

SILABO DEL CURSO TÉCNICAS DE PROGRAMACIÓN ORIENTADA A OBJETOS

I. INFORMACIÓN GENERAL

Facultad	Ingeniería	Carrera profesional	Ingeniería de Sistemas Computacionales	Ciclo	5°	Créditos		5
Código de curso	SIST1202A	Requisitos	Fundamentos de Programación	Horas	HT	HP	HL	PC
					1	0	2	6
Tipo de curso	Obligatorio	Modalidad del curso	Presencial	Periodo lectivo	2024-4			
El curso aporta a la(s) competencia(s) general(es):		<ul style="list-style-type: none">Resolución de problemasResponsabilidad social y ciudadanía						
El curso aporta a la(s) competencia(s) específica(s):		<ul style="list-style-type: none">Análisis de problemaDiseño y desarrollo de soluciones						
El curso desarrolla el componente:		<ul style="list-style-type: none">Responsabilidad social y formación ciudadana						

II. SUMILLA

El curso es de naturaleza teórico-práctico y tiene como propósito brindar los conocimientos del Paradigma Orientado a Objetos aplicados a la programación, desarrollando en el estudiante la capacidad de implementar modelos de clases en un lenguaje de programación orientado a objetos. Los temas principales son: fundamentos de programación orientado a objetos, representación de clases y relaciones entre clases, programación visual y acceso a base de datos.

III. LOGRO DEL CURSO

Al finalizar el curso, el estudiante desarrolla un proyecto, utilizando el lenguaje Java y aplicando el paradigma de programación orientado a objetos, interfaces gráficas y acceso a datos para resolver problemas específicos; demostrando lógica, habilidad, legibilidad y buenas prácticas en la implementación.

IV. METODOLOGÍA DE ENSEÑANZA APRENDIZAJE

Para alcanzar el logro de aprendizaje del curso y de las unidades, el docente integra métodos activos, estrategias y técnicas de manera reflexiva y crítica, buscando motivar, estimular y guiar el aprendizaje del estudiante.

Las estrategias y técnicas didácticas que se utilizan son: resolución de casos en forma de ejercicios propuestos de manera individual y en equipo, para los cuales los estudiantes analizan el contexto, abstraen requerimientos de software, analizan y generan diagramas de clases y generan programas utilizando lenguaje de programación orientado a objetos y documentos básicos de desarrollo de software. Además, se realizan exposiciones individuales y grupales, donde muestran progresivamente avances del proyecto de aplicación, orientado a brindar solución a un problema. Asimismo, se desarrollan actividades grupales de responsabilidad social y analizan el impacto de las mismas.

El docente soporta su práctica pedagógica en un sistema de multiplataformas y recursos multimedia que le permiten el desarrollo de actividades sincrónicas y asincrónicas, así como la gestión de contenidos, videoconferencias y el uso de diversas herramientas tecnológicas para generar experiencias formativas y brindar orientaciones que promuevan el aprendizaje y el desarrollo de competencias generales y específicas en los estudiantes.

V. ORGANIZACIÓN DE UNIDADES DE APRENDIZAJE

UN	NOMBRE / LOGRO DE UNIDAD	SEM	SABERES ESENCIALES
I	Fundamentos de programación orientada a objetos Al finalizar la unidad, el estudiante desarrolla casos, utilizando adecuadamente los principios de programación orientada a	1	Presentación del sílabo del curso. Difusión del protocolo de seguridad del laboratorio. Paradigma Orientado a Objetos. Fundamentos de la programación Orientada a Objetos. Java como herramienta de programación orientada a objetos. Principales sentencias en Java. Introducción a la programación orientada a objetos.

HT=Horas de desarrollo teórico
 HP= Horas de desarrollo práctico
 HL= Horas de desarrollo práctico en laboratorio
 PC= Horas de práctica de campo

UN	NOMBRE / LOGRO DE UNIDAD	SEM	SABERES ESENCIALES
II	Representación de clases y relaciones entre clases Al finalizar la unidad, el estudiante desarrolla casos, haciendo uso correcto de diagramas y buenas prácticas de programación orientado a objetos, demostrando creatividad y responsabilidad en su trabajo.		Clase - Encapsulamiento Principios, evolución, abstracción de datos. Ocultamiento y Encapsulamiento. Necesidad de encapsular datos. Clases y Objetos. Ciclo de vida.
			<u>Constructores - Destructores</u> Constructores y destructores. Métodos de instancia, Métodos de clase. Categorías de métodos: gestión, implementación y ayuda.
		2	Introducción al Lenguaje Unificado de Modelado UML, Representación de las clases. Diagramas de Clase. Sobrecarga de métodos. Sobrecarga de operadores. Colecciones.
		3	Relación de herencia simple. Clases abstractas. Relación de herencia múltiple. Polimorfismo y arreglo de objetos. Casting.
		4	Diagramas de clases y sus relaciones. Relación Binaria y Reflexiva.
III	Programación visual y acceso a base de datos Al finalizar la unidad, el estudiante será capaz de desarrollar aplicaciones de programación visual y con acceso a datos que, para almacenamiento de información, demostrando creatividad y buenas prácticas de programación.	5	Evaluación T1 Relación de agregación y composición.
		6	Componentes SWING y programación de componentes gráficos. Excepciones. Manejo de errores. Programación de eventos. JDBC: Java DataBase Connectivity: JDBC. Base de Datos. Conectividad JDBC. Conectando con SQL.
		7	JDBC: Puente JDBC-ODBC y Lenguaje SQL Consultas SQL Manipulación de Base de Datos. CRUD a tablas. Evaluación T2
		8	Reportes Evaluación final
		(-)	No aplica evaluación sustitutoria.

VI. SISTEMA DE EVALUACIÓN

EVALUACIÓN	PESOS	SEM	DESCRIPCIÓN DE LA EVALUACIÓN (Acción + Producto de la evidencia que debe presentar el estudiante)
T1 (a)	30%	5	Desarrollo de casos
T2 (a)	30%	7	Evaluación de práctica de campo
Evaluación Final (a)	40%	8	Presentación y exposición de proyecto final

(a) Los calificativos deben ser publicados en el sistema de acuerdo con el Calendario Académico establecido para el presente Semestre.

VII. BIBLIOGRAFÍA BÁSICA

Nº	AUTOR	TÍTULO	AÑO	ENLACE URL
1	Blasco, Francisco	Programación orientada a objetos en Java	2019	https://elibro.bibliotecaupn.elogim.com/es/lc/upnorte/titulos/127125

a) BIBLIOGRAFÍA COMPLEMENTARIA (Opcional a la necesidad de la carrera)

HT=Horas de desarrollo teórico
 HP= Horas de desarrollo práctico
 HL= Horas de desarrollo práctico en laboratorio
 PC= Horas de práctica de campo

N°	AUTOR	TÍTULO	AÑO	ENLACE URL
1	Vélez Serrano, J.	Diseñar y programar, todo es empezar: una introducción a la Programación Orientada a Objetos usando UML y Java	2011	https://elibro.bibliotecaupn.elogim.com/es/lc/upnorte/titulos/63076
2	Blasco, Francisco	Programación en Java	2020	https://digitalia.bibliotecaupn.elogim.com/a/110219

VIII. INFORMACIÓN COMPLEMENTARIA

REFERENCIA	ENLACE URL
Biblioteca Virtual UPN	https://biblioteca.upn.edu.pe/
Agile Mindset	https://conectaempleo-formacion.fundaciontelefonica.com/web/pe-agile-mindset-ed-1
Presentaciones en Público y Digitales	https://conectaempleo-formacion.fundaciontelefonica.com/web/pe-presentaciones-en-publico-y-digitales-ft-ed-6
JF Java Fundamentals Alumno - Español	https://myacademy.oracle.com/

ANEXO: Ficha de Horas de Práctica de Campo – Técnicas de programación orientada a objetos

SEMA NA	LOGRO DE UNIDAD	ACTIVIDADES PRÁCTICAS DE CAMPO	EVIDENCIA DE PRÁCTICA DE CAMPO (Acción + Producto)	ENTREGA DE EVIDENCIA
1	I Fundamentos de programación orientada a objetos	Realizar una guía documentada de las sintaxis que Java utiliza para el manejo de instrucciones secuenciales, condicionales y repetitivas y ejemplifica al menos 2 casos. Abstractar al menos 10 clases para diferentes escenarios.	Presentar informe que incluya los requerimientos de PC descritos.	SEMANA 2
2		En grupos de Proyecto: Identificar y formular el problema, diagrama de Ishikawa, documentar antecedentes adecuadamente (Cita APA) Identificar restricciones realistas que afectan al proyecto y propone alternativas de solución en base a ellas.		
3	II Representación de clases y relaciones entre clases	Elaborar el EDT en base al ciclo de vida del software y presentar al menos 40 requerimientos del proyecto. Elaborar historias de usuario y crear el diagrama de clases del proyecto.	Presentar informe y software que incluya los requerimientos y formatos de PC descritos.	SEMANA 4
4		Investigar sobre colecciones y archivos para tratamiento de datos en Java. Programar escenarios de herencia del proyecto. Documentar los criterios de aceptación de cada requerimiento.		
5	III Programación visual y acceso a base de datos	Programar al menos 01 escenario por cada tipo de relación estudiada (binaria, reflexiva, agregación, composición). Investigar sobre principios de usabilidad y diseño de prototipos centrado en el Usuario Desarrollar actividad de responsabilidad social	Presentar informe y software que incluya los requerimientos y formatos de PC descritos.	SEMANA 6
6		Implementar todos los requerimientos del proyecto y evidenciar el cumplimiento de los criterios de aceptación.		
7	RETROALIMENTACIÓN Y REGISTRO DE EVIDENCIAS EN EL GESTOR DE PRÁCTICA CAMPO			
8	Cada estudiante debe cargar en Blackboard la captura de pantalla de haber subido la evidencia de práctica de campo en el gestor.			