

## GUÍA DOCENTE 2020-2021

### DATOS GENERALES DE LA ASIGNATURA

|   |                                 |
|---|---------------------------------|
| <b>ASIGNATURA:</b>                      | Ingeniería de Software I        |
| <b>PLAN DE ESTUDIOS:</b>                | Grado en Ingeniería Informática |
| <b>FACULTAD:</b>                        | Escuela Politécnica Superior    |
| <b>CARÁCTER DE LA ASIGNATURA:</b>       | Obligatoria                     |
| <b>ECTS:</b>                            | 6                               |
| <b>CURSO:</b>                           | Tercero                         |
| <b>SEMESTRE:</b>                        | Primero                         |
| <b>IDIOMA EN QUE SE IMPARTE:</b>        | Español                         |
| <b>PROFESORADO:</b>                     | Lázaro Javier Hernández         |
| <b>DIRECCIÓN DE CORREO ELECTRÓNICO:</b> | lazarohernandez@uneatlantico.es |

### DATOS ESPECÍFICOS DE LA ASIGNATURA

|  |
|--|
| <b>REQUISITOS PREVIOS:</b>   |
|  |
| <b>CONTENIDOS:</b>   |
| <ul style="list-style-type: none"> <li>• Tema 1: Introducción a la Ingeniería de Software.</li> <li>• Tema 2: Elementos fundamentales de la producción de Software.             <ul style="list-style-type: none"> <li>- Equipos de Desarrollo de Software.</li> <li>- Metodologías de Desarrollo.</li> </ul> </li> <li>• Tema 3: Ingeniería de Requerimientos.             <ul style="list-style-type: none"> <li>- Modelado del Negocio</li> <li>- Captura de Requerimientos</li> <li>- Mockups.</li> <li>- Análisis Textual</li> </ul> </li> <li>• Tema 4: Estimación de un Proyecto de Software</li> </ul> |

## COMPETENCIAS

### COMPETENCIAS GENERALES:

- CG1 Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.
- CG2 Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática.
- CG3 Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.
- CG5 Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad,
- CG9 Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

### COMPETENCIAS ESPECÍFICAS:

- CE07 Capacidad para 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.
- CE08 Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- CE10 Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- CE22 Capacidad de conocer y aplicar los principios, metodologías y ciclos de vida de la ingeniería de software.
- CE26 Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.
- CE28 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.
- CE30 Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.

### RESULTADOS DE APRENDIZAJE:

- Conocer y aplicar las técnicas de ingeniería de requisitos
- Conocer y aplicar los distintos tipos de modelos de ciclos de vida del software

- Conocer los procesos de software a profundidad y cómo evaluarlos para mejorarlos
- Conocer las técnicas más comunes de análisis y diseño de software
- Conocer y saber aplicar aspectos de calidad en el desarrollo de software como la usabilidad, accesibilidad, seguridad, fiabilidad, etc.
- Dirigir un proyecto basándose en los principios de análisis, diseño y gestión.

## METODOLOGÍAS DOCENTES Y ACTIVIDADES FORMATIVAS

### METODOLOGÍAS DOCENTES:

- MD1 - Método expositivo
- MD2 - Estudio y análisis de casos
- MD3 - Resolución de ejercicios
- MD4 - Aprendizaje basado en problemas
- MD5 - Aprendizaje orientado a Proyectos
- MD6 - Aprendizaje cooperativo / Trabajo en grupos
- MD7 - Trabajo autónomo

### ACTIVIDADES FORMATIVAS:

A partir de las metodologías docentes especificadas anteriormente, en esta asignatura, el alumno participará en las siguientes actividades formativas:

| Actividades formativas          |                                      |
|---------------------------------|--------------------------------------|
| <b>Actividades dirigidas</b>    | Clases expositivas                   |
|                                 | Clases prácticas                     |
|                                 | Seminarios y Talleres                |
| <b>Actividades supervisadas</b> | Supervisión de actividades           |
|                                 | Tutorías (individual / en grupo)     |
| <b>Actividades autónomas</b>    | Preparación de clases                |
|                                 | Estudio personal y lecturas          |
|                                 | Elaboración de trabajos              |
|                                 | Trabajo individual en campus virtual |

El primer día de clase, el profesor proporcionará información más detallada al respecto.

## SISTEMA DE EVALUACIÓN

### CONVOCATORIA ORDINARIA:

En la convocatoria ordinaria de esta asignatura se aplican los siguientes instrumentos de evaluación:

|                     | Actividades de evaluación                           | Ponderación |
|---------------------|---|-------------|
| Evaluación continua | 1 Examen Parcial                                    | 25 %        |
|                     | Entregas de Portfolios y Ejercicios                 | 20 %        |
|                     | Interés y participación del alumno en la asignatura | 5 %         |
| Evaluación final    | Examen Teórico-Práctico                             | 50 %        |

La calificación del instrumento de la evaluación final (tanto de la convocatoria ordinaria como de la extraordinaria, según corresponda) **no podrá ser inferior, en ningún caso, a 4,0 puntos** (escala 0 a 10) para aprobar la asignatura y consecuentemente poder realizar el cálculo de porcentajes en la calificación final.

### CONVOCATORIA EXTRAORDINARIA:

La convocatoria extraordinaria tendrá lugar durante el mes de julio (consúltase el calendario académico fijado por la universidad). Esta consistirá en la realización de uno o dos Exámenes Teórico-Prácticos con un valor de hasta el 50% de la nota final de la asignatura. El resto de la nota se complementará con la calificación obtenida en la evaluación continua de la convocatoria ordinaria.

## BIBLIOGRAFÍA Y RECURSOS DE REFERENCIA GENERALES

### BIBLIOGRAFÍA BÁSICA:

- Somerville, I. (2017) *Ingeniería de Software*. Pearson, Ciudad de México.
- Schach, Sthepen R. (2006) *Ingeniería de Software clásica y orientada a objetos*. McGrawHill

### BIBLIOGRAFÍA COMPLEMENTARIA:

- Baracho, R. A. (2016). Organização e recuperação da informação pilares da arquitetura da informação. *Tendências da Pesquisa Brasileira em Ciência da Informação*, 9(1).
- Cáceres, P., Marcos, E., & Kybele, G. (2001). Procesos ágiles para el desarrollo de aplicaciones Web. Taller de Web Engineering de las Jornadas de Ingeniería del Software y Bases de Datos de, 2001.
- Chatterjee, S., Xiao, X., Elbanna, A., & Saker, S. (2017, January). The Information Systems Artifact: A Conceptualization Based on General Systems Theory. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.
- Corona, B., Muñoz, M., Miramontes, J., Calvo-Manzan, J. A., & San Feliu, T. (2016). Estado de arte sobre métodos de evaluación de metodologías ágiles en las pymes. *ReCIBE*, 5(1).
- Corona, B., Muñoz, M., Miramontes, J., Calvo-Manzan, J. A., & San Feliu, T. (2017). Estado de arte sobre métodos de evaluación de metodologías ágiles en las pymes (Art state about evaluation methods of agile methodologies in pymes). *ReCIBE*, 5(1).
- de Oliveira, F. G., & Seabra, J. M. P. (2015). METODOLOGIAS DE DESENVOLVIMENTO DE SOFTWARE: UMA ANÁLISE NO DESENVOLVIMENTO DE SISTEMAS NA WEB. *TECNOLOGIAS EM PROJEÇÃO*, 6(1), 20-34.
- Enterprise Agile: Enterprise Architecture. (2017). [enterpriseunifiedprocess.com](http://www.enterpriseunifiedprocess.com). Retrieved 13 July 2017, from <http://www.enterpriseunifiedprocess.com/essays/enterpriseArchitecture.html>
- Grupo IWT2. (2017). Manuales de NDT Y Ejemplos | Grupo IWT2. [online] Available at: <http://iwt2.org/actividad-grupo/investigacion/resultados/ndt/aprendiendo-ndt/manuales-de-ndt-y-ejemplos/> [Accessed 19 Jul. 2017].
- InfoQ. (2017). Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch. [online] Available at: <https://www.infoq.com/articles/standish-chaos-2015> [Accessed 15 Jul. 2017].
- Jacobson, I., Spence, I., & Kerr, B. (2011). Use-case 2.0. Ivar Jacobson International.
- Klap diseño producto + Lean Startup. (2017). Klap Lean Startup + Design. Retrieved 21 July 2017, from <http://klap.es> ("Klap diseño producto + Lean Startup", 2017)

- Nazareno, R., Leone, H. P., & Gonnet, S. M. (2013). Trazabilidad de procesos ágiles: un modelo para la trazabilidad de procesos SCRUM. In XVIII Congreso Argentino de Ciencias de la Computación.
- Peronti, G. G., Veiga, M., & da Silva, A. A. B. (2016). A representação do espaço de arquitetura por meio de dispositivos táteis: uma revisão conceitual e tecnológica. *Blucher Design Proceedings*, 3(1), 561-566.
- Pinto, A. L., da Silva, A. M., & Sena, P. M. B. (2017). Ontologias baseadas na visualização da informação das redes sociais. *Prisma. com*, (13).
- Pletsch, L. C., Leone, H., Bollati, V. A., & Gonnet, S. (2016). Desarrollo de softwares dirigidos en entorno ágiles.
- Pressman, R. S. (2010). *Ingeniería del Software Un enfoque práctico*, Séptima edición ed. McGrawHill.
- Rojas, F. M., Medina, I. I. S., Núñez, J. M., & Medina, J. M. C. (2016, November). Sistema de información para administrar fincas productoras de café mediante el método de desarrollo SCRUM. In *Memorias de Congresos UTP* (Vol. 1, No. 1, pp. 41-44).
- Salinas, C. J. T. (2017). *A mature agile approach in web engineering contexts* (Doctoral dissertation, Universidad de Sevilla).
- Silva, L. (2016). *Sistemas de Informação WEB para gestão de stock* (Bachelor's thesis).
- Souza, N. M. D., Simão, D. D., Oliveira, L. B. R. D., Lana, C. A., Nakagawa, E. Y., & Maldonado, J. C. (2017). *Relação entre arquitetura de software e teste de software: um mapeamento sistemático*.

#### WEBS DE REFERENCIA:

- <http://www.SoftwareEngineering-9.com>

#### OTRAS FUENTES DE CONSULTA:

- PAe - Métrica v.3. (2017). [Administracionelectronica.gob.es](http://administracionelectronica.gob.es). Retrieved 15 May 2017, from [http://administracionelectronica.gob.es/pae\\_Home/pae\\_Documentacion/pae\\_Metodolog/pae\\_Metrica\\_v3.html#.WRmZBNwIGUI](http://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Metrica_v3.html#.WRmZBNwIGUI).