

## PROGRAMA ANALÍTICO

### 1. DATOS INFORMATIVOS

<b>DEPARTAMENTO:</b> CIENCIAS DE LA COMPUTACION		<b>ÁREA DE CONOCIMIENTO:</b> PROGRAMACION	
<b>NOMBRE DE LA ASIGNATURA:</b> PROG ORIENT A OBJETOS		<b>PERIODO ACADÉMICO:</b>	
<b>CÓDIGO:</b> MVU01		<b>No. CREDITOS:</b> 4	<b>NIVEL:</b> PREGRADO
<b>FECHA ELABORACIÓN:</b>	<b>EJE DE FORMACIÓN</b>	<b>HORAS / SEMANA</b>	
	BÁSICA	<b>TEÓRICAS:</b>	<b>PRÁCTICAS/LABORATORIO</b>
<b>DESCRIPCIÓN DE LA ASIGNATURA:</b> La materia Programación Orientada a Objetos, es una asignatura del eje de formación profesional, que se caracteriza por contribuir a la formación de los elementos de competencia y fortalecer las unidades de competencia en análisis, diseño, y construcción de aplicaciones de software, basado en el paradigma orientado a objetos, sus fundamentos y principios, como el encapsulamiento, la abstracción, la herencia, el polimorfismo apoyados, por el lenguaje de programación Java. Esta asignatura se enfoca principalmente en la resolución de problemas complejos del mundo real, y en producir aplicaciones de calidad, empleando principios y prácticas, tales como pruebas de unidad, patrones de diseño y los "SOLID Principales". Se fortalece también con el uso de interfaces gráficas de usuario, que permiten la adecuada interacción entre el usuario y el computador.			
<b>CONTRIBUCIÓN DE LA ASIGNATURA A LA FORMACIÓN PROFESIONAL:</b> La asignatura contribuye al resultado de aprendizaje del nivel y es parte sustancial de la formación profesional, los componentes son la solución a problemas orientados a la integración de diferentes aplicaciones e infraestructura tecnológica existente en las organizaciones, bajo el sustento de la programación de computadores.			
<b>RESULTADO DE APRENDIZAJE DE LA CARRERA (UNIDAD DE COMPETENCIA):</b> Los estudiantes en este nivel conocerán los principios de la programación orientada a objetos para el desarrollo de aplicaciones, con estándares que permitan la reutilización del software.			
<b>OBJETIVO DE LA ASIGNATURA:</b> Formar ingenieros en Tecnologías de la Información capaces de seleccionar, crear, aplicar, integrar y administrar eficientemente las tecnologías de la información orientadas a satisfacer las necesidades de los usuarios dentro de un entorno social, organizacional y humanista.			
<b>RESULTADO DE APRENDIZAJE DE LA ASIGNATURA: (ELEMENTO DE COMPETENCIA):</b> Conceptuales: Conoce los conceptos de la programación Orientada a Objetos. Describe la relación entre la estructura estática de la clase y la estructura dinámica de la instancia de la clase. Procedimentales: Diseña, implementa, prueba y depura programas sencillos en un lenguaje de programación orientado a objetos. Diseña, implementa y prueba la aplicación relacionada entre objetos utilizando una clase mediante la jerarquía y herencia. Utiliza iteraciones para acceder a los elementos de un contenedor. Actitudinales: Participa activamente en un equipo de trabajo desarrollando aplicaciones que empleen conocimientos de programación orientada a objetos.			

### 2. SISTEMA DE CONTENIDOS Y RESULTADOS DEL APRENDIZAJE

UNIDADES DE CONTENIDOS	
<b>Unidad 1</b>  PRINCIPIOS DE DISEÑO ORIENTADO A OBJETOS	<b>Resultados de Aprendizaje de la Unidad 1</b>  Aplica, utiliza técnicas de programación orientada a objetos (POO), emplea UML para el modelamiento de Clases (Diagramas de clases) y de requerimientos (diagramas de casos de uso), utilizando una herramienta de modelado y un lenguaje POO.
<b>Sistemas de control de versionamiento (VCS)</b> Software VCS: Git, GitHub.	
<b>Paradigmas de programación</b> Transición de paradigma	
<b>Entorno de Desarrollo</b> Características e Instalación. Administración y configuración del área de trabajo Líneas de Comando	
<b>Revisión de conceptos generales de la POO.</b>	

# PROGRAMA ANALÍTICO

## UNIDADES DE CONTENIDOS

Principios Generales de la Programación Orientada a Objetos.

Definición de clases, objetos, atributos y métodos.

### Modelamiento de clases y Objetos

UML Diagramas de Casos de Uso, Clases.

Identificar clases de un sistema. Uso correcto de identificadores.

Modificadores de Acceso

Implementación de clases

### Estructura general de un programa

Creación de un programa básico P.O.O

Tipos de datos, primitivos y referencia dos.

### Constructores

Tipos de constructores

Instanciación

### Métodos getters, setters.

Definición

Aplicación.

## Unidad 2

### Resultados de Aprendizaje de la Unidad 2

diseña aplicaciones enfocadas a la solución de problemas computacionales con técnicas de programación orientada a objetos, con el uso y tratamiento de errores basadas en validaciones y testing, realiza pruebas de unidad y depuraciones de la aplicación

## RELACIONES ENTRE CLASES

### Lectura escritura de datos por consola

Entrada de datos

Salida de datos

### Excepciones

Definición

Excepciones y errores

Clases de excepción

Tipos de excepciones

Propagación de excepciones

Lanzamiento de excepciones

Métodos para el control de excepciones

Excepciones personalizadas

### Arreglos estáticos y dinámicos.

Arreglos de datos primitivos

Arreglos de objetos

Colecciones

### Relaciones entre clases

Asociación, Agregación / Composición: modelado e implementación

Dependencia: Modelado

### Generalización/Especialización

Herencia: Definición, ventajas, nomenclatura, reglas y modelado.

Implementación de herencia.

### Código limpio

Estándares de implementación, buenas prácticas de programación.

Atributos de calidad de código

# PROGRAMA ANALÍTICO

## UNIDADES DE CONTENIDOS

### Gestión de Defectos (testing).

Verificación y Validación  
Pruebas Vs Depuración.  
Pruebas de unidad

<b>Unidad 3</b>  TÉCNICAS AVANZADAS DE P.O.O	<b>Resultados de Aprendizaje de la Unidad 3</b>  Diseña aplicaciones enfocadas a la ingeniería con técnicas de programación orientada a objetos con el uso de interfaz gráfica amigable para el usuario y persistencia de datos.
<p><b>Encapsulamiento</b></p> <ul style="list-style-type: none"><li>Definición</li><li>Clases internas</li><li>Paquetes</li></ul> <p><b>Polimorfismo</b></p> <ul style="list-style-type: none"><li>Definición y ventajas</li><li>Sobrecarga de métodos</li><li>Sobre escritura de métodos</li><li>Asignación de objetos a variables de su superclase.</li></ul> <p><b>Abstracción</b></p> <ul style="list-style-type: none"><li>Definición</li><li>Modelado</li><li>Clases abstractas</li><li>Métodos abstractos</li></ul> <p><b>Interfaces de programación</b></p> <ul style="list-style-type: none"><li>Definición</li><li>Modelado</li><li>Declaración e implementación</li><li>Interfaces y polimorfismo</li></ul> <p><b>Persistencia de datos</b></p> <ul style="list-style-type: none"><li>Manipulación de archivos</li><li>Lectura y escritura de datos primitivos</li><li>Lectura y escritura de objetos: Serialización.</li></ul>	
<p><b>Componentes y objetos gráficos</b></p> <ul style="list-style-type: none"><li>Widgets (componentes gráficos)</li><li>Formularios</li><li>Menús, tablas</li><li>Gestión de eventos</li><li>Integración de componentes gráficos y clases</li></ul>	

### 3. PROYECCIÓN METODOLÓGICA Y ORGANIZATIVA PARA EL DESARROLLO DE LA ASIGNATURA

#### (PROYECCIÓN DE LOS MÉTODOS DE ENSEÑANZA - APRENDIZAJE QUE SE UTILIZARÁN)

- 1 Estudio de Casos
- 2 Clase Magistral
- 3 Resolución de Problemas
- 4 Prácticas de Laboratorio

# PROGRAMA ANALÍTICO

## PROYECCIÓN DEL EMPLEO DE LA TIC EN LOS PROCESOS DE APRENDIZAJE

- 1 Aula Virtual
- 2 Material Multimedia
- 3 Herramientas Colaborativas (Google, drive, onedrives, otros)

### 4. TÉCNICAS Y PONDERACIÓN DE LA EVALUACIÓN

- En este espacio se expresarán las técnicas utilizadas en la evaluación del proceso de enseñanza aprendizaje o evaluación formativa y sumativa.
- Las técnicas que se recomienda usar son: Resolución de ejercicios, Investigación Bibliográfica, Lecciones oral/escrita, Pruebas orales/escrita, Laboratorios, Talleres, Solución de problemas, Prácticas, Exposición, Trabajo colaborativo, Examen parcial, Otras formas de evaluación.
- Recordar que mientras más técnicas utilicen, la evaluación será más objetiva y el desempeño del estudiante se reflejará en su rendimiento (4 o 5 técnicas).
- Para evaluar se deberá aplicar la rúbrica en cada una de las técnicas de evaluación empleadas. Se debe expresar en puntaje de la nota final sobre 20 puntos. No debe existir una diferencia mayor a dos puntos entre cada técnica de evaluación empleada.
- En la modalidad presencial existen tres parciales en la modalidad a distancia existen dos parciales, toda la planificación de periodo académico se la realiza en función del número de parciales de cada modalidad.
- La ponderación a utilizarse en la evaluación del aprendizaje del estudiante será la misma en las tres parciales.
- Para la aprobación de una asignatura se debe tener una nota final promedio de 14/20, en los tres o dos

### 5. BIBLIOGRAFÍA BÁSICA/ TEXTO GUÍA DE LA ASIGNATURA

Titulo	Autor	Edición	Año	Idioma	Editorial
Programador Java 2 certificado : curso práctico	Martín Sierra, Antonio J.	-	2007	Español	Alfaomega
Cómo programar en java	Deitel, Paúl	9	2012	spa	earson
Java para estudiantes	Bell, Douglas	-	2011	Español	México, D. F. : Prentice Hall
Java a fondo : estudio del lenguaje y desarrollo de aplicaciones	Sznajdleder, Pablo Augusto	-	2010		Buenos Aires : Alfaomega Grupo Editor