ESTRATEGIAS DE ENSEÑANZA Y APRENDIZAJE:

- Clases Prácticas.
- Método Basado en problemas del contexto.
- Técnicas de presentación, trabajo en equipo, retroalimentación etc.
- Motivación al autoaprendizaje.
- Revisión de internet.
- Aprendizaje basado en proyectos

V. RECURSOS DIDÁCTICOS:

Los recursos didácticos que se estarían utilizando para impartir adecuadamente la materia de contabilidad empresarial I. serían los siguientes:

- Multimedia
- Bibliografía digital
- Fotocopias de libros e Impresos,
- Pizarra, marcadores
- Recursos NTIC's
- Libros y recursos web

VI. SISTEMA DE EVALUACIÓN:

El sistema de evaluación a implementar estará basado en los siguientes aspectos:

¿Qué evaluar? Las competencias, criterios de verificación y evidencias.

¿Cómo evaluar? Según las Evidencias: **De conocimiento** (pruebas escritas, orales); **de hacer** (Mapas Mentales, mapas conceptuales, simulaciones,); **de ser y convivir** (Actitud proactiva, responsabilidad, puntualidad, sociodramas, autovaloración y covaloración); **de producto** (Informes finales, portafolios, investigaciones, Matrices de evaluación, etc.)

¿Cuándo evaluar? Según el Momento:

- **Evaluación Diagnóstica:** Con el propósito de evaluar los conocimientos previos, se realizará una evaluación antes de empezar cada clase para determinar si los estudiantes comprendieron el avance de la materia.
- **Evaluación Formativa:** Para evaluar las evidencias y generar la retroalimentación (metacognición).
- **Evaluación de Promoción:** Con el fin de determinar el nivel de aprendizajes, en base al logro de las competencias y su comunicación.

¿Quién debe evaluar en el proceso?

- **Auto-valoración:** Realizado por los mismos estudiantes, tiene un carácter metacognitivo.

- **Co-valoración:** Realizado por los demás estudiantes, o equipo de trabajo.
- **Hetero-valoración:** Realizado por el docente, con el propósito de determinar el nivel de dominio de los aprendizajes, según criterios de verificación, y demostración de evidencias.

- **Escala Valorativa Cuantitativa**

ESCALA CONCEPTUAL	ESCALA NUMÉRICA SOBRE 100
A (Aprobado)	Entre 85 y 100 Puntos: Estratégico Entre 70 y 84 Puntos: Autónomo Entre 69 y 51 Puntos: Resolutivo
R (Reprobado)	Entre 25 y 50 Puntos: Receptivo Entre 1 y 24 puntos: Preformal

- **Matrices de Evaluación de Competencias:**

- **Matriz 1**

MICROCOMPETENCIA: Escribe programas de aplicaciones web con acceso a bases de datos para plantear soluciones a problemas del contexto en base a lenguajes de programación del lado del servidor considerando técnicas de optimización de algoritmos y paradigmas de programación.		VALORACIÓN 30
NIVELES DE DOMINIO	CRITERIOS DE VERIFICACIÓN: Aplica la POO en la creación de una pequeña aplicación en base a el lenguaje de programación PHP.	
Pre-formal (1-7)	Identifica las características del lenguaje PHP	
Receptivo (8-14)	Reconoce la sintaxis del lenguaje de programación PHP	
Resolutivo (o básico) (15-19)	Utiliza las estructuras de control del lenguaje de programación PHP para realizar programas	
Autónomo (20-25)	Implementa programas utilizando el paradigma de la POO en el lenguaje PHP	
Estratégico (26-30)	Crea aplicaciones pequeñas utilizando POO con código óptimo, limpio y claro facilitando su reutilización.	
Total		
Logros Alcanzados:		
Dificultadas encontradas:		
Acciones para mejorar:		

Matriz 2

Micro Competencia: Escribe programas de aplicaciones web con acceso a bases de datos para plantear soluciones a problemas del contexto en base a lenguajes de programación del lado del servidor considerando técnicas de optimización de algoritmos y paradigmas de programación.		Valoración 40
NIVELES DE DOMINIO	CRITERIOS DE VERIFICACIÓN: Crea soluciones webs conectadas a gestores función a base de datos mostrando información veraz y concisa.	
Pre-formal (1-10)	Identifica las diferentes maneras de conectarse a una base de datos usando PHP	
Receptivo (11-20)	Reconoce el manejo de la base de datos MySQL	
Resolutivo (o básico) (21-29)	Describe la forma de administración del servidor que soporta la base de datos MySQL y efectúa consultas a la BD	
Autónomo (30-34)	Implementa soluciones básicas que realizan operaciones de gestión de la base de datos.	
Estratégico (35-40)	Implementa soluciones completas de gestión de la base de datos considerando niveles de seguridad y restricción de usuarios.	
Logros Alcanzados:		
Dificultadas encontradas:		
Acciones para mejorar:		

- Matriz 3

Micro Competencia: Escribe programas de aplicaciones web con acceso a bases de datos para plantear soluciones a problemas del contexto en base a lenguajes de programación del lado del servidor considerando técnicas de optimización de algoritmos y paradigmas de programación.		Valoración 30
NIVELES DE DOMINIO	CRITERIOS DE VERIFICACIÓN: Implementa servicios web en concordancia a la tecnología AJAX para descargar información.	
Pre-formal (1-7)	Identifica los servicios web, su utilidad y tecnologías asociadas	
Receptivo (8-14)	Comprende la forma de leer y generar resultados en XML y JSON.	
Resolutivo (o básico) (15-19)	Comprende la lógica de las aplicaciones Cliente para consumo de datos y la lógica de las interfaces Servidor para proveer datos.	

Autónomo (20-25)	Maneja la comunicación con AJAX en la implementación de servicios web.	
Estratégico (26-30)	Implementa soluciones completas CRUD utilizando tecnología AJAX y servicios web.	
Logros Alcanzados:		
Dificultadas encontradas:		
Acciones para mejorar:		

V. BIBLIOGRAFÍA

BÁSICA:

Santiago Rodríguez de la Fuente, Fernando Pérez et al - Thompson (2009). Programación de Aplicaciones Web. Ed. Paraninfo

Beati, Hernán, PHP (2016)- Creación de páginas Web dinámicas 2a ed. Alfaomega

Sklar, D. and Trachtenberg, A. (2014). PHP Cookbook: Solutions & Examples for PHP Programmers. Edit. O'Reilly Media.

Uceda O. C. (2013). Desarrollo Web con PHP: Aprende PHP paso a paso. Edit. ocapunay.

Patel, S.K. (2014). Developing Responsive Web Applications with AJAX and jQuery. Ed.

Packt Publishing Ltd. Birmingham, Inglaterra.

Zea, R. (2015) Mastering Responsive Web Design. Ed. Packt Publishing Ltd. Birmingham, Inglaterra

COMPLEMENTARIA:

Javier Eguíluz Pérez (2008) Introducción a AJAX – Ed. IT Campus Academy

Gonzalo, M.A. (2013). Responsive Design. Desarrolla webs sensitivas con Bootstrap. Ed.

IT Campus Academy.