

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

COORDINACIÓN GENERAL DE FORMACIÓN BÁSICA  
COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA  
PROGRAMA DE UNIDAD DE APRENDIZAJE

## I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Mexicali; Facultad de Ingeniería, Arquitectura y Diseño, Ensenada y Facultad de Ciencias Químicas e Ingeniería, Tijuana.
- 2. Programa Educativo:** Ingeniero en Computación
- 3. Plan de Estudios:** 2020-1
- 4. Nombre de la Unidad de Aprendizaje:** Análisis y Diseño de Sistemas
- 5. Clave:** 36289
- 6. HC: 00 HL: 02 HT: 02 HPC: 00 HCL: 00 HE: 00 CR: 04**
- 7. Etapa de Formación a la que Pertenece:** Disciplinaria
- 8. Carácter de la Unidad de Aprendizaje:** Obligatoria
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



### Equipo de diseño de PUA

Haydeé Meléndez Guillén  
Thelma Violeta Ocegueda Miramontes

### Vo.Bo. de Subdirectores de Unidades Académicas

Alejandro Mungaray Moctezuma  
Humberto Cervantes de Ávila  
Rocío Alejandra Chávez Santoscoy

**Fecha:** 17 de octubre de 2019

## **II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE**

El análisis y diseño de sistemas es relevante debido a que proporciona los conocimientos necesarios y establece los lineamientos a seguir para llevar a cabo un proceso de desarrollo de sistemas de software-hardware de calidad.

El alumno adquirirá conocimientos sobre metodologías de análisis y diseño de sistemas, así como las herramientas necesarias para el desarrollo de sistemas; y habilidades para el análisis y co-diseño de software-hardware; fortaleciendo la actitud analítica, trabajo colaborativo, el liderazgo y la innovación.

Esta asignatura es obligatoria de la etapa disciplinaria y corresponde al área de conocimiento Diseño en Ingeniería. Es recomendable haber acreditado los cursos de Ingeniería de Procesos, Ingeniería de Requerimientos y Sistemas de Información previamente.

## **III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE**

Desarrollar un modelo de sistema que sea robusto y consistente, mediante la aplicación de técnicas y herramientas establecidas para el modelado de sistemas software/hardware, para contribuir a la calidad del proceso de desarrollo de sistemas software/hardware y a la calidad del producto final, con creatividad, proactividad y actitud colaborativa.

## **IV. EVIDENCIA(S) DE DESEMPEÑO**

Entrega un documento que contenga el análisis y diseño de un sistema de software/hardware, este documento debe contener portada, índice, introducción, justificación, desarrollo, conclusiones y bibliografía, presenta frente al grupo de forma creativa y con apoyo de medios audiovisuales.

## V. DESARROLLO DE CONTENIDOS

1. Fundamentos del análisis y diseño de sistemas.
2. Lenguaje Unificado de Modelado (UML)
3. Arquitecturas
4. Codiseño hardware-software

## VI. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

No. de Práctica	Competencia	Descripción	Material de Apoyo	Duración
1	Distinguir las diferencias entre los diversos tipos de sistemas, mediante el análisis de sus características, uso, ventajas y desventajas, para identificar la utilidad de cada tipo de sistema con una actitud crítica y reflexiva.	<p>El docente explica los fundamentos del análisis y diseño de sistemas.</p> <p>El alumno elabora una técnica de aprendizaje como un cuadro comparativo donde logre identificar las características, uso, ventajas y desventajas de los tipos de sistemas que existen.</p>	Computadora, internet, libros, pintarrón, proyector y plumones.	8 horas
2	Analizar los diferentes tipos de diagramas con los que cuenta UML, mediante el análisis de sus características, para comprender la utilidad de cada tipo de diagrama en el proceso de diseño de un sistema, con una actitud analítica y de colaboración.	<p>El docente explica los diferentes tipos de diagramas que existen en UML, y propone un sistema como ejemplo para que sea representado mediante el uso de diagramas UML.</p> <p>El alumno elabora una técnica de aprendizaje como un cuadro comparativo donde identifique las características de cada tipo de diagrama. Además, desarrolla un pequeño modelo del sistema propuesto por el docente, mediante el uso de diagramas UML.</p>	Computadora, internet, libros, pintarrón, proyector y plumones.	8 horas
3	Revisar los tipos de arquitecturas existentes, mediante el análisis de sus características, con la finalidad de seleccionar la apropiada para implementarla en un sistema propuesto por el docente, con una actitud reflexiva, proactiva y colaborativa.	<p>El docente explica los diferentes tipos de arquitecturas y sus características. Además, propone un sistema de ejemplo para que el alumno trabaje con éste.</p> <p>El alumno analiza los tipos de arquitecturas y selecciona uno de</p>	Computadora, internet, libros, pintarrón, proyector y plumones.	8 horas

		estos para aplicarlo en el sistema propuesto por el docente.		
4	Examinar las metodologías y herramientas de codiseño hardware-software, mediante el análisis y discusión de sus características, para identificar las técnicas apropiadas a implementar en un sistema propuesto por el docente, con una actitud innovadora, colaborativa y de respeto.	<p>El docente explica los conceptos básicos de codiseño hardware-software, así como sus metodologías de diseño y las herramientas de desarrollo que se utilizan en su implementación. Además, proporciona ejemplos para que el alumno trabaje con ellos.</p> <p>El alumno analiza las metodologías de codiseño hardware-software y sus herramientas de desarrollo, y mediante una discusión de grupo selecciona la metodología y la herramienta adecuada para utilizar para cada ejemplo que el docente proporciona.</p>	Computadora, internet, libros, pintarrón, proyector y plumones.	8 horas

## VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No. de Práctica	Competencia	Descripción	Material de Apoyo	Duración
1	Analizar las diferencias entre los diversos tipos de sistemas, mediante el análisis de sus características, uso, ventajas y desventajas, para elegir un tipo de sistema a analizar y diseñar, con una actitud crítica, reflexiva y de compañerismo.	El alumno selecciona en equipo, a través de una lluvia de ideas y la diferenciación de los tipos de sistemas y sus características, un sistema para desarrollar en el semestre, y elabora una descripción detallada del mismo.	Computadora, internet, libros.	8 horas
2	Aplicar los diferentes diagramas con los que cuenta UML, mediante el análisis de sus características y el tipo de sistema a realizar, para crear un modelo del sistema seleccionado, con una actitud analítica, creativa y de colaboración.	El alumno, junto con su equipo, selecciona un conjunto de diagramas que considere útiles para diseñar el tipo de sistema que seleccionó en la unidad anterior y procede a realizar el modelo del mismo.	Computadora, internet, libros.	8 horas
3	Elegir un tipo de arquitectura, mediante el análisis de las características de los mismos y las del sistema seleccionado, para modelar la arquitectura del sistema, con una actitud reflexiva, proactiva y colaborativa.	El alumno, junto con su equipo, selecciona un tipo de arquitectura acorde a las características deseadas y lo aplica a el sistema que seleccionó para diseñar.	Computadora, internet, libros.	8 horas
4	Seleccionar la metodología y las herramientas de codiseño hardware-software, mediante el análisis y discusión de sus características, para aplicarlas en la implementación del sistema seleccionado, con una actitud innovadora, colaborativa y de respeto.	El alumno, junto con su equipo, selecciona la metodología de codiseño hardware/software y las herramientas a utilizar considerando el análisis de sus características para implementar el sistema que seleccionó.	Computadora, internet, libros.	8 horas

## VII. MÉTODO DE TRABAJO

**Encuadre:** El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

### **Estrategia de enseñanza (docente)**

- Explica conceptos y ejemplos.
- Propiciar el trabajo en equipo
- Proporcionar las referencias bibliográficas
- Elaborar y aplicar evaluaciones

### **Estrategia de aprendizaje (alumno)**

- Leer, analizar e investigar.
- Trabajar en equipo con los compañeros de clase.
- Desarrollar la evidencia de desempeño.
- Desarrollar diversas estrategias de aprendizaje como cuadros comparativos, lluvias de ideas, trabajo colaborativo, y exposiciones.

## VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

### Criterios de acreditación

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

### Criterios de evaluación

- Evaluaciones ..... 20%
  - Prácticas ..... 20%
  - Tareas ..... 20%
  - Evidencia de desempeño..... 40%  
(Documento con el análisis y diseño  
de un sistema de software/hardware)
- Total..... 100%**

## IX. REFERENCIAS

Básicas	Complementarias
<p>Dennis, A., Wixom, B. H., y Roth, R. M. (2018). <i>Systems Analysis and Design</i>. Estados Unidos: Wiley.</p> <p>Dennis, A., Wixom, B. H., y Tegarden, D. (2015). <i>Systems analysis and design: An object-oriented approach with UML</i>. Estados Unidos: John Wiley &amp; Sons.</p> <p>Ha, S., y Teich, J. (2017). <i>Handbook of hardware/software codesign</i>. Holanda: Springer Publishing Company, Incorporated.</p> <p>Jiménez de Parga, C. (2015). <i>UML. Aplicaciones En Java Y C++</i>. España: Ra-Ma.</p> <p>Rajaraman, V. (2018). <i>Analysis and design of information systems</i>. PHI Learning Pvt. Ltd.</p> <p>Tilley, S., y Rosenblatt H. J. (2016). <i>Systems Analysis and Design (Shelly Cashman Series)</i>. Estados Unidos: Cengage Learning</p>	<p>Balarin, F., Giusto, P., Jurecska, A., Chiodo, M., Hsieh, H., Passerone, C., y Suzuki, K. (1997). <i>Hardware-software co-design of embedded systems: the POLIS approach</i>. Estados Unidos: Springer Science &amp; Business Media. [clásica]</p> <p>Chapman, W. (2018). <i>Engineering modeling and design</i>. Routledge.</p> <p>Davis, W. S. y Yen, D. C. (2018). <i>The information system consultant's handbook: Systems analysis and design</i>. CRC press.</p> <p>DeMicheli, G. y Sami, M. G. (Eds.). (2013). <i>Hardware/software Co-design (Vol. 310)</i>. Estados Unidos: Springer Science &amp; Business Media. [clásica]</p> <p>Martín, J. y López, L. (2014). <i>UML práctico: aprende UML paso a paso</i>. Edición Kindle.</p> <p>Miles, R., y Hamilton, K. (2008). <i>Learning UML 2.0</i>. Estados Unidos: O'Reilly Media, Inc. [clásica]</p> <p>Niemann, R. (1998). <i>Hardware/software co-design for data flow dominated embedded systems</i>. Estados Unidos: Springer Science &amp; Business Media. [clásica]</p> <p>Schaumont, P. R. (2012). <i>A practical introduction to hardware/software codesign</i>. Estados Unidos: Springer Science &amp; Business Media. [clásica]</p> <p>Staunstrup, J. y Wolf, W. (2013). <i>Hardware/software co-design: principles and practice</i>. Estados Unidos: Springer Science &amp; Business Media. [clásica]</p>

## **X. PERFIL DEL DOCENTE**

El docente que imparta esta asignatura debe contar con título en Ingeniero en Computación o de alguna otra carrera afín, preferentemente con posgrados en Ingeniería o Ciencias de Computación. Se sugiere experiencia laboral y docente mínima de tres años. Debe tener facilidad de palabra, dirigir proyectos, fomentar la innovación, creatividad y análisis, además deber responsable y tener interés para actualizarse constantemente tanto en el área disciplinar como en la pedagogía.