



1.- Identificación de la Unidad de Aprendizaje

Nombre de la Unidad de Aprendizaje					
Programación Orientada a Objetos					
Clave de la UA	Modalidad de la UA	Tipo de UA		Valor de créditos	Área de formación
IH051	Presencial	Curso taller		8	Básica Común Obligatoria
Hora semana		Horas teoría/semestre	Horas práctica/semestre	Total de horas:	Seriación
4		40	40	80	F0092
Departamento			Academia		
Cs. de la Información y Desarrollos Tecnológicos			Ingeniería de Software		
Presentación					
<p>La programación orientada a objetos permite simular escenarios de la realidad, por lo cual al cursarla el estudiante será capaz de modelar clasificaciones de objetos con características de entidades del mundo real, los cuales se podrán relacionar y comunicar para trabajar cooperativamente y así lograr un fin, lo que facilita la construcción de sistemas complejos a partir de componentes y su desarrollo en equipo.</p> <p>Las clases se podrán organizar con relaciones de herencia o jerarquía, agregación, asociación y uso entre ellas. Al definir la tipificación o clase se puede echar mano de la reutilización de código ya que se podrán instanciar tantas entidades u objetos de una u otra clase como se requieran. Modelar una clase representativa de un objeto real implica consideraciones de polimorfismo en su comportamiento, permitiendo que objetos de distintas clases responda de manera congruente a su naturaleza a un mismo llamado o mensaje, Por lo que el programador deberá desarrollar habilidades de abstracción (capturar características y comportamiento de un objeto) y encapsulamiento (ocultar y proteger información del objeto) para definir las clases y sus relaciones, en un contexto cooperativo con sus compañeros.</p>					
Tipos de saberes					
Saber (Conocimientos)	Saber hacer (Habilidades)		Saber ser (Actitudes y valores)		
Elementos esenciales del entorno de desarrollo del lenguaje de programación.	Abstrae y sintetiza escenarios o problemáticas de una realidad.		Hace trabajo colaborativo		
Declaración de datos primitivos y sus operadores	Aplica los conocimientos teóricos en la práctica.		Compromiso en su desempeño.		
Control del flujo de ejecución.	Concibe, diseña y desarrolla soluciones informáticas en el lenguaje orientado a objetos propuesto en clase.		Hace trabajo colaborativo		
			Receptivo y empático para conocer los distintos tipos de usuario (personas de la 3ª edad, niños, con capacidades diferentes, con		



<p>Manejo de conjuntos de datos</p> <p>Generación y manejo de la documentación.</p> <p>Análisis y diseño orientado a objetos.</p> <p>Diseño en el Lenguaje Unificado de Modelado</p> <p>Diseña y construye soluciones informáticas explotando las ventajas de las distintas relaciones que pueden crear entre objetos.</p> <p>Aplica control de excepciones a las aplicaciones desarrolladas evitando así rupturas en tiempo de ejecución.</p> <p>Construye aplicaciones con interfaces de usuario gráficas y por eventos.</p> <p>Respalda información para su posterior uso en memoria secundaria.</p>	<p>Capacidad creativa e innovadora en el proceso de diseño y desarrollo de aplicaciones.</p> <p>Organización y planificación.</p> <p>Diseña e implementa soluciones con base en el paradigma orientado a objetos.</p> <p>Diseña e implementa soluciones validadas a entradas inesperadas.</p> <p>Trabajar en equipo.</p> <p>Proactivo y organizado.</p> <p>Aplica conocimientos de otras disciplinas con el objetivo de crear aplicaciones con un alto grado de utilidad y usabilidad dirigido a un tipo de usuario o público específico.</p>	<p>problemas de aprendizaje) que podría tener la aplicación informática.</p>
Competencia genérica	Competencia profesional	
<p>Realiza trabajo de manera individual y en equipo de manera interdisciplinaria y multidisciplinaria.</p> <p>Aplica herramientas de programación y principios de las ciencias computacionales en el modelado y diseño de sistemas de información.</p>	<p>Sintetiza escenarios y problemáticas para desarrollar soluciones en trabajo colaborativo en el diseño e implementación de sistemas, así como organización y planificación del tiempo.</p> <p>Desarrolla y opera soluciones informáticas basándose en principios de ingeniería y estándares de calidad para optimizar procesos y soluciones.</p>	



	<p>Desempeña diferentes roles en proyectos informáticos para desarrollar habilidades y liderazgo en trabajos colaborativos.</p> <p>Aplica fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación para simplificar el proceso en la modelación y diseño de soluciones informáticas.</p> <p>Identifica oportunidades a través del uso eficiente y eficaz de soluciones informáticas, con responsabilidad y compromiso social, para mejorar el desempeño de las organizaciones.</p>
Saberes previos del alumno	
El alumno para llevar ésta unidad de aprendizaje deberá contar con conocimientos como son: lógica de programación, programación modular o por funciones, estructuras de control del flujo, abstraer y sintetizar problemáticas, trabajo colaborativo.	
Perfil de egreso al que se abona	
Desarrollo de sistemas de información y soluciones a problemas, habilidad de trabajo individual y en equipo, capacidad de análisis y diseño de soluciones a problemas siguiendo distintas metodologías.	
Perfil deseable del docente	
Formación profesional.	
El docente deberá contar preferentemente con un posgrado y/o al menos con una licenciatura afín al área de cómputo, deberá tener amplia experiencia en desarrollo de software orientado a objetos, además de actualizarse periódicamente en las versiones del lenguaje utilizado en la materia.	
Habilidades.	
Experiencia en: Elaboración de planeaciones didácticas, Diseño y evaluación de estrategias y actividades de aprendizaje, Uso de tecnologías de la información y la comunicación.	

2.- Contenidos temáticos

Contenido
1. Introducción al lenguaje 1.1 Entorno en general 1.2 Tipos primitivos, variables 1.3 Operadores básicos 1.4 Instrucciones condicionales 1.5 Control del flujo de ejecución 1.6 Arreglos 1.7 Documentación



2. Principios de la Orientación a objetos

- 2.1 Objetos y Clases
- 2.2 Métodos
- 2.3 Constructores
- 2.4 Instancias
- 2.5 Llamadas a métodos
- 2.6 Referencias y Valores
- 2.7 Análisis orientado a objetos con UML

3. Relaciones entre objetos

- 3.1 Control de acceso
- 3.2 Ámbito de una clase
- 3.3 Agregación
- 3.4 Herencia
- 3.5 Paquetes
- 3.6 Java API (ArrayList, Sets (TreeSet, HashSet), Maps)
- 3.7 Control de excepciones
- 3.8 Modificador final
- 3.9 Modificador abstract

4. Aplicaciones gráficas, eventos y archivos

- 4.1 Interfaces gráficas
- 4.2 Debugin
- 4.3 Interfaces Listener, clase Adapter
- 4.6 Manejo de archivos
- 4.7 Serialización de objetos

Estrategias generales para impartir la unidad de aprendizaje

En esta unidad de aprendizaje se pueden aplicar con éxito las estrategias de resolución de problemas, aprendizaje orientado a proyectos, estudio de casos, simulando y modelando situaciones reales.

Módulo I

Introducción al lenguaje de programación

Competencia Específica

Sintetiza escenarios y problemáticas para desarrollar soluciones en trabajo colaborativo en el diseño e implementación de sistemas, así como organización y planificación del tiempo.

Tipos de saberes

Saber (Conocimientos)	Saber hacer (Habilidades)	Saber ser (Actitudes y valores)
Elementos esenciales del entorno de desarrollo del lenguaje de programación.	Abstrae y sintetiza escenarios o problemáticas de una realidad.	Hace trabajo colaborativo



Declaración de datos primitivos y sus operadores	Aplica los conocimientos teóricos en la práctica.	Muestra una actitud receptiva a nuevos entornos de programación.
Control del flujo de ejecución.	Concibe, diseña y desarrolla soluciones informáticas en el lenguaje orientado a objetos propuesto en clase.	
Manejo de conjuntos de datos		
Generación y manejo de la documentación.		
Módulo II		
Principios de la Orientación a Objetos		
Competencia Específica		
Desarrolla y opera soluciones informáticas basándose en principios de ingeniería y estándares de calidad para optimizar procesos y soluciones.		
Desempeña diferentes roles en proyectos informáticos para desarrollar habilidades y liderazgo en trabajos de equipo.		
Tipos de saberes		
Saber (Conocimientos)	Saber hacer (Habilidades)	Saber ser (Actitudes y valores)
Análisis y diseño orientado a objetos. Diseño en el lenguaje unificado de modelado.	Capacidad creativa e innovadora en el proceso de diseño y desarrollo de aplicaciones. Organización y planificación.	Disposición en el trabajo en equipo aportando experiencias en el desarrollo.
Módulo III		
Relaciones entre objetos		
Competencia Específica		
Aplica fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la computación para simplificar el proceso en la modelación y diseño de soluciones informáticas.		
Tipos de saberes		
Saber (Conocimientos)	Saber hacer (Habilidades)	Saber ser (Actitudes y valores)
Diseña y construye soluciones informáticas explotando las ventajas de las distintas relaciones que pueden crear entre objetos.	Diseña e implementa soluciones con base en el paradigma orientado a objetos.	Compromiso en su desempeño. Hace trabajo colaborativo.



Control de excepciones a las aplicaciones desarrolladas para evitar rupturas en tiempo de ejecución.	Diseña e implementa soluciones validadas a entradas inesperadas. Trabajar en equipo. Proactivo y organizado	
Módulo IV		
Aplicaciones gráficas, eventos y archivos		
Competencia Específica		
Identifica oportunidades a través del uso eficiente y eficaz de soluciones informáticas, con responsabilidad y compromiso social, para mejorar el desempeño de las organizaciones.		
Tipos de saberes		
Saber (Conocimientos)	Saber hacer (Habilidades)	Saber ser (Actitudes y valores)
Construye aplicaciones con interfaces de usuario gráficas y por eventos. Respalda información para su posterior uso en memoria secundaria.	Aplica conocimientos de otras disciplinas con el objetivo de crear aplicaciones con un alto grado de utilidad y usabilidad dirigido a un tipo de usuario o público específico.	Receptivo y empático para conocer los distintos tipos de usuario (personas de la 3ª edad, niños, con capacidades diferentes, con problemas de aprendizaje) que podría tener la aplicación informática.
Bibliografía básica		
Oracle. (2017). The Java™ Tutorials, https://docs.oracle.com/javase/tutorial/		
Medina, N. (2016). Programación Orientada a Objetos con Java. La novela. Editorial Alfaomega. ISBN: 9786076225967.		
Blasco, F. (2019). Programación Orientada a Objetos en Java. Editorial Ra-Ma. ISBN: 978-84-9964-805-7		
Ceballos-Sierra, F. (2015) JAVA. Interfaces Graficas y aplicaciones para Internet, Editorial Ra-Ma. ISBN 9788499645223.		
Perry, J. S. (2014). Introducción a la programación Java, parte 1: Conceptos básicos del lenguaje Java. https://www.ibm.com/developerworks/ssa/java/tutorials/j-introjava1/		
Bibliografía complementaria		
Jones E. y Marcus A. y Wu. E. (2010). 6.092 Introduction to Programming in Java. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu		
McCaffrey, C. y Paluska J, Mendel L. y He R. y Toscano R., Ostler S.(2006). 6.092 Java Preparation for 6.170. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu License: Creative Commons BY-NC-SA.		



Schildt H. (2011). *Java. A Beginner's Guide* (5th Edition). ISBN-10:0071606327.

Deitel P.J y Deitel H.M. (2012), *Java. How to Program* (9th Edition). Prentice Hall. ISBN 978-0132575669.

Liang Y. D. (2012). *Introduction to Java Programming, Comprehensive, 9e, features comprehensive coverage ideal for a one-, two-, or three-semester CS1 course.*

3.-Evaluación

Criterios de Evaluación (% por criterio)

Evaluación diagnóstica

Plantear problemáticas en escenarios apegados a la realidad donde se involucren los saberes del curso previo y que son necesarios para dar continuidad en el presente curso, utilizando diagramas de flujo y/o pseudocódigo y/o código.

Evaluación Formativa

Diseño en UML e implementación en Java de programas sencillos donde se apliquen los conceptos del módulo en turno, dichos desarrollos serán considerados como prácticas. La cantidad de prácticas será dependerá del ritmo de avance que presente el grupo en turno.

Dos exámenes parciales donde se confirme el conocimiento teórico y su identificación en código. El primer parcial considerara los módulos I,II y III; el segundo parcial el módulo IV.

Los trabajos de investigación serán profundizar y/o comparar algunos temas tratados en clase, estos serán asignados aleatoriamente a los alumnos durante el modulo en curso.

Evaluación Sumatoria

- Exámenes parciales ordinarios 40%
- Trabajos de investigación 10%
- Prácticas 40%
- Autoevaluación y coevaluación 10%

4.-Acreditación

De acuerdo al **REGLAMENTO GENERAL DE EVALUACIÓN Y PROMOCIÓN DE ALUMNOS DE LA UNIVERSIDAD DE GUADALAJARA** que señala:

Artículo 5. El resultado final de las evaluaciones será expresado conforme a la escala de calificaciones centesimal de 0 a 100, en números enteros, considerando como mínima aprobatoria la calificación de 60. Las materias que no son sujetas a medición cuantitativa, se certificarán como acreditadas (A) o no acreditadas (NA).

Artículo 20. 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 H. Consejo General Universitario, se requiere: I. Estar inscrito en el plan de estudios y



curso correspondiente, y II. Tener un mínimo de asistencia del 80% a clases y actividades registradas durante el curso.

Artículo 25. La evaluación en periodo extraordinario se calificará atendiendo a los siguientes criterios: **I.** La calificación obtenida en periodo extraordinario, tendrá una ponderación del 80% para la calificación final; **II.** La calificación obtenida por el alumno durante el periodo ordinario, tendrá una ponderación del 40% para la calificación en periodo extraordinario, y **III.** La calificación final para la evaluación en periodo extraordinario será la que resulte de la suma de los puntos obtenidos en las fracciones anteriores.

Artículo 27. Para que el alumno tenga derecho al registro de la calificación en el periodo extraordinario, se requiere: **I.** Estar inscrito en el plan de estudios y curso correspondiente. **II.** Haber pagado el arancel y presentar el comprobante correspondiente. **III.** Tener un mínimo de asistencia del 65% a clases y actividades registradas durante el curso.

Artículo 33. El alumno que por cualquier circunstancia no logre una calificación aprobatoria en el periodo extraordinario, deberá repetir la materia en el ciclo escolar inmediato siguiente en que se ofrezca, teniendo la oportunidad de acreditarla durante el proceso de evaluación ordinario o en el periodo extraordinario, excepto para alumnos de posgrado.

En caso de que el alumno no logre acreditar la materia en los términos de este artículo, será dado de baja.

5.- Participantes en la elaboración

Código	Nombre
9814671	Marisela Mireles Mercado

6.- Fecha de elaboración

Julio de 2019