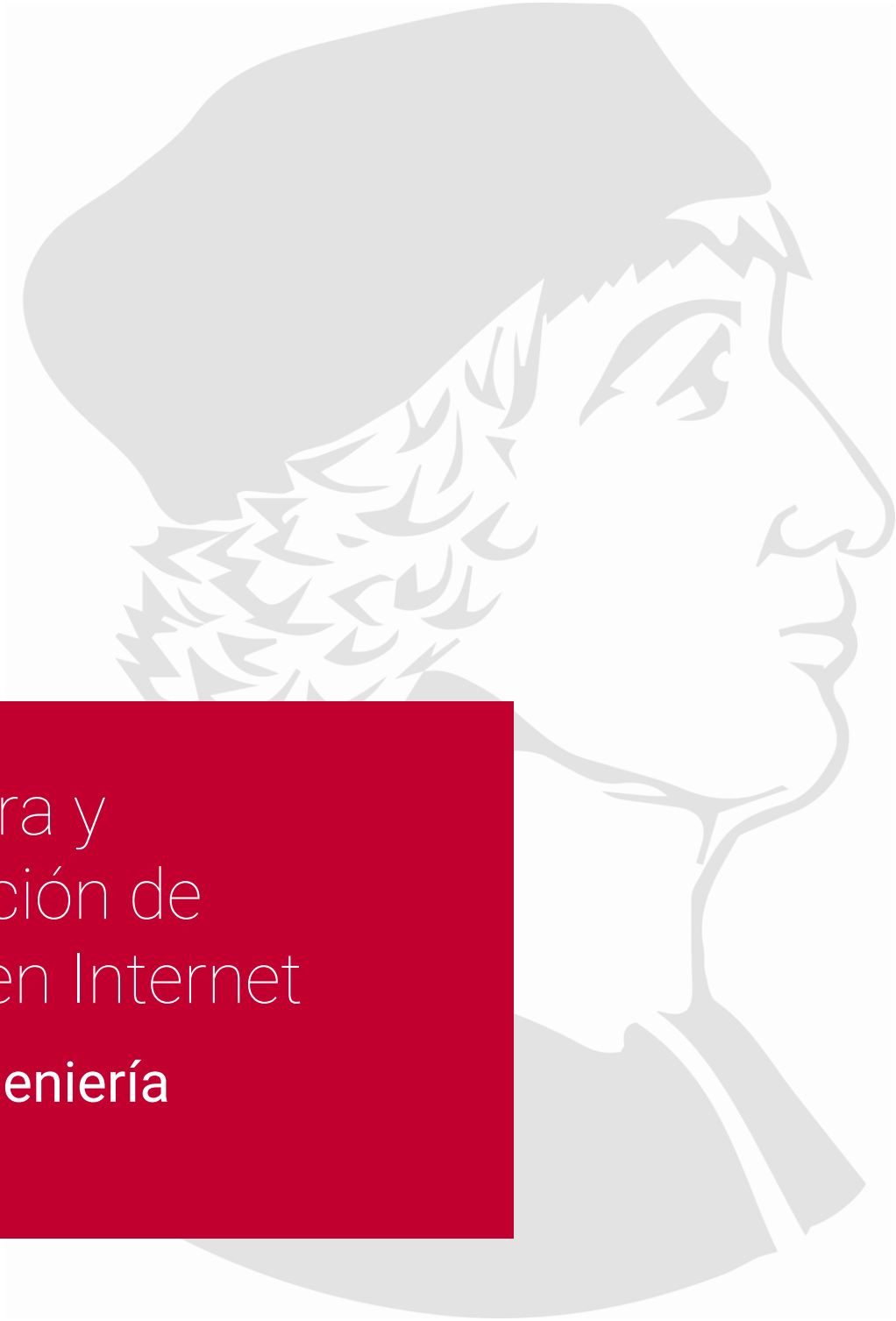

Arquitectura y programación de sistemas en Internet

Grado en Ingeniería Informática

UNIVERSIDAD NEBRIJA

GUÍA DOCENTE

Asignatura: Arquitectura y programación de sistemas en Internet

Titulación: Grado en Ingeniería Informática

Carácter: Obligatoria

Idioma: Español

Modalidad: Presencial

Créditos: 6

Curso: 3º

Semestre: 1º

Profesores/Equipo docente: Dr. D. Alberto Valero Gómez, D. Javier Yebel de Blas Fernández (prácticas), D. Fernando Murua Alcázar (prácticas)

COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

1.1. Competencias

Competencias básicas

- CB1, CB2, CB3, CB4, CB5

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.
- 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.
- 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.

1.2. Resultados de aprendizaje

Al terminar con éxito esta asignatura los estudiantes serán capaces de:

- Aplicar procedimientos algorítmicos básicos de las tecnologías informáticas.
- Diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- Determinar los tipos y estructuras de datos avanzadas más adecuados a la resolución de un problema.
- Emplear métodos de programación imperativa y de programación orientada a objetos para desarrollar soluciones a problemas concretos.
- Interpretar código realizado por terceros para su reutilización.
- Evaluar diferentes diseños de aplicaciones para seleccionar el más apropiado para resolver un problema.

2. CONTENIDOS

2.1. Requisitos previos

Conocimiento solido 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. **Arquitecturas de aplicaciones. Software as a Service (SAAS).**
   - Desarrollo de APIs.
   - Desarrollo Front.
   - Persistencia de datos.
2. **Programación sistemas Internet en Javascript/Typescript**
   - Programación JavaScript/Typescript.
   - Persistencia de datos con MongoDB.
   - GraphQL.
   - API REST.
3. **Autenticación y seguridad.**
4. **Despliegue de aplicaciones web.**
   - Docker.
   - Kubernetes.
   - Integración Continua (CI).
5. **Unit testing.**

2.4. Actividades dirigidas

Durante el curso se desarrollarán actividades dirigidas en forma de prácticas orientadas al aprendizaje de los temas vistos en las sesiones teóricas. Las prácticas serán individuales o en grupo.

2.5 Actividades formativas

CÓDIGO	ACTIVIDAD FORMATIVA	HORAS	PORCENTAJE DE PRESENCIALIDAD
AF1	Clases de teoría, evaluación y problemas	45	100%
AF2	Tutorías	12,5	100%
AF3	Prácticas	15	100%
AF4	Estudio individual	77,5	0%

3. SISTEMA DE EVALUACIÓN

3.1. Sistema de calificaciones

El sistema de calificaciones (R.D. 1125/2003, de 5 de septiembre) será el siguiente:

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. Su número no podrá exceder del cinco por ciento de los alumnos matriculados en una materia en el correspondiente curso académico, salvo que el número de alumnos matriculados sea inferior a 20, en cuyo caso se podrá conceder una sola «Matrícula de Honor».

3.2. Criterios de evaluación

Convocatoria ordinaria

Sistemas de evaluación	Porcentaje
SE1 Prueba parcial	15%
SE2 Examen final	55%
SE3 Evaluación de la participación del estudiante	10%
SE4 Actividades dirigidas, prácticas y memorias de prácticas, trabajos (obligatorios y voluntarios) y proyectos a realizar	20%

Convocatoria extraordinaria

Sistemas de evaluación	Porcentaje
SE2 Examen final	80%
SE4 Actividades dirigidas, prácticas y memorias de prácticas, trabajos (obligatorios y voluntarios) y proyectos a realizar	20%

3.3. Restricciones

Calificación mínima

La ponderación tanto del examen parcial como de los conceptos de participación y trabajos escritos/prácticas, sólo se aplicará si el alumno obtiene al menos 5,0 puntos en el examen final (ordinaria/extraordinaria).

La no superación de las actividades dirigidas supone el suspenso automático de la asignatura en la convocatoria ordinaria y extraordinaria. NO se conservará la nota de las prácticas para la convocatoria posterior.

Las actividades dirigidas que no hayan sido aprobadas no podrán ser entregadas de nuevo en la convocatoria extraordinaria.

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 autoría 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

Bibliografía básica

- 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.

Bibliografía complementaria

- 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/.