**Tecnológico Nacional de México**

**Subdirección Académica**

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

|  |  |
| --- | --- |
| Periodo | Enero – Junio 2018 |

|  |  |
| --- | --- |
| Nombre de la Asignatura: | Tópicos Avanzados de Programación |
| Plan de Estudios: | ISIC-2010-224 |
| Clave de la Asignatura: | SCD-1027 |
| Horas teoría-horas prácticas-Créditos: | 2-3-5 |

1. **Caracterización de la asignatura:**

|  |
| --- |
| Esta asignatura apoya en la implementación de aplicaciones computacionales para solucionar problemas de diversos contextos, integrando diferentes tecnologías, plataformas o dispositivos, por medio del desarrollo de software utilizando programación concurrente, acceso a datos, que soporte interfaz gráfica de usuario e incluya programación móvil.    Para el logro de los objetivos es necesario que el estudiante tenga competencias previas en cuanto a paradigmas de programación, el uso de metodologías para la solución de problemas mediante la construcción de algoritmos utilizando un lenguaje de programación orientada a objetos, el manejo de conceptos básicos de Hardware y Software, construcción de modelos de software empleando diagramas de clases.    Para adquirir la competencia planteada en esta asignatura es necesario que el estudiante haya acreditado la asignatura de Programación Orientada a Objetos y evitar cursarla en semestres muy avanzados tomando en cuenta que en esta asignatura el estudiante desarrollará competencias necesarias para cursos posteriores entre los que se encuentran los talleres de bases de datos y programación web |

1. **Intención didáctica:**

|  |
| --- |
| Se organiza el temario, en cinco unidades, en la primera tema se estudia la Interfaz Gráfica de Usuario (GUI), dependiendo de las herramientas utilizadas en los cursos anteriores se puede elegir la misma herramienta o una distinta con el fin de que el estudiante aprenda a utilizar los componentes gráficos que brinde el entorno de desarrollo, que incluya el manejo de eventos y que desarrolle nuevos componentes derivados de los ya existentes pero que incluyan nuevas propiedades.    Es recomendable considerar los conocimientos previos del grupo y las herramientas de desarrollo con las que están familiarizados. La resolución de problemas como una herramienta resulta eficaz para aprender estos conocimientos, partiendo de la definición de un problema real que pueda ser resuelto utilizando la programación de eventos y requiera de una interfaz gráfica.    El segundo tema se enfoca al estudio de la creación y manejo de librerías y componentes (estos términos pueden variar según el lenguaje de programación utilizado). Se entiende como librería a un conjunto de bloques de códigos de programación normalmente compilados, que pueden ser incluidos en una aplicación para su uso. Y a un componente como una clase de uso específico, lista para usar, que puede ser configurada o utilizada de forma visual, desde el entorno de desarrollo. Este tema debe enfatizar la creación de nuevas librerías y componentes y evitar el estudio exhaustivo de las que incluya el entorno de desarrollo, a estas últimas enfocarse solo en revisar la forma de utilizarlos.  En el tercer tema se aborda el tema de programación concurrente requiere de iniciar con el estudio a nivel conceptual sobre los hilos y su funcionamiento, y después ir implementando aplicaciones multihilos. Uno de los puntos más sensibles es la sincronización por lo que deben hacerse hincapié en una buena implementación. Para este tema es recomendable hacer prácticas sencillas para comprender la funcionalidad de los hilos, el manejo de sus estados y la sincronización, para finalmente desarrollar aplicaciones que usen la programación concurrente en la resolución de problemas reales.    En el cuarto tema se aborda el tema de acceso a datos, donde se requiere aprender cómo realizar la conexión con diferentes orígenes de datos, su manipulación y visualización. El quinto tema se refiere al estudio de la programación de dispositivos móviles, la intención de este tema es realizar un estudio a nivel introductorio sobre las distintas tecnologías que hay en el mercado, y desarrollar aplicaciones sencillas para esta clase de dispositivos. |

1. **Competencia de la asignatura:**

|  |
| --- |
| Desarrolla soluciones de software para resolver problemas en diversos contextos utilizando programación concurrente, acceso a datos, que soporten interfaz gráfica de usuario y consideren dispositivos móviles. |

1. **Análisis por competencias específicas:**

|  |  |  |  |
| --- | --- | --- | --- |
| Competencia No. | 2. Componentes y librerías | Descripción | Diseña e implementa componentes y librerías para lograr la reutilización de código |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Temas y subtemas para desarrollar la competencia específica | Actividades de aprendizaje | Actividades de enseñanza | Desarrollo de competencias genéricas | Horas teórico-práctica |
| 2.1 Definición conceptual de componentes, paquetes / librerías.  2.2 Uso de librerías proporcionadas por el lenguaje.  2.3 Creación de componentes (visuales y no visuales) definidos por el usuario  2.4 Creación y uso de paquetes/librerías definidas por el usuario. | Realizar prácticas que permitan familiarizarse con el uso de los componentes y elaborar su correspondiente reporte.  Realizar una investigación sobre la forma de crear nuevos componentes, elaborando un mapa de secuencias, discutirlo en plenaria.  Presentar los avances de los proyectos al grupo que incluyan GUI, eventos, componentes y librerías para enriquecerlos con las opiniones del docente y de los estudiantes. | El docente explicará la importancia de crear controles de usuario  Planteara programas en WPF que utilicen controles de usuario para mostrar información | Solución de problemas. Toma de decisiones. Capacidad crítica y autocrítica.  Capacidad de trabajar en equipo.  Capacidad de comunicar sus ideas.  Capacidad de aplicar los conocimientos en la práctica.  Capacidad de aprender. Capacidad de generar nuevas ideas (creatividad).  Habilidad para trabajar en forma autónoma. |  |

|  |  |
| --- | --- |
| Indicadores de Alcance | Valor de Indicador |
| 1. Crea bibliotecas de clases y las referencia desde un proyecto | 20% |
| 1. Crear controles de usuario | 40% |
| 1. Utiliza controles de usuario para representar información en programas orientados a objetos | 40% |

Niveles de desempeño:

|  |  |  |  |
| --- | --- | --- | --- |
| Desempeño | Nivel de desempeño | Indicadores de Alcance | Valoración numérica |
| Competencia Alcanzada | Excelente | Cumple con todos los indicadores de alcance | 95-100 |
| Notable | Cumple con los indicadores B y C y parcialmente A | 85-94 |
| Bueno | Cumple con los indicadores A y B y parcialmente C | 75-84 |
| Suficiente | Cumple con el indicador B y parcialmente A y C | 70-74 |
| Competencia No Alcanzada | Insuficiente | No cumple con el indicador B | N. A. |

Matriz de Evaluación:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Evidencia de Aprendizaje** | **%** | **Indicador de Alcance** | | | | | **Evaluación formativa de la competencia** |
| A | B | C | D | N |
| Proyecto | 40% |  | 20% | 20% |  |  | Realiza un proyecto de un programa orientada a objetos donde utiliza controles de usuario |
| Practicas | 60% | 20% | 20% | 20% |  |  | Realiza practicas donde utiliza controles de usuario |
|  |  |  |  |  |  |  |  |
| Total | | 20% | 40% | 40% |  |  |  |

Fuentes de información y apoyos didácticos:

|  |  |
| --- | --- |
| Fuentes de información: | Apoyos didácticos |
| 1. Aguilar, L. J. (2010). Programacion en c/c++ java y UML. México: McGraw Hill. 2. Bell, D. (2011). Java para estudiantes. México: Pearson. 3. Ceballos, F. J. (2010). JAVA 2: Curso de programación. Madrid: RA-MA. 4. Dean, J. (2009). Introducción a la programación con Java. México: McGraw Hill. 5. Deitel, D. y. (2010). Java Cómo Programar. México: Prentice Hall. 6. Friesen, J. (2011). Java para desarrollo android. España: Anaya Multimedia. 7. Huddleston, R. (2011). Android para todos. España: Anaya Multimedia. 8. Lauren Darcey, S. C. (2012). Android 4. Madrid: Anaya Multimedia. 9. Soriano, J. E. (2011). Android: Programación de dispositivos móviles a través de ejemplos. México: Marcombo, S.A. |  |

1. Calendarización de evaluación en semanas:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Semana | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| TP |  |  |  | EF | EF | 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 | 23 de Enero de 2018 |

|  |  |  |
| --- | --- | --- |
| Ing. Héctor Javier Padilla Lara |  | M.C. Adrián Alberto Treviño Becerra |
| Nombre y firma del (de la) profesor(a) |  | Nombre y firma del(de la) Jefe(a) de Departamento Académico |