|  |  |  |
| --- | --- | --- |
|  | **FUNDACIÓN UNIVERSITARIA UNINPAHU** | |
| **VICERRECTORIA ACADÉMICA** | |
| **FORMATO DE ESPACIO ACADÉMICO** | |
| **FACULTAD: INGENIERÍA Y TECNOLOGÍAS DE LA INFORMACIÓN** | |
| **PROGRAMA: TECNOLOGÍA EN COMPUTACIÓN Y DESARROLLO DE SOFTWARE** | |
| **TIPO DE FORMACIÓN: Tecnológica** | |
| **NOMBRE DEL ESPACIO ACADÉMICO: Metodología de Software** | |
| **Carácter del espacio académico: TEÓRICO: TEÓRICO-PRÁCTICO: X PRÁCTICO:** | |
| **Código: SI1142** | **Código: SI1142** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **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:** | | Desarrollo de Software | | | | |
| **JUSTIFICACIÓN** | | | | | | |
| En esta guía pretende mediante una información precisa y clara definir los contenidos, la metodología, las estrategias y actividades pedagógicas para que el estudiante desarrolle las habilidades y aptitudes requeridas que le garanticen un desempeño óptimo en esta disciplina.  La ingeniería de software es una disciplina o área de la Informática o Ciencias de la Computación que ofrece métodos y técnicas para desarrollar y mantener software de calidad a costos razonables que resuelvan problemas de todo tipo. Con el trabajo adelantado en esta asignatura el alumno desarrollara competencias en lo relacionado con la Ingeniería de Software y en lo particular lo concerniente en la construcción de Sistemas de Información. | | | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **INTENCIONALIDAD PEDAGÓGICA** | | | | | | | | | |
| **Objetivos de aprendizaje** | | | | | **Objetivos de enseñanza** | | | | |
| Al finalizar el curso el estudiante estará en capacidad de:  CONOCER   * Los fundamentos básicos de la gestión y planificación de proyectos de software.   HACER   * Desarrollar habilidades en las técnicas de análisis y diseño de sistemas de información.   SER   * Plantear una solución a un problema planteado como analista y desarrollador de software. | | | | | Propiciar un espacio de aprendizaje y manipulación acerca de la de la ingeniería del software, tomando referencia los métodos y tecnologías modernas de desarrollo y los principios y procesos definidos en los modelos internacionales de calidad de software. | | | | |
| **METODOLOGÍA** | | | | | | | | | |
| Las sesiones se realizarán en equipos de trabajo e individualmente, de acuerdo al tema, estos trabajos deben ser socializados ante el grupo y el docente durante el tiempo de acompañamiento directo.  Acompañamiento directo es el trabajo conjunto de los estudiantes y el docente durante una sesión, en este tiempo el docente explica a los estudiantes los temas propuestos y resuelve preguntas a las inquietudes de los estudiantes, es importante anotar que algunas sesiones serán de trabajo dirigido, esto con el fin de observar el desempeño de los estudiantes en talleres propuestos dentro del aula de clase.  Acompañamiento indirecto es el apoyo y la orientación del docente al trabajo independiente del estudiante.  Trabajo independiente son todas aquellas actividades (lecturas, talleres, ensayos, reseñas, entre otras) que debe realizar el estudiante para hacer aprehensión de los temas tratados durante el acompañamiento directo o para preparar con antelación.  Practica: Talleres de laboratorio de los temas avanzados y de redes de computadoras | | | | | | | | | |
| **UNIDADES Y TEMATICAS** | | | | | | | | | |
| **Unidad** | **Tema** | | | | | **Semanas** | | **Tipo de contenido (actitudinales, conceptuales o procedimentales)** | |
| 1 | **LA EVOLUCION DEL SOFTWARE**  Que es el software  Clasificación del software  Evolución del software  Características del software  Competitividad del Software  Ventajas  Aplicaciones | | | | | 1 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 2 | **INTRODUCCION A LA INGENIERIA DE SOFTWARE**  Que es la Ingeniería de software  Problemas por los cuales surge la Ingeniería de software  Evolución  Aplicaciones  Ciclo de vida  Perfil del Ingeniero de Software | | | | | 1 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 3 | **GESTION PROYECTOS DE SOFTWARE**    Que es gestionar  Diferencias entre Planificar y gestionar Software  La Arquitectura de Software  Las 4 P  Proyecto  Proceso  Proceso  Producto  Estimación de Proyectos de Software.  Gestión, control y garantía de la calidad del Software.  Modelo de Madurez | | | | | 2 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 4 | **GESTION RIESGOS**  Identificación del Riesgo  Análisis del Riesgo  Control del Riesgo  Planes de Contingencia | | | | | 1 | | CONCEPTUALES U PROCEDIMENTALES | |
| 5 | **FACTIBILIDADES**  Factibilidad Técnica  Factibilidad Económica  Factibilidad Operacional  Factibilidad de Cronograma | | | | | 1 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 6 | **DIAGRAMAS**  Diagramas de contexto  Diagrama de Flujo  Diagramas de entidad-relación | | | | | 2 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 7 | **INTRODUCCION A UML**  Introducción a UML  Diagramas de UML  Objetos  EL PARADIGMA DE OBJETOS  El Modelo de Objetos: Evolución, Fundamentos y Elementos del modelo de  Objetos – Aplicación del modelo de objetos.  Clases y Objetos: La naturaleza de los Objetos – Relaciones entre objetos – La naturaleza de una Clase – Relaciones entre clases – La Interacción entre clases y objetos.  Clasificación: La importancia de una clasificación correcta – Identificando clases y objetos – enfoques Clásicos – Análisis del Comportamiento – Análisis de Dominios – Análisis de Casos de Uso – Otros enfoques.  La Notación: Elementos de la Notación – Diagramas de Clases - Diagramas de Objetos - Ejemplos | | | | | 3 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 8 | **ETAPA DE DISEÑO, CODIFICACIÓN Y PRUEBAS**  La Especificación D: Diseño Estructural – Diseño Detallado – Técnicas de Diseño Script – Fichas Técnicas de Servicios de Objeto.  Diseño de Datos: Modelo Lógico de Datos – Modelo Físico de Datos – Documentación.  Codificación: Elección del Lenguaje – Documentación del Código – Reglas Generales.  Pruebas: Test de Caja Negra – Test de Caja Blanca – Test de Integración – Test de Aceptación. | | | | | 2 | | CONCEPTUALES Y PROCEDIMENTALES | |
| 9 | **DOCUMENTACION**  Tipos de manuales | | | | | 1 | | CONCEPTUALES Y PROCEDIMENTALES | |
| **EVALUACIÓN DEL APRENDIZAJE** | | | | | | | | | |
| Competencia Global (Tobón (2004)) | | Saberes esenciales | Unidades de Competencia / Rango de aplicación | | | | | | |
| Insuficiente  (0-29) | Aceptable  (30-39) | | | Sobresaliente  (40-45) | | Excelente  (46-50) |
| Conoce, evalúa y aplica enfoques modernos en el desarrollo de software, como tecnología de orientación por objetos, patrones y componentes, que permitan el desarrollo de aplicaciones robustas, flexibles y reutilizables. | | HACER  CONOCER  SER | No cumple con los requerimientos mínimos de la unidad de competencia | Emplea conceptos de análisis desarrollar ciertos procesos | | | Conoce y maneja adecuadamente el análisis para abordar desarrollos en el lenguaje de programación. | | En todas las situaciones planteadas, el estudiante emplea correctamente el análisis de sistemas de información. |
| Evaluar el proceso de desarrollo de software e implantar los principios de gestión y aseguramiento de calidad de software de acuerdo a las exigencias de modelos internacionales de calidad como SPICE (ISO TR 15504), CMM del SEI e ISO 9001. | | HACER  CONOCER  SER | No cumple con los requerimientos mínimos de la unidad de competencia | Aplica las técnicas y normas planteadas en UML. | | | Aplica los procedimientos adecuados y de manera correcta en la resolución de problemas mediante el uso de UML. | | Resuelve clara y correctamente los ejercicios planteados, seleccionando el procedimiento más apropiado y pertinente acorde con el lenguaje UML. |
| Adquirir principios y herramientas para la planeación, supervisión y control de proyectos de desarrollo de software. | | HACER  CONOCER  SER | No cumple con los requerimientos mínimos de la unidad de competencia | Define los parámetros en el análisis y diseño de sistemas de información. | | | El estudiante se relaciona activamente para comunicar y compartir sus conocimientos. | | Expresa de manera clara y precisa los conocimientos aprendidos, empleando un lenguaje adecuado. |
| **PREGUNTAS ORIENTADORAS** | | | | | | | | | |
| ¿La aplicación de las normas y técnicas para el desarrollo de sistemas de información previstas en UML corresponden a las exigencias habituales para el desarrollo de sistemas de información?    Es suficiente y necesario identificar claramente los procesos o etapas para la gestión y desarrollo de un sistema de información?  Involucrar a los usuarios finales en el proceso de análisis, diseño y gestión del sistema de información garantiza el cumplimiento de los objetivos previstos? | | | | | | | | | |
|  | | | | | | | | | |
|  | | | | | | | | | |
| **RELACIÓN CON LA INVESTIGACIÓN Y LA PROYECCCIÓN SOCIAL DEL PROGRAMA** | | | | | | | | | |
| * Las competencias desarrolladas permiten el desempeño del profesional en líneas de investigación utilizando las metodologías para el desarrollo de un proyecto informático. * El desarrollo de aplicaciones informáticas con las metodologías propuestas corresponden con las necesidades de los usuarios y empresas del mercado actual. | | | | | | | | | |
| **BIBLIOGRAFÍA, HEMEROGRAFÍA Y WEBGRAFÍA COMPLEMENTARIA** | | | | | | | | | |
| Bibliografía básica:  BOOCH G., "Análisis y Diseño Orientado a Objetos con Aplicaciones" - Segunda Edición - Editorial Addison-Wesley/Díaz de Santos - 1996.  PRESSMAN R., "Ingeniería del Software, un Enfoque Práctico" - Tercera Edición - Editorial Mc Graw-Hill - 1993.  RUMBAUGH J., "Modelado y Diseño Orientado a Objetos" – Editorial Prentice Hall – 1997.  RUMBAUGH J., JACOBSON I., BOOCH G., "El Lenguaje Unificado de Modelado. Manual de Referencia" - Editorial Addison-Wesley - 2000. | | | | | | | | | |
| Bibliografía complementaria:  LARMAN C., "UML y Patrones" - Segunda Edición - Editorial Prentice-Hall – 2003.  ODSTRCIL M., "Apuntes de Conceptos de Sistemas y Análisis Orientado a Objetos" - Primera Edición - FACET, UNT - 2001.  YOURDON, E., "Análisis Estructurado Moderno" - Primera Edición - Editorial Prentice Hall - 1993 | | | | | | | | | |
| Cybergrafía  <http://www.istpargentina.edu.pe/wp-content/uploads/2015/12/Metodologias_de_desarrollo_de_software.pdf>  <http://www.rete.mx/attachments/article/4/RETE%20-%20Metodolog%C3%ADa%20de%20desarrollo%20de%20aplicaciones.pdf>  <http://www.codecompiling.net/files/slides/IS_clase_13_metodos_y_procesos.pdf> | | | | | | | | | |
| Audiovisuales  https://www.youtube.com/watch?v=fcd7p1jaKI4 | | | | | | | | | |
| Enlaces de Internet  <http://www.mintic.gov.co/portal/604/w3-channel.html>  [www.acac.org](http://www.acac.org) Asociación Colombiana para el avance de la Ciencia: A.C.A.C  [www.colciencias.gov.co](http://www.colciencias.gov.co) Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología Francisco José de Caldas: COLCIENCIAS: | | | | | | | | | |