



# Guía de Asignatura

## ASIGNATURA: Diseño de software

**Título:** Máster de Formación Permanente en Desarrollo Ágil de Software y DevOps

**Materia:** *DEV-Técnicas avanzadas de ingeniería software*

**Créditos:** 6 ECTS

**Código:** 03

# Índice

|   |    |
|---|----|
| 1. Organización general.....  | 3  |
| 1.1. Datos de la asignatura.....  | 3  |
| 1.2. Introducción a la asignatura.....  | 3  |
| 1.3. Competencias y resultados de aprendizaje ( <i>transcribir la información de la memoria de verificación</i> ) ..... | 3  |
| 2. Contenidos/temario .....   | 5  |
| 3. Metodología .....  | 5  |
| 4. Actividades formativas .....   | 6  |
| 5. Evaluación .....   | 9  |
| 5.1. Sistema de evaluación.....   | 9  |
| 5.2. Sistema de calificación .....  | 10 |
| 6. Bibliografía.....  | 11 |

# 1. Organización general

## 1.1. Datos de la asignatura

|                                       |   |
|---------------------------------------|---|
| <b>TITULACIÓN</b>                     | <i>Máster de Formación Permanente en Desarrollo Ágil de Software y DevOps</i> |
| <b>ASIGNATURA</b>                     | <i>Diseño de software</i>   |
| <b>CÓDIGO - NOMBRE ASIGNATURA</b>     | <i>03_DevOps</i>  |
| <b>Carácter</b>                       | Obligatorio   |
| <b>Cuatrimestre</b>                   | Primero   |
| <b>Idioma en que se imparte</b>       | Castellano  |
| <b>Requisitos previos</b>             | No existen  |
| <b>Dedicación al estudio por ECTS</b> | <b>25 horas</b>   |

## 1.2. Introducción a la asignatura

Esta asignatura permite al estudiante obtener una base de conocimiento sólida en diseño y arquitectura software. El estudiante aprenderá a construir aplicaciones cuyo código sea limpio - fácil de leer, entender y mantener - y cuya arquitectura esté concebida de manera que la aplicación sea extensible y adaptable a nuevas necesidades. Además, la asignatura presenta técnicas de diseño que permiten al programador trabajar de manera efectiva con código legado, que es el más común en contextos empresariales donde el software puede estar en funcionamiento y evolución durante años e incluso décadas

## 1.3. Competencias y resultados de aprendizaje *(transcribir la información de la memoria de verificación)*

| CÓDIGO | COMPETENCIAS   |
|--------|--|
| C01    | Abordar problemas computacionales manejando apropiadamente la complejidad con el fin de resolver los problemas de manera efectiva y eficiente. |
| C02    | Aplicar técnicas de desarrollo de software ágil y DevOps para resolver problemas en un contexto empresarial.                                   |
| C03    | Aplicar metodologías de desarrollo de software ágil en concordancia con la forma de trabajar demandada por las empresas actuales.              |

|               |   |
|---------------|---|
| C04           | Aplicar las prácticas más relevantes en el ámbito del desarrollo de software ágil, como el Desarrollo Dirigido por Tests (TDD), el Desarrollo Dirigido por Dominio (DDD), la programación por parejas o la automatización de pruebas.                         |
| C05           | Aplicar las prácticas más relevantes en el ámbito de la computación de la nube y la cultura DevOps, como el despliegue continuo, la contenedorización de aplicaciones, o el manejo y automatización de la infraestructura software por medio de código (IaC). |
| C06           | Aplicar principios de diseño software simple, como DRY, YAGNI, Ley de Demeter, etc.   |
| C07           | Diseñar e implementar arquitecturas centradas en el dominio, como la arquitectura hexagonal.  |
| C08           | Probar software de manera manual y también automática.  |
| C09           | Elaborar diferentes tipos de pruebas automáticas: unitarias, de integración, de sistema, etc.   |
| C10           | Automatizar despliegues de infraestructura por medio de herramientas como Terraform o Ansible.  |
| C11           | Trabajar con sistemas de control de versiones como Git o SVN.   |
| C12           | Gestionar dependencias de proyectos software por medio de herramientas como Maven o Gradle.   |
| <b>CÓDIGO</b> | <b>HABILIDADES O DESTREZAS</b>  |
| H01           | Utilizar buenas prácticas de ingeniería software para escribir código de calidad que sea fácilmente mantenible y comprensible.  |
| H02           | Desarrollar software de manera colaborativa aplicando los principios y valores de la cultura DevOps.  |
| H03           | Operar y desplegar aplicaciones en la nube (es decir, en entornos cloud como AWS o Azure).  |
| H04           | Diseñar, implementar y probar programas Python.   |
| H05           | Trabajar con código legado de manera efectiva.  |
| H06           | Manejar y configurar sistemas de monitorización como Elastic Stack.   |
| H07           | Manejar y configurar sistemas de alertas y visualización de datos como Graphana.  |
| H08           | Manejar y configurar bases de datos orientadas a métricas, como Prometheus.   |
| H09           | Manejar y configurar herramientas de contenedorización de aplicaciones, como Docker, Containerd o Podman.   |
| H10           | Manejar y configurar herramientas de orquestación de contenedores, como Kubernetes.   |
| H11           | Manejar y configurar entornos de integración y entrega continua, como Jenkins o GitLab.   |
| <b>CÓDIGO</b> | <b>CONOCIMIENTOS O CONTENIDOS</b>   |
| CC1           | Dominar las herramientas más relevantes en el ámbito de la computación de la nube y la cultura DevOps, como Docker, Terraform o Kubernetes.   |
| CC2           | Dominar de conceptos avanzados de programación, como la programación orientada a objetos, la programación funcional o los patrones de diseño.   |
| CC3           | Seleccionar las estructuras de datos apropiadas que permitan producir programas eficientes.   |

|     |   |
|-----|---|
| CC4 | Analizar la eficiencia de los programas tanto en términos de memoria como en tiempo de ejecución. |
| CC5 | Mejorar la estructura del código aplicando técnicas de refactoring.                               |

## 2. Contenidos/temario

- Código limpio
  - ¿Qué es? ¿Qué aspecto tiene el código de baja calidad?
  - El poder de las variables.
  - Layout y estilo: organización del código fuente.
  - Comentarios en el código.
  - Funciones y clases de alta calidad.
  - Principios de diseño simple: DRY, YAGNI, Ley de Demeter, etc.
  - Code smells.
  - La regla del Scout.
- Diseño y arquitectura software.
  - Patrones de diseño.
  - Principios SOLID.
  - Técnicas de refactoring.
  - Domain-Driven Design (DDD).
  - Arquitecturas centradas en el dominio: hexagonal y arquitectura limpia.
  - Técnicas de trabajo efectivo con código legado.

## 3. Metodología

La modalidad de enseñanza propuesta para el presente título guarda consonancia con la Metodología General de la Universidad Internacional de Valencia, aprobada por el Consejo de Gobierno Académico de la Universidad y de aplicación en todos sus títulos.

Este modelo, que vertebra el conjunto de procesos de enseñanza y aprendizaje de la institución, combina la naturaleza síncrona (mismo tiempo-diferente espacio) y asíncrona (diferente tiempo-diferente espacio) de los entornos virtuales de aprendizaje, siempre en el contexto de la modalidad virtual.

El elemento síncrono se materializa en sesiones de diferente tipo (clases expositivas y prácticas, tutorías, seminarios y actividades de diferente índole durante las clases online) donde el profesor y el estudiante comparten un espacio virtual y un tiempo determinado que el estudiante conoce con antelación.

Las actividades síncronas forman parte de las actividades formativas necesarias para el desarrollo de la asignatura y, además, quedan grabadas y alojadas para su posterior visualización.

Por otro lado, estas sesiones no solamente proporcionan espacios de encuentro entre estudiante y profesor, sino que permiten fomentar el aprendizaje colaborativo, al generarse grupos de trabajo entre los estudiantes en las propias sesiones.

Los elementos asíncronos del modelo se desarrollan a través del Campus Virtual, que contiene las aulas virtuales de cada asignatura, donde se encuentran los recursos y contenidos necesarios para el desarrollo de actividades asíncronas, así como para la interacción y comunicación con los profesores y con el resto de departamentos de la Universidad.

## 4. Actividades formativas

La metodología VIU, basada en la modalidad virtual, se concreta en una serie de actividades formativas y metodologías docentes que articulan el trabajo del estudiante y la docencia impartida por los profesores.

Durante el desarrollo de cada una de las asignaturas, se programan una serie de actividades de aprendizaje que ayudan a los estudiantes a consolidar los conocimientos trabajados en cada una de las asignaturas. A continuación, listamos las actividades genéricas que pueden formar parte de cada asignatura, dependiendo de las competencias a desarrollar en los estudiantes en cada asignatura.

### 1. Clases presenciales

### 2. Clases virtuales síncronas

Constituyen el conjunto de acciones formativas que ponen en contacto al estudiante con el profesor, con otros expertos y con compañeros de la misma asignatura en el mismo momento temporal a través de herramientas virtuales. Las actividades recurrentes (por ejemplo, las

clases) se programan en el calendario académico y las que son ocasionales (por ejemplo, sesiones con expertos externos) se avisan mediante el tablón de anuncios del campus. Estas actividades se desglosan en las siguientes categorías:

**a. Clases expositivas:** El profesor expone a los estudiantes los fundamentos teóricos de la asignatura.

**b. Clases prácticas:** El profesor desarrolla junto con los estudiantes actividades prácticas que se basan en los fundamentos vistos en las clases expositivas. En términos generales, su desarrollo consta de las siguientes fases, pudiéndose adaptar en función de las necesidades docentes:

I. La primera fase se desarrolla en la sala principal de la videoconferencia, donde el profesor plantea la actividad.

II. A continuación, divide a los estudiantes en grupos de trabajo a través de las salas colaborativas y se comienza con la actividad. En esta fase el profesor va entrando en cada sala colaborativa rotando los grupos para resolver dudas, dirigir el trabajo o dar el feedback oportuno. Los estudiantes también tienen posibilidad de consultar al profesor en el momento que consideren necesario.

III. La tercera fase también se desarrolla en la sala principal y tiene como objetivo mostrar el ejercicio o explicar con ejemplos los resultados obtenidos. Por último, se ponen en común las conclusiones de la actividad realizada.

No obstante, el profesor puede utilizar otras metodologías activas y/o herramientas de trabajo colaborativo en estas clases.

**c. Seminarios:** En estas sesiones un experto externo a la Universidad acude a presentar algún contenido teórico-práctico directamente vinculado con el temario de la asignatura. Estas sesiones permiten acercar al estudiante a la realidad de la disciplina en términos no sólo profesionales, sino también académicos. Todas estas sesiones están vinculadas a contenidos de las asignaturas y del programa educativo.

### 3. Actividades asíncronas supervisadas

Se trata de un conjunto de actividades supervisadas por el profesor de la asignatura vinculadas con la adquisición por parte de los estudiantes de los resultados de aprendizaje y el desarrollo de sus competencias. Estas actividades, diseñadas con visión de conjunto, están relacionadas entre sí para ofrecer al estudiante una formación completa e integral. Esta categoría se desglosa en el siguiente conjunto de actividades:

**a. Actividades y trabajos prácticos:** se trata de un conjunto de actividades prácticas realizadas por el estudiante por indicación del profesor que permiten al estudiante adquirir las competencias del título, especialmente aquellas de carácter práctico. Estas actividades, entre otras, pueden ser de la siguiente naturaleza: actividades vinculadas a las clases prácticas (resúmenes, mapas conceptuales, one minute paper, resolución de problemas, análisis reflexivos, generación de contenido multimedia, exposiciones de trabajos, test de autoevaluación, participación en foros, entre otros). Estas actividades serán seleccionadas por el profesor en función de las necesidades docentes. Posteriormente, estas actividades son revisadas por el profesor, que traslada un feedback al estudiante sobre las mismas, pudiendo formar parte de la evaluación continua de la asignatura.

**b. Actividades guiadas con recursos didácticos audiovisuales e interactivos:** se trata de un conjunto de actividades en las que el estudiante revisa o emplea recursos didácticos (bibliografía, videos, recursos interactivos) bajo las indicaciones realizadas previamente por el profesor; con el objetivo de profundizar en los contenidos abordados en las sesiones teóricas y prácticas. Estas sesiones permiten la reflexión o práctica por parte del estudiante, y pueden complementarse a través de la puesta en común en clases síncronas o con la realización de actividades y trabajos prácticos. Posteriormente, estas actividades son revisadas por el profesor, que traslada un feedback al estudiante sobre las mismas, pudiendo formar parte de la evaluación continua de la asignatura.

#### **4. Tutorías**

En esta actividad se engloban las sesiones virtuales de carácter síncrono y las comunicaciones por correo electrónico o campus virtual destinadas a la tutorización de los estudiantes. En ellas, el profesor comparte información sobre el progreso del trabajo del estudiante a partir de las evidencias recogidas, se resuelven dudas y se dan orientaciones específicas ante dificultades concretas en el desarrollo de la asignatura. Pueden ser individuales o colectivas, según las necesidades de los estudiantes y el carácter de las dudas y orientaciones planteadas. Tal y como se ha indicado, se realizan a través de videoconferencia y e-mail.

Se computan una serie de horas estimadas, pues, aunque existen sesiones comunes para todos los estudiantes, éstos posteriormente pueden solicitar al docente tantas tutorías como estimen necesarias.

Dado el carácter mixto de esta actividad formativa, se computa un porcentaje de sincronía estimado del 30%.

#### **5. Estudio autónomo**

En esta actividad el estudiante consulta, analiza y estudia los manuales, bibliografía y recursos propios de la asignatura de forma autónoma a fin de lograr un aprendizaje significativo y superar la evaluación de la asignatura de la asignatura. Esta actividad es indispensable para adquirir las competencias del título, apoyándose en el aprendizaje autónomo como complemento a las clases y actividades supervisadas.

#### **6. Examen final**

Como parte de la evaluación de cada una de las asignaturas (a excepción de las prácticas y el Trabajo fin de título), se realiza una prueba o examen final. Esta prueba se realiza en tiempo real (con los medios de control antifraude especificados) y tiene como objetivo evidenciar el nivel de adquisición de conocimientos y desarrollo de competencias por parte de los estudiantes. Los exámenes o pruebas de evaluación final se realizan en las fechas y horas programadas con antelación y con los sistemas de vigilancia online (proctoring) de la universidad.



## 5. Evaluación

### 5.1. Sistema de evaluación

El Modelo de Evaluación de estudiantes en la Universidad se sustenta en los principios del Espacio Europeo de Educación Superior (EEES), y está adaptado a la estructura de formación virtual propia de esta Universidad. De este modo, se dirige a la evaluación de competencias.

| Sistema de Evaluación | Ponderación |
|-----------------------|-------------|
| Portafolio*           | 60 %        |
| Sistema de Evaluación | Ponderación |

|                      |             |
|----------------------|-------------|
| <b>Prueba final*</b> | <b>40 %</b> |
|----------------------|-------------|

**\*Es requisito indispensable para superar la asignatura aprobar cada apartado (portafolio y prueba final) con un mínimo de 5 para ponderar las calificaciones.**

Los enunciados y especificaciones propias de las distintas actividades serán aportados por el docente, a través del Campus Virtual, a lo largo de la impartición de la asignatura.

Atendiendo a la Normativa de Evaluación de la Universidad, se tendrá en cuenta que la utilización de **contenido de autoría ajena** al propio estudiante debe ser citada adecuadamente en los trabajos entregados. Los casos de plagio serán sancionados con suspenso (0) de la actividad en la que se detecte. Asimismo, el uso de **medios fraudulentos durante las pruebas de evaluación** implicará un suspenso (0) y podrá implicar la apertura de un expediente disciplinario.

## 5.2. Sistema de calificación

La calificación de la asignatura se establecerá en los siguientes cálculos y términos:

| Nivel de aprendizaje | Calificación numérica | Calificación cualitativa |
|----------------------|-----------------------|--------------------------|
| Muy competente       | 9,0 - 10              | Sobresaliente            |
| Competente           | 7,0 - 8,9             | Notable                  |
| Aceptable            | 5,0 -6,9              | Aprobado                 |
| Aún no competente    | 0,0 -4,9              | Suspenso                 |

Sin detrimento de lo anterior, el estudiante dispondrá de una **rúbrica simplificada** en el aula que mostrará los aspectos que valorará el docente, como así también los **niveles de desempeño que tendrá en cuenta para calificar las actividades vinculadas a cada resultado de aprendizaje.**

La mención de «**Matrícula de Honor**» podrá ser otorgada a estudiantes que hayan obtenido una calificación igual o superior a 9.0. Su número no podrá exceder del cinco por ciento de los estudiantes matriculados en una materia en el correspondiente curso académico, salvo que el número de estudiantes matriculados sea inferior a 20, en cuyo caso se podrá conceder una sola «Matrícula de Honor».

## 6. Bibliografía