

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

COORDINACIÓN GENERAL DE FORMACIÓN BÁSICA

COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

1. **Unidad Académica:** Facultad de Ingeniería, Mexicali; Facultad de Ingeniería y Negocios, Tecate; y Escuela de Ciencias de la Ingeniería y Tecnología, Valle de las Palmas.
2. **Programa Educativo:** Ingeniero en Mecatrónica
3. **Plan de Estudios:**
4. **Nombre de la Unidad de Aprendizaje:** Programación Visual
5. **Clave:**
6. **HC:** 02 **HL:** 03 **HT:** 00 **HPC:** 00 **HCL:** 00 **HE:** 02 **CR:** 07
7. **Etapas de Formación a la que Pertenece:** Disciplinaria
8. **Carácter de la Unidad de Aprendizaje:** Optativa
9. **Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



Equipo de diseño de PUA

José Manuel Villegas Izaguirre
Araceli Celina Justo López
Félix Francisco Reyna Beltrán

Firma

Vo.Bo. de Subdirectores de Unidades Académicas

Alejandro Mungaray Moctezuma
Angélica Reyes Mendoza
María Cristina Castañón Bautista

Firma

M. CRISTINA CASTAÑÓN BAUTISTA

Fecha: 01 de junio de 2018

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

Esta unidad de aprendizaje cumple con la parte teórico-práctico en donde el estudiante adquiere la habilidad de interpretar la idea central de un problema, materializarla y construir los mensajes e interfaz gráfica que permita la comunicación usuario-computadora; desarrollando la capacidad de identificar, interpretar y traducir la información a códigos visuales o auditivos para la manipulación de hardware a través de Interfaces Gráficas capaz de gestionar información en base de datos.

Es necesario que los estudiantes al ingresar al curso tengan conocimientos de programación básica, además de una actitud crítica, honesta, responsable y ordenada, estos conocimientos podrán ser aplicados en la elaboración de sistemas de cómputo en un lenguaje de programación visual.

Esta asignatura es de carácter optativo de la etapa disciplinaria y pertenece al área de ciencias de la ingeniería.

III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE

Elaborar aplicaciones de software con interfaz gráfica de usuario, utilizando un lenguaje de programación visual y un sistema gestor de base de datos, para solucionar problemas industriales de adquisición de datos de hardware capaz de gestionar la información por medio de base de datos, con una actitud creativa, ordenada y responsable.

IV. EVIDENCIA(S) DE DESEMPEÑO

1. Portafolio del código fuente con comentarios descriptivos de los algoritmos utilizados, de un programa con interfaz gráfica de usuario, para una aplicación asociada a hardware, que haga uso de base de datos, así como su presentación audiovisual ante grupo.
2. Proyecto final con el desarrollo de una aplicación para dar solución a un caso dado por el docente. Se debe entregar reporte que incluya portada, introducción, desarrollo del programa y conclusiones.

V. DESARROLLO POR UNIDADES

UNIDAD I. Ambiente de programación visual

Competencia:

Identificar los diversos ambientes de programación visual, comparando los elementos principales en los variados lenguajes de programación (Java, VisualBasic, C#, VisualC, ó Matlab), para comprender el concepto de interfaz gráfica de usuario y justificar la importancia de que la interfaz sea amigable para el usuario, con una actitud crítica.

Contenido:**Duración:** 2 horas

- 1.1. Interfaz Gráfica de Usuario (GUI)
 - 1.1.1. Fundamentos de la Interfaz Grafica
 - 1.1.2. Principios del diseño de interfaz graficas de usuario (GUI)
- 1.2. Lenguajes de programación para el desarrollo de interfaces graficas de usuario
 - 1.2.1. Componentes de GUI.
 - 1.2.2. Comparación de editores GUI en diversos ambientes de programación visual

UNIDAD II. Diseño de Interfaz Gráfica

Competencia:

Comprender los conceptos básicos de programación orientada a objetos, mediante la aplicación de los componentes de interfaz gráfica de usuarios, para desarrollar aplicaciones software en un entorno de escritorio que den solución a problemas industriales, con una actitud creativa, analítica y responsable.

Contenido:

Duración: 12 horas

2.1. Conceptos básicos de la Programación Orientada a Objetos

- 2.1.1. Objetos
- 2.1.2. Clase
- 2.1.3. Propiedades
- 2.1.4. Métodos
- 2.1.5. Constructores
- 2.1.6. Manejo de Eventos

2.2. Componentes de interfaz Gráfica de Usuario

- 2.2.1. Contenedores
 - 2.2.1.1. Ventanas
 - 2.2.1.2. Paneles
 - 2.2.2. Etiquetas, cajas de texto y áreas de texto
 - 2.2.3. Botones
 - 2.2.4. Cuadros combinados (combobox)
 - 2.2.5. Casillas de Verificación
 - 2.2.5.1. Check-box
 - 2.2.5.2. Radio-buttons
 - 2.2.6. Menús
 - 2.2.7. Controles deslizantes (Slider)
 - 2.2.8. Tablas
 - 2.2.9. Otros componentes
- 2.3. Crear propio componente
 - 2.4. Validaciones de cajas de texto
 - 2.5. Cajas de Dialogo

UNIDAD III. Acceso a base de datos

Competencia:

Comprender y elaborar modelos de datos, a partir del análisis de requerimientos de información, para generar el diseño lógico y físico de base de datos en un sistema gestor de base de datos, así como la gestión de la información en una aplicación de software, con una actitud creativa, responsable y honesta.

Contenido:**Duración: 4 horas**

- 3.1. Modelo Entidad-Relación
- 3.2. Modelo relacional
 - 3.2.1. Normalización de Tablas
- 3.3. Lenguaje SQL
- 3.4. Sentencias de datos
 - 3.4.1. Sentencia SELECT
 - 3.4.2. Sentencia INSERT
 - 3.4.3. Sentencia DELETE
 - 3.4.4. Sentencia UPDATE
 - 3.4.5. Consultas
- 3.5. Conexión de JDBC para acceder a datos desde el lenguaje de programación
- 3.6. Manipulación de resistiros desde la interfaz de Usuario

UNIDAD IV. Aplicaciones Multitarea

Competencia:

Desarrollar aplicaciones multitarea, mediante la aplicación del concepto de hilos, para solucionar problemas industriales, con una actitud analítica, creativa y propositiva.

Contenido:**Duración: 8 horas**

- 4.1. Conceptos de hilos
- 4.2. Tipos de hilos, estados y prioridades
- 4.3. Creación de hilos
- 4.4. Programación de hilos
- 4.5. Aplicaciones prácticas de multitarea

UNIDAD V. Usos de recursos De Hardware

Competencia:

Comprender la estructura y concepto de puertos, para diseñar programas que permitan la manipulación de puertos de computadora, a través de un lenguaje de programación visual, con una actitud creativa y propositiva.

Contenido:

Duración: 6 horas

- 5.1. Definición de Puertos
- 5.2. Tipos de puertos
- 5.3. Librerías para manipular puertos
 - 5.3.1. Adquisición de datos
 - 5.3.1.1. Lectura de datos
 - 5.3.1.2. Escritura de datos

VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No. de Práctica	Competencia	Descripción	Material de Apoyo	Duración
1	Diseñar programas de aplicación, a través del uso de elementos de la programación de interfaces gráficas de usuario, para resolver problemas industriales, con creatividad y responsabilidad.	Desarrolla una aplicación con interfaz gráfica en la que se introduzca un número entero y se presente como resultado el factorial de dicho número y entrega el código por el medio electrónico que designe el docente (correo electrónico, Google classroom u otro).	Apuntes de vistos en clase y Software de programación.	12 horas
2		Desarrolla una aplicación con interfaz gráfica que permita utilizar la ley de ohm en circuitos paralelos y entrega el código por el medio electrónico que designe el docente (correo electrónico,	Apuntes de vistos en clase y Software de programación.	12 horas

		Google classroom u otro). En la aplicación se deben introducir valores de dos de las variables eléctricas y se obtendrá como resultado la tercera.		
3	Aplicar distintos elementos de la programación de interfaces gráficas de usuario, mediante el uso de programas especializados en lenguajes de programación visual, para la manipulación de hardware y el uso de bases de datos, con creatividad.	Desarrolla una aplicación con interfaz gráfica que controle un grupo de 8 leds por medio del puerto serie y entrega el código por el medio electrónico que designe el docente (correo electrónico, Google classroom u otro).	Apuntes de vistos en clase e investigación.	12 horas
4		Desarrolla una aplicación con interfaz gráfica que permita interactuar con un sistema de adquisición de datos para el registro de temperaturas y almacenar las mediciones tomadas en una base de datos para su posterior procesamiento y entrega el código por el medio electrónico que designe el docente (correo electrónico, Google classroom u otro).	Apuntes de vistos en clase e investigación.	12 horas

VII. MÉTODO DE TRABAJO

Encuadre: El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

Estrategia de enseñanza (docente)

El profesor presentará los conceptos fundamentas de cada tema y desarrollará ejemplos de aplicación para que el alumno resuelva las prácticas. Además utilizando herramientas didácticas guiará en las horas de laboratorio y aplicará exámenes de conocimiento. También utiliza herramientas virtuales (google classroom, blackboard, etc.) para facilitar el aprendizaje.

Estrategia de aprendizaje (alumno)

El estudiante hará investigación de tareas, ejercicios prácticos y en la etapa terminal del curso se desarrollará un proyecto con aplicaciones mecánicas que involucre interfaz gráfica, adquisición de datos través de un puerto y manipulación de información en base de datos.

VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

Criterios de acreditación

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Tareas 10%
- Evaluaciones parciales 40%
- Evidencia de desempeño 1..... 30%
(Portafolio de código fuente)
- Evidencia de desempeño 2..... 20%
(Proyecto final: desarrollo de aplicación)

Total..... 100%

IX. REFERENCIAS

Básicas	Complementarias
Ceballos, J. (2017). <i>Visual Basic.NET. Curso de Programación ESPAÑA</i>: RA-MA. Horton, I. (2014). <i>Ivor Horton's Beginning Visual C++ 2013</i>. U.K.: Wrox Press. Johnson, B. (2017). <i>Professional Visual Studio</i> (1st ed). USA: Wrox. Joyanes, L. (2008). <i>Estructuras de Datos En Java</i>. E.U.: McGraw Hill/Interamericana. [clásica] Joyanes, L. (2014). <i>Programación en C, C++, JAVA Y UML</i>. (2^a ed.). E.U.: McGraw-Hill Interamericana. Marinilli, M. (2006). <i>Professional Java User Interfaces</i>. E.U.: Wiley. [clásica] Martín, A. (2010). <i>Programador Certificado JAVA 2: Curso Practico</i> (3^a ed.). España: RA-MA. [clásica] Sharp, J. (2018). <i>Microsoft Visual C# Step by Step (Developer Reference)</i> (9th ed.). USA: Intermediate.	Deitel, P., & Deitel, H. (2016). <i>Visual C# How to Program</i> (6th ed.). España: Pearson. Zukowski, J. (2005). <i>The definitive guide to Java Swing</i>. USA: A press. [clásica]

X. PERFIL DEL DOCENTE

El docente que imparta esta asignatura debe contar con título en Ingeniero en Computación, Sistemas Computacionales o afín. Preferentemente con posgrado relacionado al área de programación; experiencia docente deseable de dos años en asignaturas de programación y de un año en el desarrollo de programas para la industria bajo plataformas de programación visual. Con facilidad para transmitir el conocimiento, ordenado, proactivo y responsable.