



Programación didáctica del módulo profesional ENTORNOS DE DESARROLLO (Código 0487)

**CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB**

Departamento de informática y Comunicaciones

I.E.S. Juan de Herrera



ÍNDICE

1.- OBJETIVOS Y COMPETENCIAS DEL MÓDULO.....	3
2.- CONTENIDOS DEL MÓDULO	3
3.- EVALUACIÓN	7
3.1.- Resultados de aprendizaje y criterios de evaluación.....	7
3.2.- Criterios de evaluación mínimos para superar el módulo.....	9
3.3.- Procedimientos y herramientas de evaluación	10
3.4.- Criterios de calificación	11
3.4.1.- Convocatoria ordinaria	11
3.4.2.- Convocatoria extraordinaria	12
3.4.3.- Recuperación de un módulo pendiente del curso anterior.....	13
3.4.4.- Relación del módulo con las unidades de competencia.....	14



1.- OBJETIVOS Y COMPETENCIAS DEL MÓDULO

La formación de este módulo, en el ciclo formativo DAW, contribuye a alcanzar los **objetivos generales** del ciclo d), e), h), i) y j) y las **competencias profesionales**, personales y sociales d), f), h), i) y j) del título, recogidos en el RD: **686/2010, de 20 de mayo** (BOE 12 de junio de 2010).

Para completar la formación del módulo, además de los objetivos del ciclo formativo, se han de alcanzar los objetivos específicos de este módulo, extraídos de los Reales Decretos anteriores y expuestos en el apartado 6.1 como resultados de aprendizaje.

2.- CONTENIDOS DEL MÓDULO

A continuación, se detallan las diferencias entre los contenidos mínimos (replicados en el BOE 450/2010, de 16 de abril y en el BOE 686/2010, de 20 de mayo), y los contenidos generales (recogidos en el Decreto 3/2011, de 13 de enero, BOCM de 31 de enero de 2011)

Contenidos básicos (BOE 55 sesiones)	Contenidos generales (BOCM 65 sesiones)
<p>Desarrollo de software.</p> <ul style="list-style-type: none"> • Concepto de programa informático. • Código fuente, código objeto y código ejecutable; máquinas virtuales. • Tipos de lenguajes de programación. • Características de los lenguajes más difundidos. • Fases del desarrollo de una aplicación: análisis, diseño, codificación, pruebas, • Documentación, explotación y mantenimiento, entre otras. • Proceso de obtención de código ejecutable a partir del código fuente; herramientas implicadas. 	<p>Desarrollo de software:</p> <ul style="list-style-type: none"> • Concepto de programa informático. Instrucciones y datos. • Ejecución de programas en ordenadores: <ul style="list-style-type: none"> ○ Datos y programas. ○ Hardware vs. Software. ○ Estructura funcional de un ordenador: procesador, memoria. ○ Tipos de software. BIOS. Sistema. Aplicaciones. ○ Código fuente, código objeto y código ejecutable; máquinas virtuales. • Lenguajes de programación: <ul style="list-style-type: none"> ○ Tipos de lenguajes de programación. ○ Características de los lenguajes más difundidos. • Introducción a la ingeniería del software: <ul style="list-style-type: none"> ○ Proceso software y ciclo de vida del software. ○ Fases del desarrollo de una aplicación: análisis, diseño, codificación, pruebas, documentación, explotación y mantenimiento, entre otras. ○ Modelos de proceso de desarrollo software (cascada, iterativo, evolutivo). ○ Metodologías de desarrollo software. Características. Técnicas. Objetivos. Tipos.



Contenidos básicos (BOE 55 sesiones)	Contenidos generales (BOCM 65 sesiones)
<p>Instalación y uso de entornos de desarrollo.</p> <ul style="list-style-type: none">• Funciones de un entorno de desarrollo.• Instalación de un entorno de desarrollo.• Uso básico de un entorno de desarrollo:<ul style="list-style-type: none">○ Edición de programas.○ Generación de ejecutables.	<p>Instalación y uso de entornos de desarrollo:</p> <ul style="list-style-type: none">• Funciones de un entorno de desarrollo.• Tipos de entornos de desarrollo. Entornos de desarrollo libres y propietarios. Características.• Instalación de un entorno de desarrollo.• Uso básico de un entorno de desarrollo:<ul style="list-style-type: none">○ Uso de herramientas y asistentes.○ Creación de proyectos.○ Incorporación de elementos a proyectos.○ Edición de programas. Sintaxis y formateo de código.○ Compilación de programas. Detección de errores.○ Generación de ejecutables.○ Ejecución de programas.○ Paneles y vistas.○ Importación y exportación de ficheros.○ Personalización.○ Acceso a documentación.○ Instalación y desinstalación de aplicaciones, módulos y plugins adicionales.○ Configuración de actualizaciones.○ Automatización de tareas.

Contenidos básicos (BOE 55 sesiones)	Contenidos generales (BOCM 65 sesiones)
--------------------------------------	---



<p>Diseño y realización de pruebas.</p> <ul style="list-style-type: none">• Planificación de Pruebas.• Tipos de pruebas: Funcionales, estructurales, regresión...• Procedimientos y casos de prueba.• Pruebas de Código: Cubrimiento, valores límite, clases de equivalencia...• Pruebas unitarias; herramientas.	<p>Diseño y realización de pruebas:</p> <ul style="list-style-type: none">• Pruebas en el proceso de desarrollo de software:<ul style="list-style-type: none">○ Planificación de Pruebas a lo largo del ciclo de desarrollo.○ Tipos de pruebas: funcionales, estructurales, regresión, caja negra...○ Procedimientos y casos de prueba.Pruebas de código:<ul style="list-style-type: none">○ Cubrimiento, valores límite, clases de equivalencia...○ Pruebas unitarias de clases y funciones.○ Uso de herramientas integradas en los entornos de desarrollo para realizar pruebas unitarias.○ Automatización de pruebas unitarias.○ Pruebas de integración.○ Diseño y documentación casos de prueba.Depuración de programas:<ul style="list-style-type: none">○ Herramientas de depuración integradas en los entornos de desarrollo.○ Puntos de ruptura y seguimiento en tiempo de ejecución.○ Examinadores de variables
--	--

Contenidos básicos (BOE 55 sesiones)	Contenidos generales (BOCM 65 sesiones)
---	--



<p>Optimización y documentación.</p> <ul style="list-style-type: none"> • Refactorización. Concepto. Limitaciones. Patrones de refactorización más usuales. Refactorización y pruebas. Herramientas de ayuda a la refactorización. • Control de versiones. Estructura de las herramientas de control de versiones. Repositorio. Herramientas de control de versiones. • Documentación. Uso de comentarios. Alternativas. 	<p>Optimización y documentación:</p> <ul style="list-style-type: none"> • Refactorización: <ul style="list-style-type: none"> ○ Concepto. Limitaciones. ○ Patrones de refactorización más usuales. ○ Refactorización y pruebas. ○ Herramientas de ayuda a la refactorización. • Control de versiones: <ul style="list-style-type: none"> ○ Desarrollos colectivos ○ Herramientas de control de versiones. Utilidad. Características. Estructura (cliente/servidor). Repositorio. ○ Clientes de control de versiones. Descarga de ficheros inicial. Modificación de ficheros. Actualización de ficheros en local. Actualización de ficheros en el repositorio. Diferencias entre versiones. Restauración de versiones anteriores. Resolución de conflictos. Historial de versiones. • Documentación: <ul style="list-style-type: none"> ○ Uso de comentarios. ○ Herramientas integradas en el entorno de desarrollo para generar documentación automática de clases. ○ Alternativas.
<p>Introducción al lenguaje unificado de modelado (UML UnifiedModelingLanguage):</p> <ul style="list-style-type: none"> • Características. • Versiones. • Diagramas UML. 	<p>Introducción al lenguaje unificado de modelado (UML UnifiedModelingLanguage):</p> <ul style="list-style-type: none"> • Características. • Versiones. • Diagramas UML. • Utilización en metodologías de desarrollo orientado a objetos. • Herramientas CASE con soporte UML.
<p>Contenidos básicos (BOE 55 sesiones)</p>	<p>Contenidos generales (BOCM 65 sesiones)</p>



<p>Elaboración de diagramas de clases.</p> <p>Clases. Atributos, métodos y visibilidad</p> <ul style="list-style-type: none"> Objetos. Instanciación. Relaciones. Herencia, composición, agregación. Notación de los diagramas de clases 	<p>Elaboración de diagramas de clases:</p> <ul style="list-style-type: none"> Notación de los diagramas de clases: <ul style="list-style-type: none"> Clases. Atributos, métodos y visibilidad. Objetos. Instanciación. Relaciones. Asociación, herencia, composición, agregación, dependencia, navegabilidad. Clases abstractas. Interfaces. Paquetes. Grado de detalle. Utilización de herramientas CASE para elaborar diagramas de clases. Módulos integrados en entornos de desarrollo para elaborar diagramas de clases. Creación de código a partir de diagramas de clases. Generación de diagramas de clases a partir de código (ingeniería inversa).
<p>Elaboración de diagramas de comportamiento.</p> <ul style="list-style-type: none"> Tipos. Campo de aplicación Diagramas de casos de uso. Actores, escenario, relación de comunicación. Diagramas de secuencia. Línea de vida de un objeto, activación, envío de mensajes Diagramas de colaboración. Objetos, mensajes. 	<p>Elaboración de diagramas de comportamiento:</p> <ul style="list-style-type: none"> Tipos. Campo de aplicación Diagramas de casos de uso. Actores, casos de uso, escenario, asociaciones (relación de comunicación entre actores y casos de uso), relaciones entre casos de uso. Diagramas de secuencia. Línea de vida de un objeto/actor, activación, envío de mensajes. Diagramas de colaboración. Objetos/actores, mensajes. Otros diagramas: <ul style="list-style-type: none"> Diagramas de actividades. Diagramas de estado. Utilización de herramientas CASE para elaborar diagramas de comportamiento. Módulos integrados en entornos de desarrollo para elaborar diagramas de comportamiento.

3.- EVALUACIÓN

3.1.- Resultados de aprendizaje y criterios de evaluación

Resultados de aprendizaje	Criterios de evaluación
1.Reconoce los elementos y herramientas	a) Se ha reconocido la relación de los programas con los



<p>que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento.</p>	<p>componentes del sistema informático: memoria, procesador, periféricos, entre otros. Se han identificado las fases de desarrollo de una aplicación informática. b) Se han diferenciado los conceptos de código fuente, objeto y ejecutable. c) Se han reconocido las características de la generación de código intermedio para su ejecución en máquinas virtuales. d) Se han clasificado los lenguajes de programación. e) Se ha evaluado la funcionalidad ofrecida por las herramientas utilizadas en programación.</p>
<p>2. Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutables.</p>	<p>a) Se han instalado entornos de desarrollo, propietarios y libres. b) Se han añadido y eliminado módulos en el entorno de desarrollo. c) Se ha personalizado y automatizado el entorno de desarrollo. d) Se ha configurado el sistema de actualización del entorno de desarrollo. e) Se han generado ejecutables a partir de código fuente de diferentes lenguajes en un mismo entorno de desarrollo. f) Se han generado ejecutables a partir de un mismo código fuente con varios entornos de desarrollo. g) Se han identificado las características comunes y específicas de diversos entornos de desarrollo.</p>
<p>3. Verifica el funcionamiento de programas diseñando y realizando pruebas.</p>	<p>a) Se han identificado los diferentes tipos de pruebas. b) Se han definido casos de prueba. c) Se han identificado las herramientas de depuración y prueba de aplicaciones ofrecidas por el entorno de desarrollo. d) Se han utilizado herramientas de depuración para definir puntos de ruptura y seguimiento. e) Se han utilizado las herramientas de depuración para examinar y modificar el comportamiento de un programa en tiempo de ejecución. f) Se han efectuado pruebas unitarias de clases y funciones. g) Se han implementado pruebas automáticas. h) Se han documentado las incidencias detectadas.</p>
<p>4. Optimiza código empleando las herramientas disponibles en el entorno de desarrollo.</p>	<p>a) Se han identificado los patrones de refactorización más usuales. b) Se han elaborado las pruebas asociadas a la refactorización.</p>



	<ul style="list-style-type: none"> c) Se ha revisado el código fuente usando un analizador de código. d) Se han identificado las posibilidades de configuración de un analizador de código. e) Se han aplicado patrones de refactorización con las
	<p>herramientas que proporciona el entorno de desarrollo.</p> <ul style="list-style-type: none"> f) Se ha realizado el control de versiones integrado en el entorno de desarrollo. g) Se han utilizado herramientas del entorno de desarrollo para documentar las clases.
<p>5. Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.</p>	<ul style="list-style-type: none"> a) Se han identificado los conceptos básicos de la programación orientada a objetos. b) Se ha instalado el módulo del entorno integrado de desarrollo que permite la utilización de diagramas de clases. c) Se han identificado las herramientas para la elaboración de diagramas de clases. d) Se ha interpretado el significado de diagramas de clases. e) Se han trazado diagramas de clases a partir de las especificaciones de las mismas. f) Se ha generado código a partir de un diagrama de clases. g) Se ha generado un diagrama de clases mediante ingeniería inversa.
<p>6. Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.</p>	<ul style="list-style-type: none"> a) Se han identificado los distintos tipos de diagramas de comportamiento. b) Se ha reconocido el significado de los diagramas de casos de uso. c) Se han interpretado diagramas de interacción. d) Se han elaborado diagramas de interacción sencillos. e) Se ha interpretado el significado de diagramas de actividades. f) Se han elaborado diagramas de actividades sencillos. g) Se han interpretado diagramas de estados. h) Se han planteado diagramas de estados sencillos.

3.2.- Criterios de evaluación mínimos para superar el módulo

Como mínimos exigibles para poder superar el módulo se consideran los siguientes:

- a. El alumno reconoce los elementos, las herramientas y las fases que intervienen en el desarrollo de un programa informático.



- b. El alumno utiliza entornos integrados de desarrollo para editar código fuente y generar ejecutables.
- c. El alumno diseña y realiza pruebas para verificar el funcionamiento de programas.
- d. El alumno debe conocer los conceptos de optimiza código y generar documentación empleando para ello las herramientas disponibles en el entorno de desarrollo.
- e. El alumno genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.
- f. El alumno genera diagramas de comportamiento (sólo diagramas de casos de uso) valorando su importancia en el desarrollo de aplicaciones y empleando las herramientas disponibles en el entorno.

3.3.- Procedimientos y herramientas de evaluación

Para que la evaluación del módulo pueda llevarse a cabo según el procedimiento ordinario, requiere por parte del alumnado, la asistencia regular a las clases y actividades programadas.

Evaluación Inicial

- El objetivo de la evaluación inicial es el de conocer el nivel que tienen los alumnos, lo que influirá en el desarrollo posterior del módulo. La evaluación inicial tiene carácter informativo y en ningún caso su resultado influirá en la calificación del alumno.

Evaluación del aprendizaje a lo largo del curso

Se evaluará el grado de aprendizaje del alumnado teniendo en cuenta:

Pruebas individuales de carácter práctico/teórico en cualquier formato.

Pruebas prácticas documentadas (individuales y/o en grupo) para la resolución de supuestos propuestos que el alumnado deberá entregar dentro del plazo establecido.

Exposición de trabajos voluntarios.

La asistencia a clase, la actitud e iniciativa, interés, y el esfuerzo personal

Exámenes finales, en convocatoria ordinaria y/o extraordinaria.

Las pruebas de evaluación a realizar:

Todas las pruebas y/o prácticas indicadas anteriormente.

Examen de evaluación **final del trimestre**.

Examen **final en convocatoria ordinaria**.

Examen **final en convocatoria extraordinaria**

Pérdida de evaluación continua. Según el R.R.I. del centro respecto a la inasistencia a clase, se establece lo siguiente:

"El número máximo de faltas injustificadas que se permite para no perder el derecho a la evaluación por el procedimiento ordinario queda establecido en un 20% sobre las horas totales de cada módulo.

Cuando se haya superado el límite establecido, el alumno deberá realizar una prueba final única (Convocatoria Ordinaria), en la cual demuestre que ha adquirido las competencias que se requieren para la superación del



módulo. En ningún caso se evaluará por el proceso ordinario de evaluación en un módulo a un alumno cuando el total de faltas de asistencia (justificadas o no) exceda del 40% de las horas totales del mismo"

Al tratarse de un módulo de 62 sesiones, la pérdida de la evaluación continua podrá hacerse efectiva cuando el alumno supere las 13 faltas de asistencia sin justificar y en cualquier caso, cuando supere las 26 faltas, sean estas justificadas o no.

3.4.- Criterios de calificación

La Orden 2694/2009, modificada por la orden 11783/2012, actualmente en vigor, no menciona ni la evaluación continua ni su pérdida. Como cabe la posibilidad de que el alumno pierda el derecho a ser evaluado por el procedimiento ordinario, se hará una distinción en los criterios a aplicar.

Cada una de las pruebas, prácticas, exposiciones, trabajos, se superarán si se obtiene una calificación **superior o igual a 5**.

Para obtener la nota final, en cada evaluación (parcial o final), sin decimales (valor entero) se realizará con el siguiente criterio de redondeo: las notas menores a cinco se truncan, las mayores a cinco se redondean.

3.4.1.- Convocatoria ordinaria

Evaluación por el procedimiento ordinario

Para superar el **módulo**, el alumno deberá aprobar las **tres** evaluaciones parciales.

En cada evaluación el alumno deberá realizar las pruebas escritas que indique el profesor, así como posibles trabajos prácticos que se soliciten a lo largo de la misma. Dichos trabajos y prácticas individuales y en grupos se entregarán en la fecha y forma solicitadas por el profesor.

Al final de cada evaluación se realizará una prueba escrita (o en ordenador). La fecha de dicha evaluación la establecerá Jefatura de Estudios. Dicha prueba se calificará de 0 a 10. **No se considerará superada la evaluación cuando la calificación correspondiente a la prueba de la misma sea inferior a 5 o no se hayan entregado en tiempo y forma los trabajos (prácticas) pedidos por el profesor. Además, dichos trabajos deberán haber sido superados.**

Se aplicarán los siguientes porcentajes de calificación en cada evaluación:

Pruebas escritas o en ordenador (exámenes) : 60%

Trabajos y prácticas desarrollados individualmente o en grupo: 30%

Asistencia, puntualidad y comportamiento en clase: 10%

Los trabajos y prácticas individuales y en grupo solo serán evaluadas si se entregan en la fecha y forma solicitadas por el profesor.

En el caso que el profesor no solicite la presentación de trabajos y prácticas, el porcentaje faltante se sumará al porcentaje que se aplica a la prueba del examen.

Para superar el **módulo**, el alumno deberá aprobar las tres evaluaciones parciales.

La nota final del módulo será la media aritmética de las evaluaciones parciales (aplicando el criterio anterior de redondeo), contando para ello con la nota final (valor decimal) obtenida en cada trimestre. Para obtener la



nota final sin decimales (valor entero) del módulo se realizará con el siguiente criterio de redondeo: las notas menores a cinco se truncan, las mayores a cinco se redondean.

Recuperación de alumnos que son evaluados según el procedimiento ordinario

El profesor determinará si hay posibilidad ó no de recuperar una evaluación parcial en caso de no haber sido superada, ya sea mediante una prueba escrita o mediante la realización de prácticas. En caso de no hacer recuperación de las evaluaciones parciales, el profesor podrá determinar qué evaluaciones parciales (no superadas) debe recuperar el alumno en la prueba ordinaria de junio.

Caso de no superar alguna de las 3 evaluaciones, la evaluación final ordinaria (junio) versará sobre todos los contenidos del curso o bien sobre las evaluaciones que el profesor considere oportuno.

Si el alumno no supera el módulo en la convocatoria ordinaria de junio (los 3 trimestres de los que consta el curso), podrá realizar la recuperación del mismo en la convocatoria extraordinaria (que también se celebrará durante el mes de junio), que versará sobre los contenidos de todo el curso o bien, a criterio del profesor, sólo de los contenidos correspondientes a los trimestres no superados en la convocatoria ordinaria.

Entre la convocatoria ordinaria y extraordinaria, se destinarán 2 semanas de clases (indicadas por Jefatura de Estudios) donde los profesores tutorizarán a los alumnos que deban presentarse a la convocatoria extraordinaria, pudiendo realizar las actividades de recuperación que estimen oportunas.

Recuperación de alumnos sin derecho al procedimiento de evaluación ordinaria

La pérdida de evaluación por el procedimiento ordinario supone, desde el momento de la misma:

- La imposibilidad de presentarse a ninguna prueba parcial y evaluación.
- La obligatoriedad de presentarse a la prueba final con todos los contenidos del curso, perdiéndose cualquier nota positiva obtenida en pruebas parciales o trimestrales anteriores a la citada pérdida de evaluación.
- La obligatoriedad de presentar las prácticas que el profesor le solicite.

Los porcentajes de calificación serán:

- Examen final: 70%
- Trabajos y prácticas desarrollados individualmente: 30%

Los trabajos y prácticas individuales sólo serán evaluadas si se entregan en la fecha y forma solicitadas por el profesor.

En el caso que el profesor no solicite la presentación de trabajos y prácticas, el porcentaje faltante se sumará al porcentaje que se aplica a la prueba del examen final.

3.4.2.- Convocatoria extraordinaria

Si el alumno no supera el módulo en la convocatoria ordinaria de junio, deberá realizar la recuperación del mismo en la convocatoria extraordinaria (que también se celebrará durante el mes de junio), que versará sobre contenidos de todo el curso o a criterio del profesor sólo sobre los contenidos de aquellos trimestres que



no haya superado, siempre y cuando el alumno no haya perdido el derecho a la evaluación continua durante el curso.

3.4.3.- Recuperación de un módulo pendiente del curso anterior

La forma de recuperar un módulo pendiente de un curso anterior queda reflejada la Orden 2694/2009, modificada por la orden 11783/2012.

Dado que la carga horaria semanal es de 2 sesiones, si sólo se tuviera este módulo pendiente, se podría realizar la FCT, al no figurar además en el anexo V B) del RD del Título del Ciclo Formativo que establece la correspondencia de los módulos profesionales con las unidades de competencia para su acreditación, **SIEMPRE QUE ASÍ LO DECIDA EL EQUIPO DOCENTE**, como marca la Orden 2694/2009 modificada por la orden 11783/2012 en su **artículo 23**. En este caso el alumno podría presentarse a la convocatoria extraordinaria de diciembre.

Si tuviese pendientes otros módulos que impidiesen la realización de la FCT en el periodo de septiembre a diciembre, se realizaría una evaluación en el mes de febrero con el resto de módulos de primero pendientes.

Según el RRI se establece las siguientes consideraciones respecto a la recuperación de módulos pendientes:

- Las pruebas serán únicas para el turno de la mañana y el de la tarde, elaboradas por los profesores titulares del módulo correspondiente junto con el Jefe de Departamento o, si no fuera posible, por dos profesores que lo hayan impartido, cuya designación se hará de forma rotatoria. Además, serán evaluados por los profesores encargados y en caso de discrepancia en la calificación resolverá el Jefe de Familia Profesional o del Departamento Didáctico correspondiente.

Para el examen correspondiente a esta convocatoria se recomienda hacer uso de pruebas objetivas, mayormente de tipo test, y/o la realización de una prueba práctica en el ordenador, siempre a juicio de la comisión evaluadora.

Además de la prueba escrita, a juicio del profesor que evalúe a los alumnos, estos podrán tener que realizar una prueba práctica en los ordenadores o un trabajo de tipo práctico, cuya memoria habrán de realizar y entregar el día que se les convoque para ello.

En caso de no superar esta convocatoria, si no han agotado las cuatro convocatorias posibles dispondrán de una convocatoria extraordinaria en junio. La prueba de esta convocatoria será similar a la de diciembre o febrero.

Las fechas para estas pruebas se fijan según criterio de la Jefatura de Estudios del Centro.

Actividades de recuperación

La Jefatura de Estudios podrá organizar durante el curso clases de recuperación dirigidas a los alumnos con módulos pendientes. El profesor que las imparta realizará un repaso de contenidos y actividades de recuperación encaminadas a que los alumnos adquieran las competencias y resultados de aprendizaje necesarios que les permitan superar con éxito el módulo, bien en la convocatoria ordinaria o bien en la convocatoria extraordinaria.

Para los alumnos que no reciban clases de recuperación se propone permitir que estos alumnos asistan de oyentes a las clases del módulo en algún grupo de 1º en turno contrario, siempre y cuando haya puestos libres en el grupo de 1º.



En el caso de que no existan dichas clases de recuperación será responsabilidad de cada alumno que tenga el módulo pendiente el ponerse en contacto con el profesor titular del módulo y pedirle asesoramiento en la forma de realizar el repaso de los contenidos y de las actividades del módulo. Como sugerencia, el profesor podría dar acceso al alumno al aula virtual del curso para que pueda seguir y realizar de forma autónoma las actividades propuestas.

En cualquier caso, el alumno deberá repasar de forma autónoma todos los contenidos del módulo y realizar todas las actividades propuestas en cada una de las unidades de la programación. Para conocerlas deberá preguntar al profesor titular del módulo.

3.4.4.- Relación del módulo con las unidades de competencia

El módulo no guarda correspondencia con unidades de competencia del Catálogo Nacional de Cualificaciones Profesionales para su acreditación; por tanto, un alumno que no lo supere podrá realizar el módulo de FCT. (ver en el BOE del Título, anexo V).