

BASIC DETAILS:

Subject:	TECNOLOGÍAS AVANZADAS		
Id.:	30495		
Programme:	GRADUADO EN INGENIERÍA INFORMÁTICA (ITINERARIO DE ADAPTACIÓN)		
Module:	APLICACIONES DISTRIBUIDAS		
Subject type:	OBLIGATORIA		
Year:	4	Teaching period:	Primer Cuatrimestre
Credits:	6	Total hours:	150
Classroom activities:	16	Individual study:	134
Main teaching language:	Inglés	Secondary teaching language:	Castellano
Lecturer:		Email:	

PRESENTATION:

To move forward as a system architecture engineer, the student needs to gain experience in advanced web technologies. This subject deals with some Javascript technologies used often in Internet applications that large enterprise organizations require, and also provides knowledge that can be harnessed to prepare a certification as architect of NodeJS systems. API REST (Representational State Transfer) is designed to take advantage of existing protocols. While REST can be used over nearly any protocol, it usually takes advantage of HTTP when used for Web APIs Web Services.

Since many of us are now writing or maintaining RESTful microservices and these services/ APIs are exposed to the web and distributed over different networks, they are vulnerable to risks and security threats which affect the processes based on them. Hence, testing becomes necessary to ensure they perform correctly. To test these APIs, it's very important to automate REST API test cases instead of relying on manual testing.

PROFESSIONAL COMPETENCES ACQUIRED IN THE SUBJECT:

General programme competences	G04	Capacidad para trabajar siempre con responsabilidad y compromiso, creando un alto sentido del deber y el cumplimiento de las obligaciones
	G10	Capacidad crítica y analítica en la evaluación de información, datos y líneas de actuación
	G11	Habilidad para desenvolverse en un entorno multicultural e internacional, relacionándose con personas de distintas nacionalidades, idiomas y culturas
	G12	Capacidad para desarrollar las actividades profesionales con integridad respetando normas sociales, organizacionales y éticas
	G13	Capacidad para utilizar estrategias de aprendizaje individuales orientadas a la mejora continua en el ejercicio profesional y para emprender estudios posteriores de forma autónoma
	G14	Capacidad de abstracción para manejar diferentes modelos complejos de conocimiento y aplicarlos al planteamiento y resolución de problemas
	G15	Capacidad para estructurar la realidad relacionando objetos, situaciones y conceptos a través del razonamiento lógico matemático
Specific programme competences	E06	Capacidad para aplicar procesos de aseguramiento de la calidad en procesos y productos
	E07	Capacidad para trabajar eficazmente en equipos de proyecto, asumiendo en su caso responsabilidades directivas, y considerando los aspectos humanos, tecnológicos y financieros
	E09	Capacidad para mantener las competencias profesionales mediante el aprendizaje autónomo y la mejora continua
	E12	Capacidad para gestionar la complejidad a través de la abstracción, el modelado, las "best practices", los patrones, los estándares y el uso de herramientas apropiadas
	E15	Capacidad para comprender y alinearse con los objetivos estratégicos de la empresa en la que se desarrollan las actividades profesionales
	E17	Capacidad para identificar y analizar las necesidades de los usuarios con el objetivo de diseñar soluciones IT efectivas y usables que puedan integrarse en el entorno operativo del usuario.
	E18	Capacidad para identificar y definir los requisitos que deben ser satisfechos por los sistemas IT para satisfacer las necesidades planteadas por organizaciones o individuos
	E21	Capacidad para realizar pruebas que verifiquen la validez del proyecto (funcional, integridad de los datos, rendimiento de las aplicaciones informáticas, equipos, comunicaciones, etc.)
	E22	Capacidad para desarrollar labores de implementación en las que se requiera un alto grado de conocimientos técnicos en diversos ámbitos (programación, configuración de equipos hardware y de comunicaciones, etc.).
E23	Capacidad para diseñar e implementar políticas de seguridad con el objetivo de preservar la integridad de los entornos operativos	

	E24	Capacidad para trazar y desarrollar planes de proyecto efectivos para sistemas basados en tecnologías de la información y comunicaciones
	E25	Capacidad para analizar la viabilidad, proyectar planes de desarrollo, estimar los recursos, liderar y controlar la ejecución de proyectos de ingeniería intensivos en software
Learning outcomes	R01	Entender el enfoque orientado a objetos a través de objetos remotos.
	R02	Utilizar la estructura básica de Servlets
	R03	Desarrollar aplicaciones JSP que realicen servicios Web.
	R04	Aplicar la tecnología AJAX
	R05	Manejar la plataforma Java Enterprise Edition para desarrollar grandes aplicaciones

PRE-REQUISITES:

The student should have passed the subjects: Oriented Object Programming I and II

SUBJECT PROGRAMME:

Subject contents:

1 - Basic resources
2 - Overview
2.1 - History
2.2 - Introduction to API design
2.3 - Maturity model
3 - Resources and representation
4 - The design procedure
5 - Status codes & headers
6 - Authentication & authorization
7 - Testing

Subject planning could be modified due unforeseen circumstances (group performance, availability of resources, changes to academic calendar etc.) and should not, therefore, be considered to be definitive.

TEACHING AND LEARNING METHODOLOGIES AND ACTIVITIES:

Teaching and learning methodologies and activities applied:

Day by day the student must compose his portfolio of the subject. It include lectures summary and solved exercises. According to the previous paragraph calendar, several exercises will be proposed and its solutions will be discussed some time later. The solutions of everyday exercises will be included in the portfolio and sent to the PDU when they are required. The lecturer will upload his lectures notes on the PDU (Plataforma Docente Universitaria). Self-learning: after class, students have to finish the task has left uncompleted in class or do remain ones. All task must be uploaded to PDU in the time ordered. Students are recommended to consult the lecturer by e-mail.

At the end of the term students must pass a test. It could be composed of some questions, exercises, or oral presentation.

Student work load:

Teaching mode	Teaching methods	Estimated hours
Classroom activities	Master classes	14
	Assessment activities	2
Individual study	Tutorials	11
	Individual study	55
	Individual coursework preparation	44
	Portfolio	24
	Total hours:	150

ASSESSMENT SCHEME:

Calculation of final mark:

Written tests:	40 %
Individual coursework:	20 %
Group coursework:	15 %
Final exam:	25 %
TOTAL	100 %

*Las observaciones específicas sobre el sistema de evaluación serán comunicadas por escrito a los alumnos al inicio de la materia.

BIBLIOGRAPHY AND DOCUMENTATION:

Basic bibliography:

RESTful Web APIs by Leonard Richardson and Mike Amundsen (O'Reilly). Copyright 2013 Leonard Richardson and Amundsen.com, Inc., and Sam Ruby. 978-1-449-35806-8.

Recommended bibliography:

RESTful Web Clients by Mike Amundsen (O'Reilly). Copyright 2017 Mike Amundsen.com, Inc., and Sam Ruby. 978-1-491-92190-6

Recommended websites:

NodeJS	http://nodejs.org
Docker	https://www.docker.com
PostgreSQL	https://www.postgresql.org
Kitematic	https://kitematic.com
pgAdmin	https://www.pgadmin.org
Visual studio code	https://code.visualstudio.com
Postman	https://www.getpostman.com

* Guía Docente sujeta a modificaciones