

## 1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura:	<b>Tópicos Avanzados de Programación.</b>
Carrera:	<b>Ingeniería en Sistemas Computacionales</b>
Clave de la asignatura:	<b>SCD-1027</b>
(Créditos) SATCA <sup>1</sup>	<b>2-3-5</b>

## 2.- PRESENTACIÓN

### **Caracterización de la asignatura.**

Esta materia aporta al perfil la competencia para desarrollar soluciones de software utilizando programación concurrente, programación de eventos, que soporte interfaz gráfica y comunicación con dispositivos móviles.

Se inicia estudiando la programación concurrente desde la conceptualización del hilo, su creación, control y destrucción, así como la sincronización de varios hilos. Finalmente la materia se complementa con la introducción a la programación móvil, a través de la revisión de las distintas tecnologías, selección de una de ellas y desarrollo de una aplicación básica.

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 el lenguaje de modelado unificado (UML).

Debido a las competencias que el estudiante requiere como base para el desarrollo de las planteadas en este programa, la asignatura esta considerada cursarse cuando el estudiante haya desarrollado la competencia de programar, es recomendable cursarla inmediatamente después de haber aprobado el curso de programación orientada a objetos, y evitar cursarla en semestres muy avanzados tomando en cuenta que en esta materia el estudiante desarrollará competencias necesarias para cursos posteriores entre los que se encuentran los talleres de bases de datos y programación web.

### **Intención didáctica.**

Se organiza el temario, en cinco unidades, la primera unidad aborda el tema de

---

<sup>1</sup> Sistema de asignación y transferencia de créditos académicos

programación de eventos, en donde el objetivo es que el estudiante logre que la estructura y ejecución del programa dependan de los sucesos (eventos) que ocurran en el sistema o que ellos mismos provoquen. El estudiante debe desarrollar la habilidad de definir los eventos a los cuales el programa reaccionará y las acciones que seguirá al presentarse cada uno.

En la segunda unidad 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.

Las unidades uno y dos pueden ser estudiadas simultáneamente considerando que están estrechamente relacionadas, para ello 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.

La tercer unidad 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. Esta unidad 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 la cuarta unidad se aborda un tema complicado por lo que requiere de un tiempo razonable dentro del curso para su estudio, 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 esta unidad 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.

La quinta unidad se refiere al estudio de la programación de dispositivos móviles, la intención de esta unidad 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.

El enfoque sugerido para la materia requiere que las actividades prácticas

promuevan el desarrollo de habilidades para la experimentación, tales como: identificación, manejo y control de variables y datos relevantes; planteamiento de hipótesis; trabajo en equipo; asimismo, propicien procesos intelectuales como inducción-deducción y análisis-síntesis con la intención de generar una actividad intelectual compleja; por esta razón varias de las actividades prácticas se han descrito como actividades previas al tratamiento teórico de los temas, de manera que no sean una mera corroboración de lo visto previamente en clase, sino una oportunidad para conceptualizar a partir de lo observado.

También se busca desarrollar habilidades creativas y emprendedoras, dando un espacio al estudiante para que detecte áreas de oportunidad en su entorno y genere el proyecto que desarrollara en el transcurso del curso. Del mismo modo por la naturaleza de la materia debe promoverse la cultura de ética y respeto a los derechos de autor, tanto en las aplicaciones desarrolladas como en el uso de las herramientas utilizadas.

En las actividades prácticas sugeridas, es conveniente que el profesor busque sólo guiar a sus estudiantes para que ellos hagan la elección de las variables a controlar y registrar. Para que aprendan a planificar, que no planifique el profesor todo por ellos, sino involucrarlos en el proceso de planeación.

En el transcurso de las actividades programadas es muy importante que el estudiante aprenda a valorar las actividades que lleva a cabo y entienda que está construyendo su hacer futuro y en consecuencia actúe de una manera profesional; de igual manera, aprecie la importancia del conocimiento y los hábitos de trabajo; desarrolle la precisión y la curiosidad, la puntualidad, el entusiasmo y el interés, la tenacidad, la flexibilidad y la autonomía.

Es necesario que el profesor ponga atención y cuidado en estos aspectos en el desarrollo de las actividades de aprendizaje de esta asignatura.

### 3.- COMPETENCIAS A DESARROLLAR

<b>Competencias específicas:</b>  Desarrollar soluciones de software utilizando programación concurrente, programación de eventos, que soporte interfaz grafica e incluya dispositivos móviles.	<b>Competencias genéricas</b>  <b>Competencias instrumentales:</b> <ul style="list-style-type: none"><li>• Capacidad de análisis y síntesis</li><li>• Capacidad de organizar y planificar</li><li>• Comunicación oral y escrita</li><li>• Habilidad para buscar y analizar información proveniente de fuentes diversas.</li><li>• Solución de problemas.</li></ul>
---	--

	<ul style="list-style-type: none"> <li>• Toma de decisiones.</li> <li>• Capacidad de crear modelos de programación</li> <li>• Capacidad de manejar un lenguaje de programación orientado a objetos</li> </ul> <p><b>Competencias interpersonales:</b></p> <ul style="list-style-type: none"> <li>• Capacidad crítica y autocrítica</li> <li>• Capacidad de trabajar en equipo</li> <li>• Capacidad de comunicar sus ideas</li> <li>• Capacidad de liderazgo</li> </ul> <p><b>Competencias sistémicas:</b></p> <ul style="list-style-type: none"> <li>• Capacidad de aplicar los conocimientos en la práctica</li> <li>• Habilidades de investigación</li> <li>• Capacidad de aprender</li> <li>• Capacidad de adaptarse a nuevas situaciones</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Liderazgo</li> <li>• Habilidad para trabajar en forma autónoma</li> <li>• Preocupación por la calidad</li> <li>• Búsqueda del logro</li> </ul>
--	---

#### 4.- HISTORIA DEL PROGRAMA

<b>Lugar y fecha de elaboración o revisión</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Saltillo Fecha 9 de octubre de 2010	Representantes de los Institutos Tecnológicos de:	Reunión nacional de Diseño e innovación curricular de la carrera de Ingeniería en
Institutos Tecnológicos de: Mexicali y Tijuana. Institutos Tecnológicos	Representantes de las academias de sistemas computacionales	Análisis, enriquecimiento y elaboración del programa de estudio propuesto en la Reunión Nacional de Diseño

Superiores de: Coatzacoalcos y Xalapa.		Curricular de la carrera de
Fecha 12 de octubre de 2009 al 19 de febrero de 2010		
Instituto Tecnológico de fecha	Representantes de los Institutos Tecnológicos participantes en el diseño de la carrera de Ingeniería	Reunión nacional de consolidación de la carrea de ingeniería en

### 5.- OBJETIVO(S) GENERAL(ES) DEL CURSO (competencias específicas a desarrollar en el curso)

Desarrollar soluciones de software utilizando programación concurrente, programación de eventos, que soporte interfaz grafica e incluya dispositivos móviles.

### 6.- COMPETENCIAS PREVIAS

Diseñar e implementar objetos de programación que permitan resolver situaciones reales y de ingeniería

### 7.- TEMARIO

Unidad	Temas	Subtemas
I	Eventos	1.1. Tipos de eventos 1.2. Generación y propagación de eventos 1.3. Métodos de control de eventos 1.4. Creación de eventos
II	Interfaz grafica de usuario (GUI)	2.1. Creación de interfaz grafica para usuarios. 2.2. Computación Grafica
III	Componentes y librerías	3.1. Definición conceptual de componentes, paquetes/librerías 3.2. Uso de componentes (visuales y no visuales) proporcionados por el lenguaje 3.3. Uso de librerías proporcionadas por el lenguaje 3.4. Creación de componentes (visuales y no visuales) definidos por el usuario

IV	Programación concurrente (MultiHilos)	<p>3.5. Creación y uso de paquetes/librerías definidas por el usuario.</p> <p>4.1. Concepto de hilo.</p> <p>4.2. Comparación de un programa de flujo único contra uno de flujo múltiple</p> <p>4.3. Creación y control de hilos</p> <p>4.4. Sincronización de hilos</p>
V	Programación de dispositivos móviles	<p>5.1. Introducción</p> <p>5.2. Dispositivos móviles</p> <p>5.3. Clasificación y aplicaciones de los dispositivos móviles</p> <p>5.4. Ambientes de las aplicaciones</p> <p>5.5. Protocolo de Interfaz entre aplicaciones</p> <p>5.6. Programación de teléfonos móviles y PDA</p>

## 8.- SUGERENCIAS DIDÁCTICAS (desarrollo de competencias genéricas)

El profesor debe:

Ser competente en la disciplina que está bajo su responsabilidad y aplicar los conceptos de la asignatura. Desarrollar la capacidad para coordinar y trabajar en equipo; orientar el trabajo del estudiante y potenciar en él la autonomía, el trabajo cooperativo y la toma de decisiones. Mostrar flexibilidad en el seguimiento del proceso formativo y propiciar la interacción entre los estudiantes. Tomar en cuenta el conocimiento de los estudiantes como punto de partida y como obstáculo para la construcción de nuevos conocimientos.

- Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes y explicarlo mediante un mapa conceptual, mental o cuadro sinóptico.
- Propiciar la utilización de distintos lenguajes de programación orientados a objetos (JAVA, C#, Visual Net) para el desarrollo de los programas.
- Fomentar el uso de la terminología orientada a objetos. (Diagramas de caso de uso, secuencia, entre otros) para la planeación, organización, documentación y desarrollo de programas.
- Proponer un caso de estudio en el cual el estudiante determine las diferentes fases del mismo, para posteriormente, discutirlo en grupos de trabajo y proponer soluciones mediante la programación concurrente.

- Fomentar la participación del estudiante mediante tormenta de ideas, mesas redondas, exposiciones que permita que propicie el uso adecuado de conceptos, y de terminología de programación.
- Proponer problemas que permitan al estudiante la integración de contenidos de la asignatura y entre distintas asignaturas, para su análisis y solución.
- Propiciar en el estudiante la lectura y reflexión de artículos relacionados con la asignatura y el impacto ambiental, social y laboral que ella tiene.
- Proporcionar al estudiante la relación de los contenidos de temáticos con el desarrollo de aplicaciones para dar solución a problemas distribuidos y/o la creación de sistemas operativos.
- Asignar proyectos finales que integren los temas de este programa de estudio.
- Exponer los proyectos finales

## 9.- SUGERENCIAS DE EVALUACIÓN

Se sugiere que el estudiante proponga un proyecto, preferentemente que atienda un problema real, que de acuerdo a las especificaciones integre los puntos estudiados en cada una de las unidades de aprendizaje. Se recomienda que los proyectos sean desarrollados por equipos de trabajo cuidando la participación activa de cada uno de los integrantes. También debe de fomentarse y evaluarse la investigación e incluir los resultados de las mismas como sustento en la toma de decisiones en el desarrollo del proyecto. La evaluación debe ser continua y formativa por lo que se debe considerar el desempeño en cada una de las actividades de aprendizaje, haciendo especial énfasis en:

- Ensayos, de las observaciones hechas durante las actividades, así como de las conclusiones obtenidas de dichas observaciones.
- Información obtenida durante las investigaciones solicitadas plasmada en documentos escritos.
- Reportes escritos de otras experiencias concretas que podrían realizarse adicionalmente.
- Exámenes escritos para comprobar el manejo de aspectos teóricos y conceptuales.
- Exámenes prácticos para comprobar que el conocimiento esta siendo aplicado
- Prácticas de programación por unidad.
- Proyecto final integrador: Desarrollo de una aplicación de software concurrente con interfaz grafica, manejo de eventos con comunicación con algún dispositivo móvil.

## 10.- UNIDADES DE APRENDIZAJE

### Unidad 1: Eventos

Competencia específica a desarrollar	Actividades de Aprendizaje
--------------------------------------	----------------------------

<p>Resolver problemas utilizando programación de eventos.</p>	<ul style="list-style-type: none"> <li>• Realizar una investigación sobre la programación orientada a eventos y analizarla en el grupo buscando llegar a conclusiones</li> <li>• Partiendo de problemas reales plantear soluciones e identificar cuales son los eventos que se involucran</li> <li>• Presentar problemas y analizarlos de forma grupal para proponer soluciones y seleccionar la mas factible</li> <li>• Realizar aplicaciones simples que involucren los eventos de Mouse</li> <li>• Realizar prácticas que permitan definir nuevos eventos y asociarlos con funciones.</li> <li>• Presentar los avances de los proyectos al grupo para enriquecerlos con las opiniones de maestro y de los estudiantes.</li> </ul>
---	--

## Unidad 2: Interfaz grafica de usuario (GUI)

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
<p>Construir aplicaciones con interfaz gráfica amigable enfocadas a problemas reales.</p>	<ul style="list-style-type: none"> <li>• Realizar una investigación sobre las distintas plataformas de desarrollo visual y seleccionar una a partir de un análisis.</li> <li>• Realizar una investigación sobre los requerimientos que debe de cumplir una interfaz gráfica.</li> <li>• Realizar una investigación sobre las facilidades que ofrece la plataforma seleccionada para desarrollo de la GUI.</li> <li>• Realizar prácticas que permitan familiarizarse con el desarrollo de GUI.</li> <li>• Presentar problemas y analizarlos de forma grupal para proponer soluciones y seleccionar la mas factible</li> <li>• Presentar los avances de los proyectos al grupo para enriquecerlos con las opiniones de maestro y de los estudiantes.</li> </ul>

## Unidad 3: Componentes y librerías

<b>Competencia específica a</b>	<b>Actividades de Aprendizaje</b>
---------------------------------	-----------------------------------



<b>desarrollar</b>	
Diseñar e implementar componentes y librerías.	<ul style="list-style-type: none"> <li>• Realizar una investigación sobre los componentes visuales y no visuales que la IDE seleccionada tiene.</li> <li>• Realizar prácticas que permitan familiarizarse con el uso de los componentes.</li> <li>• Realizar una investigación sobre la forma de crear nuevos componentes</li> <li>• Presentar problemas y analizarlos de forma grupal para proponer soluciones y seleccionar la mas factible</li> <li>• Presentar los avances de los proyectos al grupo para enriquecerlos con las opiniones de maestro y de los estudiantes.</li> </ul>

#### Unidad 4: Programación concurrente (MultiHilos)

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
Resolver problemas utilizando programación concurrente.	<ul style="list-style-type: none"> <li>• Realizar una investigación sobre los conceptos fundamentales de la programación concurrente</li> <li>• Presentarles a los estudiantes un código funcional que permita el análisis del funcionamiento de una aplicación multi-hilo</li> <li>• Realizar prácticas que permitan familiarizarse con el uso de hilos.</li> <li>• Realizar una investigación sobre los problemas mas comunes en las aplicaciones multihilos y la forma de resolverlos</li> <li>• Presentar problemas y analizarlos de forma grupal para proponer soluciones y seleccionar la mas factible</li> <li>• Presentar los avances de los proyectos al grupo para enriquecerlos con las opiniones de maestro y de los estudiantes.</li> </ul>

#### Unidad 5: Programación de dispositivos móviles

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
Desarrollar aplicaciones básicas para dispositivos móviles.	<ul style="list-style-type: none"> <li>• Realizar una investigación sobre los diversos tipos de dispositivos móviles que</li> </ul>

	<p>existen en la actualidad y que tienen mayor impacto en el mercado y por lo tanto en la sociedad.</p> <ul style="list-style-type: none"> <li>• Investigar sobre los diferentes entornos de desarrollo de aplicaciones móviles y analizar de manera grupal las ventajas y desventajas que cada uno de ellos presenta.</li> <li>• Tomando como base la investigación de entornos de desarrollo elegir de manera grupal dos entornos de desarrollo a utilizar en clase. Dicha elección se deberá basar en el entorno de desarrollo que más impacto tenga en la sociedad y en las empresas, con la finalidad de que el estudiante tenga el conocimiento sobre los mismos y sea competente al momento de egresar de la institución.</li> <li>• Desarrollar aplicaciones en dispositivos móviles que manipulen archivos de bases de datos.</li> <li>• Desarrollar e implementar aplicaciones móviles que hagan uso de algún tipo de conectividad (Wi-Fi, Bluetooth, etc).</li> <li>•</li> </ul>
--	---

## 11.- FUENTES DE INFORMACIÓN

### Fuentes impresas (libros)

- Sharp, John, Visual C# 2008, 1 Edición, Anaya Multimedia, España, 2008
- Herbert Schildt, Java Manual De Referencia 7a. Ed., 7ª Edición, Mcgraw Hill, México, 2008
- Luis Joyanes Aguilar, Programación En C/C++Java Y Uml, 1, Mcgraw Hill, México, 2010
- John Dean, Introducción A La Programación Con Java, 1, Mcgraw Hill, México, 2009
- Deitel Y Deitel, Java Cómo Programar, 7ª Edición, Prentice Hall, México, 2008
- Deitel Y Deitel, C / C++ Y Java Cómo Programar, 4ta Edición, Prentice Hall, México, 2004
- Andrea, Edgar D, Java 2: Curso De Iniciación, 1era Edición, Inforbook's Ediciones, España, 2006
- [http://www.programacion.com/java/tutorial/ags\\_j2me/](http://www.programacion.com/java/tutorial/ags_j2me/)
- <http://leo.ugr.es/J2ME/TOOLS/index.html>

## 12.- PRÁCTICAS PROPUESTAS

- Editar, compilar y ejecutar distintos programas que incluyan el uso de formularios, botones, etiquetas y cajas de texto.
- Desarrollar aplicaciones que manejen gráficos en un componente tipo canvas.
- Desarrollar aplicaciones que manejen componentes visuales y no visuales proporcionados por la IDE.
- Desarrollar aplicaciones que manejen librerías proporcionados por la IDE.
- Desarrollar componentes visuales o no visuales a partir de requerimientos previamente definidos y aplicarlos en el diseño de aplicaciones.
- Desarrollar librerías a partir de requerimientos previamente definidos y aplicarlos en el diseño de aplicaciones.
- Desarrollar aplicaciones que incluyan la programación de eventos.
- Desarrollar aplicaciones que incluyan la generación de nuevos eventos.
- Analizar las diferencias de funcionalidad entre programas de flujo único contra programas de flujo múltiple.
- Desarrollar programas que implementen el uso de un hilo y manipulen sus distintos estados.
- Desarrollar programas que implementen el uso de varios hilos que compartan recursos y estén sincronizados.
- Desarrollar una aplicación con programación concurrente que resuelva un problema real.
- Desarrollar aplicaciones donde utilice el envío y recepción de datos.
- Desarrollar un juego para un dispositivo móvil.