

SÍLABO

Programación Orientada a Objetos

Código	ASUC01482	Carácter	Obligatorio	
Prerrequisito	Fundamentos de Programación			
Créditos	4			
Horas	Teóricas	2	Prácticas	4
Año académico	2024			

I. Introducción

Programación Orientada a Objetos es una asignatura obligatoria de especialidad, ubicada en el cuarto periodo académico de la carrera profesional de la Ingeniería de Sistemas e Informática y tiene como prerrequisito Fundamentos de Programación. Es prerrequisito de Construcción de Software. Con esta asignatura se desarrolla en un nivel intermedio la competencia transversal Conocimientos de Ingeniería; y en un nivel inicial la competencia específica uso de Herramientas modernas. La relevancia reside en elaborar programas de entorno visual haciendo uso de los principios fundamentales de la programación orientada a objetos.

Los contenidos generales que la asignatura desarrolla son: Conceptos básicos de programación orientada a objetos: clase, objeto, herencia, encapsulamiento, polimorfismo. Entorno visual. Funciones avanzadas de entorno visual. Conexión a base de datos. Fundamentos de interacción hombre-computador: interacción, paradigmas, proceso de diseño, modelos y teorías. Diseño centrado en el usuario.

II. Resultado de aprendizaje de la asignatura

Al finalizar la asignatura, el estudiante será capaz de implementar aplicaciones usando los fundamentos de la programación orientada a objetos.

III. Organización de los aprendizajes

Unidad 1 Fundamentos de Programación Orientada a Objetos		Duración en horas	24
Resultado de aprendizaje de la unidad:	Al finalizar la unidad, el estudiante será capaz de analizar el uso de los fundamentos de la programación orientados a objetos en el desarrollo de aplicaciones.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Tipos de datos y estructuras de programación 2. Objetos y Clases. Constructores y Destructores 3. Herencia. Clases Abstractas. Polimorfismo 4. Clases internas e interfaces 		

Unidad 2 Fundamentos de la Interacción Hombre-Computador		Duración en horas	24
Resultado de aprendizaje de la unidad:	Al finalizar la unidad, el estudiante será capaz de aplicar fundamentos de la Interacción Hombre – Computador en la construcción de interfaces usuario-computador.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Principios Generales de IHC y usabilidad. 2. Diseño y prototipo centrado en el usuario 3. Evaluación de la interfaz de usuario 4. Uso de Layouts y distribución de componentes gráficos. 		

Unidad 3 Programación Visual y Basada en Eventos		Duración en horas	24
Resultado de aprendizaje de la unidad:	Al finalizar la unidad, el estudiante será capaz de desarrollar aplicaciones de programación visual y basada en eventos para mejorar la interacción con el usuario		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Excepciones. Manejo de errores. 2. Programación multihilos. Multithreading vs Multiprocessing. 3. Programación visual. Componentes Gráficos. 4. Programación con eventos. 		

Unidad 4 Acceso a Base de Datos		Duración en horas	24
Resultado de aprendizaje de la unidad:	Al finalizar la unidad, el estudiante será capaz de implementar aplicaciones con acceso a base de datos para el almacenamiento de información.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Conexión a base de datos. 2. API JDBC. 3. Consultas SQL 4. Manipulación de Base de Datos. CRUD a tablas. 		

IV. Metodología
Modalidad Presencial

La metodología por utilizarse en la asignatura se basa en el aprendizaje colaborativo y experiencial y es orientada en proyectos y como parte de su aplicación se desarrollará un trabajo grupal, el cual implicará la implementación de un aplicativo. La asignatura utiliza sesiones de teoría, prácticas de laboratorio, resolución de ejercicios, problemas y desarrollo de proyecto.

Modalidad Semipresencial - Blended, A Distancia

La metodología por utilizarse en la asignatura se basa en el aprendizaje colaborativo y experiencial y es orientada en proyectos y como parte de su aplicación se desarrollará un trabajo individual o grupal, el cual implicará la implementación de un aplicativo. La asignatura utiliza sesiones de teoría, prácticas de laboratorio, resolución de ejercicios, problemas y desarrollo de proyecto, presencial y virtual.

V. Evaluación
Modalidad Presencial

Rubros	Unidad por evaluar	Fecha	Entregable/Instrumento	Peso Parcial	Peso Total
Evaluación de entrada	Prerrequisito	Primera sesión	- Evaluación individual teórico-práctico / Prueba objetiva	0 %	
Consolidado 1 C1	1	Semana 1-4	- Evaluación individual teórico-práctico / Prueba de desarrollo	60 %	20 %
	2	Semana 5-7	- Evaluación individual práctico en laboratorio / Prueba Mixta	40 %	
Evaluación parcial EP	1 y 2	Semana 8	- Trabajo grupal de un proceso de la realidad / Rúbrica de evaluación	25 %	
Consolidado 2 C2	3	Semana 9-12	- Evaluación individual teórico-práctico / Prueba de desarrollo	60 %	20 %
	4	Semana 13-15	- Evaluación individual práctico en laboratorio / Prueba Mixta	40 %	
Evaluación final EF	Todas las unidades	Semana 16	- Trabajo grupal de un proceso de la realidad / Rúbrica de evaluación	35 %	
Evaluación sustitutoria *	Todas las unidades	Fecha posterior a la evaluación final	Aplica		

* Reemplaza la nota más baja obtenida en los rubros anteriores.

Modalidad Semipresencial - Blended

Rubros	Unidad por evaluar	Fecha	Entregable/Instrumento	Peso Parcial	Peso Total
Evaluación de entrada	Prerrequisito	Primera sesión	- Evaluación individual teórica / Prueba objetiva	0 %	
Consolidado 1 C1	1	Semana 2	- Actividades virtuales	15 %	20 %
			- Evaluación individual teórico-práctica / Prueba de desarrollo	85 %	
Evaluación parcial EP	1 y 2	Semana 4	- Trabajo individual o grupal de un proceso de la realidad / Rúbrica de evaluación	25 %	
Consolidado 2 C2	3	Semana 6	- Actividades virtuales	15 %	20 %
			- Evaluación individual teórico-práctica / Prueba de desarrollo	85 %	
Evaluación final EF	Todas las unidades	Semana 8	- Trabajo individual o grupal de un proceso de la realidad / Rúbrica de evaluación	35 %	
Evaluación sustitutoria	Todas las unidades	Fecha posterior a la evaluación final	Aplica		

* Reemplaza la nota más baja obtenida en los rubros anteriores.

Modalidad A Distancia

Rubros	Unidad por evaluar	Fecha	Entregable/Instrumento	Peso
Evaluación de entrada	Prerrequisito	Primera sesión	- Evaluación individual teórica / Prueba objetiva	0 %
Consolidado 1 C1	1	Semana 2	- Evaluación individual teórico-práctica / Prueba de desarrollo	20 %
Evaluación parcial EP	1 y 2	Semana 4	- Trabajo individual o grupal de un proceso de la realidad / Rúbrica de evaluación	25 %
Consolidado 2 C2	3	Semana 6	- Evaluación individual teórico-práctica / Prueba de desarrollo	20 %
Evaluación final EF	Todas las unidades	Semana 8	- Trabajo individual o grupal de un proceso de la realidad / Rúbrica de evaluación	35 %
Evaluación sustitutoria	Todas las unidades	Fecha posterior a la evaluación final	Aplica	

* Reemplaza la nota más baja obtenida en los rubros anteriores.

Fórmula para obtener el promedio final:

$$PF = C1 (20 \%) + EP (25 \%) + C2 (20 \%) + EF (35 \%)$$

VI. Bibliografía

Básica

Gervais, L. (2019). *Aprender la programación orientada a objetos con el lenguaje Java*. Ediciones ENI. <https://bit.ly/2NRAXlz>

Complementaria:

Deitel, P., Deitel H. (2016). *Java. Como programar*. (10.ª ed.). México. Pearson Education.
Schildt, H. (2017). *Java: The Complete Reference*. (11.º ed.). New York, USA: McGraw- Hill Education

Recursos digitales

Bizagi. (2023). *Folder en Google Drive*.

https://drive.google.com/drive/folders/1QYyRtswudgddb_b6DwZfr5Oo4uvKyynv?usp=share_link

NetBeans. (2023). *Folder en Google Drive*.

https://drive.google.com/drive/folders/1QYyRtswudgddb_b6DwZfr5Oo4uvKyynv?usp=share_link

Oracle Xpress. (2023). *Folder en Google Drive*.

https://drive.google.com/drive/folders/1QYyRtswudgddb_b6DwZfr5Oo4uvKyynv?usp=share_link

Visual Paradigm. (2023). *Folder en Google Drive*.

https://drive.google.com/drive/folders/1QYyRtswudgddb_b6DwZfr5Oo4uvKyynv?usp=share_link