

Guía docente de la asignatura

## Cloud Computing: Fundamentos e Infraestructuras

Fecha última actualización: 12/07/2021  
Fecha de aprobación por la Comisión Académica: 21/07/2021

**Máster**

Máster Universitario en Ingeniería Informática

**MÓDULO**

Tecnologías Informáticas 1

**RAMA**

Ingeniería y Arquitectura

**CENTRO RESPONSABLE DEL TÍTULO**

Escuela Internacional de Posgrado

**Semestre**

Primero

**Créditos**

6

**Tipo**

Obligatorio

**Tipo de enseñanza**

Presencial

### PRERREQUISITOS Y/O RECOMENDACIONES

Los estudiantes deberán estar familiarizados con las técnicas fundamentales para desarrollo software que se usan en la actualidad, incluyendo

- Uso de sistemas de control de fuentes, así como alojamiento de las mismas en GitHub
- Diseño de una aplicación desde cero.
- Nociones básicas de arquitecturas de software.
- Lenguajes de programación de scripting, tales como Python o Javascript.
- Servicios avanzados de aplicaciones: almacenamiento de datos, monitorización y logs, y otros servicios similares.

### BREVE DESCRIPCIÓN DE CONTENIDOS (Según memoria de verificación del Máster)

Los contenidos que se impartirán en la asignatura son los siguientes:

- comparación de la arquitectura virtual con la arquitectura física, y se compararán los centros de procesamiento de datos reales y virtuales.
- Se hablará del hardware presente en los servidores y su virtualización, así como la diferencia entre almacenamiento y otros recursos computacionales reales y virtuales.
- Se explicará como se encaja la infraestructura con el departamento de sistemas y cómo se puede engranar con los desarrolladores de una empresa.

Finalmente se mostrarán ejemplos y aplicaciones, con una metodología centrada en el estudiante donde, tras una explicación inicial, el estudiante irá formulando un proyecto desarrollado a lo



largo del año que tendrá que alcanzar una serie de objetivos.

## COMPETENCIAS

### COMPETENCIAS BÁSICAS

- CB6 - Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.
- CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio.
- CB8 - Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios.
- CB9 - Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan a públicos especializados y no especializados de un modo claro y sin ambigüedades.
- CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

### COMPETENCIAS GENERALES

- G01 - Capacidad para proyectar, calcular y diseñar productos, procesos e instalaciones en todos los ámbitos de la ingeniería informática.
- G04 - Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos relacionados con la Ingeniería en Informática.
- G08 - Capacidad para comprender y aplicar la responsabilidad ética, la legislación y la deontología profesional de la actividad de la profesión de Ingeniero en Informática.

### COMPETENCIAS ESPECÍFICAS

- CE04 - Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.
- CE05 - Capacidad de comprender y saber aplicar el funcionamiento y organización de Internet, las tecnologías y protocolos de redes de nueva generación, los modelos de componentes, software intermediario y servicios.
- CE06 - Capacidad para asegurar, gestionar, auditar y certificar la calidad de los desarrollos, procesos, sistemas, servicios, aplicaciones y productos informáticos.
- CE07 - Capacidad para diseñar, desarrollar, gestionar y evaluar mecanismos de certificación y garantía de seguridad en el tratamiento y acceso a la información en un sistema de procesamiento local o distribuido.
- CE08 - Capacidad para analizar las necesidades de información que se plantean en un entorno y llevar a cabo en todas sus etapas el proceso de construcción de un sistema de información.



- CE09 - Capacidad para diseñar y evaluar sistemas operativos y servidores, y aplicaciones y sistemas basados en computación distribuida.
- CE10 - Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.
- CE11 - Capacidad de diseñar y desarrollar sistemas, aplicaciones y servicios informáticos en sistemas empotrados y ubicuos.
- CE15 - Capacidad para la creación y explotación de entornos virtuales, y para la creación, gestión y distribución de contenidos multimedia.

### COMPETENCIAS TRANSVERSALES

- CT01 - Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos
- CT02 - Capacidad de organización y planificación así como capacidad de gestión de la información.
- CT03 - Capacidad para el uso y aplicación de las TIC en el ámbito académico y profesional.
- CT04 - Capacidad de comunicación en lengua extranjera, particularmente en inglés.
- CT05 - Capacidad de trabajo en equipo.
- CT06 - Capacidad para innovar y generar nuevas ideas
- CT08 - Capacidad para proyectar los conocimientos, habilidades y destrezas adquiridos para promover una sociedad basada en los valores de la libertad, la justicia, la igualdad y el pluralismo

### RESULTADOS DE APRENDIZAJE (Objetivos)

- Conocer la historia de la Computación Virtual, sus orígenes y razones de su existencia.
- Conocer los conceptos relacionados con el proceso de virtualización tanto de software como de hardware.
- Comprender la diferencia entre infraestructura virtual y digital.
- Justificar la necesidad de procesamiento virtual frente a real en el contexto de una infraestructura TIC de una organización.
- Diseñar, construir y analizar las prestaciones de un centro de proceso de datos virtual.
- Conocer las diferentes tecnologías y herramientas de virtualización tanto para procesamiento, comunicación y almacenamiento.
- Instalar, configurar, evaluar y optimizar las prestaciones de un servidor virtual.
- Configurar los diferentes dispositivos físicos para acceso a los servidores virtuales: acceso de usuarios, redes de comunicaciones o entrada/salida.
- Diseñar, implementar y construir un centro de procesamiento de datos virtual.
- Documentar y mantener una plataforma virtual.
- Optimizar aplicaciones sobre plataformas virtuales.
- Conocer diferentes tecnologías relacionadas con la virtualización (Computación Nube, Utility Computing, Software as a Service, o Google AppSpot).
- Realizar tareas de administración en infraestructura virtual.
- Poner en práctica una infraestructura en conjunción con un equipo de desarrolladores (DevOps)

### PROGRAMA DE CONTENIDOS TEÓRICOS Y PRÁCTICOS



## TEÓRICO

Dado que se sigue la metodología de clase invertida y se trata de llevar a cabo un proyecto durante la asignatura, todas las clases son prácticas y basadas en proyecto, se hacen delante de un equipo informático, y se dedican a resolver dudas bajo demanda del estudiante, incluyendo pequeñas clases magistrales de unos 20-30 minutos al principio de un tema, cuando se inicia un hito, y cuando se entrega el hito. No hay diferencia, por tanto, entre un temario “teórico” y temario “práctico”

### TEMARIO:

- Arquitecturas software para la nube
- Gestión de configuraciones.
- Creación de infraestructuras virtuales reproducibles.
- Calidad de servicio y código en despliegues en la nube.
- Orquestación de máquinas virtuales.
- Uso y desarrollo usando contenedores.

### SEMINARIOS:

- Desarrollo colaborativo de software y despliegue en infraestructura nube.
- Infraestructura virtual de Azure y su uso práctico.
- DevOps y Big Data.
- Instalación práctica de máquinas virtuales.
- Medición de prestaciones en servicios web

## PRÁCTICO

Ver arriba, no hay distinción entre contenidos teóricos y prácticos, todos los contenidos impartidos están encaminados a ser puestos en práctica en una metodología docente basada en proyecto y centrada en el estudiante.

## BIBLIOGRAFÍA

### BIBLIOGRAFÍA FUNDAMENTAL

- El material de la asignatura, <https://jj.github.io/CC>.
- Curso de desarrollo ágil, con material, presentaciones y vídeos <https://jj.github.io/curso-tdd>
- What is DevOps, de Mike Loukides, O'Reilly.
- Cloud computing, Nayan Ruparelia (en la biblioteca de la UGR).
- Cloud Native Infrastructure, Justin Garrison, Kris Nova.

### BIBLIOGRAFÍA COMPLEMENTARIA

- Docker up and running, Sean P. Kane, O'Reilly
- Free Docker Books: <https://github.com/TechBookHunter/Free-Docker-Books>



## ENLACES RECOMENDADOS

- Cuentas de Twitter relacionadas con AWS, Azure, Docker, Podman, Google Cloud...
- Artículo en la Wikipedia [http://en.wikipedia.org/wiki/Cloud\\_computing](http://en.wikipedia.org/wiki/Cloud_computing).

## METODOLOGÍA DOCENTE

- MD01 Clases Teóricas-Expositivas
- MD02 Resolución de Problemas
- MD03 Resolución de Casos Prácticos
- MD04 Aprendizaje basado en Proyectos
- MD05 Prácticas en Laboratorio
- MD06 Taller de Programación
- MD09 Demos
- MD10 Exposición de Trabajos Tutelados
- MD11 Conferencias
- MD16 Tutorías Académicas

## EVALUACIÓN (instrumentos de evaluación, criterios de evaluación y porcentaje sobre la calificación final)

### EVALUACIÓN ORDINARIA

Se utilizarán las siguientes técnicas de evaluación:

- El 70% de la asignatura se evaluará a partir de los objetivos alcanzados en el desarrollo del proyecto y los hitos que se vayan entregando del mismo. Cada hito tendrá una ponderación sobre la nota final diferente, dependiendo del trabajo necesario para llevar a cabo cada uno de ellos.
- La ponderación del trabajo autónomo será el 30%. Para evaluar este trabajo autónomo hace falta una evidencia del mismo, sea en forma de presencia en las actividades que se planteen, o de colaboración a través del grupo de Telegram con el resto de los compañeros, así como, posiblemente, entrega continua y revisada de los ejercicios de autoevaluación. En su caso, la parte de trabajo autónomo y los seminarios se evaluarán teniendo en cuenta
  - originalidad en las herramientas usadas en el proyecto anterior
  - la participación de seminarios, incluyendo impartición de alguno de ellos,
  - los problemas propuestos que hayan sido resueltos y entregados por los alumnos, en su caso,
  - las actividades para crédito adicional propuestas a lo largo del curso
  - las entrevistas efectuadas durante el curso
  - y la presentación oral de los trabajos desarrollados.

En esta asignatura hay tolerancia cero con el plagio, y podrá poner en peligro la evaluación anterior. Dado que el estudiante tiene que desarrollar su propio trabajo dentro de su propio proyecto con un lenguaje y una serie de herramientas elegidas ex profeso, se considerará plagio si hay una cantidad sustancial de código copiada literalmente de un ejemplo en un tutorial, de cualquier otro sitio en la web o en un libro, o si se ha realizado una parte sustancial del mismo en colaboración con compañero o compañeros (lo que se detectará comparando el código/texto entregado por los mismos). La detección de plagio podrá tener las siguientes implicaciones, sin



que se descarten las acciones legales que figuran en los Estatutos de la Universidad:

1. En el primer caso, en la mayor parte de los casos se apercibirá al estudiante.
2. En la segunda ocasión se considerará el objetivo correspondiente no superado y el estudiante tendrá que rehacer el trabajo, evidentemente sin que se haga, en esta ocasión, ningún tipo de plagio.
3. En la tercera ocasión se pondrá en conocimiento de la coordinación del máster, con toda la documentación correspondiente, y se considerarán los objetivos de la asignatura no alcanzados en la convocatoria ordinaria, con la nota que en ese momento se haya alcanzado en la asignatura en acta.

La calificación global será la suma de los dos apartados anteriores.

Todo lo relativo a la evaluación se registrará por la normativa sobre planificación docente y organización de exámenes vigente en la Universidad de Granada.

### EVALUACIÓN EXTRAORDINARIA

Se llevará a cabo de la misma forma que la convocatoria ordinaria, con un 70% de nota para evaluación del proyecto, 30% de trabajo autónomo. El trabajo autónomo se evaluará de la misma forma que en la convocatoria extraordinaria.

En los diferentes hitos se podrán mantener las notas de los que se hayan entregado en la convocatoria ordinaria, o se podrá reenviar para subir nota, a elección del estudiante. El estudiante recibirá instrucciones sobre cómo llevar a cabo esa entrega para que se diferencie de las entregas ordinarias.

### EVALUACIÓN ÚNICA FINAL

Se hará un examen práctico sobre el ordenador, que consistirá en el despliegue en la nube de un proyecto aportado por el profesor. En este despliegue se tendrán que demostrar haber adquirido las diferentes competencias que se describen en la asignatura.

### INFORMACIÓN ADICIONAL

Se usará una forja de código y facilidades adicionales como sitio central de la asignatura en github.com. Los ejercicios y proyectos de los estudiantes se liberarán como software libre y se hará una introducción a la creación de aplicaciones de software libre y al desarrollo de software colaborativo usando el mismo (como seminario o bajo demanda)

