|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| logo_pucmm | | | | | | | | | |
| **PONTIFICIA UNIVERSIDAD CATÓLICA MADRE Y MAESTRA** | | | | | | | | | |
| **FACULTAD DE CIENCIAS DE LA INGENIERIA** | | | | | | | | | |
| **DEPARTAMENTO DE INGENIERÍA EN SISTEMAS Y COMPUTACIÓN** | | | | | | | | | |
| **PROGRAMA DE ASIGNATURA** | | | | | | | | | |
| **Asignatura:**  Programación II | | | | **Código:** ISC-307 | | | | | |
| **Año/Período:** 3er Año / 1er Período | | | | | |
| **Régimen:** Horas de Clases | | | | **Área a la que pertenece:** Programación | | | | | |
| **DISTRIBUCION DE HORAS DE CLASES** | | | | | |  | | | |
| **Teóricas** | | **Prácticas** | | **Total General** | |  | | | |
| **Horas** | **% Materia** | **Horas** | **% Materia** | **Horas** | **% Materia** |  | **T** | **P** | **C** |
| 42 | 60% | 28 | 40% | 70 | 100% |  | 3 | 2 | 4 |

|  |
| --- |
| **I-DESCRIPCIÓN GENERAL** |
| El Ingeniero en Sistemas diseña, desarrolla, prueba e implanta aplicaciones informáticas, utilizando distintas herramientas de programación como lenguajes informáticos, sistemas de bases de datos y entornos de desarrollo. Esta asignatura trata los aspectos fundamentales de la programación orientada a objetos dentro de un paradigma de programación visual apoyándose para lograrlo en la plataforma de desarrollo .NET y más concretamente en el lenguaje C#.  La asignatura se encarga por un lado de proporcionar al estudiante métodos de diseño de interfaces gráficas de usuarios y ubicarlas dentro de una metodología de análisis y diseño de aplicaciones informáticas; y por otro lado, de proporcionar los conocimientos básicos necesarios para desarrollar aplicaciones cliente-servidor, en aras de comenzar a desarrollarse en el ámbito productivo. |

|  |  |  |
| --- | --- | --- |
| **II-OBJETIVOS** | | |
| Objetivos Generales: | Potenciar en los estudiantes los fundamentos para el desarrollo de aplicaciones de software de producción utilizando el paradigma de programación visual. | |
| Objetivos Específicos: | Que los estudiantes sean capaces de:   1. Generar un análisis orientado a objetos de una aplicación informática, utilizando como base las necesidades expresadas por el usuario. 2. Definir y establecer las tareas que debe cubrir una interfaz gráfica de usuario. 3. Utilizar y combinar los patrones que conduzcan a la solución de problemas, codificándolos en el lenguaje de programación objeto de estudio. 4. Generar prototipos de aplicaciones para su evaluación. 5. Utilizar los recursos de ADO.NET, GDI+, y COM de la plataforma de desarrollo .NET. 6. Diseñar y desarrollar aplicaciones Cliente-Servidor utilizando modelos multicapas. 7. Diseñar reportes de usuarios finales con metodología adecuada. | |
| **III-INTERRELACIÓN CON OTRAS ASIGNATURAS** | | |
| Prerrequisitos: | | ISC-212 |
| Correquisitos: | | ISC-347 |
| Asignaturas que se aconseja debe haber cursado: | | ISC-213 |

|  |  |
| --- | --- |
| **IV-PROFESORADO** | |
| Profesor responsable: |  |
| Claustro de profesores: |  |
| **V-METODOLOGÍA** | |
| El curso será impartido a través de cátedras, desarrollo de prácticas y exposiciones que indicará el profesor. El proceso de aprendizaje será controlado a través de las prácticas, tareas y exámenes desarrollados durante el curso.  La distribución porcentual de los puntos asociados a las evaluaciones debe estar distribuido en: 70%(Eval. Parciales)+30%(Eval. Final). Las evaluaciones que cubran el 50% de los puntos de la asignatura deben realizarse previo a las fechas de retiro parcial y total. Los exámenes parciales tendrán una duración de 3-2 horas, deben ser confeccionados abordando contenidos teóricos y/o prácticos.  Las tareas serán orientadas con 15 días de anticipación a la fecha de entrega programada y su evaluación estará dividida en dos partes: la entrega formal del trabajo y la exposición oral en el curso de las soluciones encontradas, valorando su eficiencia y claridad.  La evaluación final (30%) deberá ser evaluada a través del desarrollo de un **Proyecto de Producción**, que involucre la asignatura de bases de datos, para lograr una vinculación y aplicación de los conocimientos adquiridos. El trabajo debe ser individual o como máximo grupos de 2 estudiantes; debe incluir la entrega de un documento (en el formato definido para la carrera) que describa el trabajo realizado y los ficheros de código fuente; la exposición de los trabajos debe realizarse a puertas abiertas con estudiantes y profesores del departamento como invitados. | |

|  |
| --- |
| **VI-PERFIL DEL EGRESADO** |
| Esta asignatura contribuye al perfil del egresado dándole las herramientas teóricas y prácticas necesarias para que desarrolle aplicaciones visuales enfocadas a la producción en una tecnología actual basada en un lenguaje de programación orientada a objetos y una definición cliente-servidor.  Competencias logradas:   * Capacidad de análisis y síntesis. * Capacidad de organización y planificación. * Capacidad de aplicar los conocimientos en la práctica. * Capacidad para trabajar en equipo. * Habilidad para trabajar de forma autónoma. * Capacidad para diseñar y debatir diferentes soluciones de un mismo problema. * Capacidad para analizar los puntos fuertes y débiles del diseño de interfaces gráficas de usuario. |

|  |
| --- |
| **VII-CONTENIDO DE LA TEORÍA** |
| **Lenguaje de programación propuesto: C#** |
| Tema 1. Diseño de interfaces gráficas de usuario.   1. Interacción Hombre-Máquina. 2. Modelo de procesamiento humano y modelos conceptuales. 3. Reseña histórica. Interfaces físicas. 4. Estándares diseño: usabilidad y principios universales.    1. Diseño centrado en el usuario.    2. Modelos de gestión de ventanas.    3. Normas de estilos. 5. Elementos de una interfaz: ventanas, formularios, menús, barras de herramientas y controles. 6. Concepción y diseño de ayudas de usuarios. 7. Estándares relacionados con el diseño de interfaces (ISO, IEEE). Metodologías relacionadas (UML, METRICA).   Tema 2. Fundamentos de la tecnología .NET.   1. Los productos: herramientas de desarrollo, servidores especializados, servicios web y dispositivos. 2. Componentes de la plataforma .NET: sistema operativo, servidores especializados, .NET framework, servicios componentes y Visual Studio .NET. 3. Arquitectura del .NET framework. 4. El Common Language Runtime (CLR). 5. El entorno del CLR. 6. Estructura de los ejecutables CLR. 7. Papel de la metadata. 8. Assemblies y manifests. 9. El lenguaje intermediador (Intermediate Language, IL). 10. El sistema de tipos común (Common Type System, CTS) y la especificación común de lenguajes (Common Language Specification, CLS). 11. Proceso de ejecución de los assemblies. 12. Comparación del CLR y la Java Virtual Machine.   Tema 3. **Principios del uso de patrones de diseño.**   1. Origen de la técnica de patrones. 2. Los patrones como componentes de diseño. 3. ¿Cómo se vincula el uso de patrones con los criterios de OOAD?. 4. Enfoque de la realización de patrones. 5. Los patrones fundamentales.   **Tema 4. Diseño y Desarrollo de Aplicaciones.**   1. Introducción. 2. Arquitectura Cliente-Servidor. 3. Modelo de Programación Multicapas. 4. Comparación con el modelo tradicional de programación. Ventajas. 5. Capa de Datos.    1. Evolución del acceso a datos: ODBC, RDO, ODBCDirect, OLE DB, ADO, RDS.    2. ADO.NET Características fundamentales y componentes. 6. Capa Intermedia o Reglas de Negocios. 7. Capa Interfaz.   **Tema 5: Reportes de salida.**   1. Conceptos y definiciones. 2. Características básicas. 3. Normas de estilos. 4. Componentes que existen para la elaboración de reportes. 5. Tipos de reportes. Generación de diferentes tipos de reportes: Listas simples, Listas agrupadas, Subreportes, Cross-tab, Gráficos. 6. Exportar información a diferentes formatos. |

|  |  |
| --- | --- |
| **VIII-GUIA DE LAS PRÁCTICAS** | |
| Las clases de laboratorio tienen como objetivo central la adquisición de habilidades prácticas. Es importante tener en cuenta los objetivos y distribución de contenidos de la teoría, para planificar los laboratorios.  **Guía para los laboratorios:**   1. Asignación de temas para el Proyecto Final. Análisis y diseño del proyecto. Desarrollo de la documentación necesaria: Modelo E/R, Casos de Usos, Esbozo de Interfaz, etc. Utilizar una herramienta para la documentación de sistemas como por ejemplo Balsamiq. 2. Prácticas para la familiarización con el entorno de .NET. 3. Prácticas para el aprendizaje en la creación y manipulación de patrones. Identificar y crear los patrones necesarios para el proyecto final. 4. Desarrollo de actividades para adquirir habilidades en el acceso a datos. Utilizar diferentes tipos de servidores de bases de datos. 5. Desarrollar actividades para identificar y subdividir el desarrollo de aplicaciones informáticas en un modelo de 3 capas. Definir esta división en el Proyecto Final. 6. Actividades para la programación de la capa de datos. 7. Actividades para la creación de la capa intermedia. 8. Actividades para la creación de la capa interfaz. 9. Desarrollar prácticas que aborden el tema de creación de reportes finales.   Durante el desarrollo de las actividades prácticas se debe ir controlando el desarrollo del proyecto final a través de tareas específicas vinculadas con los temas de la asignatura, de modo que el estudiante pueda ir desarrollando su proyecto en un orden lógico de procesos. | |
| **IX-EVALUACIÓN** | |
| Normas de evaluación: | La calificación final de cada estudiante será calculada sumando las calificaciones obtenidas en las prácticas, tareas, evaluaciones diarias, parciales y examen final.  Los aspectos a evaluar en cada caso estarán regidos por:   1. La eficiencia de la solución presentada. 2. El cumplimiento del procedimiento establecido. 3. La entrega en el marco de tiempo fijado. 4. La presentación, claridad y organización a la entrega. |
| Guía de Evaluaciones: | |  |  | | --- | --- | | 1ra Tarea  (10 ptos) | Diseño y concepción de una aplicación informática enfocada al mercado actual. Documentación del proyecto. | | Examen #1  (20 ptos) | Aspectos teóricos y prácticos relacionados con los patrones de diseño. | | 2da Tarea  (15 ptos) | Problemas para la programación de la capa de datos y la capa intermedia. | | 3ra Tarea  (15 ptos) | Continuación de la tarea#2, ahora con la programación de la capa interfaz y la realización de reportes. | | Proyecto Final  (40 ptos) | Desarrollar una aplicación informática que aborde los contenidos estudiados en clase. El tema debe estar enfocado a la solución de problemas en el mercado actual.  Requerimientos mínimos: Utilización de un modelo en 3 capas, acceso a una base de datos y desarrollo de 4 tipos diferentes de reportes. | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X-BIBLIOGRAFÍA** | | | | |
| Bibliografía Básica: | Biblioteca | Internet | | |
| * Design Patterns. Elements of Reusable Object-Oriented Software / Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides Addison-Wesley, 1994 ISBN 0-201-63361-2 |  |  | | | |
| * .NET Framework Essentials by Thuan Thai, Hoang Q. Lam O'Reilly First Edition June 2001 ISBN: 0-596-00165-7, 320 p. |  |  | | | |
| * Programación visual C# .NET./ Francisco Charte Ojeda. Ed. ANAYA. |  |  | | | |
| * Programming Microsoft ADO.NET 2.0 core reference / David Sceppa. (Redmond, Wash., Microsoft Press, 2006.) | 005.3 S289p |  | | | |
| Bibliografía de Prácticas: | Biblioteca | Internet | |
| * LA BIBLIA C#. Jeff Ferguson, Brian Patterson, Jason Beres. Ed. ANAYA. |  |  | |
| * Profesional visual studio 2005. Andrew Parsons, Nick Randolph. Ed. Wrox |  |  | |
| Bibliografía Complementaria: | Biblioteca | Internet | | |
| * David Chappell. Understanding .NET: A Tutorial and Analysis. Addison Wesley Professional, 2002. | |  | |  | | | |
| * Introduction to Design Patterns in C# by James W. Cooper. | |  | |  | | | |
| * Joel Spolsky. User Interface Design for Programmers. Springer-Verlag, New York, 2000 | |  | |  | | | |

Última actualización en fecha 02/08/2010