|  |  |  |
| --- | --- | --- |
|  | **FUNDACION UNIVERSITARIA UNINPAHU** | |
| **VICERRECTORIA ACADÉMICA** | |
| **FORMATO DE ESPACIO ACADÉMICO** | |
| **FACULTAD: Ingeniería y Tecnologías de la Información** | |
| **PROGRAMA: Ingeniería de Software** | |
| **TIPO DE FORMACIÓN:** | |
| **NOMBRE DEL ESPACIO ACADÉMICO:** Lenguaje de Programación (Associate) | |
| **Carácter del espacio académico: TEÓRICO: TEÓRICO-PRÁCTICO: X PRÁCTICO:** | |
| **Código: IS1621** | **Periodo Académico: 20152** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Número Total de Créditos**  **Académicos:** | **Número Total de Horas**  **Semanales:** | | **Número Total de Horas**  **Semanales de**  **Trabajo Directo:** | **Número Total de Horas Semanales de**  **Trabajo**  **Independiente:** | **Modalidad** | |
| Presencial | X |
| 3 | 9 | | 3 | 6 | Distancia |  |
| Virtual |  |
| **Componente al que pertenece:** | |  | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **JUSTIFICACIÓN** | | | | | | | | | |
| En el marco de la ingeniería de Software, uno de los principales artefactos esperados por parte del mercado son las aplicaciones multipropósito (software a medida, aplicaciones de uso general, juegos, etc.) que se requieren por diferentes sectores, y/o que representan una oportunidad genuina de negocio.  Para dar solución a esta necesidad, una de las principales técnicas utilizadas en la actualidad, es el uso del paradigma de programación orientada a objetos y su utilización en las diferentes plataformas que se encuentran disponibles en la actualidad. De igual manera, en la actualidad existe un sinnúmero de lenguajes de programación que permiten dar solución a muchos tipos de problemas con el uso de las técnicas mencionadas anteriormente.  Con base en lo anterior, es necesario que cada aprendiz pueda extender el uso de sus conocimientos previos en programación orientada a objetos para así diseñar e implementar soluciones integrales a problemáticas reales en diferentes campos de acción. | | | | | | | | | |
| **INTENCIONALIDAD PEDAGÓGICA** | | | | | | | | | |
| **Objetivos de aprendizaje** | | | | **Objetivos de enseñanza** | | | | | |
| Al finalizar el curso de Lenguaje de Programación (Associate), el aprendiz será capaz de:  Generar soluciones en diferentes plataformas mediante la utilización del paradigma de programación orientada a objetos con al menos un lenguaje de programación de alto nivel. | | | | Orienta y articula la teoría y práctica de los conceptos en el marco de la programación orientada a objetos. | | | | | |
| **METODOLOGÍA** | | | | | | | | | |
| Este curso se desarrolla mediante la utilización de talleres teórico-prácticos que se pueden manejar de diversas maneras, entre las que se encuentran la exposición por parte del docente de los conceptos del tema para luego proponer un ejercicio en el que se apliquen dichos conceptos; por otro lado, es posible hacer uso del video beam para mostrar la manera de utilizar una determinada solución y/o manera de codificación con el objetivo de que los aprendices puedan realizar a la par del docente el proceso de codificación y/o solución de un determinado problema propuesto.  Del mismo modo, cada una o dos semanas (según el tema lo amerite) se propondrá una situación problema que permitirá al aprendiz aplicar los conceptos vistos anteriormente y adicionalmente realizar un proceso de investigación para promover y mejorar su capacidad de solución de problemas. Estos talleres se entregarán a través del aula virtual y serán parte fundamental del proceso de enseñanza-aprendizaje de este curso. | | | | | | | | | |
| **UNIDADES Y TEMATICAS** | | | | | | | | | |
| **Unidad** | **Tema** | | | | | **Semanas** | | **Tipo de contenido (actitudinales, conceptuales o procedimentales)** | |
| 1 | Conceptos principales de la programación orientada a objetos | | | | | 4 | | * Herencia. * Encapsulamiento. * Polimorfismo. | |
| 2 | Separación del código por capas y responsabilidades (serialización en XML) | | | | | 4 | | * Arquitectura de aplicaciones por capas. * Escritura de archivos planos. * Serialización de objetos en XML. | |
| 3 | Manejo de excepciones, globalización y localización | | | | | 4 | | * Manejo de excepciones: bloque try, catch, finally. * Uso de throw. * Excepciones personalizadas. * Uso de recursos para la globalización y localización de aplicaciones. | |
| 4 | Aplicaciones web y servicios web | | | | | 4 | | * Diferencia entre una aplicación web y un servicio web. * Principales características de una aplicación web: ciclo de vida de una página. * Principales características de un servicio web: contratos y data annotations. | |
| **EVALUACIÓN DEL APRENDIZAJE** | | | | | | | | | |
| **Competencia Global** | | **Saberes esenciales** | **Unidades de Competencia / Rango de aplicación** | | | | | | |
| **Insuficiente**  **(0-29)** | | **Aceptable**  **(30-39)** | | **Sobresaliente**  **(40-45)** | | **Excelente**  **(46-50)** |
| Diseña e implementa soluciones mediante la programación orientada a objetos. | | HACER  CONOCER  SER | No reconoce los componentes principales de la programación orientada a objetos. | | Hace uso parcial de las principales características de la programación orientada a objetos, aunque no conceptualiza plenamente todos los conceptos. | | Reconoce y hace uso de los principales conceptos de la programación orientada a objetos como herencia, encapsulamiento, polimorfismo, abstracción y reutilización de código entre otros. | | Es capaz de proponer soluciones orientadas a objetos e implementarlas en algún lenguaje de programación. |
| Diseña artefactos de software separados en capas de responsabilidades. | | HACER  CONOCER  SER | No implementa soluciones orientadas a objetos por capas. | | Diseña soluciones en diferentes capas, aunque la implementación no es completa. | | Diseña e implementa soluciones en capas en las que están completamente diferenciadas las responsabilidades (presentación y negocio) aunque el acceso a datos no es completo. | | Diseña e implementa soluciones completas en tres capas, haciendo uso de la serialización en XML como capa de datos. |
| Diseña e implementa soluciones en diferentes idiomas y culturas con un manejo adecuado del control de excepciones. | | HACER  CONOCER  SER | No realiza la implementación de aplicaciones en múltiples idiomas y culturas con un manejo apropiado del control de excepciones. | | Diseña e implementa aplicaciones multi-idioma/cultura, aunque no realiza un manejo adecuado del control de excepciones. | | Diseña e Implementa aplicaciones multi-idioma/cultura, con un manejo parcial del control de excepciones. | | Diseña e Implementa aplicaciones multi-idioma/cultura, con un manejo adecuado del control de excepciones. |
| Conoce los conceptos básicos de la implementación de aplicaciones y servicios web. | | HACER  CONOCER  SER | No identifica las diferentes herramientas para la implementación de aplicaciones y servicios web. | | Reconoce parcialmente los diferentes elementos para la implementación de aplicaciones y servicios web. | | Reconoce los diferentes elementos para la implementación de aplicaciones y servicios web. | | Reconoce y domina los conceptos principales para la implementación de aplicaciones y servicios web. |
| **PREGUNTAS ORIENTADORAS** | | | | | | | | | |
| ¿Qué conocimientos y habilidades se requieren para dar solución a los diferentes problemas que se presentan en el mercado de la Ingeniería de Software? | | | | | | | | | |
| **RELACIÓN CON LA INVESTIGACIÓN Y LA PROYECCCIÓN SOCIAL DEL PROGRAMA** | | | | | | | | | |
| * **Relación del espacio académico con la investigación en el programa:** este curso fortalece las habilidades de cada uno de los aprendices para dar solución a problemas con diferentes estrategias dentro de la programación orientada a objetos. * **Relación del espacio académico con la proyección social del programa**: este curso brinda una iniciación al aprendiz a la creación de soluciones integrales a la medida, así como también la generación de oportunidades de negocio. | | | | | | | | | |
| **BIBLIOGRAFÍA, HEMEROGRAFÍA Y WEBGRAFÍA COMPLEMENTARIA** | | | | | | | | | |
| **Bibliografía básica**   * Dimitrios Kalemis. The Fundamental Concepts of Object-Oriented Programming. CreateSpace Independent Publishing Platform (October 7, 2013). * Jim Keogh (Author), Mario Giannini (2004). OOP Demystified 1st Edition. McGraw-Hill Education; 1 edition (March 8, 2004) | | | | | | | | | |
| **Bibliografía complementaria**  Dan Clark. Beginning C# Object-Oriented Programming (Expert's Voice in .NET) 2nd Edition. Apress; 2 edition (March 26, 2013).  Bart Baesens, Aimee Backie, Seppe vanden Broucke. Beginning Java Programming: The Object-Oriented Approach 1st Edition. Wrox; 1 edition (March 2, 2015)  Peter Coad, Jill Nicola. Object-Oriented Programming 1st Edition. Prentice Hall; 1 edition (February 13, 1993)  Microsoft Official Academic Course. Exam 98-361 MTA Software Development Fundamentals 1st Edition. Wiley; 1 edition (May 3, 2011). | | | | | | | | | |
| **Cybergrafía:**   * [**Fundamentos de la POO**](http://biblat.unam.mx/pt/revista/tecnologia-y-comunicacion-educativas/articulo/principios-de-diseno-instruccional-de-entornos-de-aprendizaje-apoyados-con-tic-un-marco-de-referencia-sociocultural-y-situado)   https://msdn.microsoft.com/es-es/library/bb972232.aspx   * **Globalización y localización.**   https://msdn.microsoft.com/es-es/library/ms753931(v=vs.110).aspx | | | | | | | | | |
| **Audiovisuales:**  <http://tecnologias.inpahu.edu.co/moodle/>  <https://www.youtube.com/channel/UCppoTYKswPwFoZi-tvGEYrA> | | | | | | | | | |
| **Enlaces en Internet**   * Fundamentos de C# para absolutos principiantes <http://www.microsoftvirtualacademy.com/training-courses/c-fundamentals-for-absolute-beginners_esn> * Curso Aprendiendo a Programar <http://www.microsoftvirtualacademy.com/training-courses/curso-aprendiendo-a-programar> * Software Development Fundamentals. <https://www.microsoftvirtualacademy.com/en-US/training-courses/software-development-fundamentals-8248> * Ejemplos de Serialización XML. <https://msdn.microsoft.com/es-es/library/58a18dwa(v=vs.120).aspx> * Serialización de objetos en XML con C#.NET. <http://blog.jorgeivanmeza.com/2008/09/serializacion-de-objetos-en-xml-con-cnet/> * Uso de XML serializar y deserializar. <http://programadorjavaweb.blogspot.com/2012/02/uso-de-xml-serializar-y-deserializar.html> * Serialización de objetos Java en XML. <http://josedeveloper.com/2012/03/04/serializacion-de-objetos-java-en-xml/> * Fundamentos de funcionamiento de una aplicación web. <http://www.devjoker.com/contenidos/catss/518/Fundamentos-de-funcionamiento-de-una-aplicacion-web.aspx> * Web Services: Fundamentos. http://certified-es.blogspot.com/2009/05/web-services-fundamentos.html | | | | | | | | | |