

I. Identificación del Curso

Carrera:	Desarrollo de Software			Modalidad:	Presencial	Asignatura UAC:	Programación orientada a objetos			Fecha Act:	Diciembre, 2018
Clave:	18MPBDS0412	Semestre:	4	Créditos:	7.20	División:	Informática y Computación			Academia:	Computación
Horas Total Semana:	4	Horas Teoría:	2	Horas Práctica:	2	Horas Semestre:	72	Campo Disciplinar:	Profesional	Campo de Formación:	Profesional Básico

Tabla 1. Identificación de la Planificación del Curso.

II. Adecuación de contenidos para la asignatura

Propósito de la Asignatura (UAC)
Que el estudiante desarrolle programas computacionales utilizando un enfoque orientado a objetos a partir de requerimientos para dar solución a problemas técnicos y prácticos en el ámbito del desarrollo de software
Competencias Profesionales a Desarrollar (De la carrera)
Construye sistemas o soluciones informáticas confiables, de carácter innovador, personal o a la medida empleando una metodología y una tecnología de desarrollo de software que sea adecuada y sustentada en normas y estándares nacionales e internacionales.

Tabla 2. Elementos Generales de la Asignatura



III. Competencias de la UAC

Competencias Genéricas.*

- 5. Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.
- 5.6 Utiliza las tecnologías de la información y comunicación para procesar e interpretar información.

Competencias Disciplinarias Básicas**

CO-12 Utiliza las tecnologías de la información y comunicación para investigar, resolver problemas, producir materiales y transmitir información.

Competencias Disciplinarias Extendidas***

COE-11 Aplica las tecnologías de la información y la comunicación en el diseño de estrategias para la difusión de productos y servicios, en beneficio del desarrollo personal y profesional.



Competencias Profesionales Básicas	Competencias Profesionales Extendidas
<p>- Emplea la metodología orientada a objetos como una herramienta de desarrollo de aplicaciones informáticas como base para el uso de diferentes plataformas de desarrollo de software en el ámbito de su carrera y de desarrollo profesional.</p>	<p>- Crea aplicaciones para resolver problemáticas a la medida mediante técnicas y procedimientos orientados a objetos.</p>

Tabla 3. Competencias de la Asignatura.

* Se presentan los atributos de las competencias Genéricas que tienen mayor probabilidad de desarrollarse para contribuir a las competencias profesionales, por lo cual no son limitativas; usted puede seleccionar otros atributos que considere pertinentes. Estos atributos están incluidos en la redacción de las competencias profesionales, por lo que no deben desarrollarse explícitamente o por separado.

** Las competencias Disciplinarias no se desarrollarán explícitamente en la UAC. Se presentan como un requerimiento para el desarrollo de las competencias Profesionales.

*** Cada eje curricular debe contener por lo menos una Competencia Disciplinar Extendida.



IV. Habilidades Socioemocionales a desarrollar en la UAC*4

Dimensión	Habilidad
Relaciona T	Colaboración

Tabla 4. Habilidades Construye T

*Estas habilidades se desarrollarán de acuerdo al plan de trabajo determinado por cada plantel. Ver anexo I.



V. Aprendizajes Clave

Eje Disciplinar	Componente	Contenido Central
Desarrollo de sistemas y tratamiento de la información.	Paradigmas de la programación.	<ol style="list-style-type: none"> 1. El paradigma de programación orientado a objetos (POO), su metodología y sus propiedades en el desarrollo de aplicaciones informáticas. 2. Implementación del paradigma orientado a objetos, utilizando un lenguaje de programación. 3. La herencia, el polimorfismo y las plantillas como métodos de manipulación y reutilización de clases. 4. Técnicas de comunicación y almacenamiento de datos para la creación de aplicaciones en red.



VI. Contenidos Centrales de la UAC

Contenido Central	Contenidos Específicos	Aprendizajes Esperados	Proceso de Aprendizaje	Productos Esperados
1. El paradigma de programación orientado a objetos (POO), su metodología y sus propiedades en el desarrollo de aplicaciones informáticas.	<ul style="list-style-type: none"> - El paradigma de la programación orientada a objetos. - La importancia de la representación gráfica o escrita de los conceptos del paradigma orientado a objetos. 	<ul style="list-style-type: none"> - Identifica las propiedades básicas de un objeto (abstracción, encapsulación, herencia y polimorfismo). - Identifica los conceptos de la POO, como son clase, objeto, propiedad, evento, método y mensaje para implementarlos en el desarrollo de soluciones a la medida. - Representa gráficamente la solución de un problema mediante diagramas UML. - Implementa propiedades y conceptos de la POO en el desarrollo de clases y jerarquías, para la solución de un problema informático. - Compara la diferencia entre la programación orientada a objetos y otros paradigmas. 	<ul style="list-style-type: none"> - Revisa fuentes de información virtual o física para generar un esquema donde se incluyan los conceptos de abstracción, encapsulación, herencia y polimorfismo. - Realiza una clase magistral, en donde se exponen los componentes de la POO, empleando elementos audiovisuales, a su vez se crea un organizador gráfico con la información expuesta. - Representa gráficamente las características obtenidas de un requerimiento con UML. - Revisa diferentes fuentes de información virtual o física acerca de los diferentes paradigmas de programación y genera un organizador gráfico con la información obtenida. 	<ul style="list-style-type: none"> - Cuadro sinóptico referente a las propiedades de un objeto. - Organizador gráfico con los componentes de la POO. - Diagrama UML con la representación de los requerimientos para solución de un problema. - Cuadro comparativo donde se representa la diferencia entre el paradigma de programación orientada a objetos y otros paradigmas de programación.



<p>2. Implementación del paradigma orientado a objetos, utilizando un lenguaje de programación.</p>	<ul style="list-style-type: none"> - La sintaxis que provee un lenguaje de programación para la representación clases y objetos. - La implementación de los diferentes tipos de métodos y propiedades empleados en la programación orientada a objetos. 	<ul style="list-style-type: none"> - Aplica la metodología de programación orientada a objetos. - Desarrolla clases y sus componentes; propiedades y métodos aplicando los diferentes tipos de accesos. - Define la independencia de instancias, como característica de un objeto. - Utiliza los diferentes métodos de una clase (estáticos, en línea, constantes, constructor y destructor por defecto), para complementar la funcionalidad de un objeto. - Aplica la sobrecarga de operadores, funciones y constructores para la reutilización de código. - Emplea el apuntador this como herramienta de referencia de un objeto. 	<ul style="list-style-type: none"> - Implementa un programa donde ejemplifique cada uno de las propiedades y conceptos de la POO, empleando la sintaxis proporcionada por un lenguaje de programación. - Diseña e implementa un conjunto de programas orientados a objetos, donde se integren y practiquen las propiedades y los conceptos de la POO, empleando un problema listado en requerimientos y un lenguaje de programación. - Revisa fuentes de información virtual o física acerca de los distintos tipos de métodos (estáticos, en línea, constantes, constructor y destructor por defecto) organizándolas por uso, descripción, sintaxis, ciclo de vida y normatividad. - Implementa un conjunto de programas ejemplo donde se utilicen los diferentes tipos de métodos. 	
		<ul style="list-style-type: none"> - Implementa los diferentes tipos de herencia (simple y múltiple). 	<ul style="list-style-type: none"> - Diseña e implementa un conjunto de programas que generen una solución que utilice algunos de los diferentes tipos de métodos. - Revisa una investigación acerca de los métodos de reducción de sintaxis y reuso de código (sobrecarga de operadores y 	



métodos) creando un organizador gráfico con la información obtenida, para con ella generar un sistema que emplee dichos métodos.

- Realiza un programa que refleje el comportamiento del apuntador this.

- Conjunto de programas basados en clases, constructores y métodos sobre cada uno de los aprendizajes esperados.

- Cuadro comparativo sobre los diferentes tipos de métodos en el diseño de clases.

- Cuadro sinóptico sobre las normas y sintaxis de los métodos y operadores sobrecargados.



<p>3. La herencia, el polimorfismo y las plantillas como métodos de manipulación y reutilización de clases.</p>	<ul style="list-style-type: none"> - La importancia del polimorfismo en la fabricación de aplicaciones orientadas a objetos. - Clases parametrizadas como medio de reutilización de clases. 	<ul style="list-style-type: none"> - Implementa el polimorfismo y la especialización de métodos a través de las funciones virtuales. - Crea e implementa clases abstractas usando funciones virtuales puras y apuntadores. - Crea e implementa plantillas para el almacenamiento genérico de datos. 	<ul style="list-style-type: none"> - Realiza una exposición magistral empleando un método audiovisual o TICs, para la creación de clases heredadas (simples y múltiples) indicando si es clase nueva o subtipo. - Diseña e implementa un conjunto de programas orientados a objetos, en donde se practique el uso de los diferentes tipos de herencia. - Ejemplifica el concepto de especialización (métodos virtuales y virtuales puros), empleando un programa pretexto. - Ejemplifica el concepto de polimorfismo, empleando un programa pretexto. - Diseña e implementa un conjunto de programas orientados a objetos, en donde se practique el uso de la especialización y el polimorfismo. - Realiza una exposición magistral empleando un método audiovisual o TICs, para la creación de clases parametrizadas. - Diseña e implementa un conjunto de programas orientados a objetos, en donde se practique el uso de clases parametrizadas. 	<ul style="list-style-type: none"> - Conjunto de programas sobre cada uno de los aprendizajes esperados, como son: herencia simple, herencia múltiple, métodos virtuales. - Diseños de cada programa, aplicando los elementos de la programación orientada objetos como son: herencia simple y múltiple, polimorfismo y funciones virtuales.
---	---	--	---	--



<p>4. Técnicas de comunicación y almacenamiento de datos para la creación de aplicaciones en red.</p>	<ul style="list-style-type: none"> - La representación estructurada de información. - El socket como medio de transferencia de datos. 	<ul style="list-style-type: none"> - Representa información empleando XML como medio estructurado de datos. - Almacena información en archivos empleando XML como medio alternativo a un almacenamiento binario. - Define las características y estados de un socket. - Crea un sistema que envía, recibe y procesa información entre distintos equipos empleando sockets e información estructurada en XML. 	<ul style="list-style-type: none"> - Revisa fuentes de información virtual o física y genera un ejemplo que incluya las características y nomenclatura de la información en formato XML. - Diseña y codifica un programa que utilice información en formato XML en un archivo. - Realiza una exposición con un medio audiovisual y una demostración de uso (con el lenguaje de programación) de un socket con la finalidad de realizar un sistema que envíe información en formato XML a través de un socket. 	<ul style="list-style-type: none"> - Cuadro descriptivo sobre los componentes XML. - Un sistema de almacenamiento usando XML. - Un sistema de comunicación con XML. - Diseños de cada sistema aplicando las metodologías orientadas a objetos.
---	---	--	--	--



VII. Recursos bibliográficos, hemerográficos y otras fuentes de consulta de la UAC

Recursos Básicos:

- Dean, J. y Dean, R. (2009). Introducción a la programación con Java. México: McGraw Hill
- Deitel, P. y Deitel, H. (2012). Cómo programar en Java. México: Prentice Hall
- López, A. (2007). Introducción al desarrollo de programas con Java. México: UNAM
- Stroustrup, B. (2014). Programming: Principles and Practice using C++. USA: Addison-Wesley

Recursos Complementarios:

- Ceballos, J. (2007). Java 2 Lenguaje y aplicaciones. España: Alfaomega
- Deitel, P. y Deitel, H. (2014). Cómo programar C++. México: Pearson
- Eckel, B. (2009). Thinking in Java. EUA: Prentice Hall
- Froufe, A. (2008). Java 2 Manual de usuario y tutorial. España: Alfaomega
- Joyanes, L. y Zahonero, I. (2011). Programación en Java 6. España: McGraw Hill
- Weiss, M. (2010). Estructura de datos en JavaTM: compatible con JavaTM2. México: Ed. Addison Wesley.
- Wu, T. (2008). Introducción a la Programación Orientada a Objetos con Java. España: Mc GrawHill

VIII. Perfil profesiográfico del docente para impartir la UAC

Recursos Complementarios:

Área/Disciplina: Informática

Campo Laboral: Servicios

Tipo de docente: Profesional.

Formación Académica: Licenciatura o Ingeniería, en Electrónica, Sistemas Computacionales e Informática y carreras afines.

Constancia de participación en los procesos establecidos en la Ley General del Servicio Profesional Docente, COPEEMS, COSDAC u otros.



XI. Fuentes de Consulta

Fuentes de consulta utilizadas*

- Acuerdo Secretariales relativos a la RIEMS.
- Planes de estudio de referencia del componente básico del marco curricular común de la EMS. SEP-SEMS, México 2017.
- Guía para el Registro, Evaluación y Seguimiento de las Competencias Genéricas, Consejo para la Evaluación de la Educación del Tipo Medio Superior, COPEEMS.
- Manual para evaluar planteles que solicitan el ingreso y la promoción al Padrón de Buena Calidad del Sistema Nacional de Educación Media Superior PBC-SINEMS (Versión 4.0).
- Normas Generales de Servicios Escolares para los planteles que integran el PBC. SINEMS
- Perfiles profesiográficos COPEEMS-2017
- SEP Modelo Educativo 2016.
- Programa Construye T



ANEXO II. Vinculación de las competencias con Aprendizajes esperados

Aprendizajes Esperados	Productos Esperados	Competencias Genéricas con Atributos	Competencias Disciplinarias	Competencias profesionales
<ul style="list-style-type: none"> - Identifica las propiedades básicas de un objeto (abstracción, encapsulación, herencia y polimorfismo). - Identifica los conceptos de la POO, como son clase, objeto, propiedad, evento, método y mensaje para implementarlos en el desarrollo de soluciones a la medida. - Representa gráficamente la solución de un problema mediante diagramas UML. - Implementa propiedades y conceptos de la POO en el desarrollo de clases y jerarquías, para la solución de un problema informático. - Compara la diferencia entre la programación orientada a objetos y otros paradigmas. 	<ul style="list-style-type: none"> - Cuadro sinóptico referente a las propiedades de un objeto. - Organizador gráfico con los componentes de la POO. - Diagrama UML con la representación los requerimientos para solución de un problema. - Cuadro comparativo donde se representa la diferencia entre el paradigma de programación orientada a objetos y otros paradigmas de programación. 	<p>5.Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.</p> <p>5.6.Utiliza las tecnologías de la información y comunicación para procesar e interpretar información.</p>	<p>CO-12 Utiliza las tecnologías de la información y comunicación para investigar, resolver problemas, producir materiales y transmitir información.</p>	<p>Básica:</p> <ul style="list-style-type: none"> - Emplea la metodología orientada a objetos como una herramienta de desarrollo de aplicaciones informáticas como base para el uso de diferentes plataformas de desarrollo de software en el ámbito de su carrera y de desarrollo profesional.



<ul style="list-style-type: none"> - Aplica la metodología de programación orientada a objetos. - Desarrolla clases y sus componentes; propiedades y métodos aplicando los diferentes tipos de accesos. - Define la independencia de instancias, como característica de un objeto. - Utiliza los diferentes métodos de una clase (estáticos, en línea, constantes, constructor y destructor por defecto), para complementar la funcionalidad de un objeto. - Aplica la sobrecarga de operadores, funciones y constructores para la reutilización de código. - Emplea el apuntador this como herramienta de referencia de un objeto. - Implementa los diferentes tipos de herencia (simple y múltiple). 	<ul style="list-style-type: none"> - Conjunto de programas basados en clases, constructores y métodos sobre cada uno de los aprendizajes esperados. - Cuadro comparativo sobre los diferentes tipos de métodos en el diseño de clases. - Cuadro sinóptico sobre las normas y sintaxis de los métodos y operadores sobrecargados. 	<p>5.Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.</p> <p>5.6.Utiliza las tecnologías de la información y comunicación para procesar e interpretar información.</p>	<p>CO-12 Utiliza las tecnologías de la información y comunicación para investigar, resolver problemas, producir materiales y transmitir información.</p>	<p>Básica:</p> <ul style="list-style-type: none"> - Emplea la metodología orientada a objetos como una herramienta de desarrollo de aplicaciones informáticas como base para el uso de diferentes plataformas de desarrollo de software en el ámbito de su carrera y de desarrollo profesional.
---	---	--	--	--



<ul style="list-style-type: none"> - Implementa el polimorfismo y la especialización de métodos a través de las funciones virtuales. - Crea e implementa clases abstractas usando funciones virtuales puras y apuntadores. - Crea de plantillas para el almacenamiento genérico de datos. 	<ul style="list-style-type: none"> - Conjunto de programas sobre cada uno de los aprendizajes esperados, como son: herencia simple, herencia múltiple, métodos virtuales. - Diseños de cada programa, aplicando los elementos de la programación orientada objetos como son: herencia simple y múltiple, polimorfismo y funciones virtuales. 	<p>5.Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.</p> <p>5.6.Utiliza las tecnologías de la información y comunicación para procesar e interpretar información.</p>	<p>CO-12 Utiliza las tecnologías de la información y comunicación para investigar, resolver problemas, producir materiales y transmitir información.</p> <p>COE-11 Aplica las tecnologías de la información y la comunicación en el diseño de estrategias para la difusión de productos y servicios, en beneficio del desarrollo personal y profesional.</p>	<p>Básica:</p> <ul style="list-style-type: none"> - Emplea la metodología orientada a objetos como una herramienta de desarrollo de aplicaciones informáticas como base para el uso de diferentes plataformas de desarrollo de software en el ámbito de su carrera y de desarrollo profesional. <p>Extendida:</p> <ul style="list-style-type: none"> - Crea aplicaciones para resolver problemáticas a la medida mediante técnicas y procedimientos orientados a objetos.
--	--	--	--	---



<ul style="list-style-type: none"> - Representa información empleando XML como medio estructurado de datos. - Almacena información en archivos empleando XML como medio alternativo a un almacenamiento binario. - Define las características y estados de un socket. - Crea un sistema que envía, recibe y procesa información entre distintos equipos empleando sockets e información estructurada en XML. 	<ul style="list-style-type: none"> - Cuadro descriptivo sobre los componentes XML. - Un sistema de almacenamiento usando XML. - Un sistema de comunicación con XML. - Diseños de cada sistema aplicando las metodologías orientadas a objetos. 	<p>5.Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos.</p> <p>5.6.Utiliza las tecnologías de la información y comunicación para procesar e interpretar información.</p>	<p>CO-12 Utiliza las tecnologías de la información y comunicación para investigar, resolver problemas, producir materiales y transmitir información.</p> <p>COE-11 Aplica las tecnologías de la información y la comunicación en el diseño de estrategias para la difusión de productos y servicios, en beneficio del desarrollo personal y profesional.</p>	<p>Básica:</p> <ul style="list-style-type: none"> - Emplea la metodología orientada a objetos como una herramienta de desarrollo de aplicaciones informáticas como base para el uso de diferentes plataformas de desarrollo de software en el ámbito de su carrera y de desarrollo profesional. <p>Extendida:</p> <ul style="list-style-type: none"> - Crea aplicaciones para resolver problemáticas a la medida mediante técnicas y procedimientos orientados a objetos.
--	--	--	--	---

