**Tecnológico Nacional de México**

**Subdirección Académica**

***Instrumentación Didáctica para la Formación y Desarrollo de Competencias Profesionales***

|  |  |
| --- | --- |
| Periodo | Agosto-Diciembre 2017 |

|  |  |
| --- | --- |
| Nombre de la Asignatura: | Fundamentos de Ingeniería de Software |
| Plan de Estudios: | Ingeniería en Sistemas Computacionales |
| Clave de la Asignatura: | SCC-1007 |
| Horas teoría-horas prácticas-Créditos: | 2-2-4 |

1. **Caracterización de la asignatura:**

|  |
| --- |
| Es una introducción a la Ingeniería de Software que involucra la comprensión de conceptos, metodologías, técnicas y herramientas para la elaboración del análisis de un proyecto a partir de un modelo de negocios |

1. **Intención didáctica:**

|  |
| --- |
| La asignatura debe ser abordada desde un enfoque teórico práctico, aplicando los conocimientos de las fases y metodologías del desarrollo de software, a fin de obtener el modelo de negocios y el modelo de análisis para un proyecto que servirá de base en las siguientes asignaturas del área de Ingeniería de Software. Los temas del curso comprenden los siguientes aspectos: Conceptos básicos, modelo de negocios, modelo de análisis y calidad de software. |

1. **Competencia de la asignatura:**

|  |
| --- |
| Realiza el análisis de un proyecto de software, a partir de la identificación del modelo de negocios de la organización que permita alcanzar estándares y métricas de calidad. |

1. **Análisis por competencias específicas:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Competencia No. |  | Fundamentos de Ingeniería de software | Descripción | Identifica y comprende el entorno de aplicación de la ingeniería de software |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Temas y subtemas para desarrollar la competencia específica | Actividades de aprendizaje | Actividades de enseñanza | Desarrollo de competencias genéricas | Horas teórico-práctica |
| 1.1. Conceptos básicos  1.2. Fases de la Ingeniería de software.  1.3 Metodologías de desarrollo de software (Clásicas y Ágiles)  1.4. Importancia de las herramientas CASE en la Ingeniería de software. | El alumno define el concepto de Ingeniería del software.  El alumno elabora un resumen de las fases de la ingeniería de software | El docente junto al alumno analiza diferentes software y ver las funcionalidades que ofrece  El docente hace diferencia entre las diferentes metodologías existentes para el desarrollo de software | Capacidad de análisis y síntesis  Capacidad de organizar y planificar  Comunicación oral y escrita  Habilidad para buscar y analizar información proveniente de fuentes  diversas  Trabajo en equipo Compromiso ético |  |

|  |  |
| --- | --- |
| Indicadores de Alcance | Valor de Indicador |
| 1. Comprende el concepto de Ingeniería de Software | 10% |
| 1. Conozca las metodologías de desarrollo de Software | 30% |
| 1. Comprende la metodología de desarrollo de Software:Ágil | 40% |
| 1. Comprende la diferencia entre metodologías ágiles y clásicas | 20% |

Niveles de desempeño:

|  |  |  |  |
| --- | --- | --- | --- |
| Desempeño | Nivel de desempeño | Indicadores de Alcance | Valoración numérica |
| Competencia Alcanzada | Excelente | Cumple en su totalidad con las competencias señaladas. | 95-100 |
| Notable | Cumple con la totalidad de A, B, C y parcialmente D | 85-94 |
| Bueno | Cumple con la totalidad de A, B, C y parcialmente C | 75-84 |
| Suficiente | Cumple con la totalidad de A y B y parcialmente C y D | 70-74 |
| Competencia No Alcanzada | Insuficiente | Cumple con la totalidad de A, B, C y parcialmente C | N. A. |

Matriz de Evaluación:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Evidencia de Aprendizaje** | **%** | **Indicador de Alcance** | | | | | **Evaluación formativa de la competencia** |
| A | B | C | D | N |
| Examen | 60% | 10% | 10% | 20% | 20% |  | El alumno responderá completa y correctamente un examen escrito relacionado con el contenido de la materia. |
| Investigación | 20% |  | 10% | 10% |  |  | El alumno elaborará una investigación con los temas relacionados a la competencia. |
| Cumplimiento | 20% |  | 10% | 10% |  |  | El alumno entregará completos y en el tiempo solicitado la investigación |
| Total | | 10% | 30% | 40% | 20% |  |  |

1. Fuentes de información y apoyos didácticos:

|  |  |
| --- | --- |
| Fuentes de información: | Apoyos didácticos |
| Braude, E. (2003). Ingeniería de Software, una perspectiva orientada a objetos. México: Alfaomega.  Bruegge, B., Dutoit, A. (2002). Ingeniería de Software Orientada a Objetos. México: Prentice Hall.  Fontela, C. (2011). UML Modelado de Software para Profesionales. Argentina: Alfaomega.  Fowler, M. (2000). UML Gota a Gota. (1ª Ed.). México: Pearson.  Jacobson, I., Booch, G., Rumbaugh, J. (2003). El Proceso Unificado de Desarrollo de Software. España: Addison Wesley.  Kendall, E. (2011). Análisis y Diseño de Sistemas. (8ª Ed.). México: Pearson Educación.  Laudon, K. (2004) Sistemas de Información Gerencial. (8ª Ed.). México: Pearson Educación.  Minguet, M. (2003). La Calidad del Software y su Medida. España: Universitaria Ramón Areces. | Internet |

1. Calendarización de evaluación en semanas (6)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Semana | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| TP | ES | ES |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| TR |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| SD |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

TP: Tiempo Planeado

ED: Evaluación diagnóstica

TR: Tiempo Real

EFn: Evaluación formativa (Competencia específica n)

SD: Seguimiento departamental

ES: Evaluación sumativa

|  |  |
| --- | --- |
| Fecha de elaboración | 18 de Agosto de 2017 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| M.C. Adriana Ramírez Hernández |  | M.C. ADRIÁN A. TREVIÑO BECERRA |