



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingeniería de
Sistemas Informáticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

615000255 - Evolucion y Mantenimiento del Software

PLAN DE ESTUDIOS

61IW - Grado En Ingeniería Del Software

CURSO ACADÉMICO Y SEMESTRE

2019/20 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	2
4. Competencias y resultados de aprendizaje.....	2
5. Descripción de la asignatura y temario.....	3
6. Cronograma.....	5
7. Actividades y criterios de evaluación.....	8
8. Recursos didácticos.....	12

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	615000255 - Evolucion y Mantenimiento del Software
No de créditos	6 ECTS
Carácter	Obligatoria
Curso	Tercero curso
Semestre	Sexto semestre
Período de impartición	Febrero-Junio
Idioma de impartición	Castellano
Titulación	61IW - Grado En Ingeniería Del Software
Centro responsable de la titulación	61 - Escuela Tecnica Superior de Ingeniería de Sistemas Informáticos
Curso académico	2019-20

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Jessica Diaz Fernandez (Coordinador/a)	1119	yesica.diaz@upm.es	Sin horario.
Juan Manuel Garitagoitia Martin	1214	juanmanuel.garitagoitia@upm.es	Sin horario.

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Conocimientos previos recomendados

3.1. Asignaturas previas que se recomienda haber cursado

- Ingeniería De Requisitos Y Modelado
- Bases De Datos
- Fundamentos De Ingeniería Del Software
- Ingeniería Del Proceso Software Y Construcción

3.2. Otros conocimientos previos recomendados para cursar la asignatura

- Conocimientos básicos de programación en .NET (ASP.NET)
- Conocimientos básicos de programación en Java

4. Competencias y resultados de aprendizaje

4.1. Competencias

CC5 - Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.

CE1 - 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 prácticos de la Ingeniería del Software.

CT5 - Organización y planificación: Identificar y definir eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.

4.2. Resultados del aprendizaje

RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos

RA69 - Utiliza entornos y herramientas de desarrollo

RA105 - Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de teorías, modelos y técnicas actuales.

RA98 - Mantiene soluciones atendiendo a la evolución del software

RA107 - Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

La asignatura tiene como objetivo global dotar al alumno de los conocimientos fundamentales, teóricos y prácticos, de los procesos de mantenimiento y evolución del software.

La asignatura tiene como objetivos específicos que el alumno sea capaz de poner en práctica patrones de mantenibilidad del software así como procesos de gestión de la configuración empleando técnicas y herramientas avanzadas para la gestión del cambio, control de versiones e integración continua. El alumno aplicará técnicas de reingeniería e ingeniería inversa para recuperar sistemas legados.

5.2. Temario de la asignatura

1. Proceso de Mantenimiento y Evolución

- 1.1. Introducción al mantenimiento y evolución del software
- 1.2. El cambio y las leyes del cambio
- 1.3. Tipos de mantenimiento
- 1.4. Modelos de proceso software. Proceso de Evolución. Estándares - ISO 14764
- 1.5. Gestión de la configuración del Software
 - 1.5.1. Gestión del cambio
 - 1.5.2. Control de Versiones
 - 1.5.3. Gestión de la compilación, despliegue y entrega
- 1.6. Gestión de la Configuración - DevOps
 - 1.6.1. Integración Continua
 - 1.6.2. Entrega Continua
 - 1.6.3. Despliegue Continuo

2. Control de Versiones

- 2.1. Gestión de la configuración y Control de Versiones
- 2.2. Tipos de Sistemas de Control de Versiones
- 2.3. Sistema de Control de Versiones Git y Github

3. Reingeniería. Sistemas legados

- 3.1. Reingeniería. Ingeniería inversa e Ingeniería directa
- 3.2. Ingeniería Inversa - enfoque práctico
- 3.3. Reingeniería de sistemas legados: Wrapping & Migración
- 3.4. Estrategias de migración

4. Mantenibilidad del Software

- 4.1. Mantenibilidad & Refactorización
- 4.2. Bad smells y Análisis de Código Estático
- 4.3. Principios de mantenibilidad
- 4.4. Deuda técnica (Technical Debt)

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad presencial en aula	Actividad presencial en laboratorio	Otra actividad presencial	Actividades de evaluación
1	Case Teoría de Presentación Duración: 02:00 LM: Actividad del tipo Lección Magistral Clase Teoría Tema 1 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
2	Clase Teoría Tema 1 Duración: 02:00 LM: Actividad del tipo Lección Magistral Trabajo 1.1. ISO14764 Duración: 02:00 PR: Actividad del tipo Clase de Problemas			
3	Trabajo 1.2. CI (Jenkins) Duración: 02:00 PR: Actividad del tipo Clase de Problemas	Trabajo 1.2. CI (Jenkins) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
4	Trabajo 1.3. CI/CD (Azure DevOps) Duración: 02:00 PR: Actividad del tipo Clase de Problemas	Trabajo 1.3. CI/CD (Azure DevOps) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
5	Clase Teoría Tema 2 Duración: 02:00 LM: Actividad del tipo Lección Magistral	Trabajo 1.4. CI/CD (Clarive) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		EC -Trabajo 1 TG: Técnica del tipo Trabajo en Grupo Evaluación continua Duración: 00:00
6	Trabajo 2. Uso de Git/GitHub Duración: 02:00 PR: Actividad del tipo Clase de Problemas	Trabajo 2. Uso de Git/Github Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
7	Clase Teoría Tema 2 Duración: 02:00 LM: Actividad del tipo Lección Magistral	Trabajo 2. Uso de Git/Github Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
8	Clase Teoría Tema 3 Duración: 02:00 LM: Actividad del tipo Lección Magistral	Laboratorio Tema 2 (Reingeniería) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		EC - Test COMÚN Git/Github EP: Técnica del tipo Examen de Prácticas Evaluación continua Duración: 01:30
9	Práctica: Reingeniería de un sistema legado Duración: 02:00 PR: Actividad del tipo Clase de Problemas	Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		

10		<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
11	<p>Clase Teoría Tema 4 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>	<p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
12	<p>Clase Teoría Tema 4 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>	<p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		<p>EC - Trabajo 3 Patrones de Mantenibilidad TG: Técnica del tipo Trabajo en Grupo Evaluación continua Duración: 00:00</p>
13	<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PR: Actividad del tipo Clase de Problemas</p>	<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
14		<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
15	<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>	<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
16	<p>Repaso - Test Duración: 02:00 PR: Actividad del tipo Clase de Problemas</p>	<p>Práctica: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		<p>EC - Práctica: Reingeniería de un sistema legado TG: Técnica del tipo Trabajo en Grupo Evaluación continua Duración: 00:00</p>
				<p>EC - Test Global Asignatura EX: Técnica del tipo Examen Escrito Evaluación continua Duración: 01:00</p> <p>EF - Test Global Asignatura EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Duración: 01:00</p> <p>EF - Test Trabajo de Git/Github EP: Técnica del tipo Examen de Prácticas Evaluación sólo prueba final Duración: 01:00</p>

17				<p>EF - Práctica: Reingeniería de un sistema legado TG: Técnica del tipo Trabajo en Grupo Evaluación sólo prueba final Duración: 00:00</p> <p>EF . Trabajo 3 Patrones de Mantenibilidad TG: Técnica del tipo Trabajo en Grupo Evaluación sólo prueba final Duración: 00:00</p> <p>EF - Test Prácticas EP: Técnica del tipo Examen de Prácticas Evaluación sólo prueba final Duración: 01:00</p>
----	--	--	--	---

Las horas de actividades formativas no presenciales son aquellas que el estudiante debe dedicar al estudio o al trabajo personal.

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

* El cronograma sigue una planificación teórica de la asignatura y puede sufrir modificaciones durante el curso.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación continua

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
5	EC -Trabajo 1	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	10%	/ 10	CC5
8	EC - Test COMÚN Git/Github	EP: Técnica del tipo Examen de Prácticas	Presencial	01:30	15%	/ 10	CC5
12	EC - Trabajo 3 Patrones de Mantenibilidad	TG: Técnica del tipo Trabajo en Grupo	No Presencial	00:00	15%	/ 10	CE1
16	EC - Práctica: Reingeniería de un sistema legado	TG: Técnica del tipo Trabajo en Grupo	No Presencial	00:00	30%	/ 10	CE1 CT5
17	EC - Test Global Asignatura	EX: Técnica del tipo Examen Escrito	Presencial	01:00	30%	4 / 10	CC5

7.1.2. Evaluación sólo prueba final

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	EF - Test Global Asignatura	EX: Técnica del tipo Examen Escrito	Presencial	01:00	30%	4 / 10	CC5
17	EF - Test Trabajo de Git/Github	EP: Técnica del tipo Examen de Prácticas	Presencial	01:00	15%	4 / 10	CC5

17	EF - Práctica: Reingeniería de un sistema legado	TG: Técnica del tipo Trabajo en Grupo	No Presencial	00:00	30%	5 / 10	CE1 CT5
17	EF . Trabajo 3 Patrones de Mantenibilidad	TG: Técnica del tipo Trabajo en Grupo	No Presencial	00:00	15%	5 / 10	CE1
17	EF - Test Prácticas	EP: Técnica del tipo Examen de Prácticas	Presencial	01:00	10%	4 / 10	CE1

7.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
CE - Test Global Asignatura	EX: Técnica del tipo Examen Escrito	Presencial	01:00	30%	4 / 10	CC5
CE - Test Trabajo de Git/Github	EP: Técnica del tipo Examen de Prácticas	Presencial	00:00	15%	4 / 10	CC5
CE - Práctica: Reingeniería de un sistema legado	PG: Técnica del tipo Presentación en Grupo	Presencial	00:00	30%	5 / 10	CC5 CE1 CT5
CE - Trabajo 3 Patrones de Mantenibilidad	PG: Técnica del tipo Presentación en Grupo	Presencial	00:00	15%	5 / 10	CE1
CE - Test Prácticas	EP: Técnica del tipo Examen de Prácticas	Presencial	01:00	10%	4 / 10	CE1

7.2. Criterios de evaluación

*La competencia transversal CT5 (Organización y Planificación) se evalúa mediante la prueba "Práctica: Reingeniería de un sistema legado" en la cual el alumno debe identificar las metas, objetivos y prioridades de un proyecto complejo a desempeñar a medio o largo plazo.

EVALUACIÓN CONTINUA

Actividad de Evaluación	Resultado de Aprendizaje
Trabajo 1 Tipos mantenimiento e ISO14764	RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales
Test Trabajo 2 Uso Git/Github	RA69 Utiliza entornos y herramientas de desarrollo
Trabajo 3 Patrones de Mantenibilidad	RA98 Mantiene soluciones atendiendo a la evolución del software
Práctica: Reingeniería de un sistema legado	RA69 Utiliza entornos y herramientas de desarrollo RA107 Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.
Test Global Asignatura	RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales

REQUISITOS:

- Obtener en el test global de la asignatura una calificación ≥ 4 .
- Desarrollar el 100% de las prácticas y trabajos de la asignatura

EVALUACIÓN SÓLO PRUEBA FINAL

Actividad de Evaluación	Resultado de Aprendizaje
Test Trabajo 2 Uso Git/Github	RA69 Utiliza entornos y herramientas de desarrollo
Trabajo 3 Patrones de Mantenibilidad Práctica: Reingeniería de un sistema legado	RA98 Mantiene soluciones atendiendo a la evolución del software RA69 Utiliza entornos y herramientas de desarrollo RA107 Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.
Test Global Asignatura Test de Prácticas	RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales

REQUISITOS:

- Obtener en el test global y prácticas de la asignatura una calificación ≥ 4 .
- Obtener en el test de Git una calificación ≥ 4
- Desarrollar el 100% de las prácticas y trabajos de la asignatura
- Obtener en la práctica y el trabajo 3 una calificación ≥ 5

El alumno que desee seguir el sistema de evaluación mediante sólo prueba final, deberá comunicarlo por escrito al coordinador de la asignatura o, por delegación de este, a los profesores de la misma mediante el procedimiento, antes abril de 2020.

CONVOCATORIA EXTRAORDINARIA

Se guarda la nota de los trabajos/prácticas **aprobados** durante evaluación continua o evaluación sólo prueba final.

REQUISITOS:

- Obtener en el test global y prácticas de la asignatura una calificación ≥ 4 .
- Obtener en el test de Git una calificación ≥ 4
- Desarrollar el 100% de las prácticas y trabajos de la asignatura
- Obtener en la práctica y el trabajo 3 una calificación ≥ 5

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Software Engineering 9 Edition	Bibliografía	Software Engineering 9 Edition., Sommerville
Software Evolution and Maintenance, a practitioner?s approach	Bibliografía	Software Evolution and Maintenance, a practitioner?s approach, Priyadarshi Tripathy, Kshirasagar Naik. Wiley 2015
Building maintainable software, C# Edition	Bibliografía	Building maintainable software, C# Edition, by Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek. O'Reilly Media 2016
Building Maintainable Software, Java Edition	Bibliografía	Building Maintainable Software, Java Edition, by Joost Visser, Sylvan Rigal, Rob van der Leek, Pascal van Eck, Gijs Wijnholds. O'Reilly Media 2016
Refactoring: Improving the Design of Existing Code	Bibliografía	Refactoring: Improving the Design of Existing Code, by Martin Fowler and Kent Beck ADDISON WESLEY 2008
Pro Git	Bibliografía	https://progit2.s3.amazonaws.com/es/2016-03-23-ea7f6/progit-es.1091.pdf

GitHub Glossary	Recursos web	https://help.github.com/articles/github-glossary/
Git Reference	Recursos web	https://git-scm.com/docs http://gitref.org/
Equipamiento aula	Equipamiento	
Equipamiento Prácticas	Equipamiento	