



Universidad Guadalajara
Centro Universitario del Sur

Programa de Estudio

1. IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE

División

Ciencias Exactas, naturales y tecnológicas

Departamento

Ciencias computacionales e innovación tecnológica

Academia

Programación

Programa(s) educativo(s)

Licenciatura en Ingeniería en Telemática

Denominación de la unidad de aprendizaje:

Programación Web II

Clave de la materia:	Horas de teoría:	Horas de práctica:	Carga horaria global:	Valor en créditos:
IG187	40	40	80	8

Tipo de curso:		Nivel en que se ubica:	Prerrequisitos:
C = curso		Técnico Medio	Programación Web I
CL = curso laboratorio		Técnico Superior	
L = laboratorio		Universitario	
P = práctica		Licenciatura	
T = taller		Especialidad	
CT = curso – taller	X	Maestría	
N = clínica		Doctorado	
M = módulo			
S = seminario			

Área de formación:

BÁSICA PARTICULAR OBLIGATORIA

Perfil docente:

1. Domina y estructura los saberes para facilitar experiencias de aprendizaje significativo.
2. Planifica los procesos de enseñanza y de aprendizaje atendiendo al enfoque por competencias, y los ubica en contexto disciplinares, curriculares y sociales.
3. Lleva a la práctica procesos de enseñanza y de aprendizaje de manera efectiva, creativa e innovadora.
4. Evalúa los procesos de enseñanza y aprendizaje con un enfoque formativo.

5. Construye ambientes para el aprendizaje autónomo y colaborativo.
6. Contribuye a la generación de ambiente que facilite el desarrollo sano.
7. Profesionistas con grado mínimo de licenciatura en áreas afines a las ciencias computacionales.
8. Tener experiencia y vocación docente de al menos dos años en el nivel superior.

Elaborado por:

Actualizado por:

Mtro. Gerardo Jiménez Haro

Mtro. Gerardo Jiménez Haro

Fecha de elaboración:

Fecha de última
actualización:

Fecha de última
evaluación:

Fecha de aprobación por
Colegio Departamental:

Enero 2023

Junio 2023

Junio 2023

2. PRESENTACIÓN DE LA UNIDAD DE APRENDIZAJE

Con el actual auge de internet y lo imprescindible que se ha vuelto el tener conocimientos acerca de la programación de sitios para internet, la presente asignatura de programación web II, da continuidad a la materia precedente de programación web I, centrándose en la programación del lado del servidor o también denominada como el **backend**.

Además, esta asignatura precede a la materia de ingeniería de software, por lo que es la penúltima de las asignaturas del área de programación, aportando al perfil del Ingeniero en Telemática las competencias profesionales necesarias para la creación de sitios web para internet haciendo uso de tecnologías del lado del servidor y del lado del cliente.

Adicionalmente, esta materia se complementa con las materias de bases de datos I y bases de datos distribuidas en el sentido que toda la información que se maneja en internet requiere de ser almacenada, procesada y distribuida, lo que conlleva unos procesos de base de datos y el uso de un navegador para realizar determinadas tareas o mostrar información.

3. OBJETIVO GENERAL/COMPETENCIA

Crear sitios web para internet haciendo uso de tecnologías del lado del servidor y del lado del cliente a través de plataformas de aprendizaje virtual y de materiales instruccionales en relación con problemáticas del entorno.

4. CAMPO DE APLICACIÓN PROFESIONAL DE LOS CONOCIMIENTOS

Aplicar eficazmente herramientas y utilidades para el desarrollo de aplicaciones web, conforme a las tecnologías tanto del lado del cliente como del lado del servidor, tanto en el ámbito público como privado.

5. SABERES:

Prácticos	<ul style="list-style-type: none"> • Identificar las características y comportamientos de las clases de un sistema. • Identificar e implementar en el diseño los elementos clave (clase, herencia, polimorfismo, sobrecarga de operadores) de la programación orientada a objetos. • Diseñar los elementos estáticos de un sistema. • Diseñar los elementos dinámicos de un sistema. • Implementar los diseños anteriores en un lenguaje de programación.
------------------	--

Teóricos	<ul style="list-style-type: none"> • Definir e identificar conceptos fundamentales de la programación orientada a objetos. • Identificar el modelado de objetos para el diseño e implementación de la solución de un problema computable. • Identificar elementos para el diseño UML de elementos estáticos y dinámicos de un programa orientado a objetos. • Identificar palabras reservadas de un lenguaje de programación para implementar los diseños orientados a objetos.
Formativos	<ul style="list-style-type: none"> • Abstracción y análisis orientada a objetos. • Fomento del trabajo colaborativo. • Fomento de autoaprendizaje. • Descubrir la necesidad de un diseño de la solución antes de la implementación en un lenguaje de programación.

6. CONTENIDO TEMÁTICO (TEÓRICO-PRÁCTICO)

1. Programación del lado del servidor

- 1.1. Introducción a la programación del lado del servidor.
- 1.2. PHP.
 - 1.2.1. Variables.
 - 1.2.2. Operadores.
 - 1.2.3. Estructuras de control y condición.
 - 1.2.4. Funciones.
 - 1.2.5. Matrices.
 - 1.2.6. Manejo de sesiones y cookies.
- 1.3. PHP y MySQL.
 - 1.3.1. Instalación y configuración del gestor o emulador.
 - 1.3.2. Conexiones a la base de datos.
 - 1.3.3. Funciones básicas de base de datos (Inserción, búsqueda, modificación y eliminación)

2. Gestores de contenido (CMS)

- 2.1. Introducción a los CMS.
 - 2.1.1. Tipos de CMS (Wikis, ecommerce, genéricos).
- 2.2. Requisitos previos a la instalación de un CMS.
 - 2.2.1. Identificación de contenidos.
 - 2.2.2. Definición de estructura.
 - 2.2.3. Recopilación de información.
- 2.3. Instalación y puesta a punto de un CMS.
- 2.4. Administración del CMS.
 - 2.4.1. Instalación de complementos.
 - 2.4.2. Administración de usuarios, perfiles, grupos y permisos.
 - 2.4.3. Respaldos del sitio.
 - 2.4.4. Migración y recuperación.

3. Posicionamiento web

- 3.1. Aspectos generales del posicionamiento web.
- 3.2. Herramientas y utilidades para el posicionamiento web.
- 3.3. Optimización.
- 3.4. Medición de resultados.

4. Seguridad en aplicaciones web

- 4.1. Elementos básicos de seguridad.
 - 4.1.1. Seguridad en el cliente.
 - 4.1.2. Seguridad en el servidor.
 - 4.1.3. Seguridad en la aplicación.

- 4.1.4. Seguridad en la comunicación.
- 4.1.5. Pruebas básicas de seguridad en aplicaciones web.

7. ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE

Modalidades:

- Aprendizaje colaborativo.
- Aprendizaje autogestivo.
- Aprendizaje basado en problemas.
- Lección magistral.
- Uso de la plataforma classroom como espacio de apoyo virtual.

Método:

- Trabajo en Grupo.
- Trabajo Individual.
- Plenarias.
- Clases prácticas en laboratorio de cómputo.
- Clases teóricas en aula de clases.

8. EVALUACIÓN DEL APRENDIZAJE

8.1. Evidencias de aprendizaje	8.2. Criterios de desempeño
Investigación y lecturas del contenido teórico práctico.	Entregar el tema(s) de investigación en forma electrónica o impresa según indique el profesor. La investigación debe contener lo siguiente: Título, índice, introducción, objetivos, desarrollo del tema, conclusiones y bibliografía y en caso de haber consultado en internet webgrafía.
Examen parcial.	Los exámenes parciales se realizarán escritos (teórico práctico) donde deberán contestar las preguntas y realizar los ejercicios de forma clara de acuerdo con las especificaciones en el examen.
Exposición de clase	El alumno desarrollara un tema en específico donde presentara el título del tema, desarrollo del tema (conceptos, ejercicios prácticos), síntesis y conclusiones ya sea de forma individual o en equipo.
Realización de prácticas.	Realizar la codificación de algoritmos y el corrimiento de estos. Dar solución a problemas computables. Entregar la práctica en electrónico y el corrimiento por escrito ambos sin excepción alguna.
Recopilación de prácticas y programas.	Dar solución a problemas especificados por el profesor de acuerdo a las instrucciones dadas. A lo largo del semestre el alumno reunirá en forma electrónica todas las prácticas y programas elaborados incluyendo archivos de reportes, archivos de programas y proyectos de cierre de unidad final. Reunirá los archivos en forma ordenada utilizando carpetas con el nombre y el número de práctica o número de programa.

9. CALIFICACIÓN

Examen teórico prácticos	25 puntos
Tareas y Actividades prácticas	50 puntos
Producto final	20 puntos
Actividades de formación integral	5 puntos
Total	100 puntos

Para acreditar la formación integral deberá comprobar al menos una de las siguientes condiciones:

- Tres constancias de actividades extra - escolares.
- Un taller deportivo o artístico.

El 5% de la formación integral sólo se contabilizará en caso de que la suma de la calificación de las otras actividades sea de 60 o más.

10. ACREDITACIÓN

<p>Periodo ordinario. De conformidad con el artículo 20 del Reglamento General de Evaluación y Promoción de Alumnos de la Universidad de Guadalajara, para que el alumno tenga derecho al registro del resultado final de la evaluación en el periodo ordinario, establecido en el calendario escolar aprobado por el Consejo General Universitario, se requiere:</p> <ol style="list-style-type: none">Estar inscrito en el plan de estudios y curso correspondiente, yTener un mínimo de asistencia del 80% a clases y actividades registradas durante el curso.	<p>Periodo extraordinario. De conformidad con el artículo 27 del Reglamento General de Evaluación y Promoción de Alumnos de la Universidad de Guadalajara, para que el alumno tenga derecho al registro de la calificación en el periodo extraordinario, se requiere:</p> <ol style="list-style-type: none">Estar inscrito en el plan de estudios y curso correspondiente.Haber pagado el arancel y presentar el comprobante correspondiente.Tener un mínimo de asistencia del 65% a clases y actividades registradas durante el curso. <p>Se exceptúan de este caso las materias de orden práctico que requerirán la repetición del curso (Art. 23 RGEYPA).</p>
--	---

11. BIBLIOGRAFÍA

BIBLIOGRAFÍA BÁSICA

- Arlow, J., Neustadt, I. (2006). UML 2. Madrid: Anaya Multimedia.
- Arnow, D. (2001). Introducción a la programación con JAVA. Un enfoque orientado a objetos. España: Pearson education.
- Booch, G. Rumbaugh, J. Jacobson, I. (1999). Madrid: Addison Wesley Iberoamericana.
- Cadenhead, R. (2012). Java 7. Madrid: Anaya Multimedia.
- Durán, F., Gutierrez, F., Pimentel, E. (2007). Programación orientada a objetos con java. Madrid: Thomson.
- Fain, Y. (2011). Programación Java. Madrid: Anaya Multimedia.
- García, J., Ortín, F. (2010). Programación orientada a objetos avanzada. Santiago de Compostela: Andavira.

- Joyanes, L. Zahonero, I. (2010). Programación en C, C++, Java y UML. México: McGraw Hill/Interamericana.
- Joyanes, J. (1998). Programación orientada a objetos. Madrid: McGraw-Hill.
- Joyanes, J. (2008). Fundamentos de programación algoritmos, estructura de datos y objetos. Madrid: McGraw-Hill/Interamericana de España.
- Larman, C. (1999). UML y patrones introducción al análisis y diseño orientado a objetos. México: Alfaomega.
- López, L. (2011). Programación estructurada y orientada a objetos un enfoque algorítmico. México: Alfaomega.
- López, L. (2008). Metodología de la programación orientada a objetos. México: Alfaomega.
- Rumbaugh, J. (1997). Modelado y diseño orientado a objetos. España: Prentice Hall Internacional.
- Moldes, F. (2011). Java 7. Madrid: Anaya Multimedia.
- Meyer, B. (1999). Construcción de software orientado a objetos. Madrid: Prentice Hall Iberia.
- Piza, H. (2008). Aprendiendo java un enfoque orientado a objetos. México: Universidad de Guadalajara Centro Universitario de los Valles.
- Pérez, J. (2006). Problemas resueltos de programación en lenguaje Java. Madrid: International Thomson Editores Spain.
- Sines, A. (2002). Aprendiendo programación orientada a objetos en 21 lecciones avanzadas. México Pearson.
- Soroka, B. (2006) Java 5 objects first. Sudbury, Massachusetts Jones and Bartlett.
- Wang, P. (2000). Java con programación orientada a objetos y aplicaciones. México: International Thomson Editores.
- Wu, Thomas. (2008). Programación en Java introducción a la programación orientada a objetos. México: McGraw-Hill-Interamericana.

BIBLIOGRAFÍA COMPLEMENTARIA

- Cairó, O. (1995). Metodología de la programación algoritmos, diagramas de flujo y programas. México: Alfaomega.
- Braude, E. (2003). Ingeniería de software una perspectiva orientada a objetos. México: Alfaomega.
- Burd, B. (2005). Beginning programming with java for dummies. Indianapolis: Wiley Publishing.
- Charte, O. (2005). Introducción a la programación. Madrid: Anaya Multimedia.
- Cowell, J. (1997). Essential java fast how to write object oriented software for the internet. London: Springer.
- Dean, J., Dean Raymond. (2009). Introducción a la programación con JAVA. México McGraw-Hill/Interamericana Editores.
- Deitel, J., Deitel, M. (2008). Cómo programar en Java. México: Pearson Prentice Hall.

12. RECURSOS COMPLEMENTARIOS (páginas web, mooc's, plataformas, objetos de aprendizaje)

Espacio en classroom:

<https://classroom.google.com/c/NjE0OTc3NzEzODQz?cjc=lx6s43a>

Firma:

Vo. Bo.

Mtro. Gerardo Jiménez Haro

Presidente de Academia

Dr. Jorge Lozoya Arandia

Jefe de Departamento