



Arquitectura y  
Programación de  
Sistemas en  
Internet  
**Grado en Ingeniería  
Informática**



UNIVERSIDAD  
**NEBRIJA**

## GUÍA DOCENTE

**Asignatura:** Arquitectura y Programación de Sistemas en Internet

**Titulación:** Grado en Ingeniería Informática

**Carácter:** Obligatoria

**Idioma:** Castellano.

**Modalidad:** Presencial

**Créditos:** 6

**Curso:** 3º

**Semestre:** 1º

**Profesores/Equipo Docente:** Dr. Alberto Valero Gómez / D. Javier Yebel de Blas Fernández / Fernando Murua Alcázar

### 1. COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

#### 1.1. Competencias

Esta asignatura contribuye a las siguientes competencias del plan de estudios:

*Competencias específicas:*

- CEB03. Comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- CEB04. Aplicar los conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CEB05. Aplicar el conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- CEC01. Diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- CEC05. Conocer, administrar y mantener sistemas, servicios y aplicaciones informáticas.
- CEC06. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CEC07. Conocer, diseñar y utilizar de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
- CEC08. Analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- CEC11. Conocer y aplicar las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- CEC13. Conocer y aplicar las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

CEC14. Conocer y aplicar los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

CEC17. Diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.

CEIC04. Diseñar e implementar software de sistema y de comunicaciones.

*Competencias generales:*

CGT1. Analizar y sintetizar la información necesaria para realizar su trabajo plasmando los resultados en informes o en la toma de decisiones en proyectos del ámbito de la ingeniería informática.

CGT2. Organizar y planificar los recursos e ideas necesarias para realizar su trabajo ideando acciones e hitos en proyectos del ámbito de la ingeniería informática.

CGT3. Comunicar de forma oral y escrita en la lengua nativa pudiendo expresar sus opiniones de forma clara para transmitir conceptos y soluciones dentro del ámbito de la ingeniería informática.

CGT5. Aplicar conocimientos de informática relativos al ámbito de estudio al trabajar en proyectos del ámbito de la ingeniería informática.

CGT6. Gestionar la información, conociendo su importancia y la forma de procesarla generando los recursos necesarios para facilitar su acceso y provisión en el ámbito de la ingeniería informática.

CGT7. Resolver problemas en el entorno de trabajo, dentro del ámbito de la ingeniería informática, enfrentándose a situaciones complejas en cuanto a problemas técnicos y a las relaciones personales y profesionales.

CGT8. Tomar decisiones basadas en la información disponible en el ámbito de la ingeniería informática.

CGS1. Aplicar la motivación por la calidad en el desarrollo de su actividad profesional poniendo el máximo cuidado en el desarrollo de los entregables dentro de proyectos de ingeniería informática.

CGS2. Razonar de forma crítica ante los problemas que surjan en el ámbito de la ingeniería informática, contando con la información disponible, y explicar dicho razonamiento.

CGS4. Aprender de forma autónoma conceptos relativos a la profesión ingenieril para facilitar la mejora continua ya sea mediante el acceso a información disponible o cualquier otro medio.

CGS5. Adaptarse a nuevas situaciones en el entorno de la ingeniería informática, reconociendo dichas situaciones y expresando formas de afrontarlas.

CGS6. Aplicar la creatividad ante las diferentes circunstancias generando soluciones novedosas dentro del ámbito de la ingeniería informática.

CGP1. Trabajar en equipo contribuyendo de forma activa al resultado de los proyectos u operaciones del ámbito de la ingeniería informática.

CGP2. Trabajar en un equipo de carácter interdisciplinar dentro de proyectos del ámbito de la ingeniería informática, entendiendo y pudiendo explicar la división de trabajo y la integración de los diferentes miembros del mismo.

## 1.2. Resultados de aprendizaje

Esta asignatura contribuye a los siguientes resultados de aprendizaje de la materia en la que se engloba:

- Arquitecturas de aplicaciones. Arquitecturas SOA (ESB, BPM, BAM).
- Programación de servidor. Lenguajes y plataformas
- Desarrollar aplicaciones y sistemas en internet.
- Despliegue de aplicaciones web.
- Sistemas software como servicios (SAAS)
- Almacenamiento de información persistente en Bases de Datos y almacenamiento web (cloud storage).
- Escalabilidad e infraestructuras web.
- Introducción a Frameworks de programación. Desarrollo de sistemas basados en JavaScript/TypeScript
- Arquitecturas cliente-servidor-base de datos.
- Seguridad de servidores de Internet.

## 2. CONTENIDOS

### 2.1. Requisitos previos

- Conocimiento sólido de los fundamentos de estructuras de datos y algoritmos.
- Conocimiento sólido de algún lenguaje de programación de alto nivel (C++, Java, Python, etc.)
- Conocimiento sólido de bases de datos.

### 2.2. Descripción de los contenidos

Arquitecturas de aplicaciones. Arquitecturas SOA (ESB, BPM, BAM). Programación de servidor. Lenguajes y plataformas. Herramientas de Portal. Gestión de Contenidos. Programación de aplicaciones web con J2EE. Servlets y JSPs. Enterprise Beans. Persistencia de datos. Introducción a Frameworks de programación. Introducción a entornos de otros lenguajes. Arquitecturas e-commerce. Seguridad de servidores de Internet.

### 2.3. Contenido detallado

Presentación de la asignatura.

Explicación de la Guía Docente.

1. Presentación de la Asignatura, del Profesor y de los Alumnos. Estructuración de los grupos de prácticas.
2. Arquitecturas de aplicaciones. Software as a Service (SAAS)
  - Desarrollo de APIs
  - Desarrollo Front.
  - Persistencia de datos.
3. Programación Sistemas Internet en Javascript/Typescript
  - Programación JavaScript/Typescript.
  - Persistencia de datos con MongoDB
  - GraphQL
  - API REST

**EXAMEN PARCIAL**

4. Autenticación y seguridad
5. Despliegue de aplicaciones web
  - Docker
  - Kubernetes
  - Integración Continua (CI)
6. Introducción a otros frameworks y lenguajes
  - Django y Python
  - J2EE y Java
  - PHP

**EXAMEN FINAL ORDINARIO**  
**EXAMEN FINAL EXTRAORDINARIO**

#### 2.4. Actividades Dirigidas

Como se indica en el contenido, durante el curso se desarrollarán actividades dirigidas en forma de prácticas orientadas al aprendizaje de los temas de con una duración total de aproximada de 15 horas. Las prácticas serán realizadas en grupos de 2 personas.

### 3. SISTEMA DE EVALUACIÓN

#### 3.1. Sistema de calificaciones

El sistema de calificaciones finales se expresará numéricamente del siguiente modo:

0 - 4,9 Suspenso (SS)

5,0 - 6,9 Aprobado (AP)

7,0 - 8,9 Notable (NT)

9,0 - 10 Sobresaliente (SB)

La mención de "matrícula de honor" podrá ser otorgada a alumnos que hayan obtenido una calificación igual o superior a 9,0.

#### 3.2. Criterios de evaluación

##### Convocatoria ordinaria

Sistemas de evaluación	Porcentaje
Evaluación de la participación del alumno	5%
Actividades dirigidas, prácticas y memorias de prácticas, trabajos (obligatorios y voluntarios) y proyectos a realizar.	20%
Prueba escrita parcial	10%
Prueba escrita final	65%

### Convocatoria extraordinaria

Sistemas de evaluación	Porcentaje
Actividades dirigidas, prácticas y memorias de prácticas, trabajos (obligatorios y voluntarios) y proyectos a realizar.	20%
Prueba escrita final	80%

### **3.3. Restricciones**

#### Calificación mínima

La ponderación tanto del examen parcial como de los conceptos de participación y trabajos escritos/prácticas, solo se aplicará si el alumno obtiene al menos un 5 en el examen final (ordinaria y extraordinaria).

La no obtención de al menos un 4 en las actividades dirigidas supone el suspenso automático de la asignatura en la convocatoria ordinaria. Se conservará la nota de las prácticas para posteriores convocatorias. No se conservará la nota del trabajo.

Las actividades dirigidas que no hayan sido aprobadas pueden NO podrán ser entregadas de nuevo en la convocatoria extraordinaria. Para la convocatoria extraordinaria NO es necesario tener al menos un 4 en las actividades dirigidas.

El examen parcial no libera materia.

#### Asistencia

El alumno que, injustificadamente, deje de asistir a más de un 25% de las clases presenciales podrá verse privado del derecho a examinarse en la convocatoria ordinaria.

#### Normas de escritura

Se prestará especial atención en los trabajos, prácticas y proyectos escritos, así como en los exámenes tanto a la presentación como al contenido, cuidando los aspectos gramaticales y ortográficos. El no cumplimiento de los mínimos aceptables puede ocasionar que se resten puntos en dicho trabajo.

### **3.4. Advertencia sobre plagio**

La Universidad Antonio de Nebrija no tolerará en ningún caso el plagio o copia. Se considerará plagio la reproducción de párrafos a partir de textos de auditoría distinta a la del estudiante (Internet, libros, artículos, trabajos de compañeros...), cuando no se cite la fuente original de la que provienen. El uso de las citas no puede ser indiscriminado. El plagio es un delito.

En caso de detectarse este tipo de prácticas, se considerará Falta Grave y se podrá aplicar la sanción prevista en el Reglamento del Alumno.

#### 4. BIBLIOGRAFÍA

##### Cursos Online

- The Modern GraphQL Bootcamp (with Node.js and Apollo) - <https://www.udemy.com/graphql-bootcamp/>
- Docker and Kubernetes: The Complete Guide - <https://www.udemy.com/docker-and-kubernetes-the-complete-guide/>

##### Libros

- Eloquent Javascript, 3rd Edition: A Modern Introduction to Programming. (inglés) – 1 ene. 2019 - <https://eloquentjavascript.net/>
- Learning GraphQL: Declarative Data Fetching for Modern Web Apps (inglés) Tapa blanda – 24 ago. 2018