GUÍA DOCENTE

ARQUITECTURAS DE SOFTWARE

Coordinación: GARCIA GONZALEZ, ROBERTO

Año académico 2020-21

Información general de la asignatura

Denominación	ARQUITECTURAS DE SOFTWARE
Código	102055
Semestre de impartición	1R Q(SEMESTRE) EVALUACIÓN CONTINUADA
Carácter	

Grado/Máster	Curso	Carácter	Modalidad
Grado en Ingeniería Informática	4	OBLIGATORIA	Presencial

Número de créditos de la asignatura (ECTS)	9
Tipo de actividad, créditos y grupos	

Tipo de actividad	PRALAB	TEORIA
Número de créditos	3	6
Número de grupos	1	1

Coordinación	GARCIA GONZALEZ, ROBERTO
Departamento/s	INFORMATICA E INGENIERIA INDUSTRIAL
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	Carga total: 225h - Presencial (40%) = 90h - Trabajo autónomo del estudiante (60%) = 135h
Información importante sobre tratamiento de datos	Consulte este enlace para obtener más información.
Idioma/es de impartición	Inglés

Profesor/a (es/as)	Dirección electrónica\nprofesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
BERGA GATIUS, ALBERT	albert.berga@udl.cat	5,4	
GARCIA GONZALEZ, ROBERTO	roberto.garcia@udl.cat	5,4	

Información complementaria de la asignatura

Para un mejor seguimiento de esta asignatura, se recomienda haber consolidado los conocimientos de ingeniería del software presentados amb anterioridad en las assignaturas de la especialización correspondiente.

Objetivos académicos de la asignatura

- Presentar la Ingeniería Web y sus patrones fundamentales de desarrollo de aplicaciones y arquitectura.
- Tener una visión global de les tecnologías existentes para implementar aplicaciones empresariales que utilicen los patrones y arquitecturas anteriores, especialmente Java i la Web.
- Poner en práctica los patrones y tecnologías presentados mediante la realización de un proyecto de aplicación utilizando Java (Spring) y Javascript (Angular), aplicando una metodologia ágil y desarrollo guiado por el comportamiento (BDD).

Competencias

Competencias estratégicas

CT2. Adquirir un dominio significativo de una lengua extranjera, especialmente del inglés

CT3. Adquirir capacitación en el uso de las nuevas tecnologías y de las tecnologías de la información y la comunicación

Competencias transversales

EPS11. Capacidad de comprender las necesidades del usuario expresadas en un lenguaje no técnico.

Competencias específicas

GII-IS1. Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.

GII-IS3. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

GII-IS4. Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.

Contenidos fundamentales de la asignatura

1. Proceso de desarrollo ágil de aplicaciones Web
    1. Desarrollo guiado por comportamiento (BDD) con Cucumber Java
    2. Scrum con GitHub
    3. Coordinación del equipo con Control de Versiones
    4. Integración Continua con TravisCI
    5. Despliegue automático con Heroku
2. Patrones de arquitectura de aplicaciones
    1. Introducción a patrones
    2. Patrones en el contexto de aplicaciones empresariales
    3. Detalles de patrones
    4. Ejemplos de patrones de aplicaciones
    5. Tecnologías para patrones de aplicaciones
3. Implementación de aplicaciones Web
    1. Lado servidor: Java (Spring)
    2. Lado cliente: TypeScript (Angular)

Ejes metodológicos de la asignatura

La metodología se basa en el Aprendizaje Basado en Proyectos y se desarrolla una aplicación de software empresarial, centrándose en las aplicaciones basadas en Web. El curso se inicia con una revisión de frameworks de desarrollo Web desde un punto de vista empresarial, analizando el número de ofertas de trabajo que lo mencionan, las preguntas en StackOverflow, menciones en LinkedIn, etc.. Partiendo de estas dimensiones, Spring se selecciona para el lado del servidor y AngularJS para el cliente.

Las primeras semanas del curso se centran en aspectos fundamentales del desarrollo de software, incluyendo ideas de gestión de proyectos y conceptos del proceso de desarrollo. Especificación, análisis, diseño y arquitectura de software se presentan desde una perspectiva de la Web, guiadas por patrones de aplicación de empresa y cliente/servidor. Estos conceptos son el punto de partida para el desarrollo de los proyectos de los estudiantes.

Los estudiantes colaboran en un proyecto común, bajo la guía del profesor, y completan una primero entrega donde se aplican estos conceptos fundamentales para especificar, analizar y diseñar su proyecto.

El resto del curso se centra en la ejecución del proyecto definido y se introducen los conceptos necesarios relacionados con el desarrollo Ágil, Spring o Angular según sea necesario para completar el desarrollo del proyecto.

Plan de desarrollo de la asignatura

Sem	Descripción	Actividad Presencial	Trabajo autónomo
1	Frameworks y Metodologías	Estudio Frameworks JVM Servidor Estudio Frameworks JavaScript Cliente	Planteamiento idea proyecto
2	Frameworks y Metodologías	SRUM y BDD	Definición proyecto en términos de la metodología
3	Sprint 1	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
4	Sprint 1	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
5	Sprint 1	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto

Sem	Descripción	Actividad Presencial	Trabajo autónomo
6	Sprint 1	Desarrollo conjunto del proyecto Desarrollo conjunto del proyecto	Desarrollo del proyecto
7	Sprint 1	Patrones de aplicaciones empresariales Desarrollo conjunto del proyecto	Análisis aplicación patrones al proyecto
8	Sprint 1	Patrones de aplicaciones empresariales Sprint Retrospective	Análisis aplicación patrones al proyecto
9		1er Parcial	Estudiar
10	Sprint 2	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
11	Sprint 2	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
12	Sprint 2	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
13	Sprint 2	Desarrollo conjunto del proyecto Sprint Retrospective	Desarrollo del proyecto
14	Sprint 3	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
15	Sprint 3	Presentación tutoriales Desarrollo conjunto del proyecto	Desarrollo del proyecto
16	Sprint 3	Desarrollo conjunto del proyecto Sprint Retrospective	Desarrollo del proyecto
17-18		2o Parcial	Estudiar

Sistema de evaluación

Acr.	Actividades de Evaluación	Ponderación	Nota Mínima	En grupo	Obligatoria	Recuperable
P1	Examen 1er Parcial	20%	NO	NO	SI	NO
P2	Examen 2o Parcial	20%	NO	NO	SI	NO
E1	1ª Entrega Proyecto	20%	NO	1 o 2	SI	NO
E2	2ª Entrega Proyecto	20%	NO	1 o 2	SI	NO
E3	3ª Entrega Proyecto	20%	NO	1 o 2	SI	NO
Nota Final = 0,2*P1 + 0,2*P2 + 0,2*E1 + 0,2*E2 + 0,2*E3						

La evaluación se basa fundamentalmente en el desarrollo de un proyecto siguiendo una metodología ágil en base a

3 iteraciones y 3 entregas:

- Sprint 1, 1ª Entrega: 20% nota
- Sprint 2, 2ª Entrega: 20% nota
- Sprint 3, 3ª Entrega: 20% nota

En cada iteración (revisión) se evaluará el desenpeño en el desarrollo de las tareas encomendadas para esa iteración y se pondrá en el contexto de cada entrega, que evaluará también el producto entregado. Se combinará en una nota para cada entrega y sus iteraciones asociadas.

La evaluación se complementa con dos pruebas individuales:

- 1er parcial: 20% nota, aspectos relacionados fundamentalmente con los aspectos básicos de los frameworks y metodologías utilizados.
- 2o parcial: 20% nota, aspectos relacionados con la experiencia de desarrollo del proyecto en su conjunto.

Bibliografía y recursos de información

- Fowler, M.; Rice, D. (2003). Patterns of Enterprise Application Architecture. Addison-Wesley.
    - https://discovery.udl.cat/iii/encore/record/C__Rb1199416
- Walls, C. (2015). Spring in Action, 4th Edition. Manning.
    - https://discovery.udl.cat/iii/encore/record/C__Rb1326886
- Carnell, J. (2017) Spring microservices in action. Manning.
    - https://discovery.udl.cat/iii/encore/record/C__Rb1354387
- Wilken, J. (2018). Angular in action. Manning.
    - https://discovery.udl.cat/iii/encore/record/C__Rb1354388
- Amuthan, G. (2014). Spring MVC: Beginner's guide Birmingham. Packt Publishing.
    - https://www.dawsonera.com/abstract/9781783284887