

<b>PROGRAMACIÓN DIDÁCTICA.</b>	
<b>I.E.S. JULIO VERNE</b>	<b>DEPARTAMENTO DE INFORMÁTICA</b>
<b>C.F.G.S. DESARROLLO DE APLICACIONES MULTIPLATAFORMA</b>	<b>2<sup>o</sup> CURSO</b>
<b>MÓDULO: <i>PROGRAMACIÓN DE SERVICIOS Y PROCESOS</i></b>	

<b>CURSO:</b>	<b>2021 – 2022</b>
<b>PROFESOR:</b>	<b>MARÍA JOSÉ LUCERO SÁNCHEZ</b>

## ÍNDICE

### 1.1 INTRODUCCIÓN.

El título de formación profesional de **Técnico Superior en Desarrollo de Aplicaciones Multiplataforma** tiene una duración de 2000 horas distribuidas en módulos que se desarrollarán durante dos cursos académicos.

La organización de los módulos de dicho título es la siguiente:

- I) Formación en centro educativo.
  - a) Módulos asociados a la competencia.
  - b) Módulos profesionales socioeconómicos.
  - c) Módulo profesional integrado.
- II) Formación en centro de trabajo.

Atendiendo a esa distribución, el módulo de **Programación de Servicios y Procesos** se enmarca dentro de "*formación en centro educativo*" y "*asociado a la competencia*".

La duración del mismo es de **63** horas lectivas impartidas durante el primer curso de los dos con los que cuenta el ciclo, repartidas en **3** horas semanales. La totalidad de las horas serán impartidas en el aula que dispone de equipamiento informático para cada alumno.

Este módulo será impartido por un profesor responsable de la asignatura sin apoyo por parte de otro profesor.

## 1.2 MARCO LEGISLATIVO

El marco legislativo que regula el ciclo formativo viene determinado por:

- Real Decreto 450/2010, de 16 de abril, del Ministerio de Educación, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y se fijan sus enseñanzas mínimas.
- Orden de 16 de junio de 2011, de la Consejería de Educación de la Junta de Andalucía, por la que se desarrolla el currículo correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma.

## 1.3 REFERENTE CONTEXTUAL

El referente contextual viene determinado por el proyecto educativo del centro.

## 1.4 OBJETIVOS GENERALES QUE SE DESARROLLAN EN EL CICLO

Recogidos en la programación del departamento.

## 1.5 OBJETIVOS GENERALES QUE SE DESARROLLAN EN EL MÓDULO

El presente título, según el real decreto 450/2010 y la orden de 16 de junio de 2011, tiene como **objetivos generales** los siguientes:

- b) Identificar las necesidades de seguridad analizando vulnerabilidades y verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados con la seguridad en el sistema.
- e) Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos.
- i) Seleccionar y emplear técnicas, motores y entornos de desarrollo,

evaluando sus posibilidades, para participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento.

j) Seleccionar y emplear técnicas, lenguajes y entornos de desarrollo, evaluando sus posibilidades, para desarrollar aplicaciones en teléfonos, PDA y otros dispositivos móviles.

l) Valorar y emplear herramientas específicas, atendiendo a la estructura de los contenidos, para crear tutoriales, manuales de usuario y otros documentos asociados a una aplicación.

n) Analizar y aplicar técnicas y librerías específicas, simulando diferentes escenarios, para desarrollar aplicaciones capaces de ofrecer servicios en red.

ñ) Analizar y aplicar técnicas y librerías de programación, evaluando su funcionalidad para desarrollar aplicaciones multiproceso y multihilo.

## **1.6 COMPETENCIAS QUE SE DESARROLLAN EN EL MÓDULO.**

La formación del módulo contribuye a alcanzar las competencias profesionales, personales y sociales de este título que se relacionan a continuación:

b) Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.

e) Desarrollar aplicaciones multiplataforma con acceso a bases de datos utilizando lenguajes, librerías y herramientas adecuados a las especificaciones.

i) Participar en el desarrollo de juegos y aplicaciones en el ámbito del entretenimiento y la educación empleando técnicas, motores y entornos de desarrollo específicos.

j) Desarrollar aplicaciones para teléfonos, PDA y otros dispositivos móviles empleando técnicas y entornos de desarrollo específicos.

l) Crear tutoriales, manuales de usuario, de instalación, de configuración y de administración, empleando herramientas específicas.

n) Desarrollar aplicaciones multiproceso y multihilo empleando librerías y

técnicas de programación específicas.

ñ) Desarrollar aplicaciones capaces de ofrecer servicios en red empleando mecanismos de comunicación.

t) Establecer vías eficaces de relación profesional y comunicación con sus superiores, compañeros y subordinados, respetando la autonomía y competencias de las distintas personas.

w) Mantener el espíritu de innovación y actualización en el ámbito de su trabajo para adaptarse a los cambios tecnológicos y organizativos de su entorno profesional.

## **1.7 ADAPTACIÓN AL ENTORNO DE LA ECONOMÍA ANDALUZA.**

El presente módulo tiene una vigencia máxima en el entorno andaluz. Puede afirmarse esto debido a que el perfil de técnico especialista capacitado para el procesado de información independientemente del método utilizado para su publicación es muy demandado en la actualidad en Andalucía debido a que el perfil de las empresas es mayoritariamente pequeña y mediana empresa.

## **1.8 RESULTADOS DE APRENDIZAJE Y SUS CRITERIOS DE EVALUACIÓN**

<b>Resultados de aprendizaje</b>	<b>Criterios de evaluación</b>
1. Desarrolla aplicaciones compuestas por varios procesos reconociendo y aplicando principios de programación paralela.	a) Se han analizado las características de los procesos y de su ejecución por el sistema operativo. b) Se han caracterizado los hilos de ejecución y descrito su relación con los procesos. c) Se han reconocido las características de la programación concurrente y sus ámbitos de aplicación. d) Se han identificado las diferencias entre programación paralela y programación

	<p>distribuida, sus ventajas e inconvenientes.</p> <p>e) Se han utilizado clases para programar aplicaciones que crean subprocesos.</p> <p>f) Se han utilizado mecanismos para sincronizar y obtener el valor devuelto por los subprocesos iniciados.</p> <p>g) Se han desarrollado aplicaciones que gestionen y utilicen procesos para la ejecución de varias tareas en paralelo.</p> <p>h) Se han depurado y documentado las aplicaciones desarrolladas.</p>
<p>2. Desarrolla aplicaciones compuestas por varios hilos de ejecución analizando y aplicando librerías específicas del lenguaje de programación.</p>	<p>a) Se han identificado situaciones en las que resulte útil la utilización de varios hilos en un programa.</p> <p>b) Se han reconocido los mecanismos para crear, iniciar y finalizar hilos.</p> <p>c) Se han programado aplicaciones que implementen varios hilos.</p> <p>d) Se han identificado los posibles estados de ejecución de un hilo y programado aplicaciones que los gestionen.</p> <p>e) Se han utilizado mecanismos para compartir información entre varios hilos de un mismo proceso.</p> <p>f) Se han desarrollado programas formados por varios hilos sincronizados mediante técnicas específicas.</p> <p>f) Se ha establecido y controlado la prioridad de cada uno de los hilos de ejecución.</p> <p>h) Se han depurado y documentado los programas desarrollados.</p>
<p>3. Programa mecanismos de comunicación en red empleando sockets y analizando el escenario de ejecución.</p>	<p>a) Se han identificado escenarios que precisan establecer comunicación en red entre varias aplicaciones.</p> <p>b) Se han identificado los roles de cliente y de servidor y sus funciones asociadas.</p> <p>c) Se han reconocido librerías y mecanismos del lenguaje de programación que permiten programar aplicaciones en red.</p> <p>d) Se ha analizado el concepto de socket, sus tipos y características.</p> <p>e) Se han utilizado sockets para programar una aplicación cliente que se comunique con un servidor.</p> <p>f) Se ha desarrollado una aplicación servidor en red y verificado su funcionamiento.</p> <p>g) Se han desarrollado aplicaciones que utilizan sockets para intercambiar información.</p> <p>h) Se han utilizado hilos para implementar los procedimientos de las aplicaciones relativos a la comunicación en red.</p>
<p>4. Desarrolla aplicaciones que ofrecen servicios en red, utilizando librerías de clases y aplicando criterios de eficiencia y disponibilidad.</p>	<p>a) Se han analizado librerías que permitan implementar protocolos estándar de comunicación en red.</p> <p>b) Se han programado clientes de protocolos estándar de comunicaciones y verificado su</p>

	<p>funcionamiento.</p> <p>c) Se han desarrollado y probado servicios de comunicación en red.</p> <p>d) Se han analizado los requerimientos necesarios para crear servicios capaces de gestionar varios clientes concurrentes.</p> <p>e) Se han incorporado mecanismos para posibilitar la comunicación simultánea de varios clientes con el servicio.</p> <p>f) Se ha verificado la disponibilidad del servicio.</p> <p>g) Se han depurado y documentado las aplicaciones desarrolladas.</p>
<p>5. Protege las aplicaciones y los datos definiendo y aplicando criterios de seguridad en el acceso, almacenamiento y transmisión de la información.</p>	<p>a) Se han identificado y aplicado principios y prácticas de programación segura.</p> <p>b) Se han analizado las principales técnicas y prácticas criptográficas.</p> <p>c) Se han definido e implantado políticas de seguridad para limitar y controlar el acceso de los usuarios a las aplicaciones desarrolladas.</p> <p>d) Se han utilizado esquemas de seguridad basados en roles.</p> <p>e) Se han empleado algoritmos criptográficos para proteger el acceso a la información almacenada.</p> <p>f) Se han identificado métodos para asegurar la información transmitida.</p> <p>g) Se han desarrollado aplicaciones que utilicen sockets seguros para la transmisión de información.</p> <p>h) Se han depurado y documentado las aplicaciones desarrolladas.</p>

## 1.9 METODOLOGÍA GENERAL

Recogida en la programación del departamento de manera general.

En particular en este módulo se concretará de la siguiente forma:

- El profesor presenta una introducción al tema apoyado en diferentes recursos.
- Se plantean problemas que el profesor resuelve y motiva a los alumnos a que resuelvan otros similares por analogía.
- Plantea ampliaciones del problema que los alumnos deberán resolver, a partir de lo practicado, investigando por su cuenta, trabajando en equipo y experimentando, de manera que construyan su propio aprendizaje mediante la práctica.
- El profesor guiará y apoyará en esa práctica, y resolverá cuestiones concretas cuando sea necesario, sobre las que los alumnos se apoyarán para seguir ampliando en conocimientos y práctica.

- Se trabajarán supuestos prácticos cercanos a la realidad que deben conocer y manejar en un entorno profesional, pero a la vez se tratará que los planteamientos sean didácticos y sirvan para introducir en nuevos ámbitos y complejidad la materia a tratar.
- Se trabajará con ejercicios que irán evolucionando a lo largo de las diferentes unidades didácticas, de manera que se vaya añadiendo conceptos nuevos, pero a la vez integren todo lo aprendido hasta ese momento.
- También se añadirá otras actividades de mayor complejidad que los alumnos irán desarrollando de manera que sirva eso como un proyecto completo donde pongan en juego todas las competencias que se pretenden adquirir de forma integral.
- Los ejercicios prácticos se realizarán en el aula de ordenadores utilizando el software relacionado con la unidad de trabajo en la que estemos trabajando.

## 1.10 CONTENIDOS BÁSICOS DEL MÓDULO

### - Programación multiproceso

- Ejecutables. Procesos. Servicios. Problemas asociados a recursos compartidos.
- Estados de un proceso. Planificación de procesos por el sistema operativo.
- Hilos.
- Programación concurrente.
- Programación paralela y distribuida.
- Creación de procesos.
- Comunicación entre procesos.
- Gestión de procesos.
- Sincronización entre procesos.
- Programación de aplicaciones multiproceso.
- Depuración y documentación.

### - Programación multihilo

- Recursos compartidos por los hilos.
- Estados de un hilo. Cambios de estado.
- Elementos relacionados con la programación de hilos. Librerías y clases.

- Gestión de hilos. Creación, inicio y finalización.
- Compartición de información entre hilos.
- Sincronización de hilos.
- Prioridades de los hilos. Gestión de prioridades.
- Programación de aplicaciones multihilo.
- Depuración y documentación.
- Programación de comunicaciones en red
  - Protocolos de comunicaciones. Puertos.
  - Comunicación entre aplicaciones.
  - Roles cliente y servidor.
  - Elementos de programación de aplicaciones en red. Librerías.
  - Sockets. Tipos y características.
  - Creación de sockets.
  - Enlazado y establecimiento de conexiones.
  - Utilización de sockets para la transmisión y recepción de información.
  - Finalización de conexiones.
  - Programación de aplicaciones cliente y servidor en red.
  - Utilización de hilos en la programación de aplicaciones en red.
- Aplicaciones de servicios en red
  - Protocolos estándar de comunicación en red a nivel de aplicación (telnet, ftp, http, pop3, smtp, entre otros)
  - Librerías de clases y componentes.
  - Utilización de objetos predefinidos.
  - Establecimiento y finalización de conexiones.
  - Transmisión de información.
  - Programación de aplicaciones cliente.
  - Programación de servidores.
  - Implementación de comunicaciones simultáneas.
  - Pruebas de la disponibilidad del servicio. Monitorización de tiempos de respuesta.
  - Depuración y documentación.

- Utilización de técnicas de programación segura
  - Prácticas de programación segura.
  - Principios criptográficos. Principales aplicaciones de la criptografía.
  - Protocolos criptográficos.
  - Criptografía de clave pública y clave privada.
  - Política de seguridad. Limitaciones y control de acceso a usuarios. Esquemas de seguridad basados en roles.
  - Programación de mecanismos de control de acceso.
  - Encriptación de información.
  - Protocolos seguros de comunicaciones.
  - Sockets seguros.
  - Programación de aplicaciones con comunicaciones seguras.
  - Depuración y documentación.

## **1.11 CONTENIDOS.**

Las unidades didácticas en las que se distribuye la asignatura son las siguientes:

UD01: Generación de servicios en red.

UD02: Programación de comunicaciones en red.

UD03: Utilización de técnicas de programación segura

UD04: Programación de procesos.

UD05: Programación de hilos.

**1.12 TEMPORIZACIÓN**

UD01: Generación de servicios en red.	15 h.
UD02: Programación de comunicaciones en red.	12 h.
UD03: Utilización de técnicas de programación segura.	10 h.
UD04: Programación de procesos.	12 h.
UD05: Programación de hilos.	14 h.
TOTAL	63 h

**1.13 TABLA DE UNIDAD CON RESULTADO APRENDIZAJE**

<b>UNIDAD</b>	<b>RESULTADOS APRENDIZAJE</b>	<b>CRITERIOS DE EVALUACIÓN</b>
UD01: Generación de servicios en red.	4	4.a, 4.b, 4.c, 4.d, 4.e, 4.f, 4.g
UD02: Programación de comunicaciones en red.	3	3.a, 3.b, 3.c, 3.d, 3.e, 3.f, 3.g, 3.h
UD03: Utilización de técnicas de programación segura.	5	5.a, 5.b, 5.c, 5.d, 5.e, 5.f, 5.g, 5.h
UD04: Programación de procesos.	1	1.a, 1.b, 1.c, 1.d, 1.e, 1.f, 1.g, 1.h
UD05: Programación de hilos.	2	2.a, 2.b, 2.c, 2.d, 2.e, 2.f, 2.g, 2.h

## 1.14 ESPECIFICACIÓN DE LAS UNIDADES DIDÁCTICAS

<b>UNIDAD DIDÁCTICA 1: Generación de servicios en red</b>	
<p>Descripción: en esta unidad se da a conocer el concepto de “servicio” y su aplicación en el contexto de la computación distribuida, así como las características de un protocolo de nivel de aplicación y los principales protocolos usados en la actualidad. El alumnado conocerá tecnologías avanzadas de comunicación para programar aplicaciones distribuidas usando el modelo de cliente/servidor como la invocación remota de servicios (RMI) y los servicios web (SOAP y REST).</p>	
<b>Criterios de evaluación:</b> 4.a, 4.b, 4.c, 4.d, 4.e, 4.f, 4.g	<b>N.º sesiones:</b> 15
Contenidos	Metodología
1 Servicios. Concepto. Servicios en red. Servicios de nivel de aplicación. 2 Programación de aplicaciones cliente y servidor. 3 Protocolos estándar de nivel de aplicación. Telnet. SSH. FTP. HTTP. POP3. SMTP. Otros. 4 Técnicas avanzadas de programación de aplicaciones distribuidas. 5 Invocación de servicios remotos (RMI). 6 Servicios web. SOAP y REST api.	<ul style="list-style-type: none"> <li>● Exposiciones teóricas.</li> <li>● Ejercicios y supuestos prácticos.</li> <li>● Prácticas de programación de aplicaciones cliente/servidor en Java.</li> </ul>
Observaciones:	

<b>UNIDAD DIDÁCTICA 2: Programación de comunicaciones en red</b>	
<p>Descripción: en esta unidad se dan a conocer los conceptos básicos de la computación distribuida, así como los protocolos básicos de comunicación entre aplicaciones y los principales modelos de computación distribuida. Y permitirá al alumnado aprender a programar aplicaciones que se comuniquen con otras en red mediante sockets, aplicando de forma práctica los principios fundamentales del modelo cliente/servidor.</p>	
<b>Criterios de evaluación:</b> 3.a, 3.b, 3.c, 3.d, 3.e, 3.f, 3.g, 3.h	<b>N.º sesiones:</b> 12
Contenidos	Metodología
1 Conceptos básicos. Comunicación entre aplicaciones. 2 Protocolos de comunicaciones: IP, TCP, UDP. 3 Sockets. 4 Modelos de comunicaciones. Cliente/Servidor. Comunicación en grupo. P2P.	<ul style="list-style-type: none"> <li>● Exposiciones teóricas.</li> <li>● Ejercicios y supuestos prácticos.</li> <li>● Prácticas de programación con Sockets en Java.</li> </ul>
Observaciones	

### UNIDAD DIDÁCTICA 3: Utilización de técnicas de programación segura

Descripción: en esta unidad se presentan los conceptos básicos relacionados con la seguridad de la información y la historia de la criptografía. Permitirá al alumnado familiarizarse con los modelos criptográficos más importantes, aprender a programar usando mecanismos de cifrado y aplicando comunicaciones seguras en la programación de aplicaciones distribuidas.

**Criterios de evaluación:** 5.a, 5.b, 5.c, 5.d, 5.e, 5.f, 5.g, 5.h

**N.º sesiones: 10**

Contenidos	Metodología
1 Conceptos básicos. 2 Características de los servicios de seguridad. 3 Modelo de clave privada. 4 Modelo de clave pública. 5 Control de acceso.	<ul style="list-style-type: none"> <li>● Exposiciones teóricas.</li> <li>● Ejercicios y supuestos prácticos.</li> <li>● Prácticas de programación segura en Java.</li> </ul>
Observaciones	

### UNIDAD DIDÁCTICA 4. Programación de procesos

Descripción: en esta unidad se dan a conocer los conceptos básicos del funcionamiento de los sistemas en lo relativo a la ejecución de diferentes programas, el concepto de concurrencia y cómo el sistema puede proporcionar multiprogramación al usuario, las políticas de planificación del sistema para proporcionar un entorno multitarea, y permitirá al alumnado familiarizarse con la programación de procesos entendiendo sus principios y formas de aplicación.

**Criterios de evaluación:** 1.a, 1.b, 1.c, 1.d, 1.e, 1.f, 1.g, 1.h

**N.º sesiones: 12**

Contenidos	Metodología
1 Conceptos básicos. 2 Programación concurrente. 3 Funcionamiento básico del sistema operativo.	<ul style="list-style-type: none"> <li>● Exposiciones teóricas.</li> <li>● Ejercicios y supuestos prácticos.</li> </ul>

4 Procesos. Estados, colas y planificación. 5 Gestión de procesos. 6 Comunicación entre procesos. 7 Sincronización de procesos. 8 Programación multiproceso.	● Prácticas de programación multiproceso en Java.
Observaciones	

### UNIDAD DIDÁCTICA 5. Programación de hilos

Descripción: en esta unidad se dan a conocer los conceptos básicos del funcionamiento de los sistemas en lo relativo a la ejecución de diferentes hilos, los conceptos de paralelismo y cómo el sistema puede proporcionar multitarea al usuario, y permitirá al alumnado familiarizarse con la programación de hilos entendiendo sus principios y formas de aplicación y aprender mecanismos de sincronización de hilos para construir aplicaciones paralelas.

**Criterios de evaluación:** 2.a, 2.b, 2.c, 2.d, 2.e, 2.f, 2.g, 2.h

**N.º sesiones:** 14

Contenidos	Metodología
1 Conceptos básicos. 2 Recursos compartidos por hilos. 3 Estados de un hilo. 4 Gestión de hilos. Planificación. 5 Sincronización de hilos. 6 Programación de aplicaciones multihilo.	● Exposiciones teóricas. ● Ejercicios y supuestos prácticos. ● Prácticas de programación multihilo en Java.
Observaciones:	

## 1.15 CONTENIDOS ACTITUDINALES

Recogidos en la programación del departamento.

## 1.16 PROCESO DE EVALUACIÓN – CALIFICACIÓN.

La calificación de cada trimestre será una nota numérica de 0 a 10 y vendrá desglosada de la siguiente manera, teniendo en cuenta que los criterios de evaluación son los que se marcan en las tablas definidas anteriormente y se valoran y cuantifican a través de una o dos pruebas en cada trimestre:

1. **1<sup>er</sup> trimestre:**
  1. Pruebas evaluables escritas o prácticas de trabajo en clase y/o en casa: Se valorará un 15%.
  2. Una prueba específica a través de un examen. Supondrá el 85% de la nota del trimestre.
  
2. **2<sup>o</sup> trimestre:**
  1. Pruebas evaluables escritas o prácticas de trabajo en clase y/o en casa: Se valorará un 10%.
  2. Una prueba específica a través de un examen. Supondrá el 85% de la nota del trimestre.

### **Criterios de calificación generales :**

Para recuperar un trimestre se establecerán unas pruebas de recuperación al principio del trimestre siguiente (exceptuando el segundo trimestre que no tiene recuperación por falta de tiempo).

En caso de que no supere alguna recuperación, se establecerán en el mes de Junio otra prueba para recuperar cada uno de los trimestres. La superación del curso se realizará cuando haya superado la totalidad de los trimestres por separado.

En caso de que el alumno desee superar la nota, se establecerán una prueba específica que se desarrollarán en Junio.

La nota final del curso vendrá determinada por la media ponderada de la notas de los dos trimestres. El porcentaje aplicable a cada trimestre será:

PRIMER TRIMESTRE: 50%

SEGUNDO TRIMESTRE: 50%

- El aprobado se establece en un 5 sobre 10.

## 1.17 ADAPTACIONES CURRICULARES.

Recogidas en la programación del departamento

- **Caso Particular:** No se ha detectado alumnos con necesidades especiales

## 1.18 RECURSOS DIDÁCTICOS.

En el caso de este módulo los recursos los dividimos en dos tipos: Humanos y Materiales.

- Recursos Humanos:** El módulo cuenta exclusivamente con un profesor titular
- Recursos Materiales:** Se pueden inventariar los siguientes:
  - Un aula taller, donde se ubican todas las clases del grupo.
  - Un proyector de vídeo/SVGA.
  - 20 ordenadores de sobremesa para el alumnado.
  - Un ordenador de sobremesa para el profesor.

- Una impresora láser.
- Una LAN que integra todos los ordenadores del aula.
- Red WiFi de soporte conectada a la red del centro.
- Plataforma Moodle de apoyo.
- Linux de libre distribución y Windows 10.
- Java (JDK 8 mínimo), Eclipse (incluido SonarLint) y GitHub.

### **1.19 BIBLIOGRAFÍA RECOMENDADA.**

- Alberto Sánchez Campos  
Programación de servicios y procesos  
Ed. Ra-Ma  
ISBN: 978-84-9964-240-6
- M.<sup>a</sup> Jesús Ramos Martín  
Programación de servicios y procesos 2<sup>a</sup> Ed.  
Ed. Garceta  
ISBN: 978-84-1728-931-7