

Universidad de Oviedo

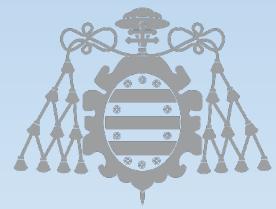
Historial Docente, Académico e Investigador

Proyecto docente e investigador

Plaza F036-570-DFA0340
Área de Lenguajes y Sistemas Informáticos

Departamento de Informática

Jose Emilio Labra Gayo



Universidad de Oviedo

Índice general

1	Preámbulo	5
I	Historial académico, docente e investigador	
2	Datos personales	9
3	Resumen	11
4	Formación académica	13
4.1	Títulos universitarios oficiales	13
5	Actividades de carácter científico-profesional	15
6	Reconocimientos	17
7	Proyectos y contratos de investigación	19
7.1	Proyectos I+D financiados en convocatorias públicas	19
7.1.1	Participación como Investigador Principal	19
7.1.2	Participación como miembro del equipo investigador	21
7.2	Contratos I+D con Empresas y/o administraciones	24
7.2.1	Participando como investigador responsable	24
7.2.2	Participando como miembro del equipo de investigación	28
8	Publicaciones y congresos	29
8.1	Artículos indexados en Journal Citation Reports	29

8.2	Otras revistas internacionales	37
8.3	Libros como autor	37
8.4	Libros como editor	38
8.5	Números especiales de revistas editados	38
8.6	Capítulos de libros	39
8.7	Publicaciones derivadas de congresos	40
8.8	Participaciones en congresos	48
8.9	Conferencias invitadas	54
8.10	Tutoriales impartidos en conferencias	55
8.11	Preprints	57
8.12	Experiencia en organización/revisión de actividades de I+D	58
8.12.1	Participación en Consejo Asesor Internacional de Libros	58
8.12.2	Revisión de libros	58
8.12.3	Actividades de revisión de artículos	59
8.12.4	Miembro del comité científico o de programa	59
8.12.5	Miembro del comité organizador	64
9	Tesis doctorales	65
9.1	Tesis doctorales dirigidas	65
9.2	Tribunales de tesis doctorales	66
10	Estancias en centros extranjeros	67
11	Actividad docente	69
11.1	Universidad de Oviedo	69
11.1.1	Asignaturas impartidas	69
11.1.2	Dirección de trabajos fin de titulación	74
11.1.3	Coordinación de asignaturas	81
11.2	Otras Universidades	82
11.3	Contribuciones de carácter docente	83
11.4	Participación en congresos orientados a la formación y a la innovación docente universitaria	83
11.5	Proyectos de innovación docente	85
11.6	Cursos y seminarios	85
11.6.1	Cursos impartidos	85
11.6.2	Cursos recibidos	89
11.7	Material docente	90
11.8	Otros méritos docentes	91
12	Actividades de Gestión	93
12.1	Ámbito académico	93
12.1.1	Órganos unipersonales	93
12.1.2	Órganos colegiados	93
12.1.3	Colaboración con Organismos de Evaluación	94

12.2	Coordinador acuerdos internacionales	94
12.3	Gestión de la investigación	95
12.4	Becas, Premios y reconocimientos	95

II

Proyecto Docente

13	Prefacio	99
14	Entorno	101
14.1	Concepto de Universidad y Espacio Europeo de Educación Superior	101
14.1.1	Créditos ECTS	103
14.1.2	Resultados de aprendizaje y sistema de calificaciones	104
14.2	La Universidad en España	104
14.2.1	Marco legislativo	105
14.2.2	ANECA	107
14.2.3	Enseñanzas de Grado	107
14.2.4	Enseñanzas de Máster	110
14.2.5	Enseñanzas de Doctorado	110
14.3	La Universidad de Oviedo	110
14.3.1	Contexto histórico	110
14.3.2	La docencia en la Universidad de Oviedo	111
14.3.3	La investigación en la Universidad de Oviedo	113
14.3.4	Escuela de Ingeniería Informática de Oviedo	115
14.3.5	Centro Internacional de Postgrado de la Universidad de Oviedo	121
14.4	Departamento y área de conocimiento	122
14.4.1	El Departamento de Informática	122
14.4.2	El área de Lenguajes y Sistemas Informáticos	124
15	Concepto y Referencias Curriculares	125
15.1	¿Qué es la Arquitectura del Software?	125
15.2	Evolución de la Arquitectura del Software	127
15.3	Recomendaciones curriculares	130
15.3.1	Informe conjunto de ACM e IEEE	131
15.3.2	Software Engineering Body of Knowledge (SWBoK)	133
15.3.3	Propuesta curricular UNESCO-IFIP	134
15.4	Otras referencias curriculares	136
15.4.1	IASA Global	136
15.4.2	iSAQB	136
15.5	Arquitectura del software en otras universidades	137
15.5.1	Arquitectura del software en universidades internacionales	137
15.5.2	Georgia Tech: Software architecture and design	138
15.5.3	Arquitectura del software en universidades nacionales	138
15.6	Relación con otras materias	141
15.6.1	Asignaturas de programación	142
15.6.2	Estructuras de datos y algoritmia	143
15.6.3	Diseño de Software	143

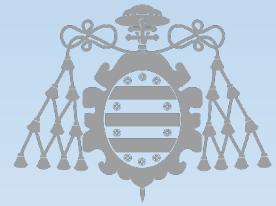
15.6.4 Ingeniería del proceso software	144
15.6.5 Software y estándares para la Web	144
15.6.6 Bases de Datos y Repositorios de información	145
15.6.7 Comunicación persona-máquina	146
15.6.8 Sistemas distribuidos e Internet	146
15.6.9 Seguridad de Sistemas Informáticos	146
15.6.10 Diseño de lenguajes de programación	147
15.6.11 Ingeniería de requisitos	147
15.6.12 Dirección y planificación de proyectos informáticos	148
15.6.13 Calidad, Validación y verificación del software	148
15.6.14 Aspectos Sociales, Legales, Éticos y Profesionales de la Informática	149
15.7 Conclusiones de la revisión de referencias curriculares	149
16 Metodología docente	151
16.1 Introducción	151
16.2 Los estudiantes	151
16.3 Modalidades de enseñanza-aprendizaje	152
16.3.1 Clases expositivas	153
16.3.2 Enseñanza por Descubrimiento	155
16.3.3 Aprendizaje basado en problemas/proyectos	156
16.3.4 Prácticas de aula y Seminarios	156
16.3.5 Clases de prácticas de laboratorio	157
16.3.6 Tutorías	157
16.3.7 Actividades complementarias	158
16.4 Evaluación de la enseñanza	160
16.4.1 Técnicas de evaluación	160
16.4.2 Evaluación del Docente	163
17 Propuesta docente	165
17.1 Competencias y Resultados de aprendizaje	165
17.1.1 Competencias	165
17.1.2 Resultados de aprendizaje	166
17.2 Contenidos	167
17.2.1 Unidad 1. Definiciones de Arquitectura del Software	168
17.2.2 Unidad 2. Comunicando la Arquitectura del Software	168
17.2.3 Unidad 3. Papel del arquitecto del software y Stakeholders	169
17.2.4 Unidad 4. Atributos de calidad	169
17.2.5 Unidad 5. Creación y validación de arquitecturas	170
17.2.6 Unidad 6. Técnicas de Arquitectura del Software	170
17.2.7 Unidad 7. Taxonomías: Construcción, mantenimiento y gestión de configuraciones	
171	
17.2.8 Unidad 8. Taxonomías: Modularidad	171
17.2.9 Unidad 9. Taxonomías: Comportamiento. Sistemas básicos y monolitos	173
17.2.10 Unidad 10. Taxonomías: Comportamiento. Sistemas distribuidos y big data ..	174
17.2.11 Unidad 11. Taxonomías: Disposición	175
17.2.12 Unidad 12. Arquitectura del software y Entorno empresarial	176

17.3 Actividades de aprendizaje	177
17.3.1 Clases expositivas	178
17.3.2 Prácticas de aula/seminario	181
17.3.3 Prácticas de laboratorio	183
17.3.4 Tutorías grupales	185
17.4 Evaluación de la asignatura	185
17.4.1 Evaluación de teoría	185
17.4.2 Evaluación de práctica	185
17.4.3 Evaluaciones alternativas	187
17.4.4 Resumen de la evaluación	187
17.4.5 Premios y concursos	188
17.5 Justificación y posibles variaciones	188
17.6 Bibliografía básica comentada	192
17.7 Anexo: Guía docente	202

III

Proyecto Investigador

18 Contexto del proyecto	213
18.1 Introducción	213
18.2 Marco de la investigación	213
18.2.1 Actividades y cooperación con otros grupos de investigación e instituciones	215
19 Proyecto ANGLIRU – PID2020-117912RB-C21	217
19.1 Memoria del proyecto	217
19.2 Resolución	253
19.3 Informe de Valoración científico-técnica	259
20 Referencias bibliográficas	271
Índice	277



Universidad de Oviedo

1. Preámbulo

De conformidad con el artículo 62 de la Ley Orgánica 6/2001, de 21 de diciembre, de Universidades, parcialmente modificada por la Ley Orgánica 4/2007, de 12 de abril, y por el Decreto-ley 10/2015, de 11 de septiembre; el artículo 3 del Real Decreto 1313/2007, de 5 de octubre, por el que se regula el régimen de los concursos de acceso a cuerpos docentes universitarios; el artículo 151 de los Estatutos de la Universidad de Oviedo, aprobados por Decreto 12/2010, de 3 de febrero, del Principado de Asturias (BOPA de 11 de febrero), y el Reglamento para los concursos de acceso a los cuerpos de funcionarios docentes universitarios de la Universidad de Oviedo, aprobado por el Consejo de Gobierno de 18 de diciembre de 2008 (BOPA de 14 de enero de 2009), modificado por Acuerdo del Consejo de Gobierno de 20 de octubre de 2020 (BOPA de 18 de noviembre) se presenta el presente documento.

En cumplimiento de lo dispuesto en la Resolución de 25 de Junio de 2021 de la Universidad de Oviedo, publicada en el BOE de 9 de Julio de 2021, por la que se convoca a concurso la plaza 16, código F036-570-DFA0340 del Cuerpo de Catedrático de Universidad/, para el área de conocimiento Lenguajes y Sistemas Informáticos, se presenta esta memoria que contiene el historial académico, investigador y de docencia, así como los proyectos docente e investigador del candidato.

El perfil del proyecto docente propuesto es [Arquitectura del Software](#). La asignatura de Arquitectura del Software es una asignatura de carácter obligatorio que se imparte en tercer curso de la Escuela de Ingeniería Informática de la Universidad de Oviedo.

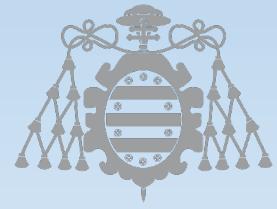
El resto del documento está organizado en tres partes. La primera de ellas desarrolla el historial académico, investigador y de docencia del candidato. La segunda el proyecto docente y la tercera el proyecto de investigación.

Oviedo,
Septiembre de 2021

Jose Emilio Labra Gayo

Historial académico, docente e investigador

2	Datos personales	9
3	Resumen	11
4	Formación académica	13
5	Actividades de carácter científico-profesional	15
6	Reconocimientos	17
7	Proyectos y contratos de investigación	19
8	Publicaciones y congresos	29
9	Tesis doctorales	65
10	Estancias en centros extranjeros	67
11	Actividad docente	69
12	Actividades de Gestión	93



Universidad de Oviedo

2. Datos personales

Apellidos y nombre: Labra Gayo, Jose Emilio

NIF: 09394072K.

Nacimiento: 09/06/1969, Oviedo, Asturias

Residencia: Asturias, Avilés, Avenida Los Telares

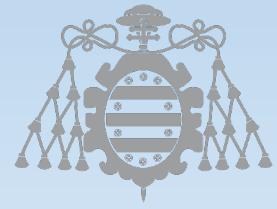
Correo electrónico y teléfono: labra@uniovi.es, 985103394

Categoría actual como docente: Profesor Titular de Universidad

Departamento o unidad docente actual: Departamento de Informática

Área de conocimiento actual: Lenguajes y Sistemas Informáticos

Facultad o escuela actual: Escuela de Ingeniería Informática, Universidad de Oviedo



Universidad de Oviedo

3. Resumen

Jose Emilio Labra Gayo es Doctor Ingeniero en Informática desde el año 2001 y profesor Titular de Universidad en la Universidad desde el año 2003. Acreditado a Catedrático de Universidad desde el año 2020. Actualmente dispone de 3 sexenios de investigación concedidos.

En el año 2004 fundó el grupo oficial de investigación WESO (Web Semantics Oviedo) del cual es el Investigador Principal. El grupo WESO ha sido evaluado positivamente por la ANEP en el año 2012 y 2018.

Investigación:

- Según scopus (ORCID: 0000-0001-8907-5348): h-index: 13. Total documentos indexados: 95 y citas: 628 de 527 documentos (fecha consulta: 15 octubre 2021).
- Según Google scholar : h-index: 21, i10-index: 46, 1859 citas.
- 35 artículos en revistas JCR (16 Q1, 3 Q2, 11 Q3, 5 Q4)
- 4 Artículos en revistas no JCR
- Libros:
 - 3 libros como autor
 - 4 libros como editor
 - 11 capítulos de libros
- Congresos: 21 publicaciones derivadas de congresos y 66 contribuciones en congreso.
- Investigador Principal en 3 proyectos del Plan Nacional de Investigación (ANGLIRU, VARSSHA y ROCAS)

Transferencia de conocimiento:

- Fundador e Investigador Principal del grupo WESO (Web Semantics Oviedo)
- Investigador Principal en 8 proyectos financiados en convocatorias públicas
- Investigador Responsable en 23 contratos con empresa a través de la Fundación Universidad de Oviedo.
- Miembro del equipo investigador en 16 proyectos financiados en convocatorias públicas.
- Director de la cátedra de empresa DXC Technology desde el año 2018
- Participación en red temática europea Multilingual Web

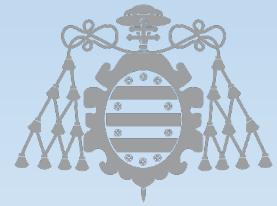
- Miembro de dos grupos de trabajo del World Wide Web Consortium (W3C): Data Shapes (2014-2017) y Web of Things (2017 – 2020).
- Miembro de W3C community groups: Shape Expressions (ShEx) , SHACL (Shapes Constraint Language)
- Chairman del W3C Community Group on Best Practices for Multilingual Linked Open Data
- Editor especificación semántica del lenguaje ShEx y del Test-suite and implementation report de SHACL

Docencia

- Ha sido coordinador de las asignaturas [Arquitectura del Software](#), [Web Semántica](#), [Nuevos avances en Web Semántica](#), [Lógica](#), [Programación Lógica y Funcional](#) y [Programación declarativa](#)
- Ha impartido docencia en los niveles de Ingeniería Técnica, Grado, Ingeniería, Máster y doctorado.
- Ha dirigido 4 tesis doctorales, 7 trabajos de investigación, 29 trabajos fin de máster en el máster de ingeniería web, 17 trabajos fin de grado en el grado de ingeniería informática del software, 18 proyectos fin de carrera en la titulación de ingeniería informática, 59 proyectos fin carrera en la titulación de ingeniería técnica informática de sistemas y 26 proyectos fin de carrera en la titulación de ingeniería técnica informática de gestión.

Gestión

- Director de la Escuela de Ingeniería Informática entre los años 2004 y 2012, (8 años)
- Secretario y Jefe de Estudios de dicha Escuela entre los años 1996 a 2000.
- Director Cátedra DXC Technology, desde Julio de 2018
- Miembro del Consejo de Gobierno de la Universidad de Oviedo, de 2004 a 2018
- Miembro del Claustro de la Universidad de Oviedo, de 2002 a 2008, y de 2017 a la actualidad.

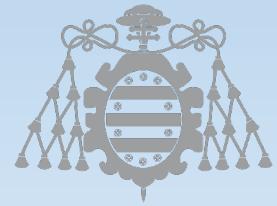


Universidad de Oviedo

4. Formación académica

4.1 Títulos universitarios oficiales

- Ingeniero Técnico en Informática de Sistemas
Escuela Universitaria de Ingeniería Técnica Informática de Oviedo, Febrero 1992
- Ingeniero en Informática
ETS Ingenieros Industriales e Informáticos. Noviembre, 1994.
- Doctor Ingeniero en Informática
Departamento de Informática de la Universidad de Oviedo. Julio, 2001.



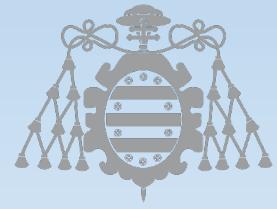
Universidad de Oviedo

5. Actividades de carácter científico-profesional

Categoría	Organismo o centro	Dedicación	Nombramiento	Cese/Fin
Profesor Titular de Universidad	Universidad de Oviedo	TC*	05/01/2004	_____
Profesor Titular de Escuela Universitaria	Universidad de Oviedo	TC*	25/07/1999	04/01/2004
Profesor Asociado tipo II	Universidad de Oviedo	TC*	01/12/1994	24/07/1999
Profesor Asociado tipo I	Universidad de Oviedo	TC*	01/10/1992	30/11/1994
Becario	Departamento de Matemáticas, Universidad de Oviedo	TP*	Oct. 1990	Feb. 1992
A. Programador	Seresco Asturiana S.A.	TC*	Junio 1990	Oct. 1990
Becario	Departamento de Matemáticas, Universidad de Oviedo	TP*	Nov. 1989	Jun. 1990

TC*: Tiempo completo

TP*: Tiempo Parcial



Universidad de Oviedo

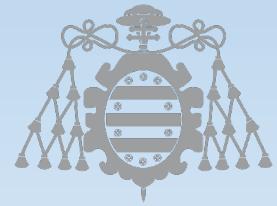
6. Reconocimientos

Sexenios de investigación

Categoría	Tramo	Desde	Hasta
Profesor Titular Escuela Universitaria	1	-	01/01/2002
Profesor Titular de Universidad	2	01/01/2003	31/12/2013
Profesor Titular de Universidad	3	01/01/2014	31/12/2019

Quinquenios docentes

Categoría	Tramo	Desde	Hasta
Profesor Titular Escuela Universitaria	1	01/10/1992	30/09/1997
Profesor Titular Escuela Universitaria	2	01/10/1997	30/09/2002
Profesor Titular de Universidad	3	01/10/2002	30/09/2007
Profesor Titular de Universidad	4	01/10/2007	30/09/2012
Profesor Titular de Universidad	5	01/10/2012	30/09/2017



Universidad de Oviedo

7. Proyectos y contratos de investigación

7.1 Proyectos I+D financiados en convocatorias públicas

7.1.1 Participación como Investigador Principal

9. **Título:** ANGLIRU: Applying kNowledge Graphs to research data interoperability and ReUsability

Referencia: PID2020-117912RB

Entidad financiadora: Proyecto coordinado - Programa Estatal de Generación de Conocimiento y Fortalecimiento Científico y Tecnológico del Sistema de I+D+i y del Programa Estatal de I+D+i Orientada a los Retos de la Sociedad, en el marco del Plan Estatal de Investigación Científica y Técnica y de Innovación 2017-2020

Entidades participantes: Grupo WESO, Universidad de Oviedo, Izertis

Duración: 2021-2023

Participación: Investigador Principal del proyecto 1 y Coordinador

Cuantía de la subvención: 65.703€ (subproyecto1) + 77.924€ (subproyecto2)

8. **Título:** Arquitectura semántica e infraestructura ontológica. Fase II. Desarrollo de la solución

Referencia: 2018/88/OT-AM

Entidad financiadora: Proyecto cofinanciado por la Universidad de Murcia a través del Fondo Europeo de Desarrollo Regional (FEDER) periodo de programación 2014-2020.

Entidades participantes: Grupo WESO, Universidad de Oviedo, Izertis

Duración: 2019-2021

Participación: Director del proyecto e Investigador Principal

Cuantía de la subvención: 701800€

7. **Título:** Arquitectura semántica e infraestructura ontológica. Fase I. Estudio de viabilidad

Referencia: 2018/88/OT-AM

Entidad financiadora: Proyecto cofinanciado por la Universidad de Murcia a través del Fondo Europeo de Desarrollo Regional (FEDER) periodo de programación 2014-2020.

Entidades participantes: Grupo WESO, Universidad de Oviedo, Izertis

Duración: 2019

Participación: Director del proyecto e Investigador Principal

Cuantía de la subvención: 7623€

6. **Título:** [VARSSHA: Validating RDF Streams using Shapes](#)

Referencia: TIN2017-88877-R

Entidad financiadora: Ministerio de Economía, Industria y Competitividad. Proyectos I+D+i (RETOS), convocatoria 2017

Entidades participantes: Grupo WESO, Universidad de Oviedo

Duración: 2017–2020

Participación: Investigador Principal

Cuantía de la subvención: 27709€

5. **Título:** [ROCAS: Razonamiento en la nube aplicando semántica](#)

Referencia: TIN2011-27871

Entidad financiadora: Ministerio de Ciencia e Innovación. Programas de Investigación fundamental no orientada. Área temática: Tecnologías informáticas

Entidades participantes: Grupo WESO, Universidad de Oviedo

Duración: 2012–2014

Participación: Investigador Principal

Cuantía de la subvención: 44000€

4. **Título:** [TagFlow, sistema Avanzado de monitoreo y tagging de Flujos de datos multimedia](#)

Referencia: TSI-020602-2012-195

Entidad financiadora: Ministerio de Industria, Turismo y Comercio, Subprograma Avanza I+D

Entidades participantes: BMAT Licensing, S. L., Universidad de Oviedo (WESO)

Duración: 2012–2014

Participación: Investigador Principal por la parte contratada a la Universidad del Proyecto

Coordinador del proyecto: BMAT Licensing S.L.

Cuantía de la subvención: 68799€

3. **Título:** [RETINAS \(REal TIme video ANalysis for Security applications\)](#)

Entidad financiadora: Plan de Ciencia Tecnología e Innovación del Principado de Asturias 2006 - 2009

Entidades participantes: Universidad Autónoma de Madrid, European Centre for Soft Computing, Universidad de Oviedo, Treelogic

Duración: 2011–2013

Participación: Investigador Principal por la parte contratada a la Universidad del Proyecto

Coordinador del proyecto: Treelogic

Cuantía de la subvención: 27000€

2. **Título:** [ORIGIN: ORganizaciones Inteligentes Globales INnovadoras](#)

Entidad financiadora: Centro para el Desarrollo Tecnológico e Industrial.

Entidades participantes: Indra Software Labs, Treelogic, Sicaman Nuevas Tecnologías, Extreme

Duración: 2010–2012

Participación: Investigador Principal por la parte contratada a la Universidad del Proyecto

Coordinador del proyecto: Indra Software Labs

Cuantía de la subvención: 62250€

1. **Título:** Prototipo de plataforma pan-europea de servicios de información para PYMES en licitación pública internacional (10DERS INFORMATION SERVICES)
Referencia: TSI-020100-2010-919
Entidad financiadora: Ministerio de Industria, Turismo y Comercio, Subprograma Avanza I+D
Entidades participantes: Gateway Strategic Consultancy Services S.L, EXIS-TI y Universidad de Oviedo
Duración: 2010–2011
Participación: Investigador Principal por la parte contratada a la Universidad del Proyecto
Coordinador del proyecto: Gateway Strategic Consultancy Services S. L.
Cuantía de la subvención: 62250€

7.1.2 Participación como miembro del equipo investigador

16. **Título:** Red temática sobre desarrollo urbano
Referencia: MINECO-15-DER2015-71345-RDT
Entidad financiadora: Ministerio de Economía y Competitividad
Entidades participantes: Universidad de Oviedo, Universidad Internacional Menéndez y Pelayo, Universitat de Girona, Universidad de Zaragoza, Universidad Complutense de Madrid, Universidad Carlos III, Universidad de Málaga, Universidad de las Islas Baleares
Duración: 2015–2017
Participación: Investigador
Investigador principal: María Rosario Alonso Ibáñez
15. **Título:** Rednova: Redes en la nube para la coproducción de valor
Referencia: MICINN-11-IPT-2011-1907-430000
Entidad financiadora: Ministerio de Ciencia e Innovación. Programas de Investigación fundamental no orientada. Área temática: Tecnologías informáticas
Entidades participantes: Universidad de Oviedo
Duración: 2011–2014
Participación: Investigador
Investigador principal: Juan Manuel Cueva Lovelle
14. **Título:** Advancing the Multilingual Web
Referencia: UE-10-MULTILINGUALWEB-250500
Entidad financiadora: Unión Europea
Entidades participantes: W3C, Aalto-Korkeakoulusaatio, Bioloom Group, Consiglio Nazionale delle Ricerche, European Commission - Directorate-General for Translation, Facebook Ireland, Fachhochschule Potsdam, Institut Jozef Stefan, Institutul de Cercetari Pentru Inteligenta Artificiala, Language Technology Centre Ltd., Lionbridge Belgium, Microsoft Ireland, Opera Software, SAP AG, TAUS, Universidad de Oviedo, Universidad Politécnica de Madrid, University of Limerick, University of Economics - Praga, Transware Ltd., XTM-international
Duración: 2010–2012
Participación: Investigador
Investigador principal: Richard Ishida (coordinador), Cristina Valdés Rodríguez por parte de la Universidad de Oviedo
13. **Título:** Plataforma ecodigital Episteme: ecosistema digital de soporte al futuro ecosistema de innovación de las PYME españolas de la cadena de valor del sector de servicios de negocio intensivos en tecnología y en conocimientos

- Referencia:** MITC-11-TSI-020302-2011-20
Entidad financiadora: Ministerio de Industria
Entidades participantes: Universidad de Oviedo
Duración: 2010–2012
Participación: Investigador
Investigador principal: Juan Manuel Cueva Lovelle
12. **Título:** [Red Temática española de Linked Data](#)
Referencia: TIN2010-10811-E
Entidad financiadora: Ministerio de Ciencia e Innovación
Entidades participantes: Universidad Politécnica de Madrid, Euskal Herriko Unibertsitatea, Universidad Rey Juan Carlos, Universidad de Deusto, Universidad Complutense de Madrid, Universidad Nacional de Educación a Distancia, Universitat de Lleida, Universidad Europea de Madrid, Universidad San Pablo CEU, Universidad de Zaragoza, Universidad de Santiago de Compostela, Universidad de Murcia, Universidad de Málaga, Universidad Carlos III de Madrid, Universidad de Extremadura, Universidad de Sevilla, Fundación CTIC, Universidad de Oviedo.
Duración: 2011
Participación: Investigador
Investigador principal: Oscar Corcho García
Cuantía de la subvención: 22000€
11. **Título:** [Muvas - Museo Virtual Asturias](#)
Referencia: TSI-070300-2008-28
Entidad financiadora: Fondos FEDER, Ministerio de Industria, Turismo y Comercio, Subprograma Avanza I+D
Entidades participantes: Universidad de Oviedo.
Duración: 2008-2010
Participación: Investigador
Participantes: 17
Investigador principal: Aquilino Adolfo Juan Fuente
Cuantía de la subvención: 430000€
10. **Título:** [Red Temática de Web Semántica](#)
Referencia: TSI2006-26928-E
Entidad financiadora: Ministerio de Educación y Ciencia
Entidades participantes: 27 grupos de investigación de toda España
Duración: 2008-2010
Participación: Investigador
Investigador principal: Asunción Gómez Pérez
Cuantía de la subvención: 15000€
9. **Título:** [Código con Demostración Asociada](#)
Referencia: TIN2004-07943-C04-04
Entidad financiadora: Ministerio de Educación y Ciencia
Entidades participantes: Universidad Complutense de Madrid, Universidad de Oviedo
Duración: 2004-2007
Participación: Investigador
Investigador principal: Ricardo Peña Marí
Cuantía de la subvención: 34500€
8. **Título:** [Plataforma reflectiva para la separación dinámica de aspectos](#)

Referencia: TIN2004-03453

Entidad financiadora: Ministerio de Educación y Ciencia

Entidades participantes: Universidad de Oviedo

Duración: 2004-2006

Participación: Investigador

Investigador principal: Juan Manuel Cueva Lovelle

Cuantía de la subvención: 51000€

7. **Título:** Aplicación de Tecnologías de Análisis Estático a la Calidad del Software

Entidad financiadora: FICYT (Fundación para el Fomento en Asturias de la Investigación Científica Aplicada y la Tecnología)

Entidades participantes: Universidad de Oviedo, Seresco Asturiana, S. A.

Duración: 2004-2007

Participación: Investigador

Investigador principal: Agustín Cernuda del Río

Cuantía de la subvención: 15678,16€

6. **Título:** Diseño e Implementación de un servicio Web para el análisis e interpretación de la información financiera en formato

Referencia: XBRL - FC-04-PC-75

Entidad financiadora: FICYT (Fundación para el Fomento en Asturias de la Investigación Científica Aplicada y la Tecnología)

Entidades participantes: Universidad de Oviedo

Duración: 2004-2007

Participación: Investigador

Investigador principal: Javier de Andrés Suárez

Cuantía de la subvención: 18816€

5. **Título:** Interoperabilidad de Objetos y Componentes mediante el sistema Integral Orientado a Objetos Oviedo3

Entidad financiadora: Universidad de Oviedo

Entidades participantes: Universidad de Oviedo

Duración: 1999

Participación: Investigador

Participantes: 15

Investigador principal: Juan Manuel Cueva Lovelle

Cuantía de la subvención: 500000ptas

4. **Título:** Proyecto L.E.I.A (dentro del proyecto ESPRIT EP24231 titulado JEDI (Java Enabled Database Access over Internet))

Referencia: ESPRIT EP 24231

Entidad financiadora: Unión Europea

Entidades participantes: Universidad de Oviedo

Duración: 1997–1999

Participación: Investigador

Participantes: 15

Investigador principal: Juan Manuel Cueva Lovelle

Cuantía de la subvención: 4640000ptas

3. **Título:** Sistema Intergral Orientado a Objetos: Oviedo3

Referencia: PBP-TIC97-01

Entidad financiadora: FICYT (Fundación para el Fomento en Asturias de la Investigación Científica Aplicada y la Tecnología)

Entidades participantes: Universidad de Oviedo

Duración: 1996-1997

Participación: Investigador

Investigador principal: Juan Manuel Cueva Lovelle

Cuantía de la subvención: 8300000ptas

2. **Título:** Variables Aleatorias Difusas y sus Aplicaciones al Establecimiento de Modelos de Uso en Economía, Ingeniería y otras Ciencias Experimentales

Referencia: DGES PB95-1049

Entidad financiadora: Dirección General de Investigación Científica y Técnica (DGICYT)

Entidades participantes: Universidad de Oviedo

Duración: 1996-1998

Participación: Investigador

Investigador principal: María Ángeles Gil Álvarez

1. **Título:** Modelización de un Sistema de Transporte de Gas mediante Algoritmos de Control Numérico basados en Redes Neuronales y Lógica Fuzzy

Referencia: PA-CYR96-02

Entidad financiadora: Principado de Asturias - FICYT

Entidades participantes: Universidad de Oviedo

Duración: 1996-1997

Participación: Investigador

Investigador principal: Luciano Sánchez Ramos

7.2 Contratos I+D con Empresas y/o administraciones

7.2.1 Participando como investigador responsable

23. **Título:** Robustifying Scholia: paving the way for knowledge discovery and research assessment through Wikidata - University of Oviedo

Referencia: FUO-21-234

Entidad financiadora: University of Virginia - Sloan Foundation

Entidades participantes: Universidad de Oviedo

Duración: 05/2021–09/2021

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 50.000\$

22. **Título:** Wikibase-SAF realisation phase in Luxembourg

Referencia: FUO-20-292

Entidad financiadora: Micelio BV – Gobierno de Luxemburgo

Entidades participantes: Universidad de Oviedo

Duración: 10/2020–12/2021

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 19375€

21. **Título:** POC: Integración de Solid y Symmetry

Referencia: FUO-301-19

Entidad financiadora: Izertis

Entidades participantes: Universidad de Oviedo

Duración: 12/2019–03/2020

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos, Daniel Fernández Álvarez, Herminio García González

Cuantía de la subvención: 6000€

20. **Título:** Generación de preguntas y respuestas a partir de grafos de conocimiento

Referencia: FUO-331-19

Entidad financiadora: Zapiens

Entidades participantes: Universidad de Oviedo

Duración: 06/2019–01/2020

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos

Cuantía de la subvención: 15000€

19. **Título:** Prototipo de buscador basado en principios Solid

Referencia: FUO-301-19

Entidad financiadora: Empathy

Entidades participantes: Universidad de Oviedo

Duración: 07/2019–07/2020

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos

Cuantía de la subvención: 6000€

18. **Título:** ICD Analytics for Symmetry

Referencia: FUO-345-18

Entidad financiadora: Izertis

Entidades participantes: Izertis, Universidad de Oviedo

Duración: 17/2018–07/2019

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos

Cuantía de la subvención: 9302€

17. **Título:** AMRDF: Reconocimiento y procesamiento semántico de conocimiento a partir de lenguaje natural

Referencia: FUO-324-18

Entidad financiadora: Zapiens Technologies S. L.

Entidades participantes: Zapiens, Universidad de Oviedo

Duración: 10/2018–12/2018

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello

Cuantía de la subvención: 7000€

16. **Título:** Diseño, Elaboración y Desarrollo de un videojuego sobre Pedro Menéndez de Avilés

Referencia: FUO-18-388

Entidad financiadora: Ayuntamiento de Avilés

Entidades participantes: Universidad de Oviedo

Duración: 12/2018–08/2019

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Jordán Pascual Espada

Cuantía de la subvención: 2545,43€

15. **Título:** SNOMED Analytics for WeDoctor

Referencia: FUO-373-17

Entidad financiadora: Izertis

Entidades participantes: Universidad de Oviedo

Duración: 11/2017–01/2018

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos

Cuantía de la subvención: 5977,12€

14. **Título:** POC SNOMED Sobre Plataforma WeDoctor

Referencia: FUO-218-17

Entidad financiadora: Izertis

Entidades participantes: Universidad de Oviedo

Duración: 05/2017–08/2017

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo, Daniel Gayo Avello, Patricia Ordóñez de Pablos

Cuantía de la subvención: 11988,52€

13. **Título:** SIGINFO: Información para la venta de activos

Referencia: FUO-239-16

Entidad financiadora: Izertis

Entidades participantes: Universidad de Oviedo

Duración: 05/2016–06/2017

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo

Cuantía de la subvención: 25000€

12. **Título:** Tratamiento de datos para A4AI-Alliance for Affordable Internet

Referencia: FUO-051-16

Entidad financiadora: The Web Foundation

Entidades participantes: The World Wide Web Foundation, Universidad de Oviedo

Duración: 2016

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo

Cuantía de la subvención: 400€

11. **Título:** LODLearning: enriqueciendo la experiencia formativa en eLearning

Entidad financiadora: Izertis

Entidades participantes: Izertis, Universidad de Oviedo

Duración: 04/2015–09/2015

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo

Cuantía de la subvención: 8621,67€

10. **Título:** Structure and information architecture of the new website

Referencia: FUO-EM-057-15

Entidad financiadora: The Web Foundation

Entidades participantes: The World Wide Web Foundation, Universidad de Oviedo

Duración: 01/2015–03/2015

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo

Cuantía de la subvención: 11985€

9. Título: [Rebuilding IFAD's land portal](#)

Entidad financiadora: IFAD (International Fund for Agricultural Development) - United Nations

Entidades participantes: SBC Consulting, Grupo WESO - Universidad de Oviedo

Duración: 2013–2014

Investigador principal: Jose Emilio Labra Gayo

Investigadores: Jose Emilio Labra Gayo

Cuantía de la subvención: 28272€

8. Título: [The Web Index - Base, LOD portal for World Wide Web Foundation](#)

Entidad financiadora: The Web Foundation

Entidades participantes: The Web Foundation, Grupo WESO - Universidad de Oviedo

Duración: 2012–2013

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 39864€

7. Título: [Investigación y desarrollo sobre aplicaciones de Web Semántica para los proyectos BCN: Historia de la Ley y Labor Parlamentaria](#)

Entidad financiadora: Biblioteca Nacional del Congreso de Chile

Entidades participantes: Grupo WESO - Universidad de Oviedo

Duración: 2012

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 19200€

6. Título: [Creación plataforma de redes sociales profesionales](#)

Entidad financiadora: R&R Professional Clinical Research

Entidades participantes: Grupo WESO - Universidad de Oviedo

Duración: 2012

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 773€

5. Título: [Curso HTML5/CSS3](#)

Entidad financiadora: Dicampus

Entidades participantes: Fundación Universidad de Oviedo

Duración: 2012

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 1150€

4. Título: [Plataforma web de producción tecnológica de contenidos informativos y entretenimiento](#)

Entidad financiadora: Freews Inc.

Entidades participantes: Fundación Universidad de Oviedo

Duración: 03/2012–09/2012

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 2645€

3. Título: [Integración Web Semántica](#)

Entidad financiadora: Fundación CTIC

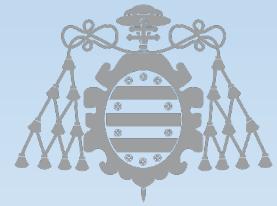
Entidades participantes: Fundación Universidad de Oviedo

Duración: 2005–2006

Investigador principal: Jose Emilio Labra Gayo

Cuantía de la subvención: 4060€

2. Título: Colaboración en el desarrollo de la taxonomía COREP**Referencia:** FUO-EM-014-05**Entidad financiadora:** Asociación XBRL España**Entidades participantes:** Fundación Universidad de Oviedo**Duración:** 02/2005–05/2005**Investigador principal:** Jose Emilio Labra Gayo**Cuantía de la subvención:** 7000€**1. Título:** Desarrollo de un Modelo de Servicios de Información para la Administración basado en la Web Semántica**Entidad financiadora:** Fundación CTIC**Entidades participantes:** Fundación Universidad de Oviedo**Duración:** 2004–2005**Investigador principal:** Jose Emilio Labra Gayo**Cuantía de la subvención:** 4000€**7.2.2 Participando como miembro del equipo de investigación****1. Título:** Extensión del estandar CSS3 que permita la adaptación multidpositivo de contenidos web**Referencia:** FUO-EM-115-05**Entidad financiadora:** Fundación CTIC**Entidades participantes:** Fundación Universidad de Oviedo**Duración:** 06/2005–06/2005**Investigador principal:** Juan Manuel Cueva Lovelle**Cuantía de la subvención:** 4000€



Universidad de Oviedo

8. Publicaciones y congresos

8.1 Artículos indexados en Journal Citation Reports

35. **Autor(es):** H. Turki, M. Ali Hadj Taieb, T. Shafee, T. Lubiana, D. Jemielniak, M. Ben Aouicha, **J. E. Labra Gayo**, E. A. Youngstrom, M. Banat, D. Das, D. Mietchen
Título: [Representing COVID-19 information in collaborative knowledge graphs: the case of Wikidata](#)
Clave: Artículo
Nombre revista: Semantic Web Journal
Fecha: 2021
Factor de impacto: 2.214 (Q1 en SCIE – Computer Science, Theory and Methods)
34. **Autor(es):** D. Fernández-Álvarez ,J. Frey, **J. E. Labra Gayo**, D. Gayo-Avello, S. Hellmann
Título: [Approaches to measure class importance in Knowledge Graphs](#)
Clave: Artículo
Nombre revista: PLOS One
Fecha: 2021
doi: <https://doi.org/10.1371/journal.pone.0252862>
Factor de impacto: 6.765 (Q1 en SCIE)
33. **Autor(es):** A. Waagmeester, E. L. Willighagen, A. I. Su, M. Kutmon, **J. E. Labra Gayo**, D. Fernández-Álvarez, Q. Groom, P. J. Schaap, L. M. Verhagen, J. J. Koehorst
Título: [A protocol for adding knowledge to Wikidata: aligning resources on human coronaviruses](#)
Clave: Artículo
Nombre revista: BMC Biology
Volumen: 19 (12)
Fecha: 2021
doi: <https://doi.org/10.1186/s12915-020-00940-y>
Factor de impacto: 6.765 (Q1 en SCIE)

32. **Autor(es):** A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutiérrez, **J. E. Labra-Gayo**, S. Kirrane, S. Neumaier, A. Polleres, R. Navigli, A. Ngonga, S. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann
Título: Knowledge graphs
Clave: Artículo
Nombre revista: ACM Computing Surveys
Volumen: 54
Número: 4
Páginas: 1–37
Fecha: 2021
doi: <https://doi.org/10.1145/3447772>
ISSN: 0360-0300
Factor de impacto: 7.990 (Q1 en SCIE)
31. **Autor(es):** S. Ruiz-Saavedra; H. García; S. Arboleya.; N. Salazar; **J. E. Labra-Gayo**; I. Díaz.; M. Gueimonde; S. González, S. de los Reyes-Gavilán
Título: Intestinal microbiota alterations by dietary exposure to chemicals from food cooking and processing. Application of Data Science for risk prediction
Clave: Artículo
Nombre revista: Computational and Structural Biotechnology Journal
Volumen: 19
Páginas: 1081-1091
Fecha: 2021
doi: <https://doi.org/10.1016/j.csbj.2021.01.037>
Factor de impacto: 6.018 (Q1 en SCIE)
30. **Autor(es):** H. García G., I. Boneva, S. Staworko, **J. E. Labra-Gayo**, J. M. Cuevas L.
Título: ShExML: improving the usability of heterogeneous data mapping languages for first-time users
Clave: Artículo
Nombre revista: PeerJ Computer Science
Fecha: 2020
doi: <https://doi.org/10.7717/peerj-cs.318>
Factor de impacto: 3.099 (Q2 en SCIE)
29. **Autor(es):** H. García González, **J. E. Labra Gayo**
Título: XMLSchema2ShEx: Converting XML validation to RDF validation
Clave: Artículo
Nombre revista: Semantic Web Journal
Volumen: 11 (2)
Páginas: 235-253
Fecha: 2020
doi: <https://doi.org/10.3233/SW-180329>
Factor de impacto: 2.214 (Q1 en SCIE – Computer Science, Theory and Methods)
28. **Autor(es):** J. A. Benítez-Andrade, I. García-Rodríguez, C. Benavides, H. Alaiz-Moretón, **J. E. Labra-Gayo**
Título: An Ontology based multi-domain model in social network analysis: experimental validation and case study

Clave: Artículo

Nombre revista: Information Sciences

Volumen: 540

Páginas: 390-413

Fecha: 2020

doi: <https://doi.org/10.1016/j.ins.2020.06.008>

Factor de impacto: 5.524 (Q1 en SCIE)

27. **Autor(es):** F. Cifuentes-Silva, D. Fernández-Álvarez, **J. E. Labra Gayo**

Título: National Budget as Linked Open Data: New Tools for Supporting the Sustainability of Public Finances

Clave: Artículo

Nombre revista: Sustainability

Volumen: 12 (11)

Fecha: 2020

doi: <https://doi.org/10.3390/su12114551>

Factor de impacto: 2.592 (Q1 en SCIE)

26. **Autor(es):** P. Ordóñez-de-Pablos, **J. E. Labra Gayo**

Título: Rethinking key issues for understanding the new challenges of disruption and digital transformation in companies and economies

Clave: Artículo

Nombre revista: Journal of Behaviour and Information Technology

Volumen: 38 (9)

Páginas: 873-875

Fecha: 2019

doi: <https://doi.org/10.1080/0144929X.2019.1641663>

Factor de impacto: 1.429 (Q3 en SCIE)

25. **Autor(es):** H. García-González, D. Fernández-Álvarez, **J. E. Labra Gayo**, P. Ordóñez-de-Pablos

Título: Applying big data and stream processing to the real estate domain

Clave: Artículo

Nombre revista: Journal of Behaviour and Information Technology

Volumen: 38 (9)

Páginas: 950-958

Fecha: 2019

doi: <https://doi.org/10.1080/0144929X.2019.1620858>

Factor de impacto: 1.429 (Q3 en SCIE)

24. **Autor(es):** F. J. Hidalgo, **J. E. Labra Gayo**, P. Ordóñez de Pablos

Título: Semantic Modeling of Administrative Procedures from a Spanish Regional Public Administration

Clave: Artículo

Nombre revista: Sustainability

Volumen: 10 (3)

Fecha: 2018

doi: <https://doi.org/10.3390/su10030633>

Factor de impacto: 1.789 (Q3 en SCIE)

23. **Autor(es):** D. Fernández Álvarez, **J. E. Labra Gayo**, D. Gayo Avello, P. Ordóñez de Pablos
Título: MERA: A Musical Entities Reconciliation Architecture Based on Semantic Technologies
Clave: Artículo
Nombre revista: International Journal on Semantic Web and Information Systems (IJSWIS)
Volumen: 13 (4)
Fecha: 2017
doi: <https://doi.org/10.4018/IJSWIS.2017100103>
Factor de impacto: 1.833 (Q3 en SCIE)
22. **Autor(es):** J. A. Benítez, **J. E. Labra Gayo**, E. Quiroga, V. Martín, I. García, P. Marques, C. Benavides
Título: A web-based tool for automatic data collection, curation and visualization of complex healthcare survey studies including social network analysis
Clave: Artículo
Nombre revista: Computational and Mathematical Methods in Medicine
Volumen: 2017
Fecha: 2017
doi: <https://doi.org/10.1155/2017/2579848>
Factor de impacto: 1.563 (Q3 en SCIE)
21. **Autor(es):** H. García González, M. del Puerto Paule, **J. E. Labra Gayo**
Título: Enhancing e-Learning content by using Semantic Web technologies
Clave: Artículo
Nombre revista: Transactions on Learning Technologies
Volumen: 10 (4)
Fecha: 2016
doi: <https://doi.org/10.1109/TLT.2016.2629475>
Factor de impacto: 2.315 (Q2 en SCIE)
20. **Autor(es):** H. Sagastegui Chigne, **J. E. Labra Gayo**, M. E. Alva Obeso, P. Ordóñez De Pablos, J. M. Cuevas Lovelle
Título: Towards the Implementation of the Learning Analytics in the Social Learning Environments for the Technology-Enhanced Assessment in Computer Engineering Education
Clave: Artículo
Nombre revista: International Journal of Engineering Education
Volumen: 32 (4)
Páginas: 1637-1646
Fecha: 2016
Factor de impacto: 0.559 (Q2 en SCIE, Engineering, Multidisciplinary)
19. **Autor(es):** **J. E. Labra Gayo**, D. Kontokostas, S. Auer
Título: Multilingual Linked Data Patterns
Clave: Artículo
Nombre revista: Semantic Web Journal
Volumen: 6 (4)
Páginas: 319-337
Fecha: 2015

doi: <https://doi.org/10.3233/SW-140136>

Factor de impacto: 1.786 (Q1 en SCIE)

18. **Autor(es):** E. Rolando Núñez-Valdez, J. M. Cueva Lovelle, A. A. Juan Fuente, **J. E. Labra Gayo**

Título: [Creating recommendations on electronic books: A collaborative learning implicit approach](#)

Clave: Artículo

Nombre revista: Computers in Human Behavior

Fecha: 2015

doi: <http://dx.doi.org/10.1016/j.chb.2014.10.057>

Factor de impacto: 2.273 (Q1 en SSCIE)

17. **Autor(es):** S. J. González Pérez, B. López Pérez, D. Machado Fernández, **J. E. Labra Gayo**

Título: [Interoperability between platforms without a defined referential model: A semi-automatic learning system for structural pairing](#)

Clave: Artículo

Nombre revista: Computers in Human Behavior

Fecha: 2015

doi: <http://dx.doi.org/10.1016/j.chb.2014.11.011>

Factor de impacto: 2.273 (Q1 en SSCIE)

16. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, A. Rodríguez González, P. Ordóñez de Pablos

Título: [Empowering the access to public procurement opportunities by means of linking controlled vocabularies. A case study of Product Scheme Classifications in the European e-Procurement sector](#)

Clave: Artículo

Nombre revista: Computers in Human Behavior

Fecha: 2014

doi: <http://dx.doi.org/10.1016/j.chb.2013.07.046>

Factor de impacto: 2.273 (Q1 en SSCIE)

15. **Autor(es):** **J. E. Labra Gayo**, J. Jeuring, J. M. Álvarez Rodríguez

Título: [Inductive Representations of RDF Graphs](#)

Clave: Artículo

Nombre revista: Science of Computer Programming

Fecha: 2014

doi: <http://dx.doi.org/10.1016/j.scico.2013.12.011>

Factor de impacto: (2014) 0.715 (Q3 en SCIE)

14. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, P. Ordóñez de Pablos

Título: [New trends on e-Procurement applying semantic technologies](#)

Clave: Artículo

Nombre revista: Computers in Industry

Volumen: 65 (5)

Páginas: 797-799

Fecha: 2014

doi: <http://dx.doi.org/10.1016/j.compind.2014.04.002>

- Factor de impacto:** (2013) 1.457 (Q3 en SCIE)
13. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, P. Ordóñez de Pablos
Título: New trends on e-Procurement applying semantic technologies: Current status and future challenges
Clave: Artículo
Nombre revista: Computers in Industry
Volumen: 65 (5)
Páginas: 800-820
Fecha: 2014
doi: <http://dx.doi.org/10.1016/j.compind.2014.04.005>
Factor de impacto: (2013) 1.457 (Q3 en SCIE)
12. **Autor(es):** C. L. Alvargonzález, J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, P. Ordóñez de Pablos
Título: ACOTA: A multilingual and Semi-automatic collaborative tagging system
Clave: Artículo
Nombre revista: Journal of Web Engineering
Volumen: 13 (1 and 2)
Páginas: 160-180
Fecha: 2014
Factor de impacto: (2013) 0.444 (Q4 en SCIE)
11. **Autor(es):** A. Montes García, J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, M. Martínez Merino
Título: Towards a Journalist-Based News Recommendation System: The Wesomender Approach
Clave: Artículo
Nombre revista: Expert Systems with Applications
Volumen: 40 (17)
Páginas: 6735–6741
Fecha: 2013
doi: <https://doi.org/10.1016/j.eswa.2013.06.032>
Factor de impacto: (2013) 1.965 (Q1 en SCIE)
10. **Autor(es):** M. García Rodríguez, J. M. Alvarez Rodríguez, D. Berrueta Muñoz, L. Polo Paredes, **J. E. Labra Gayo**, P. Ordoñez De Pablos
Título: Towards a Practical Solution for Data Grounding in a Semantic Web Services Environment
Clave: Artículo
Nombre revista: Journal of Universal Computer Science
Volumen: 18 (11)
Fecha: 2012
doi: <https://doi.org/10.3217/jucs-018-11-1576>
Factor de impacto: (2012) 0.762 (Q3 en SCIE)
9. **Autor(es):** J. de Andrés, P. Lorca, **J. E. Labra Gayo**
Título: The effects of ERP implementations on the profitability of big firms: the case of Spain

Clave: Artículo

Nombre revista: Journal of Technology Management

Volumen: 59 (1-2)

Páginas: 22–44

Fecha: 2012

doi: <https://doi.org/10.1504/IJTM.2012.047254>

Factor de impacto: (2012) 0.564 (Q3 en SCIE)

8. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, F. Cifuentes, G. Hernández, C. Sánchez, J. A. Guzmán Luna

Título: Towards a Pan-European E-Procurement platform to Aggregate, Publish and Search Public Procurement Notices powered by Linked Open Data: The MOLDEAS Approach

Clave: Artículo

Nombre revista: International Journal of Software Engineering and Knowledge Engineering (IJSEKE) Special Issue on Consuming and Producing Linked Data on Real World Applications

Volumen: 22 (3)

Páginas: 365–383

Fecha: 2012

doi: <https://doi.org/10.1142/S0218194012400086>

Factor de impacto: (2012) 0.262 (Q4 en SCIE)

7. **Autor(es):** A. Rodríguez González, **J. E. Labra Gayo**, R. Colomo P., M. A. Mayer, J. M. Gómez Berbis, A. García Crespo

Título: SeDeLo: Using Semantics and Description Logics to Support Aided Clinical Diagnosis

Clave: Artículo

Nombre revista: Journal of Medical Systems

Volumen: 36 (4)

Páginas: 2471–2481

Fecha: 2012

doi: <https://doi.org/10.1007/s10916-011-9714-1>

Factor de impacto: (2012) 0.654 (Q3 en SCIE)

6. **Autor(es):** R. González Crespo, O. Sanjuán Martínez, J. M. Cueva Lovelle, B. C. Pelayo García-Bustelo, **J. E. Labra Gayo**, P. Ordoñez de Pablos

Título: Recommendation System based on User Interaction data applied to Intelligent Electronic Books

Clave: Artículo

Nombre revista: Computers in Human Behavior

Volumen: 27 (4)

Páginas: 1445–1449

Fecha: 2011

doi: <https://doi.org/10.1016/j.chb.2010.09.012>

Factor de impacto: (2011) 2.293 (Q1 en SSCIE)

5. **Autor(es):** **J. E. Labra Gayo**, J. M. Cueva Lovelle, P. Ordoñez de Pablos

Título: WESONET: Applying Semantic Web Technologies and Collaborative Tagging to Multimedia

Clave: Artículo

Nombre revista: Computers in Human Behavior

Volumen: 26 (2)

Páginas: 205–209

Fecha: 2010

doi: <https://doi.org/10.1016/j.chb.2009.10.004>

Factor de impacto: (2011) 2.293 (Q1 en SSCIE)

4. **Autor(es):** J. Barranquero Tolosa, **J. E. Labra Gayo**, A. B. Martínez Prieto, S. Méndez Núñez, P. Ordoñez de Pablos

Título: [Interactive web environment for collaborative and extensible diagram based learning](#)

Clave: Artículo

Nombre revista: Computers in Human Behavior

Volumen: 26 (2)

Páginas: 210–217

Fecha: 2010

doi: <https://doi.org/10.1016/j.chb.2009.10.003>

Factor de impacto: (2011) 2.293 (Q1 en SSCIE)

3. **Autor(es):** Yingzhou Zhang, **J. E. Labra Gayo**, A. Cernuda del Río

Título: [A monadic program slicer](#)

Clave: Artículo

Nombre revista: ACM SIGPLAN Notices

Volumen: 41 (5)

Páginas: 30–38

Fecha: 2010

doi: <https://doi.org/10.1145/1149982.1149986>

Factor de impacto: 0.621 (Q4 en SCIE)

2. **Autor(es):** M. González Rodríguez, **J. E. Labra Gayo**, J. M. Cueva Lovelle

Título: [Web Navigability Testing with Remote Agents](#)

Clave: Artículo

Nombre revista: Lecture Notes in Computer Science

Volumen: 2016

Páginas: 311–323

Fecha: 2001

doi: https://doi.org/10.1007/3-540-45144-7_29

Factor de impacto: 0.515 (Q4 en SCIE)

1. **Autor(es):** D. Basanta Gutiérrez, M. C. Luengo Díez, R. Izquierdo Castanedo, **J. E. Labra Gayo**, J. M. Cueva Lovelle

Título: [Improving the quality of compiler construction with object-oriented techniques](#)

Clave: Artículo

Nombre revista: ACM SIGPLAN Notices

Volumen: 35 (12)

Páginas: 41–50

Fecha: 2000

doi: <https://doi.org/10.1145/369928.369944>

Factor de impacto: (2002) 0.190 (Q4 en SCIE)

8.2 Otras revistas internacionales

4. **Autor(es):** A. Rodríguez González, Á. García-Crespo, R. Colomo Palacios, **J. E. Labra Gayo**, J. Miguel Gómez-Berbís, G. Alor-Hernández

Título: [Automated Diagnosis through Ontologies and Logical Descriptions: The ADONIS Approach](#)

Clave: Artículo

Nombre revista: International Journal of Decision Support System Technology

Volumen: 3 (1)

Páginas: 21–39

Fecha: 2011

Editorial: IGI Global

doi: <https://doi.org/10.4018/jdsst.2011010102>

Factor de impacto: SCIMAGO (2011): Q3

3. **Autor(es):** M. Elena Alva, A. Belén Martínez, M. del C. Suárez, **J. E. Labra Gayo**, J. M. Cueva Lovelle, H. Sagástegui Chigne

Título: [Towards the evaluation of usability in educative websites](#)

Clave: Artículo

Nombre revista: International Journal of Technology Enhanced Learning

Volumen: 2 (1-2)

Fecha: 2010

doi: <https://dx.doi.org/10.1504/IJTEL.2010.031265>

Factor de impacto: SCIMAGO (2010): Q3

2. **Autor(es):** D. Ana Cernea, **J. E. Labra Gayo**, E. del Moral

Título: [SOAF: Semantic Indexing System Based on Collaborative Tagging](#)

Clave: Artículo

Nombre revista: Interdisciplinary Journal of E-Learning and Learning Objects

Volumen: 4 (1)

Fecha: 2008

doi: <https://doi.org/10.28945/371>

1. **Autor(es):** **J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez, A. Cernuda del Río

Título: [Modular Development of Interpreters from Semantic Building Blocks](#)

Clave: Artículo

Nombre revista: Nordic Journal of Computing

Páginas: 391–407

Fecha: 2008

8.3 Libros como autor

3. **Autor(es):** A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutiérrez, **J. E. Labra-Gayo**, S. Kirrane, S. Neumaier, A. Polleres, R. Navigli, A. Ngonga, S. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann

Título: [Knowledge graphs](#)

Editorial: Morgan & Claypool Publishers

Fecha: Previsto para finales de 2021

Url: <https://www.emse.fr/~zimmermann/KGBook/HTML-Book/>

2. **Autor(es):** **J. E. Labra Gayo**, E. Prud'hommeaux, I. Boneva, D. Kontokostas

Título: [Validating RDF data](#)

Editorial: Morgan & Claypool Publishers

Referencia: Synthesis Lectures on the Semantic Web: Theory and Technology

Volumen: 7 (1)

Páginas: 1–328

Fecha: 2017

doi: <https://doi.org/10.2200/S00786ED1V01Y201707WBE016>

ISBN: 978-1681731643

Url: <http://book.validatingrdf.com/>

1. **Autor(es):** J. E. Labra Gayo

Título: [Web Semántica](#)

Editorial: Netbiblo

Fecha: 2012

ISBN: 978-84-9745-571-8

8.4 Libros como editor

4. **Editor(es):** P. Ordóñez de Pablos, Xi Zhang, Mohammad Nabil Almunawar, **J. E. Labra Gayo**
Título: [Handbook of Research on Big Data, Green Growth, and Technology Disruption in Asian Companies and Societies](#)
Editorial: IGI Global
Fecha: 2021
ISBN: 9781799885245
doi: <https://doi.org/10.4018/978-1-7998-8524-5>
3. **Editor(es):** P. Ordóñez de Pablos, M. Lytras, **J. E. Labra Gayo**
Título: [Cases on Open-Linked Data and Semantic Web Applications](#)
Editorial: IGI Global
Fecha: 2013
ISBN: 978-1466628274
doi: <https://doi.org/10.4018/978-1-4666-2827-4>
2. **Editor(es):** P. Ordóñez de Pablos, J. M. Cueva Lovelle, **J. E. Labra Gayo**, R. Tennyson
Título: [E-Procurement Management for Successful Electronic Government Systems](#)
Editorial: IGI Global
Fecha: 2012
ISBN: 9781466621190
doi: <https://doi.org/10.4018/978-1-4666-2119-0>
1. **Editor(es):** J. M. Cueva Lovelle, M. González Rodríguez, L. Joyanes, **J. E. Labra Gayo**, M. del Puerto Paule
Título: [Web Engineering](#)
Editorial: Springer-Verlag
Referencia: Lecture Notes in Computer Science
Fecha: 2003
doi: <https://doi.org/10.1007/3-540-45068-8>

8.5 Números especiales de revistas editados

2. **Editor(es):** P. Ordóñez de Pablos, **J. E. Labra Gayo**

Título: Special issue about "Big data research for behavior, integrating services and applications: Implications for innovation, policy making and technology transfer

Nombre revista: Journal of Behaviour and Information Technology

Volumen: 38 (9)

Fecha: 2019

Factor de impacto: 1.429 (Q3 en SCIE)

1. **Editor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**, P. Ordóñez de Pablos

Título: Special issue: New trends on E-Procurement applying Semantic Technologies

Nombre revista: Computers in Industry

Fecha: 2014

Factor de impacto: 1.457 (Q3 en SCIE)

8.6 Capítulos de libros

7. **Autor(es):** **J. E. Labra Gayo**, H. García González, D. Fernández Álvarez, E. Prud'hommeaux

Título: Challenges in RDF validation

Referencia: Libro: Current Trends in Semantic Web Technologies: Theory and Practice

Editor(es): G. Alor-Hernández, J. L. Sánchez-Cervantes, A. Rodríguez-González, R. Valencia-García

Editorial: Springer-Verlag, Studies in Computational Intelligence

Fecha: 2019

doi: https://doi.org/10.1007/978-3-030-06149-4_6

ISBN: 978-3-030-06149-4

6. **Autor(es):** A. Rodríguez-González, Á. García-Crespo, R. Colomo-Palacios, **J. E. Labra Gayo**, J. M. Gómez-Berbís, G. Alor-Hernández

Título: Automated Diagnosis through Ontologies and Logical Descriptions: The ADONIS Approach

Referencia: Libro: Engineering Effective Decision Support Technologies: New Models and Applications

Editorial: IGI Global

Fecha: 2013

doi: <https://doi.org/10.4018/978-1-4666-4002-3.ch007>

5. **Autor(es):** J. L. Marín, M. Rodríguez, A. Marín, R. Calmeau, J. M. Álvarez Rodríguez, **J. E. Labra Gayo**

Título: Euroalert.net: aggregating public procurement data to deliver commercial services to SMEs

Referencia: Libro: E-Procurement Management for Successful Electronic Government Systems

Editorial: IGI Global

Páginas: 114-130

Fecha: 2013

doi: <https://doi.org/10.4018/978-1-4666-2119-0.ch007>

4. **Autor(es):** W. Jimenez, C. L. Alvargonzález, P. Abella Vallina, J. M. Álvarez Rodríguez, P. Ordóñez de Pablos, **J. E. Labra Gayo**

Título: Emergent Ontologies by Collaborative Tagging for Knowledge Management

Referencia: Libro: Advancing Information Management through Semantic Web Concepts and Ontologies

- Editorial:** IGI Global
Páginas: 54–69
Fecha: 2013
doi: <https://doi.org/10.4018/978-1-4666-2494-8.ch002>
3. **Autor(es):** S. Fernández, D. Berrueta, L. Shi, **J. E. Labra Gayo**
Título: [Mailing Lists and Social Semantic Web](#)
Referencia: Libro: Social Web Evolution: Integrating Semantic Applications and Web 2.0 Technologies
Editorial: IGI Global
Páginas: 335–349
Fecha: 2010
doi: <https://doi.org/10.4018/978-1-60566-272-5.ch004>
2. **Autor(es):** J. de Andrés, P. Lorca, **J. E. Labra Gayo**
Título: [Determinants of ERP implementations: an empirical study in Spanish companies](#)
Referencia: Libro: Emerging Topics and Technologies in Information Systems
Editorial: IGI Global
Fecha: 2009
doi: <https://doi.org/10.4018/978-1-60566-222-0.ch010>
1. **Autor(es):** J. L. Marín de la Iglesia, **J. E. Labra Gayo**
Título: [Doing Business by selling free services](#)
Referencia: Libro: Web 2.0, the Business Model
Editorial: Springer
Fecha: 2008
doi: https://doi.org/10.1007/978-0-387-85895-1_6

8.7 Publicaciones derivadas de congresos

39. **Autor(es):** Andra Waagmeester, Paul Braun, Manoj Karingamadathil, **J. E. Labra Gayo**,
 Siobhan Leachman, Katherine Thornton
Título: [Linking Data and Descriptions on Moths Using the Wikimedia Ecosystem](#)
Clave: Congreso
Nombre congreso: Biodiversity Information Sciences and Standards - TDWG21
Volumen: 5
Fecha: 2021
doi: <https://doi.org/10.3897/biss.5.73806>
38. **Autor(es):** G. Facundo Colunga, A. González Hevia, **J. E. Labra Gayo**
Título: [ShEx-Lite: Automatic generation of domain object models from a Shape Expressions subset language](#)
Clave: Congreso
Nombre congreso: International Semantic Web Conference – Demos and Industry track
Volumen: 2721
Páginas: 148–152
Fecha: 2020
Editorial: CEUR Workshop Proceedings
ISSN: 1613-0073
Factor de impacto: Demo - conferencia tipo A+ (GII-GRIN-SCIE)
37. **Autor(es):** A. González Hevia, G. Facundo Colunga, E. Rubiera Azcona, **J. E. Labra Gayo**

Título: Automatic synchronization of RDF graphs representing ontologies and Wikibase instances

Clave: Congreso

Nombre congreso: International Semantic Web Conference. Wikidata Workshop

Volumen: 2773

ISSN: 1613-0073

Fecha: 2020

Editorial: CEUR Workshop Proceedings

36. **Autor(es):** F. Cifuentes Silva, **J. E. Labra Gayo**

Título: Legislative document content extraction based on Semantic Web technologies - A use case about processing the History of the Law

Clave: Congreso

Nombre congreso: Extended Semantic Web Conference, ESWC2019, Portoroz, Slovenia

Volumen: 11503

Páginas: 558–573

ISBN: 978-3-030-21347-3

Fecha: 2019

Editorial: Springer

doi: https://doi.org/10.1007/978-3-030-21348-0_36

Factor de impacto: Conferencia tipo A (GII-GRIN-SCIE), nominado a Best Paper Award

35. **Autor(es):** K. Thornton, H. Solbrig, G. Stupp, **J. E. Labra Gayo**, D. Mietchen, E. Prud'Hommeaux, A. Waagmeester

Título: Using Shape Expressions (ShEx) to Share RDF Data Models and to Guide Curation with Rigorous Validation

Clave: Congreso

Nombre congreso: Extended Semantic Web Conference, ESWC2019, Portoroz, Slovenia

Volumen: 11503

Páginas: 606–620

ISBN: 978-3-030-21347-3

Fecha: 2019

Editorial: Springer

doi: https://doi.org/10.1007/978-3-030-21348-0_39

Factor de impacto: Conferencia tipo A (GII-GRIN-SCIE), Best Paper Award

34. **Autor(es):** F. Årup Nielsen, K. Thornton, **J. E. Labra Gayo**

Título: Validating Danish Wikidata Lexemes

Clave: Congreso

Nombre congreso: Semantics Conference - Posters and demos

Volumen: 2451 (20)

ISSN: 1613-0073

Fecha: 2019

Editorial: CEUR Workshop Proceedings

33. **Autor(es):** **J. E. Labra Gayo**, H. García González, D. Fernández Álvarez

Título: RDFShape: An RDF playground based on Shapes

Clave: Congreso

Nombre congreso: ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks

Volumen: 2180 (35)

- Fecha:** 2018
Editorial: CEUR Workshop Proceedings
URL: <http://ceur-ws.org/Vol-2180/paper-35.pdf>
Factor de impacto: Demo - conferencia tipo A+ (GII-GRIN-SCIE)
32. **Autor(es):** D. Fernández Álvarez, H. García González, J. Frey, S. Hellmann, **J. E. Labra Gayo**
Título: [Inference of Latent Shape Expressions Associated to DBpedia Ontology](#)
Clave: Congreso
Nombre congreso: ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks
Volumen: 2180 (15)
Fecha: 2018
Editorial: CEUR Workshop Proceedings
URL: <http://ceur-ws.org/Vol-2180/paper-15.pdf>
Factor de impacto: Poster en conferencia tipo A+ (GII-GRIN-SCIE)
31. **Autor(es):** I. Boneva, **J. E. Labra Gayo**, E. Prud'Hommeaux
Título: [Semantics and Validation of Shapes Schemas for RDF](#)
Clave: Congreso
Nombre congreso: International Semantic Web Conference
Volumen: 10587
Páginas: 104–120
Fecha: 2017
Editorial: Springer
doi: https://doi.org/10.1007/978-3-319-68288-4_7
Factor de impacto: Conferencia tipo A+ (GII-GRIN-SCIE)
30. **Autor(es):** V. Aravena, R. Gacitúa, H. Astudillo, **J. E. Labra Gayo**
Título: [Identifying potential suppliers for competitive bidding using Latent Semantic Analysis](#)
Clave: Congreso
Nombre congreso: Proceedings of the 2016 42nd Latin American Computing Conference
Páginas: 1–12
Fecha: 2016
Editorial: IEEE
Lugar: Valparaíso, Chile
doi: <https://doi.org/10.1109/CLEI.2016.7833360>
Factor de impacto: Conferencia tipo C (GII-GRIN-SCIE)
29. **Autor(es):** D. Fernández Álvarez, **J. E. Labra Gayo**, H. García González
Título: [Inference and Serialization of Latent Graph Schemata Using ShEx](#)
Clave: Congreso
Nombre congreso: The Tenth International Conference on Advances in Semantic Processing – SEMAPRO
Páginas: 68–70
Fecha: 2016
Editorial: IARIA XPS Press
Lugar: Venecia, Italia
ISBN: 978-1-61208-507-4

28. **Autor(es):** S. Staworko, I. Boneva, **J. E. Labra Gayo**, S. Hym, E. Prud'Hommeaux, H. Solbrig
Título: Complexity and expressiveness of ShEx for RDF
Clave: Congreso
Nombre congreso: 18th International Conference on Database Theory, ICDT
Páginas: 195–211
Fecha: 2015
Referencia: Leibniz International Proceedings in Informatics (LIPIcs)
Editorial: Dagstuhl Publishing
Editor(es): M. Aregas, M. Ugarte
doi: <https://doi.org/10.4230/LIPIcs.ICDT.2015.195>
Factor de impacto: Conferencia tipo A (GII-GRIN-SCIE)
27. **Autor(es):** **J. E. Labra Gayo**, E. Prud'Hommeaux, S. Staworko, H. Solbrig
Título: Towards an RDF validation language based on Regular Expression derivatives
Clave: Congreso
Nombre congreso: International Workshop on Linked Web Data Management (LWDM)
Volumen: 1330
Editorial: CEUR Workshop Proceedings
Páginas: 197–204
Fecha: 2015
Lugar: Bruselas, Bélgica
26. **Autor(es):** E. Prud'hommeaux, **J. E. Labra Gayo**, H. Solbrig
Título: Shape Expressions: an RDF validation and transformation language
Clave: Congreso
Nombre congreso: 10th International Conference on Semantic Systems
Editorial: ACM
Páginas: 32–40
Fecha: 2014
Lugar: Leipzig, Alemania
ISBN: 978-1-4503-2927-9
Factor de impacto: Best paper award, conferencia tipo B (GII-GRIN-SCIE)
25. **Autor(es):** **J. E. Labra Gayo**, E. Prud'hommeaux, H. Solbrig, J. M. Álvarez Rodríguez
Título: Validating linked data portals using RDF shape expressions
Clave: Congreso
Nombre congreso: 1st Workshop on Linked Data Quality
Volumen: 1215
Editorial: CEUR Workshop Proceedings
Fecha: 2014
Lugar: Leipzig, Alemania
24. **Autor(es):** **J. E. Labra Gayo**, H. Farham, J. Castro Fernández, J. M. Álvarez Rodríguez
Título: Representing Statistical Indexes as Linked Data Including Metadata about Their Computation Process
Clave: Congreso
Nombre congreso: Metadata and Semantics Research
Editor(es): S. Closs, R. Studer, E. Garoufallou, M. A. Sicilia
Páginas: 42–53
Fecha: 2014

- Editorial:** Springer
doi: https://doi.org/10.1007/978-3-319-13674-5_5
ISBN: 978-3-319-13674-5
Factor de impacto: Conferencia tipo C (GII-GRIN-SCIE)
23. **Autor(es):** J. E. Labra Gayo, J. Jeuring, J. M. Álvarez Rodríguez
Título: [Inductive Graphs: a purely functional approach to represent RDF](#)
Clave: Congreso
Nombre congreso: Graph Structures for knowledge representation and reasoning
Editor(es): M. Croitoru, S. Rudolph, S. Woltran, C. Gonzales
Páginas: 92–111
Fecha: 2014
Editorial: Springer
doi: https://doi.org/10.1007/978-3-319-04534-4_7
ISBN: 978-3-319-04534-4
22. **Autor(es):** J. M. Álvarez Rodríguez, P. Ordóñez de Pablos, M. Vafopoulos, **J. E. Labra Gayo**
Título: [Towards a Stepwise Method for Unifying and Reconciling Corporate Names in Public Contracts Metadata: The CORFU Technique](#)
Clave: Congreso
Nombre congreso: Conference on Metadata and Semantic Research
Volumen: 5288
Páginas: 315–329
Fecha: 2013
Editorial: Springer
doi: https://doi.org/10.1007/978-3-319-03437-9_3
ISBN: 978-3-319-04534-4
Factor de impacto: Conferencia C (GII-GRIN-SCIE)
21. **Autor(es):** J. González Lorenzo, **J. E. Labra Gayo**, J. M. Álvarez Rodríguez
Título: [Applying MapReduce to Spreading Activation Algorithm on Large RDF Graphs](#)
Referencia: World Summit on Knowledge Society - WSKS'11
Editor(es): M. Lytras, D. Ruan, R. D. Tennyson, P. Ordóñez De Pablos , F. García Peñalvo, L. Rusu
Editorial: Springer-Verlag, Communications in Computer and Information Science
Páginas: 601–611
Fecha: 2011
Volumen: 278
doi: https://doi.org/10.1007/978-3-642-35879-1_76
ISBN: 978-3-642-35879-1
20. **Autor(es):** J. M. Alvarez Rodríguez, **J. E. Labra Gayo**, Patricia Ordoñez De Pablos
Título: [ONTOSPREAD: A Framework for Supporting the Activation of Concepts in Graph-Based Structures through the Spreading Activation Technique](#)
Referencia: World Summit on Knowledge Society - WSKS'11
Editor(es): M. Lytras, D. Ruan, R. D. Tennyson, P. Ordóñez De Pablos , F. García Peñalvo, L. Rusu
Editorial: Springer-Verlag, Communications in Computer and Information Science
Páginas: 454–459
Fecha: 2011

- Volumen:** 278
doi: https://doi.org/10.1007/978-3-642-35879-1_55
ISBN: 978-3-642-35879-1
19. **Autor(es):** F. Cifuentes Silva, **J. E. Labra Gayo**, C. Sifaqui
Título: [Towards an architecture and adoption process for Linked Data technologies in Open Government contexts - A case study for the Library of Congress of Chile](#)
Referencia: I-Semantics, 7th International Conference on Semantic Systems
Editorial: ACM
Páginas: 79–86
Fecha: 2011
doi: <https://doi.org/10.1145/2063518.2063529>
Factor de impacto: Conferencia tipo C (GII-GRIN-SCIE)
18. **Autor(es):** J. de Andrés, M. Landajo, P. Lorca, **J. E. Labra Gayo**, P. Ordóñez de Pablos
Título: [Assessing the Liquidity of Firms: Robust Neural Network Regression as an Alternative to the Current Ratio](#)
Referencia: World Summit on Knowledge Society - WSKS'10
Editorial: Stringer-Verlag
Fecha: 2010
Volumen: 111
doi: https://doi.org/10.1007/978-3-642-16318-0_69
17. **Autor(es):** S. Fernández, D. Berrueta, M. García Rodríguez, **J. E. Labra Gayo**
Título: [TRIOO - Keeping the Semantics of Data Safe and Sound into Object-oriented Software](#)
Referencia: Proceedings of the 5th International Conference on Software and Data Technologies (ICSOFT)
Editorial: SCITE-Press
Fecha: 2010
Volumen: 1
ISBN: 978-989-8425-22-5
doi: <https://doi.org/10.5220/0003008803110320>
Factor de impacto: Conferencia tipo B (GII-GRIN-SCIE)
16. **Autor(es):** S. Méndez Núñez, **J. E. Labra Gayo**, J. de Andrés
Título: [Towards a Semantic Web Environment for XBRL](#)
Referencia: World Wide Web Conference, Developers track
Fecha: 2009
URL: <http://www2009.eprints.org/231/index.html>
15. **Autor(es):** S. Méndez Núñez, **J. E. Labra Gayo**, J. de Andrés Suárez, P. Ordóñez de Pablos
Título: [Analysis of XBRL documents containing accounting information of listed firms using Semantic Web Technologies](#)
Referencia: Metadata and Semantics
Editorial: Stringer-Verlag
Páginas: 375–381
Fecha: 2007
ISBN: 978-0-387-77745-0
doi: https://doi.org/10.1007/978-0-387-77745-0_36
14. **Autor(es):** P. Kroha, R. Janetzko, **J. E. Labra Gayo**

- Título:** [Ontologies in Checking for Inconsistency of Requirements Specification](#)
Referencia: The Third International Conference on Advances in Semantic Processing (SE-MAPRO)
Editorial: IEEE
Páginas: 375–381
Fecha: 2009
doi: <https://doi.org/10.1109/SEMAPRO.2009.11>
13. **Autor(es):** A. Rodríguez, R. Colomo, J. M. Gómez, G. Alor, R. Posada-Gomez, **J. E. Labra Gayo**, K. Vidyasankar
Título: [A proposal for a Semantic Intelligent Document Repository Architecture](#)
Referencia: Electronics, Robotics and Automotive Mechanics Conference (CERMA)
Editorial: IEEE
Páginas: 375–381
Fecha: 2009
Lugar: Mexico
doi: <https://doi.org/10.1109/CERMA.2009.26>
12. **Autor(es):** A. Rodríguez, E. Jiménez, M. Radzimski, J. M. Gómez, G. Alor, R. Posada-Gomez, **J. E. Labra Gayo**
Título: [Applying Caching Capabilities to Inference Applications based on Semantic Web](#)
Referencia: 1st International Conference on Computational Collective Intelligence - Semantic Web, Social Networks & Multiagent Systems (ICCCI)
Editorial: IEEE Studies in Computational Intelligence
Páginas: 27–37
Fecha: 2009
Lugar: Hongkong
doi: https://doi.org/10.1007/978-3-642-03958-4_3
11. **Autor(es):** A. Rodríguez, **J. E. Labra Gayo**, G. Alor, J. M. Gómez
Título: [ADONIS: Automated diagnosis system based on sound and precise logical descriptions](#)
Referencia: IEEE Computer Based Medical Systems
Editorial: IEEE
Páginas: 27–37
Fecha: 2009
Lugar: Albuquerque, USA
doi: <https://doi.org/10.1109/CBMS.2009.5255449>
10. **Autor(es):** M. E. Alva, A. B. Martínez Prieto, **J. E. Labra Gayo**, M. C. Suárez, J. M. Cuevas Lovelle, H. Sagastegui
Título: [Proposal of a Tool of Support to the Evaluation of User in Educative Web Sites](#)
Referencia: Emerging Technologies and Information Systems for the Knowledge Society - World Summit on Knowledge Society
Editorial: Springer-Verlag
Páginas: 149–157
Volumen: 5288
Fecha: 2008
doi: https://doi.org/10.1007/978-3-540-87781-3_17
9. **Autor(es):** S. Méndez Núñez, J. de Andrés Suárez, **J. E. Labra Gayo**, P. Ordóñez de Pablo

Título: A Semantic Based Collaborative System for the Interoperability of XBRL Accounting Information

Referencia: Emerging Technologies and Information Systems for the Knowledge Society - World Summit on Knowledge Society

Editorial: Springer-Verlag

Páginas: 593–599

Volumen: 5288

Fecha: 2008

doi: https://doi.org/10.1007/978-3-540-87781-3_64

8. **Autor(es):** S. Fernández Rodríguez, D. Berrueta, **J. E. Labra Gayo**

Título: A semantic web approach to publish and consume mailing lists

Referencia: IADIS International Journal on WWW/Internet

Editorial: IADIS Press

Fecha: 2008

ISSN: 1645-7641

doi: https://doi.org/10.1007/978-3-540-87781-3_64

7. **Autor(es):** D. Berrueta, **J. E. Labra Gayo**, L. Polo

Título: Searching over Public Administration Legal Documents Using Ontologies

Referencia: Joint Conference on Knowledge Based Software Engineering

Editorial: IOS Press

Volumen: 140

Páginas: 167–175

Fecha: 2006

Lugar: Tallin, Estonia

6. **Autor(es):** Y. Zhang, B. Xu, **J. E. Labra Gayo**

Título: A Formal Method for Program Slicing

Referencia: IEEE Australian Software Engineering Conference ASWEC

Editorial: IEEE

Volumen: 140

Fecha: 2005

Lugar: Brisbane, Australia

ISBN: 0-7695-2257-2

doi: <https://doi.org/10.1109/ASWEC.2005.7>

5. **Autor(es):** T. H. Sagastegui, **J. E. Labra Gayo**, J. M. Cueva Lovelle, J. Morales, M. E. Alva, E. Valdés, C. García

Título: A Collaborative E-learning Component for the IDEFIX Project

Referencia: International Conference on Web Engineering

Editorial: Springer-Verlag

Volumen: 2722

Páginas: 116–119

Fecha: 2003

ISSN: 0302-9743

doi: https://doi.org/10.1007/3-540-45068-8_21

4. **Autor(es):** **J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez, A. Cernuda del Río

Título: Specification of Logic Programming Languages from Reusable Semantic Building Blocks

- Referencia:** Electronic Notes in Theoretical Computer Science
Editorial: Elsevier
Volumen: 64
Páginas: 220–233
Fecha: 2002
doi: [https://doi.org/10.1016/S1571-0661\(04\)80352-9](https://doi.org/10.1016/S1571-0661(04)80352-9)
3. **Autor(es):** J. E. Labra Gayo, M. C. Luengo Díez, J. M. Cueva Lovelle, A. Cernuda del Río
Título: LPS: A Language Prototyping System Using Modular Monadic Semantics
Referencia: Electronic Notes in Theoretical Computer Science
Editorial: Elsevier
Volumen: 44 (2)
Páginas: 110–131
Fecha: 2001
doi: [https://doi.org/10.1016/S1571-0661\(04\)80923-X](https://doi.org/10.1016/S1571-0661(04)80923-X)
2. **Autor(es):** J. E. Labra Gayo, J. M. Cueva Lovelle, C. Luengo Díez, M. González Rodríguez
Título: A Language Prototyping Tool based on Semantic Building Blocks
Referencia: Functional Programming Workshop at EUROCAST'01
Editorial: Springer
Volumen: 2178
Páginas: 597–611
Fecha: 2001
ISBN: 978-3-540-42959-3
doi: https://doi.org/10.1007/3-540-45654-6_46
1. **Autor(es):** A. Cernuda del Río, J. E. Labra Gayo, J. M. Cueva Lovelle
Título: A Model for Integrating Knowledge into Component-Based Software Development
Referencia: Knowledge Management
Editorial: University of Paisley
Volumen: 44 (2)
Páginas: 110–131
Fecha: 2001
ISBN: 84-95721-21-X
doi: [https://doi.org/10.1016/S1571-0661\(04\)80923-X](https://doi.org/10.1016/S1571-0661(04)80923-X)

8.8 Participaciones en congresos

38. **Autor(es):** J. E. Labra Gayo, J. Barranquero Tolosa, G. Facundo Colunga, A. González Hevia, E. Rubiera Azcona, D. Ruiz Santamaría, P. Álvarez de Ron Ondina
Título: ASIO: A research management system based on semantic technologies
Clave: Congreso
Nombre congreso: International Semantic Web Conference, ISWC2020. First virtual workshop on Research data management for Linked Open Science - DaMALOS 2020
URL: <https://repository.publisso.de/resource/frl%3A6423282>
Fecha: 2020
37. **Autor(es):** J. E. Labra Gayo, J. Barranquero Tolosa, G. Facundo Colunga, A. González Hevia, E. Rubiera Azcona, D. Ruiz Santamaría, P. Álvarez de Ron Ondina
Título: Entity Schema visualization and authoring tools
Clave: Congreso

Nombre congreso: WikidataCon

URL: https://www.wikidata.org/wiki/Wikidata:WikidataCon_2019

Fecha: 2019

Lugar: Berlín, Alemania

36. **Autor(es): J. E. Labra Gayo**

Título: Data models, visualizations and Shape Expressions for RDF. Combining RDFShape with Cytoscape

Clave: Congreso

Nombre congreso: Biohackathon 2019

URL: <http://2019.biohackathon.org/>

Fecha: 2019

Lugar: Fukuoka, Japón

35. **Autor(es): J. E. Labra Gayo**

Título: ShEx Wikidata

Clave: Congreso

Nombre congreso: Wikimedia Hackathon

URL: https://www.mediawiki.org/wiki/Wikimedia_Hackathon_2019

Fecha: 2019

Lugar: Praga, República Checa

34. **Autor(es): J. E. Labra Gayo**

Título: Negation and recursion validating RDF data

Clave: Congreso

Nombre congreso: 4th Workshop on Logical Reasoning and Computation

Fecha: 2018

Lugar: A Coruña, España

33. **Autor(es): J. E. Labra Gayo, J. Jeuring, J. M. Álvarez Rodríguez**

Título: Inductive Graphs: A purely functional approach to represent RDF

Clave: Congreso

Nombre congreso: Workshop Graph Structures for Knowledge Representation and Reasoning, International Conference on Artificial Intelligence

Fecha: 2013

Lugar: Beijing, China

32. **Autor(es): F. E. Castillo-Barrera, R. C. Medina-Ramirez, J. E. Labra Gayo, S. Masoud Sadjadi**

Título: Kukulcan: Semantic Web Framework for Knowledge Management in the Domain of Digital Circuits

Clave: Congreso

Nombre congreso: IKE'13: The 12th International Conference on Information and Knowledge Engineering

Fecha: 2013

Lugar: Las Vegas, USA

31. **Autor(es): J. E. Labra Gayo**

Título: Multilingual Linked Data Patterns

Clave: Congreso

Nombre congreso: 6th W3C Workshop on Multilingual Web

Fecha: 2013

- Lugar:** Roma, USA
30. **Autor(es): J. E. Labra Gayo**
Título: Best Practices for Multilingual Linked Open Data
Clave: Congreso
Nombre congreso: 5th W3C Workshop on Multilingual Web
Fecha: 2012
Lugar: Dublin, Irlanda
Video: http://videolectures.net/w3cworkshop2012_labra_gayo_data/
29. **Autor(es): J. E. Labra Gayo, J. M. Álvarez Rodríguez**
Título: Validating statistical index data represented in RDF using SPARQL queries
Clave: Congreso
Nombre congreso: W3C RDF Validation Workshop: Practical Assurances for Quality RDF data
Fecha: 2012
Lugar: Boston, USA
URL: <https://www.w3.org/2012/12/rdf-val/>
28. **Autor(es): J. M. Álvarez Rodríguez, J. E. Labra Gayo, R. Calmeau, A. Marín, J. L. Marín**
Título: Innovative Services to Ease the Access to the public procurement notices using linking open data and advanced methods based on semantics
Referencia: 5th International Conference on Methodologies, Technologies and Tools enabling e-Government
Páginas: 267–282
Fecha: 2011
Lugar: Camerino, Italia
27. **Autor(es): A. Álvarez, J. E. Labra Gayo**
Título: SCOWT: Semantic Competencies for Worker Training
Referencia: IADIS e-Learning Conference
Fecha: 2011
Lugar: Roma, Italia
26. **Autor(es): J. M. Álvarez Rodríguez, P. Abella Vallina, W. Jiménez, J. E. Labra Gayo**
Título: Application of the Spreading Activation Technique for Recommending Concepts of well-known ontologies in Medical Systems
Referencia: I Workshop on Semantic Applied Technologies on Biomedical Systems (SATBI 2011), ACM International Conference on Bioinformatics and Computational Biology
Fecha: 2011
Lugar: Chicago, USA
25. **Autor(es): J. M. Álvarez Rodríguez, J. E. Labra Gayo, R. Calmeau, A. Marín, J. L. Marín**
Título: Query Expansion Methods and Performance Evaluation for Reusing Linking Open Data of the European Public Procurement Notices
Referencia: Workshop sobre Linked Data en España. CAEPIA
Fecha: 2011
Lugar: La Laguna, Tenerife, España
24. **Autor(es): F. Cifuentes Silva, J. M. Álvarez Rodríguez, J. E. Labra Gayo**
Título: An architecture and process of implantation for Linked Data environments
Referencia: Workshop sobre Linked Data en España. CAEPIA

- Fecha:** 2011
Lugar: La Laguna, Tenerife, España
23. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**
Título: [10ders Project](#)
Referencia: Red Temática Española de Linked Data
Fecha: 2011
Lugar: Madrid, España
22. **Autor(es):** J. M. Álvarez Rodríguez, **J. E. Labra Gayo**
Título: [Semantic Methods for Reusing Linking Open Data of the European Public Procurement Notices](#)
Referencia: PhD Symposium ESWC
Fecha: 2011
Lugar: Camerino, Italia
21. **Autor(es):** J. L. Marín, A. Marín, M. Rodríguez, J. M. Álvarez Rodríguez, **J. E. Labra Gayo**
Título: [Euroalert.net: Building a pan-European platform to aggregate public procurement data and deliver commercial services for SMEs powered by open data](#)
Referencia: Workshop Share-PSI.eu
Fecha: 2011
Lugar: Bruselas, Bélgica
20. **Autor(es):** D. Berrueta, **J. E. Labra Gayo**, I. Herman
Título: [XSLT+SPARQL: Scripting the Semantic Web with SPARQL embedded into XSLT stylesheets](#)
Referencia: 4th Workshop on Scripting the Semantic Web
Fecha: 2008
Lugar: Tenerife, España
19. **Autor(es):** **J. E. Labra Gayo**, P. Ordóñez de Pablos, J. M. Cueva Lovelle
Título: [Combining Collaborative Tagging and Ontologies in Image Retrieval Systems](#)
Referencia: 2nd International Conference on Metadata and Semantic Web Research
Fecha: 2007
Lugar: Corfú, Grecia
18. **Autor(es):** S. Fernández, D. Berrueta, **J. E. Labra Gayo**
Título: [Mailing lists meet the semantic web](#)
Referencia: Workshop on Social Aspects of the Semantic Web (SAW)
Fecha: 2007
Lugar: Poznan, Polonia
17. **Autor(es):** D. A. Cernea, E. del Moral, **J. E. Labra Gayo**
Título: [SOAF: un sistema de indexado semántico de OA basado en las anotaciones colaborativas](#)
Referencia: IV Simposio Pluridisciplinar sobre Diseño, Evaluación y Desarrollo de Contenidos Educativos Reutilizables (SPEDECE' 07)
Fecha: 2007
Lugar: Bilbao, España
16. **Autor(es):** A. Cernuda del Río, **J. E. Labra Gayo**

- Título:** Validación estática de compatibilidad de procesos de negocio en coreografías de servicios web
- Referencia:** Jornadas Científico-Técnicas de Servicios Web (Congreso Español de Informática)
- Fecha:** 2005
- Lugar:** Granada, España
- ISBN:** 84-9732-455-2
15. **Autor(es):** M. García Rodríguez, D. Fernández Lanvin, A. A. Juan Fuente, **J. E. Labra Gayo**
- Título:** Desarrollo de aplicaciones multicanal aplicando MVC
- Referencia:** IADIS Iberoamericana WWW/2005
- Fecha:** 2005
- ISBN:** 972-8924-03-8
14. **Autor(es):** H. Sagastegui Chigne, M. E. Alva Obeso, **J. E. Labra Gayo**, D. Fernández Lanvin
- Título:** Modelo de control de acceso X-RBAC para un Framework de aprendizaje colaborativo
- Referencia:** IADIS Iberoamericana WWW/2005
- Fecha:** 2005
- ISBN:** 972-8924-03-8
13. **Autor(es):** **J. E. Labra Gayo**
- Título:** Graphical Representations and Infinite Virtual Worlds in a Logic and Functional Programming Course
- Referencia:** 12th International Workshop on Functional and (Constraint) Logic Programming, Federated Conference on Rewriting, Deduction and Programming
- Fecha:** 2003
- Lugar:** Valencia, España
- ISBN:** 84-96221-02-4
12. **Autor(es):** **J. E. Labra Gayo**, J. M. Cueva Lovelle
- Título:** Reusable Semantic Specifications of Programming Languages
- Referencia:** 6th Brazilian Symposium on Programming Languages
- Fecha:** 2002
- Lugar:** Rio de Janeiro, Brasil
11. **Autor(es):** **J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez, A. Cernuda del Río
- Título:** Reusable Monadic Semantics of Object Oriented Programming Languages
- Referencia:** 6th Brazilian Symposium on Programming Languages
- Fecha:** 2002
- Lugar:** Rio de Janeiro, Brasil
10. **Autor(es):** **J. E. Labra Gayo**, J. M. Gil, R. Turrado Camblor, A. Jarero Silva
- Título:** Development of a Generic e-Learning Multi-paradigm Programming Language System
- Referencia:** 14th International Workshop on the Implementation of Functional Languages (IFL'02)
- Fecha:** 2002
- Lugar:** Madrid, España

9. **Autor(es): J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez, A. Cernuda del Río
Título: Comparación de Técnicas de Especificación Semántica de Lenguajes de Programación
Referencia: Simposio Iberoamericano de Información e Ingeniería del Software en la Sociedad del Conocimiento
Fecha: 2001
Lugar: Bogotá, Colombia
8. **Autor(es): A. Cernuda del Río, J. E. Labra Gayo**, J. M. Cueva Lovelle
Título: Applying the Itacio Verification Model a Component-Based Real-Time Sound Processing System
Referencia: CLPSE'01: Constraint Programming & Logic Programming
Fecha: 2001
Lugar: Chipre
7. **Autor(es): J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez, A. Cernuda del Río
Título: Reusable Monadic Semantics of Logic Programs with Arithmetic Predicates
Referencia: APPIA-GULP-PRODE 2001, Joint Conference on Declarative Programming
Fecha: 2001
Lugar: Évora, Portugal
6. **Autor(es): J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez
Título: Modular Development of Interpreters from Semantic Building Blocks
Referencia: 12th Nordic Workshop on Programming Theory
Fecha: 2000
Lugar: Bergen, Noruega
5. **Autor(es): J. E. Labra Gayo**, J. M. Cueva Lovelle, M. C. Luengo Díez
Título: Language Prototyping using Modular Monadic Semantics
Referencia: III Latin-American Conference on Functional Programming
Fecha: 1999
Lugar: Recife, Brasil
4. **Autor(es): M. C. Luengo Díez, J. M. Cueva Lovelle, J. E. Labra Gayo**
Título: Desarrollo de Compiladores en un Sistema Integral Orientado a Objetos
Referencia: V Congreso Internacional de Investigación en Ciencias de la Computación
Fecha: 1998
Lugar: Aguas Calientes, Mexico
ISBN: 970-13-2261-4
3. **Autor(es): J. E. Labra Gayo**
Título: An Implementation of Modular Monadic Semantics using Folds and Monadic Folds
Referencia: Workshop on Research Themes on Functional Programming, III International Summer School on Advanced Functional Programming
Fecha: 1998
Lugar: Braga, Portugal
2. **Autor(es): J. E. Labra Gayo**, L. A. Oliveira Rodríguez, J. M. Cueva Lovelle
Título: Harmony: Sistema funcional de ayuda a la Composición Musical
Referencia: II Latin American Conference on Functional Programming
Fecha: 1997
Lugar: La Plata, Argentina

1. Autor(es): J. E. Labra Gayo**Título:** Haskell: Una Aproximación Funcional a la Programación Orientada a Objetos**Referencia:** II Jornadas sobre Tecnologías Orientadas a Objetos**Fecha:** 1996**Lugar:** Oviedo, España**8.9 Conferencias invitadas****11. Autor(es): J. E. Labra Gayo****Título:** Legislative data portals and linked data quality**Referencia:** HELDIG Digital Humanities Summit 2019**Participación:** Keynote Speaker**Fecha:** 2019**Lugar:** Helsinki, Finlandia**URL:** <https://www.helsinki.fi/en/helsinki-centre-for-digital-humanities/heldig-digital-humanities-summit-2019>**10. Autor(es): J. E. Labra Gayo****Título:** Validating RDF data: Challenges and perspectives**Referencia:** 1st Iberoamerican Conference on Knowledge Graphs and Semantic Web**Participación:** Keynote Speaker**Fecha:** 2019**Lugar:** Villa Clara, Cuba**URL:** <http://kgswc.org/>**9. Autor(es): J. E. Labra Gayo****Título:** Tecnologías semánticas para validar información sobre salud**Referencia:** XV Congreso ANIS (Asociación Nacional Informadores Salud)**Participación:** Invitado**Fecha:** 2019**Lugar:** Oviedo, España**URL:** <http://anisalud.com/actualidad/agenda/eventodetalle/12108/-/xv-congreso-anis>**8. Autor(es): J. E. Labra Gayo****Título:** Datos abiertos y enlazados: caminando hacia la web semántica**Referencia:** IV Foro distrital: Buenas prácticas en gestión de la información geográfica**Participación:** Invitado**Fecha:** 2015**Lugar:** Bogotá, Colombia**7. Autor(es): J. E. Labra Gayo****Título:** Datos abiertos enlazados y portales de datos**Referencia:** XVII Congreso de Estudiantes de Estadística: CONEEST**Participación:** Invitado**Fecha:** 2015**Lugar:** Puno, Perú**6. Autor(es): J. E. Labra Gayo****Título:** RDF Data shapes and WESO Research group**Referencia:** Desayuno con W3C: tu futuro en la Web**Participación:** Invitado

Fecha: 2015

Lugar: Madrid, España

URL: <http://w3c.es/Eventos/2015/Desayuno/>

5. Autor(es): J. E. Labra Gayo

Título: [Transparencia y liderazgo: datos abiertos y enlazados](#)

Referencia: Profesionales digitales: Jornadas puertas abiertas de la Universidad de Oviedo

Participación: Invitado

Fecha: 2014

Lugar: Mieres, España

4. Autor(es): J. E. Labra Gayo

Título: [Caminando hacia la web semántica: datos abiertos y enlazados](#)

Referencia: Workshop on semantic web and linked open data: Mexican International Conference on Computer Science

Participación: Keynote Speaker

Fecha: 2013

Lugar: Morelia, Mexico

3. Autor(es): J. E. Labra Gayo

Título: [Datos abiertos: reutilización de información en el sector público](#)

Referencia: II Jornadas de Empleo y Universidad

Participación: Invitado

Fecha: 2010

Lugar: Oviedo, España

2. Autor(es): J. E. Labra Gayo

Título: [¿Qué es la Web 2.0?](#)

Referencia: VIII Jornadas de Dinamización Económica. Instituto Leonés de Desarrollo Económico, Formación y Empleo (ILDEFE)

Participación: Invitado

Fecha: 2009

Lugar: León, España

1. Autor(es): J. E. Labra Gayo

Título: [Semantic Web and Image Information Mining](#)

Referencia: ESA-EUSC (European Space Agency-European Union Satellite Center) 2008: Image Information Mining: pursuing automation of geospatial intelligence for environment and security

Participación: Keynote Speaker

Fecha: 2008

Lugar: Frascati, Italia

URL: <http://wiki.services.eoportal.org/tiki-index.php?page=ESA-EUSC-2008>

8.10 Tutoriales impartidos en conferencias

9. Autor(es): J. E. Labra Gayo

Título: [Validating RDF data using Shapes](#)

Nombre congreso: International Semantic Intelligence Conference

Fecha: Feb. 2021

Lugar: Nueva Delhi, India – Virtual

URL: <https://www.ifis.uni-luebeck.de/~groppe/isic/2021/program>

8. Autor(es): J. E. Labra Gayo

Título: [Validating RDF using Shapes](#)

Nombre congreso: International Winter School at 2nd International Conference on Knowledge Graphs and Semantic Web

Fecha: Nov. 2020

Lugar: Mexico – Virtual

7. Autor(es): J. E. Labra Gayo

Título: [Introduction to ShEx/Entity Schemas](#)

Nombre congreso: Biohackathon Europe

Fecha: Nov. 2020

Lugar: Barcelona – Virtual

URL: <https://www.biohackathon-europe.org/>

Video: <https://www.youtube.com/watch?v=gnEaa9zHSos>

6. Autor(es): J. E. Labra Gayo

Título: [Shapes applications and tools](#)

Nombre congreso: International Semantic Web Conference - tutorial de 3 horas

Fecha: Oct. 2020

Lugar: Atenas, Grecia – Virtual

5. Autor(es): J. E. Labra Gayo

Título: [Data Models, Visualizations and ShEx for Wikicite](#)

Nombre congreso: WikiCite

Fecha: Nov. 2018

Lugar: Berkeley, California, USA

4. Autor(es): J. E. Labra Gayo

Título: [Validating RDF data](#)

Nombre congreso: International Semantic Web Conference - tutorial de 3 horas

Fecha: Oct. 2018

Lugar: Monterey – California, USA

3. Autor(es): E. Prud'hommeaux, J. E. Labra Gayo, T. Baker, A. Waagmeester, K. Thornton, L. Werksmeister, G. Supp

Título: [Testing data with the Shape Expressions Language](#)

Nombre congreso: Semantic Web Applications and Tools for Healthcare and Life Sciences - SWAT4HCLS

Fecha: 2018

Lugar: Antwerp, Bélgica

2. Autor(es): J. E. Labra Gayo, E. Prud'hommeaux, I. Boneva, H. Solbrig

Título: [RDF and Linked data validation](#)

Nombre congreso: Extended Semantic Web Conference

Fecha: 2016

Lugar: Creta, Grecia

1. Autor(es): E. Prud'hommeaux, J. E. Labra Gayo

Título: [ShEx by Example](#)

Nombre congreso: SWAT4LS: Semantic Web Applications and Tools for Life Sciences

Fecha: 2014
Lugar: Berlín, Alemania

8.11 Preprints

7. **Autor(es):** J. E. Labra Gayo, A. González Hevia, D. Fernández Álvarez, A. Ammar, D. Brickley, A. J. G. Gray, E. Prud'hommeaux, D. Slenter, H. Solbrig, S. A. Hosseini, B. Fünfkstük, A. Waagmeester, E. Willighagen, L. Ovchinnikova, G. Benjaminsen, R. García González, L. J. Garcia, D. Mietchen
Título: [Knowledge graphs and wikidata subsetting](#)
Clave: Preprint
Referencia: BioHackrXiv Preprints
Fecha: 2021
doi: <https://doi.org/10.37044/osf.io/wu9et>
6. **Autor(es):** L. J. Garcia, J. Bolleman, M. Dumontier, S. Jupp, J. E. Labra Gayo, T. Liener, T. Ohta, N. Queralt-Rosinach, C. Wu
Título: [Data validation and schema interoperability](#)
Clave: Preprint
Referencia: BioHackrXiv Preprints
Fecha: 2020
doi: <https://doi.org/10.37044/osf.io/8qdse>
5. **Autor(es):** A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutiérrez, J. E. Labra-Gayo, S. Kirrane, S. Neumaier, A. Polleres, R. Navigli, A. Ngonga, S. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann
Título: [Knowledge graphs](#)
Clave: Preprint
Referencia: arXiv preprint, <https://arxiv.org/abs/2003.02320>
Versión extendida del artículo 4
Fecha: 2020
4. **Autor(es):** I. Boneva, J. Dusart, D. Fernández Álvarez, J. E. Labra Gayo
Título: [Semi Automatic Construction of ShEx and SHACL Schemas](#)
Clave: Preprint
Referencia: arXiv preprint, <https://arxiv.org/abs/1907.10603>
Fecha: 2019
3. **Autor(es):** J. E. Labra Gayo, E. Prud'hommeaux, H. Solbrig, I. Boneva
Título: [Validating and describing linked data portals using shapes](#)
Clave: Preprint
Referencia: arXiv preprint, <https://arxiv.org/abs/1701.08924>
Fecha: 2017
2. **Autor(es):** I. Boneva, J. E. Labra Gayo, E. Prud'hommeaux, S. Staworko
Título: [Shape Expressions Schemas](#)
Clave: Preprint
Referencia: arXiv preprint, <https://arxiv.org/abs/1510.05555>

Fecha: 2015

1. **Autor(es):** I. Boneva, J. E. Labra Gayo, E. Prud'hommeaux
Título: Semantics and Validation of Shapes Schemas for RDF
Clave: Preprint
Referencia: arXiv preprint, <https://arxiv.org/abs/1404.1270>
Versión inicial artículo enviado a congreso 12

Fecha: 2014

8.12 Experiencia en organización/revisión de actividades de I+D

8.12.1 Participación en Consejo Asesor Internacional de Libros

3. **Título:** Intellectual Capital Strategy Management for Knowledge-Based Organizations
Editor(es): P. Ordóñez de Pablos, R. B. Tennyson, J-Y. Zhao
Editorial: IGI Global
ISBN: 9781466636552
Fecha: 2013
doi: <https://doi.org/10.4018/978-1-4666-3655-2>
Factor de impacto: Aparece en SCOPUS
2. **Título:** Strategic Role of Tertiary Education and Technologies for Sustainable Competitive Advantage
Editor(es): P. Ordóñez de Pablos, R. B. Tennyson
Editorial: IGI Global
ISBN: 9781466642331
Fecha: 2013
doi: <https://doi.org/10.4018/978-1-4666-4233-1>
Factor de impacto: Aparece en SCOPUS
1. **Título:** Advancing Information Management through Semantic Web Concepts and Ontologies
Editor(es): P. Ordóñez de Pablos, H. O. Nigro, R. B. Tennyson, S. E. Nisaro, W. Karwowski
Editorial: IGI Global
ISBN: 9781466621176
Fecha: 2013
doi: <https://doi.org/10.4018/978-1-4666-2116-9>
Factor de impacto: Aparece en SCOPUS y ACM Digital Library

8.12.2 Revisión de libros

3. **Título:** Global Hospitality and Tourism Management Technologies: Applications and Issues
Editor(es): P. Ordóñez de Pablos, R. B. Tennyson, J-Y. Zhao
Editorial: IGI Global
ISBN: 1-61350-041-6
Fecha: 2011
doi: <https://doi.org/10.4018/978-1-61350-041-5>
2. **Título:** Semantic Web Personalization and Context Awareness: Management of Personal Identities and Social Networking
Editor(es): M. D. Lytras, P. Ordóñez de Pablos, E. Damiani
Editorial: IGI Global
ISBN: 1615209212

Fecha: 2011

doi: <https://doi.org/10.4018/978-1-61520-921-7>

1. **Título:** [Digital Culture and E-Tourism: Technologies, Applications and Management Approaches](#)

Editor(es): P. Ordóñez de Pablo, M. D. Lytras, E. Damiani, L. Díaz

Editorial: IGI Global

ISBN: 1615209239

Fecha: 2010

doi: <https://doi.org/10.4018/978-1-61520-923-1>

8.12.3 Actividades de revisión de artículos

Miembro de Editorial Board

- Revista PLOS ONE, desde 2018
- Revista Scientific Programming, desde 2017

Revisión artículos en revistas JCR

Perfil en Publons: <https://publons.com/researcher/1348378/jose-emilio-labra-gayo/peer-review/>

- Revista Knowledge Based Systems, Elsevier, revisor 1 artículo.
- Revista: Applied Intelligence, JCR, revisor de 1 artículo.
- Revista: Sensors, JCR, revisor de 4 artículos.
- Revista: Data, JCR, revisor de 1 artículos
- Revista: Applied Sciences, JCR, revisor de 1 artículo.
- Revista: Plos One, JCR, revisor de 1 artículo.
- Revista Computer Standards and Interfaces, JCR, revisor de un artículo.
- Bioinformatics, revista JCR en SCI, Revisor de 1 artículo.
- Computers and Human Behaviour, JCR en SSCI, revisor de 5 artículos.
- Computers in Industry, JCR en SCI, revisor de 2 artículos y editor de un Special Issue.
- Revista Telematics and Informatics, JCR, Revisor de 1 artículo.
- Revista Expert Systems with Applications, JCR, Revisor de 1 artículo.
- Revista: Computing. Revisor de 2 artículos.
- Engineering Applications of Artificial Intelligence. Revisor de 2 artículos.
- Information Sciences, JCR en SCI, revisor de 1 artículo.
- International Journal of Universal Computer Science, revista JCR, revisor de 1 artículo.
- Science of Computer Programming, revista JCR en SCI, revisor de 2 artículos, 2013
- Semantic Web Journal, revista JCR en SCI (2015), revisor de 2 artículos.
- Web Semantics: Science, Services and Agents on the World Wide Web, JCR en SCI, revisor de 1 artículo.
- Journal of Web Semantics, revista JCR, revisor de 2 artículos.

8.12.4 Miembro del comité científico o de programa

61. **Título:** [The ACM Web Conference, Semantics and Knowledge track](#)
Lugar: Lyon – Francia
Fecha: 2021
Ámbito: : Internacional
60. **Título:** [IJCKG. International Joint Conference on Knowledge Graphs, research track](#)
Lugar: Online
Fecha: 2021

- Ámbito:** : Internacional
59. **Título:** ISWC. International Semantic Web Conference, research track
Lugar: Online
Fecha: 2021
Ámbito: : Internacional
58. **Título:** DaMaLOS. 2nd Workshop on Data and research objects management for Linked Open Science
Fecha: 2021
Ámbito: : Internacional
57. **Título:** 1st Workshop on Data and research objects management for Linked Open Science (DaMaLOS 20)
Fecha: 2020
Ámbito: : Internacional
56. **Título:** 4th Workshop on Quality of Open Data (QOD 2021)
Fecha: 2021
Ámbito: : Internacional
55. **Título:** JIST-KG 2020 (Joint International Conference on Knowledge Graphs 2020)
Fecha: 2020
Ámbito: : Internacional
54. **Título:** KGSWC 2020 (2nd Iberoamerican Knowledge Graph and Semantic Web Conference)
Fecha: 2020
Ámbito: : Internacional
53. **Título:** 3rd Workshop on Quality of Open Data (QOD 2029)
Fecha: 2020
Ámbito: : Internacional
52. **Título:** ECAI 2020 (24th European Conference on Artificial Intelligence)
Fecha: 2020
Ámbito: : Internacional
51. **Título:** ISWC. International Semantic Web Conference, research track
Fecha: 2019
Ámbito: : Internacional
50. **Título:** 1st Iberoamerican Knowledge Graph and Semantic Web Conference
Ámbito: : Internacional
Fecha: 2019
49. **Título:** 22nd International Conference on Business Information Systems
Ámbito: : Internacional
Fecha: 2019
48. **Título:** XV Jornadas de Ciencia e Ingeniería de Servicios
Ámbito: : Nacional
Fecha: 2019

47. **Título:** 2nd International Workshop on Quality of Open Data
URL: <http://bis.ue.poznan.pl/bis2019/qod/>
Ámbito: : Internacional
Fecha: 2019
46. **Título:** 22nd International Conference on Business Information Systems (BIS 2019)
URL: <http://bis.ue.poznan.pl/bis2019/>
Ámbito: : Internacional
Fecha: 2019
45. **Título:** 32st IEEE CBMS International Symposium on Computer-Based Medical Systems
URL: <http://www.cbms2019.org/>
Ámbito: : Internacional
Fecha: 2019
44. **Título:** XV JORNADAS DE CIENCIA E INGENIERÍA DE SERVICIOS (JCIS'19)
Ámbito: : Internacional
Fecha: 2019
43. **Título:** XIV JORNADAS DE CIENCIA E INGENIERÍA DE SERVICIOS (JCIS'19)
Ámbito: : Internacional
Fecha: 2018
42. **Título:** Jornadas de Enseñanza Universitaria de Informática
Ámbito: : Nacional
Fecha: 2018
41. **Título:** International Workshop on Ontology Patterns (WOP'17)
Ámbito: : Internacional
Fecha: 2017
40. **Título:** Workshop on Linked Enterprise Data Services, Provenance, Linking and QualitY - LEDSPLaY17
Ámbito: : Internacional
Fecha: 2017
39. **Título:** Linked Data Quality Workshop (LDQ 17)
Ámbito: : Internacional
Fecha: 2017
38. **Título:** Jornadas de Enseñanza Universitaria de Informática
Ámbito: : Nacional
Fecha: 2017
37. **Título:** 4th International Workshop on Semantic Web and Linked Open Data - ISW-LOD2016
Ámbito: : Internacional
Fecha: 2017
36. **Título:** International Workshop on Ontology Patterns (WOP16)
Ámbito: : Internacional
Fecha: 2016
35. **Título:** Semantics Conference
Ámbito: : Internacional

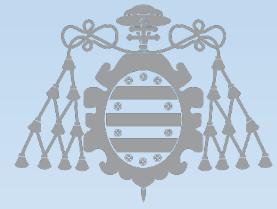
- Fecha:** 2016
34. **Título:** [Linked Data Quality Workshop \(LDQ 16\)](#)
Ámbito: : Internacional
Fecha: 2016
33. **Título:** [W3C Workshop on Multilingual Web](#)
Ámbito: : Internacional
Fecha: 2015
32. **Título:** [International Conference on Knowledge Engineering and Semantic Web](#)
Ámbito: : Internacional
Fecha: 2014
31. **Título:** [16th International Symposium on Computers in Education](#)
Ámbito: : Internacional
Fecha: 2014
30. **Título:** [Workshop on Ontology and Semantic Web Patterns \(5th edition\)](#)
Ámbito: : Internacional
Fecha: 2014
29. **Título:** [X Jornadas de Ciencia e Ingeniería de Servicios](#)
Ámbito: : Nacional
Fecha: 2014
28. **Título:** [Encuentro Nacional em Ciencias de la Computación](#)
Ámbito: : Nacional (México)
Fecha: 2014
27. **Título:** [Jornadas de Enseñanza Universitaria de Informática](#)
Ámbito: : Nacional
Fecha: 2014
26. **Título:** [Jornadas de Enseñanza Universitaria de Informática](#)
Ámbito: : Nacional
Fecha: 2013
25. **Título:** [12th International Semantic Web Conference - Evaluation Track](#)
Ámbito: : Internacional
Fecha: 2012
24. **Título:** [9th Extended Semantic Web Conference](#)
Ámbito: : Internacional
Fecha: 2012
23. **Título:** [Metadata and Semantics Research Conference, MTSR, 2012](#)
Ámbito: : Internacional
Fecha: 2012
22. **Título:** [Jornadas de Ciencia e Ingeniería de Servicios, JCIS 2012](#)
Ámbito: : Internacional
Fecha: 2012

21. **Título:** Social Data and Semantic Web, Workshop of the 10th International Conference on Semantic Web
Ámbito: : Internacional
Fecha: 2011
20. **Título:** Workshop on LInked Data en Conferencia Española de la Asociación para La Inteligencia Artificial (CAEPIA, 2011)
Ámbito: : Nacional
Fecha: 2011
19. **Título:** 7th Spanish Conference on Services Sciences and Engineering
Ámbito: : Nacional
Fecha: 2011
18. **Título:** 2º Taller sobre Ingeniería del Software y e-Learning (ISELEAR)
Ámbito: : Nacional
Fecha: 2011
17. **Título:** Social Data and Semantic Web, Workshop of the 10th International Conference on Semantic Web
Ámbito: : Internacional
Fecha: 2010
16. **Título:** 1er Taller sobre Ingeniería del Software y e-Learning (ISELEAR)
Ámbito: : Nacional
Fecha: 2010
15. **Título:** V Jornadas científico técnicas en Servicios Web y SOA
Ámbito: : Nacional
Fecha: 2009
14. **Título:** 13th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2009
13. **Título:** 12th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2008
12. **Título:** VIII Jornadas sobre Programación y Lenguajes
Ámbito: : Nacional
Fecha: 2008
11. **Título:** III Jornadas científico técnicas em Servicios Web y SOA
Ámbito: : Nacional
Fecha: 2007
10. **Título:** 11th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2007
9. **Título:** International Conference on Web Engineering
Ámbito: : Internacional
Fecha: 2006

8. **Título:** II Jornadas científico técnicas em Servicios Web
Ámbito: : Nacional
Fecha: 2006
7. **Título:** 10th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2006
6. **Título:** Jornadas Científico-Técnicas en Servicios Web
Ámbito: : Nacional
Fecha: 2005
5. **Título:** 9th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2005
4. **Título:** Fourth International Workshop on Language, Descriptions, Tools and Applications
Ámbito: : Internacional
Fecha: 2004
3. **Título:** 8th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2004
2. **Título:** International Conference on Web Engineering
Ámbito: : Internacional
Fecha: 2004
1. **Título:** 7th Brazilian Symposium on Programming Languages
Ámbito: : Internacional
Fecha: 2003

8.12.5 Miembro del comité organizador

4. **Título:** Jornadas de Enseñanza Universitaria de Informática
Ámbito: : Nacional
Fecha: 2014
Lugar: Oviedo, España
Ámbito: : Nacional
3. **Título:** International Conference on Web Engineering
Ámbito: : Internacional
Fecha: 2003
Lugar: Oviedo, España
2. **Título:** VI Congreso Español sobre Tecnologías y Lógica Fuzzy, ESTYLF 96
Ámbito: : Nacional
Fecha: 1996
Lugar: Oviedo, España
1. **Título:** II Jornadas sobre Tecnologías Orientadas a Objetos
Ámbito: : Nacional
Fecha: 1996
Lugar: Oviedo, España



Universidad de Oviedo

9. Tesis doctorales

9.1 Tesis doctorales dirigidas

4. **Programa Oficial:** Ingeniería Informática. Universidad de Oviedo
Doctor/a: Herminio García González
Título: Integración semántica de grandes fuentes de datos heterogéneas
Fecha Lectura: 25 de marzo de 2021
Calificación: Sobresaliente Cum Laude
Mención: Mención Internacional
Director/es: Jose Emilio Labra Gayo, Juan Manuel Cueva Lovelle

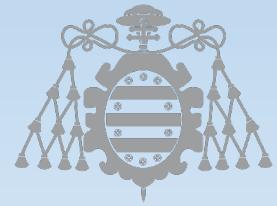
3. **Programa Oficial:** Ingeniería Informática. Universidad de León
Doctor/a: Jose Alberto Benítez Andrades
Título: Tecnologías Semánticas aplicadas al Análisis de Redes Sociales en el ámbito de la Salud
Fecha Lectura: 14 de noviembre de 2017
Calificación: Sobresaliente Cum Laude
Director/es: Jose Emilio Labra Gayo, Isaías García Rodríguez

2. **Programa Oficial:** Ingeniería Informática. Universidad de Oviedo
Doctor/a: Hernán Sagastegui Chigne
Título: Desarrollo de Tecnologías Colaborativas para Entornos virtuales de apoyo al aprendizaje
Fecha Lectura: 14 de diciembre de 2015
Calificación: Sobresaliente Cum Laude
Director/es: Jose Emilio Labra Gayo

1. **Programa Oficial:** Ingeniería Informática. Universidad de Oviedo
Doctor/a: Jose María Álvarez Rodríguez
Título: Métodos semánticos de reutilización de datos abiertos enlazados en las licitaciones públicas
Fecha Lectura: 14 de diciembre de 2012
Calificación: Sobresaliente Cum Laude
Director/es: Jose Emilio Labra Gayo

9.2 Tribunales de tesis doctorales

7. **Título:** A Framework Based on Web Services for Integrating Things into the Web of Things
Autor: Xavier Olvera González
Institución: Universidad Autónoma San Luis Potosí, México
Papel: Examinador técnico y miembro del tribunal evaluador
Fecha: 2017–2021
6. **Título:** Representing, Using and Maintaining Military Historical Linked Data on the Semantic Web
Autor: Mikko Koho
Institución: Aalto University School of Science
Papel: Opponent y Preliminary Examiner
Fecha: 2020
5. **Título:** Improving and Assessing Data Quality of Knowledge Graphs
Autor: Ben de Meester
Institución: IDLab, University of Ghent, Bélgica
Papel: Miembro del tribunal
Fecha: 2020
4. **Título:** A generic framework for integration of Cyber-physical Systems by use of intelligent systems
Autor: Andrii Berezovskyi
Institución: KTH – Royal Institute of Technology, Suecia
Papel: Midterm Phd Examnation
Fecha: 2020
3. **Título:** Metodología de medición y evaluación de la Usabilidad en Sitios Web educativos
Autor: María Elena Alva Obeso
Institución: Universidad de Oviedo, España
Papel: Miembro del tribunal de tesis
Fecha: 2005
2. **Título:** Servicios Web reflectivos y adaptables dinámicamente sobre servidores de aplicaciones soportados por máquinas virtuales
Autor: Javier Parra Fuente
Institución: Universidad Pontificia de Salamanca, Campus de Madrid, España
Papel: Miembro del tribunal de tesis
Fecha: 2005
1. **Título:** Clasificación de usuarios basada en la detección de errores usando técnicas de procesadores de lenguaje
Autor: Juan Ramón Pérez Pérez
Institución: Universidad de Oviedo, España
Papel: Miembro del tribunal de tesis
Fecha: 2006



Universidad de Oviedo

10. Estancias en centros extranjeros

9. **Centro de destino :** Universidad Técnica Federico Santa María

Lugar: Valparaíso y Santiago de Chile, Chile

Fecha: Estancias regulares de una a dos semanas desde 2005 hasta la actualidad todos los años excepto en 2020 debido a COVID

Tema: Profesor en el Magister de Tecnologías de la Información

Carácter de la estancia: Profesor invitado

8. **Centro de destino:** University of Malta

Lugar: Msida, Malta

Fecha: Sept. 2017

Propósito: Estancia a través del programa Erasmus+ Teaching Staff Mobility

7. **Centro de destino :** Utrecht University

Lugar: Utrecht, Holanda

Fecha: Curso 2012-2013, 1 año

Entidad financiadora: Programa de ayudas de movilidad a la excelencia, Universidad de Oviedo

Tema: Estancia de investigación - representaciones funcionales de grafos RDF

Carácter de la estancia: Invitado

6. **Centro de destino:** Leipzig University

Lugar: Leipzig, Alemania

Fecha: Sep.-Oct. 2012

Tema: Estancia de investigación - Web semántica

Carácter de la estancia: Invitado

5. **Centro de destino:** Charles University

Lugar: Praga, República checa

Fecha: Oct. 2012

Tema: Estancia de investigación - Web semántica

Carácter de la estancia: Invitado

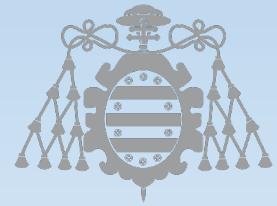
4. **Centro de destino:** Universidade do Minho

Lugar: Braga, Portugal

Fecha: 2002, Abril

Propósito: Estancia a través del programa Erasmus+ Teaching Staff Mobility

3. **Centro de destino:** Canisius College
Lugar: Búfalo, Nueva York
Fecha: Marzo–Abril, 2000
Propósito: Elaboración del Programa de Intercambio entre la Universidad de Oviedo y el Canisius College.
2. **Centro de destino :** Universidad Nacional de Luján
Lugar: Buenos Aires, Argentina
Fecha: Agosto-Sept. 1997
Tema: Programación Lógica y Funcional
Carácter de la estancia: Profesor invitado
Entidad financiadora: Programa de Cooperación Intercampus/AL.E., Agencia Española de Cooperación Iberoamericana
1. **Centro de destino :** Universidad Nacional de la Matanza
Lugar: Buenos Aires, Argentina
Fecha: Agosto-Sept. 1995
Tema: Computación lógica
Carácter de la estancia: Invitado como estudiante de doctorado
Entidad financiadora: Programa de Cooperación Intercampus/AL.E., Agencia Española de Cooperación Iberoamericana



Universidad de Oviedo

11. Actividad docente

11.1 Universidad de Oviedo

11.1.1 Asignaturas impartidas

Primer y segundo ciclos y grados

- **Arquitectura del Software**, Obligatoria de 3º del Grado en Ingeniería Informática del Software, Escuela de Ingeniería Informática de Oviedo. Se imparte desde el curso 2012-2013 que inicia su docencia con el nuevo plan de estudios hasta la actualidad.
- **Programación declarativa**, Optativa en la titulación de Ingeniería Técnica en Informática, Escuela de Ingeniería Informática de Oviedo. Se impartió de forma completa desde el curso 2004–2005 hasta el curso 2007–2008.
- **Programación lógica y funcional**, Optativa en la titulación de Ingeniería Técnica en Informática, Escuela de Ingeniería Informática de Oviedo. Se impartió de forma completa desde el curso 1994–1995 hasta el curso 2003–2004 en el cual desaparece por cambio de estudios y pasará a denominarse programación declarativa.
- **Lógica**, Obligatoria de primer curso en la titulación de Ingeniería Técnica en Informática, Escuela de Ingeniería Informática de Oviedo. Se impartió de forma completa desde el curso 1992–1993 hasta el curso 2007–2008.
- **Procesadores de lenguaje**, Obligatoria en la titulación de Ingeniero en Informática, Escuela Superior de Ingenieros Industriales e Ingenieros Informáticos de Gijón. Se impartieron 15 horas en el tema intérpretes desde los cursos 1996–1997 hasta 2004–2005
- **Tecnologías de la Información y Comunicación en Historia del Arte**, Optativa de cuarto curso en el Grado de Historia del Arte. Se impartieron 10 horas de introducción a metadatos desde el curso 2011–2012 hasta el curso 2019–2020
- **Computabilidad** Obligatoria de segundo curso, se imparte de forma completa el curso 1993–94 y de forma parcial (el tema de lógica) en el curso 2011–2012.
- **Metodología de la programación I**, obligatoria de primer curso. Se imparten las prácticas en el año 1992–1993.
- **Fundamentos de Informática**, obligatoria de primer curso en todas las ingenierías. Se imparte el curso 2013–2014.

Tercer ciclo y Máster

- **Web Semántica** asignatura obligatoria de primer curso del Máster Universitario en Ingeniería Web de la Universidad de Oviedo en la especialidad profesional. Se imparte desde el inicio del máster en

el curso 2006–2007 hasta la actualidad.

- **Nuevos avances en Web Semántica** asignatura obligatoria de primer curso del Máster Universitario en Ingeniería Web de la Universidad de Oviedo. Se imparte desde el inicio del máster en el curso 2006–2007 hasta la actualidad.
- **Web Semántica** curso de doctorado en el programa de doctorado del Departamento de Informática titulado Sistemas y Servicios Informáticos para Internet de la Universidad de Oviedo. Se imparte desde 2006–2007 hasta 2010–2011.
- **Lenguajes y estándares para la Web** asignatura obligatoria de primer curso del Máster Universitario en Ingeniería Web de la Universidad de Oviedo. Se imparte desde el inicio del máster en el curso 2006–2007 hasta el curso 2011–2012.
- **Programación Orientada a Objetos** asignatura obligatoria de primer curso del Máster Universitario en Ingeniería Web de la Universidad de Oviedo. Se imparte desde el curso 2014–2015 hasta el curso 2018–2019.

Tabla resumen de asignaturas impartidas

La siguiente tabla muestra las asignaturas impartidas. Se ha marcado con asterisco (*) las horas impartidas en inglés.

Curso aca- démico	Asignatura	Titulación	Curso	Horas teoría	Horas prácti- ca	Horas tota- les
2020- 2021	Arquitectura del Software	Grado de Ing. Informática del Software	3	21	-	21-19*
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
2019- 2020	Arquitectura del Software	Grado de Ing. Informática del Software	3	21-21*	14-67*	123
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	17,5	22,5
2018- 2019	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
	Arquitectura del Software	Grado de Ing. Informática del Software	3	21-21*	57-30*	129
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Programación Orientada a Objetos	Máster Univ. en Ingeniería Web	1	3.8	13.7	17.5
2017- 2018	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
	Arquitectura del Software	Grado de Ing. Informática del Software	3	25*	56-33*	114
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Programación Orientada a Objetos	Máster Univ. en Ingeniería Web	3,8	7,7	11,5	
2016- 2017	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
	Arquitectura del Software	Grado de Ing. Informática del Software	3	10	-	34-33*
					10*	87

Curso aca-démico	Asignatura	Titulación	Curso	Horas teoría	Horas prácti-ca	Horas tota-les
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Programación Orientada a Objetos	Máster Univ. en Ingeniería Web	3,8	7,7	11,5	
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
2015-2016	Arquitectura del Software	Grado de Ing. Informática del Software	3	36*	68 35*	- 87
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	10	15
	Programación Orientada a Objetos	Máster Univ. en Ingeniería Web	3,5	4	7,5	
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
2014-2015	Arquitectura del Software	Grado de Ing. Informática del Software	3	42*	116 (37*)	195
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	5	17,5	22,5
	Web Semántica	Máster Univ. en Ingeniería Web	1	5	17,5	22,5
	Programación Orientada a Objetos	Máster Univ. en Ingeniería Web	11,5	11	22,5	
	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	3,5		3,5	
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
2013-2014	Arquitectura del Software	Grado de Ing. Informática del Software	3	21	128	147
	Fundamentos de Informática	Grados de Ingeniería	1	42	108	150
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	10		10
	Web Semántica	Máster Univ. en Ingeniería Web	1	10		10
	Dirección y gestión de proyectos Web	Máster Univ. en Ingeniería Web	40		40	
	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	7		7	
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
2012-2013	Arquitectura del Software	Grado de Ing. Informática del Software	3	21	72	93
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5	5	10
2011-2012	Computabilidad	Grado de Ing. Informática del Software	2	10		10

Curso aca-démico	Asignatura	Titulación	Curso	Horas teoría	Horas prácti-ca	Horas tota-les
2010-2011	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	1	4	10	14
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Tecnologías de la Información y la Comunicación para Historia del Arte	Grado en Historia del Arte	4	5,5	10	
2010-2011	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	2		10	10
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	6	20	28
	Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Web Semántica	Máster Univ. en Sistemas y Servicios Informáticos para Internet	1	10	10	20
2009-2010	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	2		5	5
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Web Semántica	Máster Univ. en Ingeniería Web	1		20	20
	Servicios Web	Máster Univ. en Ingeniería Web	1	4,5		4,5
2008-2009	Web Semántica	Máster Univ. en Sistemas y Servicios Informáticos para Internet	1		20	20
	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	2	16		16
	Nuevos avances en Web Semántica	Máster Univ. en Ingeniería Web	1	10	20	30
	Web Semántica	Máster Univ. en Ingeniería Web	1		24	24
2007-2008	Servicios Web	Máster Univ. en Ingeniería Web	1	2,2		2,2
	Web Semántica	Programa doctorado: Sistemas y Servicios Informáticos para Internet	1	20		20
	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	15		15
	Programación declarativa	Ingeniero Técnico en Informática de Gestión/Sistemas	3	10	15	25
2006-2007	Arquitectura y diseño de sitios web	Máster Univ. en Ingeniería Web	2	2		2
	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	1	7		7
	Web Semántica	Máster Univ. en Ingeniería Web	1	21		21
	Servicios Web	Máster Univ. en Ingeniería Web	1	3		3
2006-2007	Web Semántica	Programa doctorado: Sistemas y Servicios Informáticos para Internet	1	20		20
	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	15		15
	Programación declarativa	Ingeniero Técnico en Informática de Gestión/Sistemas	3	10	5	15
	Arquitectura de la información en la Web	Máster Univ. en Ingeniería Web	2	12		12
	Lenguajes y estándares para la Web	Máster Univ. en Ingeniería Web	1	3,8		3,8

Curso aca- démico	Asignatura	Titulación	Curso	Horas teoría	Horas prácti- ca	Horas tota- les
	Web Semántica	Máster Univ. en Ingeniería Web	1	12		12
	Servicios Web	Máster Univ. en Ingeniería Web	1	12		12
	Web Semántica	Programa doctorado: Sistemas y Servicios Informáticos para Internet	1	20		20
2005- 2006	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	15		15
	Programación declarativa	Ingeniero Técnico en Informática de Gestión/Sistemas	3	7,5	7,5	15
	Desarrollo de aplicaciones en Sistemas Distribuidos e Internet	Ingeniero Técnico en Informática de Gestión/Sistemas	3	5		5
	Procesadores de lenguaje	Ingeniero en Informática	4	16		16
2004- 2005	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	15		15
	Programación declarativa	Ingeniero Técnico en Informática de Gestión/Sistemas	3	7,5	7,5	15
	Desarrollo de aplicaciones en Sistemas Distribuidos e Internet	Máster Univ. en Ingeniería Web	3	5		5
	Procesadores de lenguaje	Ingeniero en Informática	4	16		16
	Ingeniería Web	Programa doctorado: Avances en informática		10		10
2003- 2004	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	90	60	150
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
	Ingeniería Web	Programa doctorado: Avances en informática		10		10
2002- 2003	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	90	60	150
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
	Ingeniería Web	Programa doctorado: Avances en informática		10		10
2001- 2002	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	70	60	130
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	90	120
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
	Informática aplicada	Licenciado en Biología	1		10	10
2000- 2001	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	60	60	120
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	45	75
	Procesadores de lenguaje	Ingeniero en Informática	4	16		16
1999- 2000	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	30	30	60
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	45	75
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
1998- 1999	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	30	30	60

Curso aca- démico	Asignatura	Titulación	Curso	Horas teoría	Horas prácti- ca	Horas tota- les
1997- 1998	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	45	75
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	60	90	150
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
	Procesadores de lenguaje	Ingeniero en Informática	4	15		15
	Metodológicos de la programación	Ingeniero Técnico en Informática de Sistemas	2	4		4
1996- 1997	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	60	90	150
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
	Procesadores de lenguaje	Licenciado en Informática	4		15	15
1995- 1996	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	90	90	180
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
1994- 1995	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	90	90	180
	Programación lógica y funcional	Ingeniero Técnico en Informática de Gestión/Sistemas	3	30	30	60
1993- 1994	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	60	90	150
	Computabilidad	Ingeniero Técnico en Informática de Gestión/Sistemas	2	45	15	60
1992- 1993	Lógica	Ingeniero Técnico en Informática de Gestión/Sistemas	1	120		120
	Metodología de la programación I	Ingeniero Técnico en Informática de Gestión/Sistemas	1	120		120

11.1.2 Dirección de trabajos fin de titulación

Trabajos fin de máster - Máster de Ingeniería Web, Escuela de Ingeniería Informática de Oviedo

29. Alumno: Alejandro Montes Garcia, Tecnologías semánticas para recepción, edición y distribución de vídeo digital, 20/06/2013, Matrícula de Honor, 10
28. Alumna: María José Sánchez Doria, Modelando y consultando historias clínicas utilizando SNOMED CT y FHIR, Junio 2019, Aprobado
27. Alumno: Edita Hovhannisyan Topalyan, Validación Dinámica y Ejecución de Tareas Automatizadas en Bases de Datos Empresariales, Julio 2019, Sobresaliente
26. Alumno: Any Navarro Camero, Una propuesta de Métrica de Auditoría para la evaluación de la calidad de Datos Abiertos, junio 2018, Notable
25. Alumno Elwin Luis Huaman Quispe, Datos Abiertos Enlazados y el Turismo, Julio 2017, Notable
24. Ydalys Yaundeht Aponte Montañez, Análisis de Fuentes de datos Open Data y Big Data para el Mercado Inmobiliario, Julio 2017, Notable
23. Alumno: Herminio García González, LODLearning: Improving e-Learning by using Semantic Web, junio 2016, Sobresaliente

22. Daniel Fernandez Alvarez, MERA: Musical Entities Reconciliation Architecture (Arquitectura para la Reconciliación de Entidades Musicales), Julio 2015, Sobresaliente
21. Bernardo Martinez Garrido, Desarrollo de Aplicación Web para el reconocimiento de entidades musicales, Julio 2015, Sobresaliente
20. Alumno: Francisco Javier Collado Valle, Tecnologías de web semántica aplicadas al modelado del agarre de objetos, 17/07/2014, Sobresaliente, 9
19. Alumno: Alvaro Suárez Martínez, Aplicación web para la traducción de etiquetas asociadas a URIs, 22/07/2013, Notable, 8,5
18. Alumno: Cesar de Luis Alvargonzález, Acota: Tecnologías de etiquetado semiautomático y colaborativo, 22/06/2013, Sobresaliente, 9
17. Alumno: Sergio Sánchez Marcos, Herramienta para la gestión de la información en una empresa multinacional utilizando Cloud Computing, 24/07/2012, Sobresaliente
16. Alumno: Gonzalo Cruz Martínez, Web aumentada: plugin para scripts remotos, 23/07/2012, Notable,
15. Alumno: Weena Jiménez Nacero, Tecnologías de etiquetado colaborativo para la creación dinámica de ontologías, 15/02/2012, Matrícula de Honor
14. Alumno: Alejandro Álvarez Menéndez, Tecnologías de la web semántica para la creación de ontologías sobre competencias y habilidades, 22/07/2011, Sobresaliente, 9
13. Alumno: Pablo Abella Vallina, Redes semánticas colaborativas, 21/07/2011, Matrícula de Honor
12. Alumno: Sara Zapico Fernández, Personalización dinámica de sitios web mediante userscripts". La Web aumentada, 27/06/2011, Sobresaliente, 9
11. Alumno: Francisco Cifuentes Silva, Publicación de datos abiertos enlazados en el ámbito legislativo, 22/06/2011, Notable, 8,5
10. Alumno: Sergio Fernández López, Trío, estudio e implementación de modelos de datos RDF en lenguajes orientados a objetos, 28/07/2010, Sobresaliente, 9,5
9. Alumno: Manuel Álvarez Álvarez, Diseño y construcción de lenguajes de dominio específico asistido por ontologías, 28/07/2010, Matrícula de Honor,
8. Alumno: Carolina Chávez Bermeo, Análisis de la implantación de una taxonomía XBRL para la validación de informes de entidades financieras para una entidad reguladora de supervisión bancaria, 29/06/2010, Sobresaliente
7. Alumno: Sheila Méndez Núñez, Sistema reflectivo de inteligencia ambiental aplicado a la robótica, 15/07/2009, Matrícula de Honor 10, codirigido con Francisco Ortín Soler
6. Alumno: Iván Mínguez Pérez, Terra. técnicas de recomendación de recursos anotados semánticamente, 24/06/2009, Matrícula de Honor, 10
5. Alumno: Miguel García Rodríguez, AGUA: automatic grounding using annotations (automatización de grounding en servicios web semánticos mediante anotaciones, 09/01/2009, Sobresaliente
4. Alumno: Pablo Castaño Iglesias, Generador de código web accesible, 24/09/2008, Sobresaliente
3. Alumno: Noelia Rodríguez González, Desarrollo de un prototipo de plataforma para ofrecer e-learning a través de TDT, 25/06/2008, Sobresaliente 9,5
2. Alumno: Andrea Díaz Fernández, Gestor de contenidos para el mundo del motor, 24/09/2008, Sobresaliente
1. Alumno: Carlos Fernández Álvarez, Aplicación integral para la gestión interna y de clientes en medianas empresas en entorno web, 25/06/2008, Notable 8

Trabajos fin de grado - Grado en Ingeniería Informática del Software

17. Othmane Bakhtaoui, Sistema de sincronización entre ontologías e instancias de Wikibase, Julio, 2021
16. Jorge Álvarez Fidalgo, Integración de Shape Expressions y UML, Julio, 2020
15. Emilio Cortina Labra, Generador de preguntas a partir de grafo de conocimiento, Julio, 2020
14. Guillermo Facundo Colunga, Generación automática de modelos de dominio multilenguaje a través de un subconjunto de shape expressions, Julio, 2020, Sobresaliente propuesto para matrícula de honor, 10.
13. Astrid Gamoneda Arruñada, Buscador basado en principios SOLID, Nov. 2020
12. Ana Bravo Mediavilla, Página web con generación de problemas matemáticos, Julio, 2019, Notable
11. José Antonio García García, Estudio sobre Arquitecturas de Software Descentralizadas, Julio 2019
10. Tania Álvarez Díaz, Aplicación turística empleando tecnologías de web semántica, Julio, 2019, Sobresaliente
9. Sergio Faya Fernández, Sistema de análisis de contribuciones para GitHub, Julio 2019, Sobresaliente
8. Pablo Menéndez Suárez, Editor Visual de Shape Expressions, Julio 2019, Sobresaliente propuesto para matrícula de honor, 10.
7. Jorge Zapatero Sánchez, Bot conversacional que asiste a usuarios de un dominio dado, Julio, 2018, Sobresaliente
6. Santiago Marqués Rivera, Portal Web sobre la Constitución Española, Julio, 2018, Notable
5. David González García, Agregador de productos y comparador de precios, Escuela de Ingeniería Informática, 26/11/2017, Matrícula de Honor, 10
4. Alumno: Cristian Álvarez Belaustegui, Backend de un portal de datos e información sobre la tierra, Escuela de Ingeniería Informática – Oviedo, 23/07/2014, Matrícula de Honor, 10
3. Alumno: Herminio García González, API y visualizaciones de un sistema de información de la tierra, 23/07/2014, Sobresaliente, 10
2. Alumno: Ignacio Fuertes Bernardo, Wesearch: Generador automático de consultas basado en ontologías, Curso de Adaptación al Grado en Ingeniería Informática del Software para los Ingenieros Técnicos Informáticos de Sistemas, 23/07/2013, Sobresaliente, 9,5
1. Alumno: Alejandro Gálvez Marcos, Comunicación y Testing entre SAP y Documentum dentro del marco del proyecto Arcelor-Mittal Alyom, Curso de Adaptación al Grado en Ingeniería Informática del Software para los Ingenieros Técnicos Informáticos de Sistemas, Escuela de Ingeniería Informática – Oviedo, 23/07/2012, Sobresaliente, 9

Trabajos de investigación (doctorado) - Departamento de Informática

7. Alumno: Angel Retamar Arias, Avances para la Presentación de Información Aumentada sobre Páginas Web, Doctorado Sistemas y Servicios Informáticos para Internet, Departamento de Informática, 07/09/2011, Notable
6. Alumno: Sergio García Caso, Tecnologías de Cloud Computing y Tecnologías Semánticas Aplicadas a la Internet de las Cosas, Doctorado Sistemas y Servicios Informáticos para Internet, Departamento de Informática, 09/09/2010, Sobresaliente
5. Alumno: Diego Berrueta Muñoz, Caracterización de un Lenguaje de Transformación de RDF, Doctorado Avances en Informática, Departamento de Informática, 05/07/2007, Sobresaliente
4. Alumno: Doina Ana Cernea, La Semantica de los Recursos Multimedia para el Diseño de los Objetos del Aprendizaje, Doctorado Avances en Informática, Departamento de Informática, 05/07/2006, Sobresaliente
3. Alumno: Alberto Manuel Fernandez Alvarez, Tecnologías P2P, Doctorado Avances en Informática, Departamento de Informática, 13/07/2004, Sobresaliente

2. Alumno: Teobaldo Hernan Sagastegui Chigne, Tecnologias Web Colaborativas para el Apoyo a la Enseñanza, Doctorado Avances en Informática, Departamento de Informática, 18/09/2003, Sobresaliente
1. Alumno: Jose Manuel Morales Gil, "Aplicacion de Nuevas Tecnologías en la Evaluacion de Prácticas Desarrolladas", Doctorado Avances en Informática, Departamento de Informática, 23/09/2002, Notable

Proyectos fin de carrera - Ingeniero en Informática (Escuela Pol. de Ingenieros de Gijón)

18. Alumno: Jesús González Lorenzo, Procesamiento de datos semánticos mediante Mapreduce. Máster Universitario en Sistemas y Servicios Informáticos para Internet, 20/07/2011, Sobresaliente, 9
17. Alumno: Alejandra Fernández Martínez, Estudio de tecnologías de web semántica aplicadas a la gestión de una mediática, 22/12/2009, Matrícula de Honor, 10
16. Alumno: Miguel Martínez Álvarez, Semántica distribuida en la recuperación de información, 26/03/2009, Sobresaliente, 10
15. Alumno: Sandra Catalán García, Sistema de gestión web de los recursos turísticos del Principado de Asturias, 22/12/2008, Sobresaliente, 10
14. Alumno: Yolanda Pérez Martínez, Sistema inteligente de captación de datos a partir de múltiples fuentes RFID, 07/07/2008, Sobresaliente
13. Alumno: Victor Díaz Moriones, Extracción automática de semántica de imágenes basado en etiquetado colaborativo y ontologías, 01/07/2008, Matrícula de Honor
12. Alumno: Jose María Álvarez Rodríguez, Activación de conceptos en ontologías mediante el algoritmo de Spreading Activation, 05/11/2007, Sobresaliente
11. Alumno: Cecilia Garcia Pelayo, Herramienta de gestión financiera para la Web basada en XBRL, 29/03/2007, Sobresaliente
10. Alumno: Andrea Suárez Villar, Sistema para gestionar la grabación colaborativa de música por internet, 03/11/2006, Sobresaliente
9. Alumno: Iván Frade Ortea, Técnicas de activación de conceptos en una ontología a partir de una consulta, 03/10/2006, Sobresaliente
8. Alumno: Javier Martínez Álvarez, Sistema de Gestión y Configuración remota de Equipamiento de cliente, 26/06/2006, Sobresaliente
7. Alumno: Manuel Cañón López, Construcción de un sistema conversacional basado en servicios web para jugar al ajedrez, 14/04/2005, Sobresaliente,
6. Alumno: Diego Berrueta Muñoz, Clases de tipos en un lenguaje Lógico-funcional, 30/09/2004, Matrícula de Honor,
5. Alumno: Oscar Combarros Fernández, Desarrollo de un intérprete genérico mediante servicios web, 30/10/2003, Sobresaliente
4. Alumno: Roberto Turrado Cambor, Development of an IDE for computer aided instruction via web: problem manager, 17/10/2002, Sobresaliente
3. Alumno: Alfredo Jarero Silva, Sistema de enseñanza a través de internet de los fundamentos de la programación lógica, 28/06/2002, Sobresaliente,
2. Alumno: Isaac Prado Amores, Diseño e implementación de un sistema orientado a la creación de aplicaciones multimedia, 02/11/2000, Sobresaliente
1. Alumno: Luis Antonio Vinuesa Martínez, Sistema de cálculo topográfico y modelización digital del terreno, 05/02/1998, Sobresaliente,

Proyectos fin de carrera - Ing. téc. Informática de Sistemas - Escuela Ingeniería Informática

59. Alumno: Francisco Patallo Gago, Aplicación Web de gestión de clubes deportivos de base, Julio 2015, Notable
58. Alumno: Roberto Auro Zapico, Labels4all: Servicio web de etiquetado multilingüe, Julio 2015, Sobresaliente
57. Alumno: Jorge Yagüe París, Wesby: Visualizador genérico de datos enlazados mediante motor de plantillas, noviembre 2014, Sobresaliente
56. Alumno: Jesús Daniel Salas Campos, Experiencia profesional en sector TIC: desde el desarrollo a la dirección, 14/07/2014, Sobresaliente
55. Alumno: Pedro Pablo Viñas García, Diseño, integración y personalización de un módulo de gestión de proyectos para el ERP Integralia orientado a una empresa de innovación, 15/11/2012, Sobresaliente 9,5
54. Alumno: Héctor Álvarez López, Software de relación semántica de noticias, 15/11/2012, Aprobado, 6
53. Alumno: Andrés Vigil Rodríguez, Checking contradictions in UML specifications. 01/11/2011, Sobresaliente
52. Alumno: Aitor Rubio Díaz, Trivialpedia: sistema de preguntas tipo trivial basado en web semántica, 22/11/2010, Sobresaliente, 9
51. Tomas Suárez Lamadrid, Sistema de soporte a la decisión para recomendación semiautomática de medicinas, 30/08/2010, Sobresaliente, 9, co-dirigido con Alejandro Rodríguez González
50. Jessica Muñiz Fernández, Sistema de información de empleo mediante tecnologías de web semántica, 30/08/2010, Sobresaliente, 9
49. Sara García Pérez, Essences Finder sistema de búsqueda de imágenes combinando servicios web y ontologías, 30/08/2010, Notable, 8,5
48. Alumno: Héctor Alaiz Moretón, Sistema experto para sintonizado de PIDs basado en lenguajes de mercado, 22/07/2010, Matrícula de Honor, 10
47. Sheila Méndez Núñez, Análisis de documentos XBRL mediante tecnologías de web semántica, 04/09/2007, Sobresaliente, 9,5, co-dirigido con: Javier de Andrés Suárez
46. Alumno: José Barranquero Tolosa, Entorno web 2.0 para el diseño de diagramas de interconexión de objetos, 04/09/2007, Matrícula de Honor, 10
45. Alumno: Silvia Gonzalez Griñón, Rutoteca. gestión y comunicación entre usuarios, 07/03/2007, Notable, 7
44. Alumno: Guzmán González Menéndez, Juego P2P para controlar la evolución de una especie: interfaz gráfica y lógica del juego, 10/01/2007, Sobresaliente, 9
43. Alumno: María del Rosario Gavín Sierra, Sistema de comunicación para reuniones de trabajo colaborativo, 07/03/2007, Sobresaliente
42. Alumno: Carmen Suárez Fernández, Webbasierte Systeme, 10/01/2007, Notable 8, co-dirigido con: Frank Huch, Sebastian Fischeer
41. Alumno: Sergio Fernandez López, SWAML publicación de listas de correo en web semántica, 20/12/2006, Sobresaliente
40. Alumno: Alejandra Fernández Martínez, Servicio de validación de documentos EARL, 20/12/2006, Matrícula de Honor
39. Alumno: David José Brenes Martínez, Juego P2P para controlar la evolución de una especie. módulo de red y simulación, 19/09/2006, Sobresaliente,
38. Alumno: Miguel García Rodriguez, Infraestructura para PCC (Proof Carrying Code) sobre la plataforma .NET, 10/01/2006, Sobresaliente, co-dirigido con Francisco Ortín Soler

37. Alumno: Marta Segurola López, El color púrpura-sitio web de ayuda a mujeres maltratadas, 19/07/2006, Sobresaliente
36. Alumno: Victor Manuel González González, Interprete para Voice XML en español, 19/07/2006, Sobresaliente
35. Alumno: Iñaki Quesada Inclán, Rutoteca: biblioteca de rutas de montaña mediante sistemas de información geográfica, 05/07/2006, Sobresaliente
34. Alumno: Jose María Lago Aresti, Simulación de sistemas tácticos de baloncesto en un entorno web, 11/01/2006, Notable
33. Alumno: Jose Ramón Bobes Bascarán, Sistema gestor de contenidos aplicados a temas de salud mental, 21/12/2005, Sobresaliente
32. Alumno: Ángela Gutiérrez Ezama, Diseño de un sistema conversacional aplicado a la enseñanza, 20/07/2005, Sobresaliente
31. Alumno: Jose María Álvarez Rodríguez, Simulación de ejercicios de fitness en un entorno colaborativo, 12/01/2005, Sobresaliente, co-dirigido con: Carlos Rodríguez Álvarez
30. Alumno: Fernando Coll Fueyo, Sistema colaborativo de foros de discusión, 15/01/2004, Matrícula de Honor
29. Alumno: Micaela Prior Fernández, Herramienta para la gestión de prácticas de una asignatura a través de internet, 14/01/2004, Sobresaliente
28. Alumno: Iván Frade Ortea, Prolix: un intérprete de Prolog online, 29/09/2003, Sobresaliente
27. Alumno: Pablo Díez Vigil, Portal de información y servicios para la facultad de química con generación dinámica de contenidos, 24/09/2002, Matrícula de Honor
26. Alumno: Manuel Cañón López, Diseño de un sistema conversacional de ajedrez, 24/09/2002, Matrícula de Honor
25. Alumno: Elías Fernández-Combarro Álvarez, Sistema para la determinación de medidas difusas a partir de ejemplos usando algoritmos genéticos, 23/09/2002, Matrícula de Honor, Co-dirigido con: Pedro Miranda Menéndez
24. Alumno: Violeta Alonso Peláez, Las nuevas tecnologías en el ámbito de la pedagogía musical. aplicación al ritmo, 16/01/2002, Sobresaliente, Co-dirigido con: Juan Manuel Cueva Lovelle
23. Alumno: César Caban Fernández, Simulador de navegación por estuarios, 16/01/2002, Sobresaliente, Co-dirigido con: M. Reyes Poo Argüelles y Rafael García
22. Alumno: Ignacio Gonzalez Alonso, Democracia digital, 24/09/2001, Sobresaliente, codirigido con: J. A. López Brugos
21. Alumno: Carlos López Menéndez, Nuevas tecnologías aplicadas a la pedagogía musical, 17/01/2001, Matrícula de Honor, Codirigido con: Matilde Peiteado-Juan M. Cueva
20. Alumno: Ricardo González Campuzano, Paseo virtual por el edificio de la Facultad de Informática, 26/09/2000, Matrícula de Honor
19. Alumno: David Álvarez Velicia, Aplicación para la grabación, edición y mezcla de audio digital, 28/02/2000, Matrícula de Honor
18. Alumno: Fabián González Coto, Gestión Académica de la EUITIO, 24/09/1999, Sobresaliente
17. Alumno: Hugo Loredo Colunga, Gestión de Entidades Deportivas en Internet mediante el Lenguaje Funcional Haskell, 24/09/1999, Sobresaliente
16. Alumno: Eduardo García Galán, Sistema de Improvisación Musical, 25/09/1998, Sobresaliente tem Alumno: Alfredo Jarero Silva, Desarrollo de un Sistema Prolog con Control de la Pila de Objetivos, 24/09/1999, Sobresaliente
15. Alumno: Adrián Cabielles Merino, Algoritmos de Búsqueda en Árboles de Juego. Caso Práctico: Ajedrez, 25/07/1998, Sobresaliente

14. Alumno: Javier García Corrales, Guía Virtual de Oviedo, 24/06/1998, Sobresaliente
13. Alumno: Alejandro Gaitero González, Generación de Cuadro Horario, 18/10/1997, Sobresaliente
12. Alumno: Isaac Prado Amores, Método de Guitarra Eléctrica Multimedia, 18/10/1997, Sobresaliente
11. Alumno: Jose Angel Pedreira Junco, Caja de Ritmos, 18/10/1997, Sobresaliente
10. Alumno: Antonio Fernández Vidal, Reconocedor de Partituras, 10/06/1997, Sobresaliente
9. Alumno: Guillermo Rodríguez Díez, Guía de la Facultad de Ciencias, 10/02/1997, Sobresaliente
8. Alumno: F. J. Rodríguez Menéndez, Desarrollo de un Sistema de Formalización de Frases, 10/10/1996, Sobresaliente
7. Alumno: Jorge López Pelegrín, Sistema de Enseñanza de Lógica Asistido por Ordenador, 25/09/1996, Sobresaliente
6. Alumno: Jose Carlos López Miranda, Editor de Muestras Digitales Multipista, 25/09/1996, Sobresaliente
5. Alumno: Antonio Bautista León, Sistema para Reconocimiento Jerárquico de Zonas Poligonales a partir de un Conjunto de Líneas, 25/06/1996, Sobresaliente
4. Alumno: Almudena Areces Fernández, Sistema Interactivo de Ayuda a la Enseñanza de las Técnicas de Demostración Automática de Teoremas, 25/09/1995, Sobresaliente
3. Alumno: Jesús María Bernardo Llana, Aplicación Conversacional de Aprendizaje Adaptativo, 20/09/1994, Sobresaliente
2. Alumno: Francisco Ortín Soler, Sistema Interactivo de Dibujo Tridimensional, 20/09/1994, Sobresaliente
1. Alumno: Antonio Valdés Tronco, Documentación, Generación y Reconocimiento de Códigos de Barras, 20/09/1994, Sobresaliente

Proyectos fin de carrera - Ing. téc. Informática de Gestión - Escuela Ingeniería Informática

26. Alumno: Ignacio Fernández Pardo, Experiencia profesional en el desarrollo, mantenimiento e implantación de aplicaciones y servicios web para entidades bancarias, Julio 2016, Notable
25. Alumno: Raquel Curto Fernández, Sistema de publicación semántica de datos empresariales, Julio 2016, Notable
24. Alumno: Rhea María Bajk Garrán, Servicio web de ayuda a la gestión y monitorización de partes de incidencia para una empresa de prevención de riesgos laborales, 15/07/2012, Sobresaliente, 9
23. Alumno: Javier Alfonso Cendón, Sistema Biometrico de Autenticacion en el ámbito universitario, Escuela de Ingeniería Informática - Oviedo, 30/08/2010, Sobresaliente, 9
22. Alumno: Aurora Barrero López, XQSystems, una solucion empresarial para servicios avanzados TIC, 16/07/2008, Notable, 7,5
21. Alumno: Maríía Belén López Fernández, Sistema de ayuda a la enseñanza de demostraciones en lógica informática, 10/01/2007, Matrícula de Honor, 10, co-dirigido con: César Fernández Acebal
20. Alumno: Edita Hovhannisyan, Portal web del periódico digital Asturias con otro acento, Escuela de Ingeniería Informática - Oviedo, 04/09/2007, Sobresaliente, 9,5
19. Alumno: Pablo Figaredo Rubio, Sistemas de etiquetado de imágenes con información geográfica, Escuela de Ingeniería Informática - Oviedo, 11/01/2006, Sobresaliente
18. Alumno: Marta Martínez Valdés, Sistema de administración de reuniones para trabajo colaborativo, 08/03/2006, Matrícula de Honor, Co-dirigido con: Hernán Sagastegui
17. Alumno: Magdalena Llano Huidobro, Herramienta conversora de XBRL, 22/09/2005, Sobresaliente
16. Alumno: Gabriel Rodríguez García, Desarrollo de un sistema para la gestion de documentos, 09/03/2005, Sobresaliente,

15. Alumno: Rosa María Fernández Alfonso, Sistema de evaluación del esfuerzo de los estudiantes, 29/09/2004, Notable, co-dirigido con: Anders Sjögren
14. Alumno: Iván Maximino Velasco Martín, Diccionario tesoro multilingüe y colaborativo, 10/03/2004, Sobresaliente
13. Alumno: Diego García García, Ayudante de la navegación en la web: gestor de bookmarks, 29/09/2003, Matrícula de Honor
12. Alumno: Cecilia García Pelayo, Sistema de apoyo a la enseñanza mediante técnicas de realidad virtual, 29/09/2003, Sobresaliente
11. Alumno: Eduardo Valdés Rosillo, Sistemas para la realización de ejercicios mediante internet, 10/07/2003, Sobresaliente
10. Alumno: Jose Conde López, Aplicación web para la gestión de árboles genealógicos, 19/03/2003, Matrícula de Honor
9. Alumno: Enrique Gutiérrez Sans, Sistema de control de status remoto para terminales de autoservicio, Escuela de Ingeniería Informática - Oviedo, 24/07/2002, Matrícula de Honor
8. Alumno: Jose Manuel Basilio Pérez, Arte sacro en la villa de Cangas de Narcea, Escuela de Ingeniería Informática - Oviedo, 21/06/2001, Notable
7. Alumno: Joaquín Rubio Álvarez, Lenguaje de signos y palabra complementada, 28/07/1999, Matrícula de Honor
6. Alumno: Ana B. Menéndez del Valle, DESIGNER: Sistema Interactivo de Apoyo en la Enseñanza de Dibujo, 13/01/1998, Notable, co-dirigido con Jose Antonio López Brugos; P. Alvarez Peñín, Co-dirigido con Juan Antonio Alvarez González, Natalia Barrio Fernández, Diccionario Temático, 25/09/1998, Notable
5. Alumno: Mónica Freire Maira, Editor Interactivo de Música Polifónica, 18/09/1997, Sobresaliente
4. Alumno: Miguel Guerra Fernández, Informatización del Catálogo de Artistas Asturianos, 01/10/1996, Sobresaliente
3. Alumno: Jesús Daniel Díaz García, Métodos de Reconocimiento de Voz, 25/09/1996, Notable
2. Alumno: Luis Ángel Oliveira Rodríguez, Sistema experto de ayuda a la composición musical, 25/09/1996, Matrícula de Honor, 10
1. Alumno: Raquel Lombardía García, Gestión Integrada de Almacén de Componentes Eléctricos y Electrónicos para Calefacción, 05/10/1995, Sobresaliente

11.1.3 Coordinación de asignaturas

A continuación se indica las asignaturas de las que ha sido coordinador desde el curso 2008-2009 en el que se dispone de certificado de coordinación de docencia. Anteriormente, fue responsable de las asignaturas [Lógica](#) de primer curso, [Programación lógica y funcional](#) y [Programación declarativa](#), optativas de tercer curso en las titulaciones de Ingeniero técnico de Informática de Gestión y Sistemas de la Escuela Universitaria de Ingeniería Técnica en Informática de la Universidad de Oviedo.

Curso	Asignatura	Titulación
2020-2021	Arquitectura del Software	Grado en Ingeniería Informática del Software
	Nuevos avances en Web Semántica	Máster Universitario en Ingeniería Web
	Web Semántica	Máster Universitario en Ingeniería Web
2019-2020	Arquitectura del Software	Grado en Ingeniería Informática del Software
	Nuevos avances en Web Semántica	Máster Universitario en Ingeniería Web
	Web Semántica	Máster Universitario en Ingeniería Web

2018-2019	Arquitectura del Software Nuevos avances en Web Semántica Programación Orientada a Objetos Web Semántica	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2017-2018	Arquitectura del Software Nuevos avances en Web Semántica Programación Orientada a Objetos Web Semántica	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2016-2017	Arquitectura del Software Prácticas de empresa Proyecto fin de máster Web Semántica	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2015-2016	Arquitectura del Software Web Semántica	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web
2014-2015	Arquitectura del Software Programación Orientada a Objetos	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web
2013-2014	Arquitectura del Software Nuevos avances en Web Semántica Web Semántica	Grado en Ingeniería Informática del Software Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2012-2013	Arquitectura del Software Tecnologías de la Comunicación y de la Información aplicada a Historia del Arte Nuevos avances en Web Semántica Web Semántica	Grado en Ingeniería Informática del Software Grado en Historia del Arte Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2011-2012	Nuevos avances en Web Semántica Web Semántica	Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web
2010-2011	Nuevos avances en Web Semántica Web Semántica Web Semántica	Máster Universitario en Ingeniería Web Máster Universitario en Ingeniería Web Máster Universitario en Sistemas y Servicios Informáticos para Internet

11.2 Otras Universidades

6. **Título:** [Adding COVID Information to Wikidata](#)
Curso: CS520 – Knowledge Graphs
Lugar: Universidad de Stanford
Fecha: Abril 2021
URL curso: <https://web.stanford.edu/class/cs520/>
5. **Título:** [Sistemas y Aplicaciones Web](#)
Curso: Magister en Tecnologías de la Información
Lugar: Universidad Técnica Federico Santa María
Descripción: Curso de 16h impartido todos los años desde la primera edición del magister en el año 2005. El magister ha obtenido la acreditación de calidad por la Agencia Acredita CL
Fecha: Año 2005 – actualidad
URL: <https://www.mti.cl/>
4. **Título:** [RDF, Linked data and Semantic Web](#)
Curso: Master of Science in Artificial Intelligence
Lugar: Universidad de Malta
Fecha: Dic. 2017
3. **Título:** [Tecnologías XML y Web Semántica](#)
Curso: Máster Universitario en Dirección e Ingeniería de Sitios Web
Lugar: Universidad Internacional de la Rioja, España

Papel: Desarrollo de contenidos docentes

Fecha: 2014–2016

2. **Título:** [Web semántica: estado del arte, innovaciones futuras y XML avanzado](#)

Curso: Programa de doctorado en Ingeniería del Software

Lugar: Universidad Pontificia de Salamanca, Campus de Madrid

Descripción: Cursos de doctorado en el doctorado en Ingeniería del Software

Fecha: 2005–2007, 2006–2008

1. **Título:** [Generic Programming](#)

Curso: Escola de Engenharia

Lugar: Universidade do Minho, Portugal

Fecha: 2002

11.3 Contribuciones de carácter docente

11.4 Participación en congresos orientados a la formación y a la innovación docente universitaria

14. **Autor(es):** [J. E. Labra Gayo](#)

Título: [Bolonia en la Escuela de Ingeniería Informática](#)

Referencia: IV Jornadas de Orientación Universitaria – Bolonia en la Universidad de Oviedo

Fecha: 2011

13. **Autor(es):** J. R. Pérez Pérez, M. Paule Ruiz, D. Gayo Avello, [J. E. Labra Gayo](#), J. M. Redondo López, C. Fernández Acebal, M. Prado Vizoso

Título: [Empleo de Wikis como apoyo en desarrollo colaborativo de ejercicios](#)

Referencia: II Jornadas de Intercambio de Experiencias en Docencia Universitaria de la Universidad de Oviedo

Fecha: 2007

ISBN: 978-84-8317-689-4

12. **Autor(es):** Aquilino A. Juan, Javier de Andrés, Covadonga Nieto, Macamen Suárez, Juan Ramón Pérez, Agustín Cernuda, Candi Luengo, A. Belén Martínez, Miguel Riesco, Daniel F. Lanvin, [J. E. Labra Gayo](#), Marián D. Fondón, Jose Manuel Redondo

Título: [Definición de Competencias Específicas y Genéricas del Ingeniero en Informática.](#)

Editorial: ICE. Instituto de Ciencias de la Educación.

Fecha: 2006

ISBN: 84-88828-24-1

11. **Autor(es):** [J. E. Labra Gayo](#), D. Fernández Lanvin, J. Calvo Salvador, A. Cernuda del Río

Título: [Una experiencia de aprendizaje basado en proyectos utilizando herramientas colaborativas de desarrollo de software libre](#)

Referencia: XII Jornadas de Enseñanza Universitaria de Informática (JENUI)

Lugar: Teruel, España

Fecha: 2006

ISBN: 978-84-8317-689-4

10. **Autor(es):** M. Díaz Fondón, J. R. Pérez, M. Riesco Albizu, A. Juan Fuente, C. Nieto, A. Lubiano, C. Luengo, M. Suárez Torrente, D. Fernández Lanvín, [J. E. Labra Gayo](#), J. de Andrés, J. M. Redondo, R. García, A. M.S. Gibello

Título: [Reflexiones sobre el Proceso de Adaptación de asignaturas tras la realización de un proyecto piloto en la Escuela de Ingeniería Técnica Informática de Oviedo](#)

Referencia: Jornadas Nacionales de Intercambio de Experiencias Piloto de Implantación de Metodologías ECTS.

Lugar: Badajoz, España

Fecha: 2006

9. **Autor(es):** [J. E. Labra Gayo](#)

- Título:** Programación Declarativa utilizando XML, representaciones gráficas y mundos virtuales infinitos
Referencia: Simposio Nacional de Docencia en Informática (Congreso Español de Informática)
Lugar: Granada, España
Fecha: 2005
ISBN: 84-9732-443-9
8. **Autor(es): J. E. Labra Gayo**
Título: ¿Hay Lógica en la situación actual de las titulaciones informáticas en España?
Referencia: X Jornadas de Enseñanza Universitaria de la Informática
Lugar: Alicante, España
Editorial: Thomson
Fecha: 2004
ISBN: 84-9732-334-3
7. **Autor(es): J. E. Labra Gayo**
Título: Representaciones gráficas y mundos virtuales infinitos en las prácticas de programación lógica y funcional
Referencia: IX Jornadas de Enseñanza Universitaria de la Informática
Lugar: Cádiz, España
Editorial: Thomson
Fecha: 2003
ISBN: 84-283-2845-5
6. **Autor(es): H. Sagastegui, J. E. Labra Gayo, J. M. Cueva, J. M. Morales Gil, E. Alva Obeso, E. Valdés, C. García**
Título: Una componente e-Learning de aprendizaje colaborativo para el proyecto IDEFIX
Referencia: IX Jornadas de Enseñanza Universitaria de la Informática
Lugar: Cádiz, España
Editorial: Thomson
Fecha: 2003
ISBN: 84-283-2845-5
5. **Autor(es): J. E. Labra Gayo, J. M. Gil, R. Turrado Camblor**
Título: Desarrollo de un Entorno para la Enseñanza a Distancia de los Lenguajes de Programación
Referencia: International Symposium on Tele-education and Continuous Formation
Fecha: 2002
Lugar: La Habana, Cuba
4. **Autor(es): J. E. Labra Gayo, J. M. Gil, R. Turrado Camblor**
Título: Plataforma de enseñanza de lenguajes de programación a través de Internet: Proyecto IDEFIX
Referencia: VIII Jornadas de Enseñanza Universitaria de Informática
Fecha: 2002
Lugar: Cáceres, España
ISBN: 84-600-9782-X
3. **Autor(es): J. E. Labra Gayo, J. M. Gil, R. Turrado Camblor**
Título: Development of a Multi-paradigm Programming Language e-Learning System: IDEFIX Project
Referencia: International Conference on Information and Innovation Technologies in Education
Fecha: 2002
Lugar: Badajoz, España
2. **Autor(es): J. E. Labra Gayo, J. M. Gil, R. Turrado Camblor, A. Jarero Silva**
Título: Development of a Generic e-Learning Multi-paradigm Programming Language System
Referencia: 14th International Workshop on the Implementation of Functional Languages (IFL'02)
Fecha: 2002
Lugar: Madrid, España
1. **Autor(es): D. Gayo Avello, B. López Pérez, J. E. Labra Gayo**

Título: Desarrollo del portal Web de la E. U. de Ingeniería Técnica en Informática de Oviedo

Referencia: VII Jornadas de Enseñanza Universitaria de Informática

Fecha: 2001

Lugar: Palma de Mallorca, España

ISBN: 84-7632-657-2

11.5 Proyectos de innovación docente

6. **Título:** UBICamp: Integrated Solution to Virtual Mobility Barriers
Referencia: 526843-LLP-1-2012-ES-ERASMUS-ESMO
Entidad financiadora: Unión Europea: EAC/27/11. Action: Erasmus Multilateral Projects. Sub-Action: ERASMUS Mobility strategies and removal of barriers to mobility in higher education
Fecha: Oct. 2012–Mar. 2015
Coordinador: Aquilino Adolfo Juan Fuente
5. **Título:** Agenda electrónica del trabajo del estudiante
Referencia: C-2008
Coordinador: Juan Ramón Pérez Pérez
Fecha: 2008–2009
4. **Título:** Gestión de grupos de investigación en entornos virtuales apoyados en técnicas, recursos y objetos de aprendizaje
Referencia: EA-2005-0058
Entidad financiadora: Secretaría de Estado de Universidades e Investigación, Programa de Estudios y Análisis
Coordinador: María de los Ángeles Díaz Fondón
Fecha: 2005–2006
3. **Título:** Utilización de un entorno Web activo y colaborativo para la integración de recursos de aprendizaje
Referencia: PC-06-016
Entidad financiadora: Universidad de Oviedo
Coordinador: Juan Ramón Pérez Pérez
Fecha: 2006
2. **Título:** Incorporación asignatura *Lógica* al Aula Virtual
Referencia: PB-02-024
Entidad financiadora: Universidad de Oviedo
Coordinador: Jose Emilio Labra Gayo
Fecha: 2002
1. **Título:** Incorporación asignatura *Programación Lógica y Funcional* al Aula Virtual
Referencia: PA-01-000
Entidad financiadora: Universidad de Oviedo
Coordinador: Jose Emilio Labra Gayo
Fecha: 2001

11.6 Cursos y seminarios

11.6.1 Cursos impartidos

41. **Título:** Knowledge graphs and RDF validation
Centro: Bayer CropScience (Bélgica)
Fecha: Fecha prevista para Nov. 2021
Duración: 20h
40. **Título:** HTML5 y el Futuro de la Web
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: Julio 2013
Duración: 20h

39. **Título:** [HTML5 y el Futuro de la Web](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: Julio 2012
Duración: 18h
38. **Título:** [Cloud Computing, desarrollo de aplicaciones y minería Web](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: Julio 2010
Duración: 2h 30min
37. **Título:** [Desarrollo Web Avanzado](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2009/2010
Duración: 3h
36. **Título:** [Web Semántica y Tecnologías XML](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2009
Duración: 24h
35. **Título:** [Desarrollo de aplicaciones Web mediante Servicios Web y APIs abiertas](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2009
Duración: 3h
34. **Título:** [Creación de sitios Web con XHTML y CSS](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2009
Duración: 5h
33. **Título:** [El software libre, fundamentos e impacto en la sociedad](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2008/2009
Duración: 3h
32. **Título:** [Creación de sitios Web mediante hojas de estilo, usabilidad y accesibilidad](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2008/2009
Duración: 3h
31. **Título:** [Desarrollo de aplicaciones Web 2.0 con PHP](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2008/2009
Duración: 6h
30. **Título:** [Evolución de la Web](#)
Centro: Jornadas *Cuéntame cómo pasó: del ábaco a la Ingeniería Informática*, Escuela de Ingeniería Técnica Informática, Universidad de Oviedo
Fecha: Marzo 2009
Duración: 3h
29. **Título:** [Nuevos retos en la Web](#)
Centro: Cursos de verano, Universidad de Oviedo
Fecha: 2008
Duración: 8h
28. **Título:** [Tecnologías XML y Web Semántica](#)
Centro: Cursos de verano, Universidad de Oviedo
Fecha: 2007
Duración: 18h
27. **Título:** [Desarrollo de aplicaciones XML](#)

- Centro:** Centro del profesorado y recursos de Oviedo
Fecha: Junio 2007
Duración: 20h
26. **Título:** [XML, servicios Web y Web Semántica](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2006/2007
Duración: 19h
25. **Título:** [Tecnologías XML y Web Semántica](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2006
Duración: 16h
24. **Título:** [XML, servicios Web y Web Semántica](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2005/2006
Duración: 19h
23. **Título:** [Tecnologías XML y Web Semántica](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2005
Duración: 14h
22. **Título:** [XML, servicios Web y Web Semántica](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2004/2005
Duración: 18h
21. **Título:** [Tecnologías XML y Web Semántica](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2004
Duración: 14h
20. **Título:** [El Sistema de edición de documentos LATEX](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2004
Duración: 6h
19. **Título:** [Diseño de sitios Web mediante estándares](#)
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2004
Duración: 4h
18. **Título:** [XML, servicios Web y Web Semántica](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2003/2004
Duración: 21h
17. **Título:** [XML, servicios Web y Web Semántica](#)
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2002/2003
Duración: 16h
16. **Título:** [Internet, Curso 1º, cuatrimestre 2º](#)
Centro: Programa Universitario para Mayores, Universidad de Oviedo (PUMUO)
Fecha: 2003
Duración: 6h
15. **Título:** [Internet, Curso 2º, cuatrimestre 2º](#)
Centro: Programa Universitario para Mayores, Universidad de Oviedo (PUMUO)
Fecha: 2003

- Duración:** 3h
14. **Título:** Creación de sitios Web mediante hojas de estilo. Usabilidad, accesibilidad y arquitectura de la información
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2002/2003
Duración: 3h
13. **Título:** El sistema de edición de documentos L^AT_EX
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2003
Duración: 4h
12. **Título:** Programación de sitios Web con PHP
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2002/2003
Duración: 6h
11. **Título:** Plataforma .Net y Servicios Web
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2002
Duración: 2h
10. **Título:** Tecnologías XML para la estructuración del conocimiento en Internet
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 2002
Duración: 12h
9. **Título:** Representación de información en la Web: Hipertexto, XML y Web Semántica
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2001/2002
Duración: 9h
8. **Título:** Nuevos avances en Ingeniería Web
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: 2001/2002
Duración: 12h
7. **Título:** Técnicas Avanzadas en Desarrollo de Software en Internet
Centro: Cursos de Extensión Universitaria, Universidad de Oviedo
Fecha: Julio 2000
Duración: 2h
6. **Título:** Programación Orientada a Objetos en lenguaje C++
Centro: Cursos de Verano, Universidad de Oviedo
Fecha: 1998
Duración: 9h
5. **Título:** Informática básica para el empleo
Centro: Oficina de colocación y promoción al empleo, Universidad de Oviedo
Fecha: Dic. 1997 – Feb. 1998
4. **Título:** Programación Funcional en Lenguaje Haskell
Centro: Departamento de Ciencias Básicas, Universidad Nacional de Luján, Argentina
Fecha: Octubre 1997
Duración: 8h
3. **Título:** Programación Lógica y Funcional
Centro: Universidad Nacional de La Matanza, Buenos Aires, Argentina
Fecha: Sept. 1995
Duración: 20h
2. **Título:** Entorno de microordenadores – programación en lenguaje C

Centro: Departamento de Matemáticas de la Universidad de Oviedo en colaboración con el Instituto Nacional de Empleo

Fecha: Sept. 1991

Duración: 60h

1. **Título:** [Master de Diseño Asistido por Ordenador](#)

Centro: Departamento de Matemáticas de la Universidad de Oviedo

Fecha: Dic. 1990

Duración: 24h

11.6.2 Cursos recibidos

10. [Web of Science, nivel avanzado](#), Universidad de Oviedo, Fundación Española para la Ciencia y la Tecnología, 2015

9. Cursos en la Casa de las Lenguas, Universidad de Oviedo

- [Pronunciación y fonética correctiva del inglés](#), nivel B2, 2013
- [Recursos de expresión escrita para la docencia en inglés en el aula universitaria de Ingeniería Informática](#), nivel B2, 2012

8. Cursos del Programa de Formación para la Docencia Universitaria

- [Cómo atraer \(la atención\) hablando. Un reto para la enseñanza](#), 2011
- [Cómo elaborar y redactar un proyecto de investigación](#), 2009
- [Hacia el Espacio Europeo de Educación Superior](#), 2004
- [El trabajo en equipo como competencia y como estrategia de aprendizaje](#), 2003
- [Inglés académico para profesores universitarios: recursos de comunicación oral y escrita](#), 2003
- [Cómo afrontar el estrés laboral](#), 2003
- [¿Cómo motivar en el aula?](#), 2003
- [Cómo enseñar significativamente: un reto para el profesor](#), 2003
- [Prácticas sobre el discurso oral en el ámbito académico](#), 2002

7. [Curso de formación de los comités de evaluación](#), Plan Nacional de Evaluación de la Calidad de las Universidades, Universidad de la Rioja, 1999

6. [Principles, Logics and Implementations of Programming Languages \(PLI'99\)](#), París, Francia, 1999

5. [Third International Summer School on Advanced Functional Programming](#), Universidade do Minho, Braga, Portugal, 1998

4. Cursos de Perfeccionamiento Docente Universitario, impartidos por el Instituto de Ciencias de la Educación de la Universidad de Oviedo

- [La programación en el aula universitaria y el proyecto docente](#), Abril, 1996
- [La evaluación de aprendizajes](#), Enero, 1996
- [La metodología en el aula universitaria](#), Diciembre, 1995

3. Cursos de Lenguas Modernas impartidos por el Vicerrectorado de Estudiantes y Extensión Universitaria de la Universidad de Oviedo:

- [Inglés, nivel superior](#), Curso 2000–01
- [Inglés, nivel conversación](#), Curso 1994–95
- [Inglés 3-1](#), Curso 1993–94
- [Inglés 2-2](#), Curso 1990–91

2. Cursos organizados por la Asociación de Doctores, Licenciados e Ingenieros en Informática de Asturias:

- [Redes de computadores](#), Abril 1997
- [Diseño de pruebas de Software](#), Febrero 1996
- [Auditoría y Control de Sistemas de Información](#), Octubre 1995

1. Cursos impartidos por INTERGRAPH España S.A.

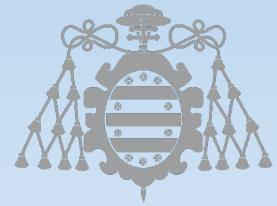
- [WTM](#) (Marzo 1991)
- [IPT](#) (Marzo 1991)

11.7 Material docente

9. **Título:** [Lenguajes, gramáticas y autómatas en Procesadores de Lenguajes](#)
Autor(es): J. M. Cueva Lovelle, R. Izquierdo, A. A. Juan Fuente, M. C. Luengo, F. Ortín Soler, **J. E. Labra**
Referencia: Cuaderno didáctico, Nº36
Editorial: Servitec
Fecha: 2003
8. **Título:** [Intérpretes y Diseño de Lenguajes de Programación](#)
Autor(es): **J. E. Labra**, J. M. Cueva Lovelle, R. Izquierdo, A. A. Juan Fuente, M. C. Luengo, F. Ortín Soler
Referencia: Cuaderno didáctico, Nº35
Editorial: Servitec
Fecha: 2003
7. **Título:** [Fundamentos de Informática General](#)
Autor(es): A. Cernuda, D. Gayo, J. M. Cueva, B. López, M. A. Díaz, L. Tajes, F. Álvarez, C. Carús, N. Cueva, M. Paule, J. A. Pérez, D. Álvarez, C. Luengo, N. García, M. González, J. R. Pérez, M. Riesco, **J. E. Labra**, A. B. Martínez
Referencia: Servicio de Publicaciones – Universidad de Oviedo
Fecha: 2000
6. **Título:** [Programación práctica en Prolog](#)
Autor(es): **J. E. Labra**
Referencia: Cuaderno didáctico Nº13
Editorial: Servitec
Fecha: 1998
ISBN: 84-8416-612-0
5. **Título:** [Lógica Proposicional Informática](#)
Autor(es): **J. E. Labra**
Referencia: Cuaderno didáctico Nº12
Editorial: Servitec
Fecha: 1998
ISBN: 84-8416-613-9
4. **Título:** [Ejercicios de Lógica Informática](#)
Autor(es): **J. E. Labra**, Ana I. Fernández Martínez
Referencia: Cuaderno didáctico Nº9
Editorial: Servitec
Fecha: 1998
ISBN: 84-8416-357-1
3. **Título:** [Introducción al lenguaje Haskell](#)
Autor(es): **J. E. Labra**
Referencia: Cuaderno didáctico Nº87
Editorial: Departamento de Matemáticas
Fecha: 1996
2. **Título:** [Lógica de predicados de primer orden](#)
Autor(es): **J. E. Labra**
Referencia: Cuaderno didáctico Nº77
Editorial: Departamento de Matemáticas
Fecha: 1994
1. **Título:** [Simulación Entorno Operativo Unix](#)
Autor(es): **J. E. Labra**, M. Riesco Albizu
Referencia: Cuaderno didáctico Nº39
Editorial: Departamento de Matemáticas
Fecha: 1992

11.8 Otros méritos docentes

9. Acreditación recibida para impartir docencia en inglés en los Grados Bilingües de la Universidad de Oviedo en 2013.
8. Presidente de la Comisión de Calidad de los Grados y Másteres Oficiales de la Escuela de Ingeniería Informática, año 2011–2012
7. Participación en los Encuentros sobre Calidad en Educación Superior, organizado por la ANECA, Universidad de La Laguna y Universidad de Oviedo, 2010
6. Participación en las Jornadas Erasmus Mundus, organizadas por la Universidad de Oviedo, 2010
5. Participación en Acción 2A, *Elaboración de Guías Docentes* en la convocatoria de Ayudas para el Desarrollo del Espacio Europeo de Educación Superior, curso 2005–2006, asínatura **Programación declarativa**
4. Participación en Acción 2A, *Elaboración de Guías Docentes* en la convocatoria de Ayudas para el Desarrollo del Espacio Europeo de Educación Superior, curso 2005–2006, asínatura **Lógica**
3. Participación en las jornadas Microsoft Academic Days, organizadas por Microsoft, 2005 en Lisboa (Portugal)
2. 2003, participación en el 34th Technical Symposium on Computer Science Education (SIGCSE 2003), en Reno, Nevada, USA
1. 2000, Miembro de la Comisión de Autoevaluación de la Escuela Universitaria de Ingñiería Técnica Informática de Oviedo



Universidad de Oviedo

12. Actividades de Gestión

12.1 Ámbito académico

12.1.1 Órganos unipersonales

3. **Cargo académico:** Director

Institución: Escuela Ingeniería Informática, Universidad de Oviedo.

Inicio: 16/06/2004

Fin: 26/04/2012

2. **Cargo académico:** Secretario académico

Institución: Escuela Universitaria de Ingeniería Técnica Informática, Universidad de Oviedo

Inicio: 19/07/1996

Fin: 09/07/2000

1. **Cargo:** Director

Institución: Cátedra DXC Technology

Inicio: 19/07/2018

12.1.2 Órganos colegiados

9. 2017 – actualidad, miembro del **Claustro** de la Universidad de Oviedo

8. Miembro del **Claustro** de la Universidad de Oviedo desde 7 de junio de 2002 hasta 29 mayo de 2008 en el colectivo de profesores doctores.

7. 2008 – 2012, miembro del **Consejo de Gobierno** de la Universidad de Oviedo

6. 2008 - 2012, miembro de la **Comisión de Asuntos Económicos, Infraestructuras e Informática** del Consejo de Gobierno de la Universidad de Oviedo

5. Desde Marzo de 2002 hasta 2012, miembro de la **comisión de doctorado** del Departamento de Informática

4. 1997 - 2002, Miembro de la **Comisión de Docencia** del Departamento de Informática de Oviedo

3. 1996 - 2000 **Coordinador del Programa ECTS** (European Credit Transfer System) en la Escuela de Ingeniería Informática de la Universidad de Oviedo

2. 1996 al 2000: Ponente de la **Comisión Asesora de Convalidaciones** de la Universidad de Oviedo

1. Desde 1992, Miembro de la Junta de Escuela de la Escuela de Ingeniería Informática de la Universidad de Oviedo

12.1.3 Colaboración con Organismos de Evaluación

8. Evaluador remoto de proyectos europeos en la convocatoria: EISMEA HORIZON-EIC Pathfinder Open 2021.
7. Evaluador expedientes en convocatoria Atracción de talento Investigador 2020, Comunidad de Madrid, área TIC.
6. Evaluador de proyectos del Fondo de Cohesión estructural Europeo, programa 2014–2020, acción 1.1.1.1 *Industry Driven Research*, Letonia, 2020.
5. Evaluador remoto de proyectos europeos en convocatoria *FET Open – Novel ideas for radically new technologies (H2020-FETOPEN-2018-2020)*, 2019
4. Evaluador de proyectos del Fondo de Cohesión estructural Europeo, programa 2014–2020, acción 1.1.1.1 *Industry Driven Research*, Letonia, 2018.
3. Miembro del Panel para la revalidación del *Computing Phd Programme* del *Cork Institute of Technology*, 2013.
2. Jurado del Concurso Universitario de Ideas Empresariales para la selección del proyecto representante de la Universidad de Oviedo al Premio Uniprojecta Portal Universia y al Premio Uniprojecta Fundación Universia, 2011 y 2013.
1. Miembro del grupo de expertos del programa europeo EQANIE (European Quality Assurance Network for Informatics Education): <https://eqanie.eu/>

12.2 Coordinador acuerdos internacionales

6. **Institución:** University of Malta
Lugar: Sliema, Malta
Tipo: Acuerdo Erasmus
Participación: Coordinador
Fecha: 2017 – Actualidad
5. **Institución:** Kungliga Tekniska Högskolan
Lugar: Estocolmo, Suecia
Tipo: Acuerdo Erasmus nº 437
Participación: Coordinador
Fecha: Oct. 2002 – Sept. 2006
4. **Institución:** Universidade do Minho
Lugar: Braga, Portugal
Tipo: Acuerdo Erasmus nº 376
Participación: Coordinador
Fecha: Oct. 1999 – Sept. 2005
3. **Institución:** Christian-Albrechts Universität zu Kiel
Lugar: Kiel, Alemania
Tipo: Acuerdo Erasmus nº 436
Participación: Coordinador
Fecha: Oct. 2002 – Sept. 2005
2. **Institución:** University of Pécs
Lugar: Hungría
Tipo: Acuerdo Erasmus nº 1056
Participación: Coordinador
Fecha: Oct. 2012 – Sept. 2015
1. **Institución:** Canisius College

Lugar: Buffalo, Nueva York, USA
Tipo: Responsable del Convenio de Cooperación
Participación: Responsable
Fecha: Oct. 1999 – Oct. 2004

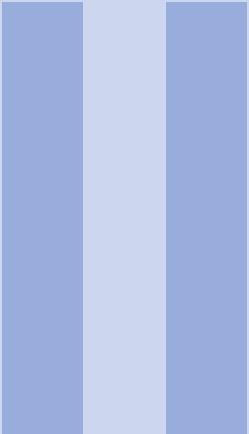
12.3 Gestión de la investigación

7. 2018 – actualidad, miembro del Editorial Board de la revista PLOS One, Factor de impacto, 2.806 (Q1)
6. 2018 – actualidad, miembro del Editorial Board de la revista Scientific Programming, Factor de impacto: 0.627, Q4
5. 2004 - actualidad, fundador e investigador principal del grupo WESO (Web Semantics Oviedo), el cual ha obtenido evaluaciones positivas como grupo oficial de investigación por parte de la Agencia Estatal de Investigación en el año 2012 y en el año 2018.
4. 2017 – 2020: Miembro del grupo de trabajo del W3C Web of Things
3. 2004 – 2017 Miembro del grupo de trabajo del W3C Data Shapes
2. Desde 2013, chairman del W3C Community group sobre Best Practices for Multilingual Linked Open Data.
1. Desde 2017 Miembro del W3C Community Group on Shape Expressions. Editor de la especificación semántica del lenguaje.

12.4 Becas, Premios y reconocimientos

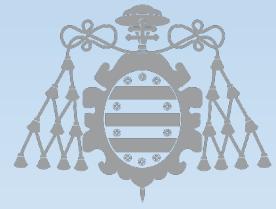
10. Valoración positiva 3 tramos de Investigación (sexenios): Comisión Nacional Evaluadora de la Actividad Investigadora. Periodos: 1995-2001, 2003-2013, 2014-2019
9. 2019, Best Paper Award, *Using Shape Expressions (ShEx) to Share RDF Data Models and to Guide Curation with Rigorous Validation*, Katherine Thornton, Harold Solbrig, Gregory Stupp, Jose Emilio Labra Gayo, Daniel Mietchen, Eric Prud'Hommeaux, Andra Waagmeester, In Extended Semantic Web Conference, ESWC2019. Lecture Notes in Computer Science - 2019. Eslovenia. Conferencia tipo A según índice GGS.
8. 2014, Best Paper Award, 10th International Conference on Semantic Systems. Paper: *Shape Expressions: an RDF Validation and transformation language*. Leipzig, Alemania. Conferencia tipo C según índice GGS
7. Keynote speaker en:
 - 1st Iberoamerican Conference on Knowledge Graphs and Semantic Web, Cuba, 2019
 - 11 Congreso Internacional de Ingeniería del Software, Sistemas de Información y Tecnologías de la Información, Perú, 2014
 - 1st Workshop on Semantic Web and Linked Open Data, Morelia, México. 2013
 - ESA-EUSC Conference on Image Information Mining, Roma, Frascatti, 2008
6. Beca para participar en los siguientes eventos:
 - Biohackathon Europe, Líder proyecto: Handling Knowledge graph subsets, 8–12 Nov. Barcelona, 2021
 - Biohackathon Europe, Líder proyecto: Knowledge graphs and wikidata subsetting, 9–13 Nov. Barcelona (virtual), 2020
 - Biohackathon, RDFShape–Cytoscape, 1–7 Sep. Fukuoka, Japón, 2019
 - Wikicite, 27–29 Nov. Berkeley, Estados Unidos, 2018
 - Wikibase summit, 19–21 Sep. Nueva York, Estados Unidos, 2018
5. 2005 Premio Sociedad de la Información de Asturias, 2005 al proyecto *Extensión del estandar CSS3 que permita la adaptación multidipositivo de contenidos web*

4. 2001 Premio Extraordinario de Doctorado del Departamento de Informática de la Universidad de Oviedo
3. 1999 Beca TMR (Training and Mobility of Researchers) de la Comunidad Europea (Actividad 4) para la asistencia al congreso Principles, Logics and Implementations of High-Level Programming Languages (PLI'99), París, 27 Sept. - 1 Oct. 1999
2. Beca del Programa de Cooperación Intercampus/ALE. 97 sobre [Programación Lógica y Funcional](#), Universidad Nacional de Luján, Buenos Aires, Argentina
1. Beca del Programa de Cooperación Intercampus/ALE. 95 sobre [Computación Lógica](#), Universidad Nacional de La Matanza, Buenos Aires, Argentina, 15 de Agosto a 15 de Septiembre de 1995



Proyecto Docente

13	Prefacio	99
14	Entorno	101
15	Concepto y Referencias Curriculares	125
16	Metodología docente	151
17	Propuesta docente	165



Universidad de Oviedo

13. Prefacio

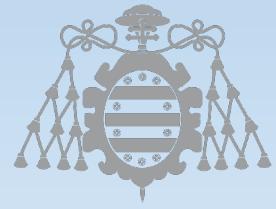
Este documento describe el Proyecto Docente cumpliendo con las directrices que rigen los concursos de acceso a las plazas de Catedrático de Universidad en la Universidad de Oviedo, y las Bases de la convocatoria de la resolución de 25 de Junio de 2021 de la Universidad de Oviedo por la que se convoca concurso de acceso a plazas de cuerpos docentes universitarios (Boletín Oficial del Estado del 25 de Junio de 2021).

El presente proyecto presenta una propuesta de programa para la asignatura *Arquitectura del Software*, siguiendo las decisiones del Consejo de Gobierno de la universidad de Oviedo del 28 de Mayo de 2021 donde se acordó que la actividad docente asociada a esta plaza se correspondería con el perfil de: *Arquitectura del software*, asignatura obligatoria de tercer curso del Grado en Ingeniería Informática del Software que se imparte en el primer semestre en la Escuela de Ingeniería Informática de la Universidad de Oviedo y cuyos descriptores se corresponden con el plan de estudios de 2011.

El capítulo 14 describe el contexto en el que se va a desarrollar la propuesta: el Espacio Europeo de Educación Superior (EEES), la Universidad de Oviedo, el departamento de Informática y la Escuela de Ingeniería Informática de esta universidad.

En el capítulo 15 se presentan unas consideraciones iniciales sobre la arquitectura del software y su evolución, seguida de unas referencias curriculares y revisión de cómo se imparte la disciplina en otras asignaturas internacionales y nacionales.

Le sigue el capítulo 16 en el que se repasan aspectos de metodología didáctica que se han considerado relevantes para la docencia de esta asignatura, finalizando con el capítulo 17 en el que se describe la propuesta docente concreta, que se plasma en una revisión de las competencias y resultados de aprendizaje, seguida de la descripción de las principales actividades de aprendizaje y la evaluación de la asignatura. También se incluye una revisión bibliográfica de los principales libros que se han considerado para la docencia, seguido de un anexo en el que se incluye la guía docente propuesta para la asignatura por el autor de esta memoria para el curso 2020-21.



Universidad de Oviedo

14. Entorno

En este capítulo se contextualiza la materia contemplada en este proyecto docente. Se comienza describiendo brevemente el contexto educativo general. Posteriormente se describe la Universidad de Oviedo para finalizar con el estudio de Grado en el que se imparte esta asignatura en la Universidad de Oviedo: Graduado en Ingeniería Informática del Software por la Universidad de Oviedo. Esta titulación ha sido verificada por el Consejo de Universidades, previo informe favorable de la ANECA en consonancia con los objetivos establecidos para la construcción del *EEES*.

14.1 Concepto de Universidad y Espacio Europeo de Educación Superior

El concepto de Universidad surge en Europa en los siglos XII y XIII. Entre las Universidades más antiguas destacan las de Bolonia, París, Oxford y Salerno. En un principio, las Universidades constaban generalmente de cuatro facultades: Artes, Leyes, Medicina y Teología [74].

Dichas facultades han sobrevivido, y se han visto complementadas con numerosas disciplinas, particularmente de Ciencias Sociales y Estudios Técnicos. La Universidad permaneció relativamente estable hasta el s. XVIII, en el que se realizan varias transformaciones por la influencia de la Ilustración. Se suelen identificar varios modelos universitarios, entre los que se puede destacar el modelo francés y el modelo alemán. El modelo francés se suele asociar con Napoleón y se caracterizaba por instituciones con un fuerte control centralizado por el estado. El modelo alemán, relacionado con Wilhelm von Humboldt promulgaba que la enseñanza debía venir guiada por la investigación actual, y que ésta debía ser imparcial e independiente de influencias religiosas, políticas, económicas o ideológicas [76].

Ortega y Gasset en 1930 vaticinaba las dos funciones principales de la Universidad [37]:

- La enseñanza de profesiones intelectuales.
- La Investigación científica y la preparación de futuros investigadores.

En el catálogo World Higher Education Database realizado Asociación Internacional de Universidades, existen unas 20,000 instituciones de educación superior repartidas en 196 países con distintos grados de calidad académica e investigadora¹.

El Espacio Europeo de Educación Superior (EEES) comenzó a gestarse a finales del siglo XX en la Unión Europea con el objetivo de facilitar un reconocimiento más sencillo de las titulaciones y mejorar la formación de los estudiantes y su integración en un mercado laboral unificado y sin fronteras dentro de la Unión Europea.

¹<http://www.whed.net/>

El factor desencadenante de esta actuación en el ámbito educativo han sido los programas de movilidad de estudiantes que fueron una iniciativa de la Comisión Europea para favorecer el desarrollo de una verdadera dimensión cultural de la Unión Europea. Estos programas determinaron la necesidad de encontrar un sistema adecuado de equivalencias y reconocimiento de estudios. La demanda de procedimientos eficaces para reconocer los estudios originó el Sistema Europeo de Transferencia de Créditos o ECTS. Este sistema pretende facilitar la transferencia y el reconocimiento de los períodos cursados en otros países. Se creó un método de trabajo que se basaba en la utilización de principios compartidos (transparencia, confianza mutua, importancia del trabajo del estudiante) y documentos con formatos normalizados.

Se puede considerar que el EEES comenzó formalmente a gestarse con la *Declaración de La Sorbona*² (25 de mayo de 1998) donde los Ministros de Educación de Francia, Alemania, Italia y Reino Unido se comprometieron en crear un marco de referencia común para la educación superior en Europa, con el objeto de promover la contratación y movilidad de los ciudadanos europeos. Un año después, el 19 de junio de 1999, 29 países europeos entre los que ya se encontraba España, ratificaron este compromiso en la *Declaración de Bolonia*³, cuya meta era crear el EEES en 2010. A partir de la Declaración de Bolonia, la Conferencia de Universidades Europeas de Salamanca, la Conferencia de Estudiantes Europeos en Goteborg (Suecia), y la Conferencia Ministerial de Praga, celebradas en 2001, y la Conferencia de Berlín en 2003 (con 40 países participantes) se sentaron las bases del EEES tal y como se conoce actualmente.

De forma bianual se celebraron Conferencias Ministeriales con el objetivo de realizar un seguimiento de los avances conseguidos y establecer los objetivos y las prioridades para los dos años siguientes. Finalmente, en 2010 se celebró una reunión conjunta en Budapest y Viena en la que se da comienzo, de manera oficial, al EEES o European Higher Education Area (EHEA)⁴.

Dos años más tarde, en abril de 2012, los ministros de educación superior de cuarenta y siete países europeos se reúnen en Bucarest para llegar a un acuerdo sobre las reformas para crear un EEES modernizado y abierto. Los ministros acordaron que las reformas deben centrarse en desarrollar la capacidad no aprovechada de la educación superior para contribuir al crecimiento y la empleabilidad.

Los ejes en los que se centra el EEES son los siguientes:

- Construcción de un espacio común de conocimiento e investigación que se basa en:
 - Promoción de un sistema de títulos comprensibles y comparables.
 - Generar una estructura educativa similar y comparable, organizando la enseñanza basada en dos ciclos principales, dando el último acceso al doctorado.
 - Desarrollo de un sistema de créditos que permita la movilidad de los estudiantes, no solo en el ámbito de la formación universitaria, también en el aprendizaje a lo largo de la vida.
 - Promover de la movilidad de estudiantes y profesorado, construyendo para ello instrumentos facilitadores. item Generar políticas de calidad del sistema educativo.

El EEES pretende proporcionar a los estudiantes una oferta más amplia y transparente de oportunidades educativas de calidad, creando un proceso simplificado de reconocimiento tanto interno como externo en todos los sistemas nacionales de enseñanza superior participantes. A lo largo de los últimos años, todos los participantes en el Proceso de Bolonia (47 países actualmente) han realizado reformas significativas de sus sistemas de enseñanza superior que han supuesto importantes avances en términos de calidad de la enseñanza, movilidad y reconocimiento académico.

En marzo de 2015 el Parlamento Europeo aprueba un informe sobre el seguimiento de la aplicación del Proceso de Bolonia (2015/2039(INI)) que establece una lista amplia de prioridades y desafíos entre los que se pueden destacar los siguientes:

²<http://ehea.info/page-sorbonne-declaration-1998>

³<http://ehea.info/page-ministerial-conference-bologna-1999>

⁴<http://ehea.info/>

- Evaluación del proceso y nuevos desafíos, con el fin de determinar adecuadamente la dirección que deben tomar las reformas de Bolonia y de identificar los ámbitos susceptibles de mejora.
- Estructura e instrumentos de Bolonia, no tanto en cuanto a la estructura de tres ciclos de Bolonia, mayoritaria en los países del **EEES**, sino más bien a la correcta aplicación del Sistema ECTS y a la expedición automática y gratuita del Suplemento Europeo al Título (que no se cumple en el 70 % de los países del **EEES**)
- Movilidad y calidad de la enseñanza superior, con el objeto de reforzar los procedimientos de reconocimiento, mejorar la calidad de los servicios para estudiantes en las universidades de acogida, etc.
- Dimensión social, fijando objetivos medibles para ampliar la participación general en la educación superior y aumentando el acceso de los grupos infrarrepresentados
- Dimensión exterior, con el fin de aumentar su atractivo y competitividad a nivel mundial,
- Prioridades para el futuro próximo, entre las que se pueden señalar la necesidad de corregir sus deficiencias y hacer que el enfoque educativo esté más orientado hacia los estudiantes y los resultados; reforzar los vínculos entre la educación superior y la investigación y, también, mejorar la sincronización de las medidas de apoyo al Proceso de Bolonia, como los programas Erasmus+ y Horizonte 2020.

En Junio de 2019 se celebró la reunión del 20 aniversario de Bolonia⁵ en la que se revisan los valores académicos del **EEES**, el aprendizaje centrado en el estudiante, el papel en el desarrollo sostenible y la dimensión social de la educación superior.

14.1.1 Créditos ECTS

La Declaración de Bolonia destaca la importancia del establecimiento de un Sistema de Créditos como elemento fundamental en la comparabilidad y la compatibilidad de los títulos. Así pues, el Sistema Europeo de Transferencia de Créditos, o ECTS (European Credit Transfer System) se ha diseñado como una forma de facilitar las tareas de reconocimiento de los estudios entre universidades de distintos países.

El Real Decreto 1125/2003, de 5 de septiembre⁶, por el que se establece el sistema europeo de créditos y el sistema de calificaciones tal y como se adopta en España define el crédito como:

El crédito europeo es la unidad de medida del haber académico que representa la cantidad de trabajo del estudiante para cumplir los objetivos del programa de estudios y que se obtiene por la superación de cada una de las materias que integran los planes de estudios de las diversas enseñanzas conducentes a la obtención de títulos universitarios de carácter oficial y validez en todo el territorio nacional. En esta unidad de medida se integran las enseñanzas teóricas y prácticas, así como otras actividades académicas dirigidas, con inclusión de las horas de estudio y de trabajo que el estudiante debe realizar para alcanzar los objetivos formativos propios de cada una de las materias del correspondiente plan de estudios.

Esto supuso pasar de un modelo educativo donde los créditos estaban asociados al trabajo del profesor, midiendo las horas de clase (1 crédito = 10 horas de clase), a un modelo donde un crédito representa las horas que el alumno necesita para el aprendizaje de la asignatura. Los créditos ECTS miden el trabajo del estudiante, centrándose en el aprendizaje y la consecución de unos objetivos predefinidos. Se valora en 60 créditos europeos el conjunto de asignaturas que un estudiante medio, dedicado a los estudios a tiempo completo, debe superar en un año, incluyendo tanto las horas presenciales como las horas de estudio y trabajo personal y la preparación y realización de exámenes. Su equivalencia en horas de trabajo, para dicho estudiante medio, tiene una cota superior aproximada de 1600 horas calculadas mediante:

$$8 \text{ horas diarias} \times 5 \text{ días a la semana} \times 40 \text{ semanas al año} = 1600 \text{ horas.}$$

Un crédito europeo representa por tanto entre 25 y 30 horas totales de trabajo del estudiante medio.

⁵<http://bolognaprocess2019.it/>

⁶<https://www.boe.es/eli/es/rd/2003/09/05/1125>

Competencias

Una *competencia* se define (según el RD 797/1995 del Ministerio de Trabajo y Seguridad Social) como la capacidad de aplicar conocimientos, destrezas y actitudes al desempeño de la ocupación que se trate, incluyendo la capacidad de respuesta a problemas, imprevistos, la autonomía, la flexibilidad, la colaboración con el entorno profesional y con la organización del trabajo. En el diseño de los planes de estudios se debe hacer énfasis en los métodos de aprendizaje de dichas competencias, así como en los procedimientos para evaluar su adquisición.

Por una parte, se encuentran las competencias genéricas, que deben ser adquiridas por cualquier estudiante universitario. Por ejemplo, cabe señalar las siguientes competencias genéricas:

- Aprender a conocer. Aprender a comprender el mundo que nos rodea, desarrollando las capacidades profesionales pertinentes.
- Aprender a hacer. Poner en práctica los conocimientos adquiridos.
- Aprender a convivir. Potenciar proyectos colaborativos para que los diferentes participantes obtengan beneficio práctico del trabajo en común.
- Aprender a ser. Se debe tener pensamiento autónomo y crítico y comportarse con responsabilidad y equidad en la vida.

Por otra parte, deben ser definidas las competencias específicas, que tienen un marco de aplicación más reducido que las genéricas, por ejemplo, las propias de cada asignatura. Es preciso tener especial cuidado en su elección, ya que todas ellas deben ser evaluadas, y por tanto, la evaluación será tanto más compleja cuanto mayor sea el número de competencias específicas.

14.1.2 Resultados de aprendizaje y sistema de calificaciones

El resultado de aprendizaje es una declaración de lo que se espera que el estudiante conozca, comprenda y sea capaz de hacer al finalizar un período de aprendizaje. A la hora de seleccionar los resultados de aprendizaje se deben tener en cuenta las competencias que el estudiante ha de adquirir, así como que el docente pueda evaluar el grado de adquisición de dichos resultados.

Uno de los aspectos más destacables de la implantación del EEES es la propuesta de aplicación de nuevas metodologías docentes, como la evaluación continua o el trabajo en grupo, que deben convivir con las cada vez menos frecuentes clases magistrales. Por tanto, los sistemas de evaluación deben estar en consonancia con la metodología empleada en cada caso, teniendo en cuenta que se deben evaluar las competencias adquiridas por el estudiante.

A la hora de presentar los resultados de la evaluación, cada país ha elaborado su propia normativa. En España, el Real Decreto 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en todo el territorio nacional, y en particular el artículo 5, indica que el nivel de aprendizaje conseguido por los estudiantes se expresará con calificaciones numéricas. Además, establece que los resultados obtenidos por el estudiante en las asignaturas se calificarán en función de la siguiente escala numérica de 0 a 10, con expresión de un dígito decimal, a la que podrá añadirse su correspondiente calificación cualitativa.

- Suspenso: 0 – 4,9.
- Aprobado: 5,0 – 6,9.
- Notable: 7,0 – 8,9.
- Sobresaliente: 9,0 - 10.

La mención de *Matrícula de Honor* podrá ser otorgada a estudiantes que hayan obtenido una calificación igual o superior a 9,0. Su número no podrá exceder del 5 % de estudiantes matriculados en una materia en el correspondiente curso académico, salvo que el número de matriculados sea inferior a 20, en cuyo caso se podrá conceder una sola *Matrícula de Honor*.

14.2 La Universidad en España

En España, las universidades más antiguas serían las de Palencia (1212), Salamanca (1215), y Valladolid (1260). Posteriormente, a finales del siglo XIII se fundaría la Universidad de Lérida. La institución universitaria española ha estado marcada por el modelo napoleónico en el que las instituciones tenían una fuerte dependencia del gobierno Central. Las instituciones universitarias han tenido que adaptarse a la

gran demanda universitaria que se produjo en la segunda mitad del siglo XX. Mientras que en 1945, el total de alumnos no superaba los 40.000 al cerca de 1 millón que se matriculaban a principios de los años 90 [35]. Siguiendo los acuerdos de Bolonia, España también realizó su correspondiente adaptación al Espacio Europeo de Educación Superior

14.2.1 Marco legislativo

Las siguientes son las principales Leyes o Reales Decretos que se han ido desarrollando en España relativas a la Universidad:

- Ley Orgánica 6/2001, de 21 de diciembre, de Universidades (LOU) (Boletín Oficial del Estado de 24 de diciembre de 2001)⁷.
- Acuerdo del Consejo de Ministros de 19 de julio de 2002, por el que se autoriza al Ministerio de Educación, Cultura y Deporte a constituir a la Fundación estatal ANECA. Esta ley derogaba la anterior Ley de Reforma Universitaria (LRU)
- Real Decreto 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en todo el territorio nacional⁸.
- Real Decreto 49/2004, de 19 de enero, sobre homologación de planes de estudios y títulos de carácter oficial y validez en todo el territorio nacional⁹.
- Ley Orgánica 4/2007, de 12 de abril, por la que se modifica la Ley Orgánica 6/2001, de 21 de diciembre, de Universidades (LOMLOU)¹⁰.
- Real Decreto 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales.¹¹
- Real Decreto 1509/2008, de 12 de septiembre, por el que se regula el Registro de Universidades, Centros y Títulos.¹²
- Real Decreto 861/2010, de 2 de julio, por el que se modifica el Real Decreto 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales.¹³
- Real Decreto 99/2011, de 28 de enero, por el que se regulan las enseñanzas oficiales de doctorado.¹⁴
- Real Decreto 1027/2011, de 15 de julio, por el que se establece el Marco Español de Cualificaciones para la Educación Superior.¹⁵
- Real Decreto 96/2014, de 14 de febrero, por el que se modifican los Reales Decretos 1027/2011, de 15 de julio, por el que se establece el Marco Español de Cualificaciones para la Educación Superior (MECES), y 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales.¹⁶
- Real Decreto 967/2014, de 21 de noviembre, por el que se establecen los requisitos y el procedimiento para la homologación y declaración de equivalencia a titulación y a nivel académico universitario oficial y para la convalidación de estudios extranjeros de educación superior, y el procedimiento para determinar la correspondencia a los niveles del marco español de cualificaciones para la educación superior de los títulos oficiales de Arquitecto, Ingeniero, Licenciado, Arquitecto Técnico, Ingeniero Técnico y Diplomado¹⁷.
- Real Decreto 22/2015, de 23 de enero, por el que se establecen los requisitos de expedición por las universidades del Suplemento Europeo al Título¹⁸.
- Real Decreto 195/2016, de 13 de mayo, por el que se establecen los requisitos para la expedición del

⁷<https://www.boe.es/eli/es/lo/2001/12/21/6>

⁸<https://www.boe.es/eli/es/rd/2003/09/05/1125>

⁹<https://www.boe.es/eli/es/rd/2004/01/19/49>

¹⁰Ley Orgánica 4/2007, de 12 de abril, por la que se modifica la Ley Orgánica 6/2001, de 21 de diciembre, de Universidades.

¹¹<https://www.boe.es/eli/es/rd/2007/10/29/1393>

¹²<https://www.boe.es/eli/es/rd/2008/09/12/1509>

¹³<https://www.boe.es/eli/es/rd/2010/07/02/861>

¹⁴<https://www.boe.es/eli/es/rd/2011/01/28/99>

¹⁵<https://www.boe.es/eli/es/rd/2011/07/15/1027>

¹⁶<https://www.boe.es/eli/es/rd/2014/02/14/96>

¹⁷<https://www.boe.es/eli/es/rd/2014/11/21/967>

¹⁸<https://www.boe.es/eli/es/rd/2015/01/23/22>

Suplemento Europeo al Título Universitario de Doctor.¹⁹

- Resolución de 11 de mayo de 2017, de la Secretaría General de Universidades, por la que se publica el Acuerdo del Consejo de Universidades de 10 de mayo de 2017, por el que se ordenan las enseñanzas universitarias oficiales de Grado²⁰.
- Real Decreto 822/2021, de 28 de septiembre, por el que se establece la organización de las enseñanzas universitarias y del procedimiento de aseguramiento de su calidad²¹. Deroga el Real Decreto 1393/2007.

El marco normativo actual para la ordenación y verificación de enseñanzas oficiales españolas viene dado por:

- La Ley Orgánica 4/2007 (LOMLOU) de abril, por la que se modifica la Ley Orgánica 6/2001 de Universidades (LOU), que establece una nueva estructura de las enseñanzas y títulos universitarios españoles en consonancia con los objetivos establecidos para la construcción del **EEES**.
- El Real Decreto 861/2010, de 2 de julio, por el que se modifica el 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales, y
- el Real Decreto 99/2011, de 28 de enero, por el que se regulan las enseñanzas oficiales de doctorado.

La Ley Orgánica 6/2001, de 21 de diciembre, de Universidades (Boletín Oficial del Estado de 24 de diciembre de 2001), conocida como LOU, define, además de las funciones y la autonomía de las universidades, su estructura y gobierno, la composición del personal docente, y la evaluación y acreditación de las enseñanzas, de las actividades docentes e investigadoras y de su profesorado, entre otras. Esta Ley Orgánica sustituye y deroga la Ley Orgánica 11/1983, de 25 de agosto, de Reforma Universitaria, comúnmente conocida como LRU (Boletín Oficial del Estado de 1 de septiembre de 1983). Una de las principales modificaciones de esta ley fue la inclusión del sistema de Habilitación para el acceso a los cuerpos docentes.

En el año 2007, la Ley Orgánica 4/2007, de 12 de abril, o Ley Orgánica de Modificación de la Ley Orgánica de Universidades (LOMLOU), modifica la LOU para adaptarla al **EEES**, estructurando las enseñanzas universitarias en tres ciclos: Grado, Máster y Doctorado, para adaptar los actuales estudios a la Declaración de Bolonia, incentivando la movilidad de la comunidad universitaria, tanto docentes como estudiantes, y armonizando los títulos españoles y los europeos. La Ley Orgánica 4/2007, de 12 de abril, modifica parcialmente la LOU para dar respuesta a cambios en elementos del entorno, tales como los acuerdos en política de educación Superior en Europa, así como para corregir deficiencias en la LOU tras cinco años de aplicación. Entre ellos, la sustitución del sistema de Habilitación (proceso presencial, competitivo) por el de Acreditación (proceso no presencial, no competitivo), que es el actualmente vigente. El Real Decreto Ley 14/2012, de 20 de abril, de medidas urgentes de racionalización del gasto público en el ámbito educativo (Boletín Oficial del Estado de 21 de abril), en su Artículo 6, también modifica la LOU. Modifica, entre otras cosas, la dedicación a la actividad docente del personal docente e investigador, ligando ésta a méritos investigadores.

El Real Decreto 1393/2007 introdujo cambios importantes en el sistema universitario como las directrices para el diseño de los títulos de Grado. En concreto, se dispuso que los planes de estudio de las titulaciones de Grado debieran tener 240 créditos. En estos créditos se incluía toda la formación teórica y práctica que el estudiante tenía que adquirir. Por otra parte, el Real Decreto 1393/2007 establece que los Sistemas de Garantía de la Calidad han de ser parte de los nuevos planes de estudios, de manera que constituyen el fundamento para que la nueva organización de las enseñanzas funcione eficientemente y para crear la confianza sobre la que descansa el proceso de acreditación de títulos. La autonomía en el diseño del título se ha de combinar con un adecuado sistema de evaluación y acreditación, que permita supervisar la ejecución efectiva de las enseñanzas e informar a la sociedad sobre la calidad de las mismas. Los sistemas de verificación y acreditación permitirán el equilibrio entre una mayor capacidad de las universidades para diseñar los títulos y la rendición de cuentas orientada a garantizar la calidad y mejorar la información a la sociedad sobre las características de la oferta universitaria. La acreditación de un título se basa en la verificación del cumplimiento del proyecto presentado por la Universidad y facilita la participación en programas de financiación específicos como, por ejemplo, de movilidad de profesores o estudiantes.

En el momento de redacción de la presente memoria se está preparando una nueva Ley Orgánica del Sistema Universitario (LOSU) cuyo anteproyecto ha sido objeto de consulta prenormativa o consulta pública

¹⁹<https://www.boe.es/eli/es/rd/2016/05/13/195>

²⁰[https://www.boe.es/eli/es/res/2017/05/11/\(1\)/con](https://www.boe.es/eli/es/res/2017/05/11/(1)/con)

²¹<https://www.boe.es/eli/es/rd/2021/09/28/822>

desde el 18 de junio al 8 de julio 2021²².

14.2.2 ANECA

ANECA es el órgano de evaluación de la calidad de la educación superior de alcance nacional encargado de realizar actividades de evaluación, certificación y acreditación, del sistema universitario español con el fin de su mejora continua y adaptación al EEEES. Para conseguir sus fines desarrolla procedimientos de evaluación de la siguiente naturaleza: enseñanzas conducentes a títulos oficiales de Grado, Máster y Doctorado; órganos y procesos académicos de carácter institucional; candidatos individuales a convertirse en profesor universitario en sus diversas categorías; y, por último a través de la Comisión Nacional de Evaluación de la Actividad Investigadora (CNEAI), de la producción investigadora individual de los profesores universitarios y personal investigador.

Las actividades de ANECA y sus procedimientos son objeto de evaluación externa cada cinco años por un panel de expertos coordinado por la Asociación Europea de Agencias de Aseguramiento de la Calidad (ENQA), para la renovación de su pertenencia al Registro Europeo de Agencias (EQAR).

Sus procedimientos se rigen por lo dispuesto en los Criterios y Directrices para el Aseguramiento de la Calidad de la Educación Superior comunes a todo el EEEES y se llevan a cabo por pares académicos con el concurso profesional y técnico del personal de la Agencia.

ANECA fue creada por la Ley Orgánica de Universidades 6/2001, de 21 de diciembre como Fundación adscrita al Ministerio de Educación y sus funciones fueron ampliadas en su modificación en posterior de 2007 (LOMLOU 4/2007, de 12 de abril). La Ley 15/2014, de 16 de septiembre, de racionalización del Sector Público y otras medidas de reforma administrativa, le confirió su estatuto definitivo como Organismo Autónomo de la Administración Central del Estado, adscrito al Ministerio de Educación, Cultura y Deporte (actualmente, al Ministerio de Ciencia, Innovación y Universidades) a través de la Secretaría General de Universidades.

En este contexto, las universidades, tanto públicas como privadas, tienen autonomía para diseñar propuestas de títulos oficiales. Sin embargo, deben adaptarse a las características dictadas por el Gobierno a raíz de las declaraciones europeas y es preceptivo obtener dos informes: Uno de autorización por parte de la Comunidad Autónoma donde pretenda impartirse el título, y otro, emitido por el Consejo de Universidades, que es el responsable de verificar que cumple con la normativa. Para ello, el programa VERIFICA, implantado por la ANECA, evalúa las propuestas de los planes de estudio de títulos diseñados en consonancia con él.

El Real Decreto 1393/2007, posteriormente modificado por el Real Decreto 43/2015 en algunos aspectos, establece que las enseñanzas universitarias conducentes a la obtención de títulos de carácter oficial y validez en todo el territorio nacional se estructuran en tres ciclos, denominados respectivamente Grado, Máster y Doctorado. A continuación, se describen las características de cada uno de estos ciclos.

14.2.3 Enseñanzas de Grado

Las enseñanzas de Grado tienen como finalidad la obtención por parte del estudiante de una formación general orientada a la preparación para el ejercicio de actividades de carácter profesional. La superación de estas enseñanzas da derecho a la obtención del título de Graduado o Graduada, con la denominación específica que figure en el Registro de Universidades, Centros y Títulos²³. En el Suplemento Europeo al Título, cuyos requisitos de expedición se establecen en el Real Decreto 22/2015, se incluirá la información unificada, personalizada para cada titulado universitario, sobre los estudios cursados, los resultados obtenidos, las capacidades profesionales adquiridas y el nivel de su titulación en el sistema nacional de educación superior.

Los planes de estudios conducentes a la obtención del título de Graduado han de tener 240 créditos, conteniendo toda la formación que el estudiante deba adquirir. Estas enseñanzas concluyen con la elaboración y defensa de un trabajo de fin de Grado. El plan de estudios debe contener un mínimo de 60 créditos de formación básica con al menos 36 vinculados a algunas de las materias básicas correspondientes a la rama de conocimiento a la que se adscribe el título. Estas materias se han de concretar en asignaturas con un mínimo de 6 créditos cada una y ofertarse en la primera mitad del plan de estudios. Los restantes, hasta 60 créditos deben estar configurados por materias básicas de la misma u otras ramas de conocimiento, o por otras materias siempre que sean de carácter básico para la formación inicial del estudiante o de carácter transversal. Las prácticas externas (en caso de programarse) deben tener una extensión máxima de 60 créditos

²²<https://www.universidades.gob.es/portal/site/universidades/>

²³<https://www.educacion.gob.es/ruct/>

y ofrecerse preferentemente en la segunda mitad del plan de estudios. Por último, el trabajo de fin de Grado ha de tener entre 6 y 30 créditos, realizarse en la fase final del plan de estudios y orientarse a la evaluación de competencias asociadas al título.

Directrices propias de los estudios de Informática

La Resolución de 8 de junio de 2009, de la Secretaría General de Universidades, por la que se da publicidad al Acuerdo del Consejo de Universidades, por el que se establecen recomendaciones para la propuesta por las universidades de memorias de solicitud de títulos oficiales en los ámbitos de la Ingeniería Informática, Ingeniería Técnica Informática e Ingeniería Química²⁴ establece que todo plan de estudios de Grado en Ingeniería Informática debe proporcionar, como mínimo, las siguientes competencias repartidas entre los módulos que se describen a continuación.

Módulo de Formación Básica.

El número de créditos ECTS de este módulo es 60, y en él deben adquirirse las siguientes competencias:

- Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
- Comprensión y dominio de los conceptos básicos de campos y ondas y electromagnetismo, teoría de circuitos eléctricos, circuitos electrónicos, principio físico de los semiconductores y familias lógicas, dispositivos electrónicos y fotónicos, y su aplicación para la resolución de problemas propios de la ingeniería.
- Capacidad para 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.
- Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- 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.
- Conocimiento adecuado del concepto de empresa, marco institucional y jurídico de la empresa. Organización y gestión de empresas.

Módulo Común a la Rama de Informática.

El número de créditos ECTS de este módulo es 60, y en él deben adquirirse las siguientes competencias:

- 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.
- 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.
- Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
- Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

²⁴[https://www.boe.es/eli/es/res/2009/06/08/\(3\)](https://www.boe.es/eli/es/res/2009/06/08/(3))

- Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
- Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.
- Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.
- Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
- Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
- Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.
- Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas. Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

Módulo de Tecnología Específica de Ingeniería del Software.

El número de créditos ECTS de este módulo es 48, y en él deben adquirirse las siguientes competencias:

- 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.
- 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.
- Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
- 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.
- Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.
- 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.

Proyecto Fin de Grado.

El número de créditos ECTS de este módulo es 12, y define un ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas, en el que el alumno debe adquirir las siguiente

competencias:

- Capacidad de integrar y sintetizar en un trabajo de naturaleza profesional individual y original los conocimientos adquiridos en el ámbito de la Ingeniería Informática.

14.2.4 Enseñanzas de Máster

Las enseñanzas de Máster tienen como finalidad la adquisición por el estudiante de una formación avanzada, de carácter especializado o multidisciplinar, orientada a la especialización académica o profesional, o bien a promover la iniciación en tareas investigadoras. La superación de estas enseñanzas da derecho a la obtención del título de Máster Universitario, con la denominación específica que figure en el Registro de Universidades, Centros y Títulos.

Los planes de estudios conducentes a la obtención de los títulos de Máster Universitario han de tener entre 60 y 120 créditos, conteniendo toda la formación teórica y práctica que el estudiante debe adquirir: materias obligatorias, materias optativas, seminarios, prácticas externas, trabajos dirigidos, actividades de evaluación y otras que resulten necesarias según las características propias de cada título. Estas enseñanzas concluyen con la elaboración y defensa pública de un trabajo de fin de Máster, que ha de tener entre 6 y 30 créditos.

14.2.5 Enseñanzas de Doctorado

Las enseñanzas de Doctorado tienen como finalidad la formación avanzada del estudiante en las técnicas de investigación, pueden incorporar cursos, seminarios u otras actividades orientadas a la formación investigadora e incluyen la elaboración y presentación de la correspondiente Tesis Doctoral, consistente en un trabajo original de investigación. La superación de estas enseñanzas da derecho a la obtención del título de Doctor o Doctora, con la denominación que figure en el Registro de Universidades, Centros y Títulos.

Para obtener el título de Doctor o Doctora es necesario haber superado un período de formación y un período de investigación organizado. Al conjunto organizado de todas las actividades formativas y de investigación conducentes a la obtención del título se denomina Programa de Doctorado. El acceso al período de formación de un Programa de Doctorado requiere las mismas condiciones que el acceso a un Máster, mientras que para acceder al período de investigación es necesario estar en posesión de un título oficial de Máster Universitario u otro del mismo nivel expedido por una institución de educación superior del EEEES. La tesis doctoral consiste en un trabajo original de investigación elaborado por el candidato en cualquier disciplina. La Universidad debe establecer los procedimientos con el fin de garantizar la calidad de las tesis doctorales tanto en su elaboración como en el proceso de evaluación.

14.3 La Universidad de Oviedo

14.3.1 Contexto histórico

El asturiano Fernando de Valdés Salas, arzobispo de Sevilla, Gran Inquisidor General y Presidente del Consejo de Castilla, expresó en su testamento su deseo de que fuera fundada una Universidad en Asturias. Despu  s de su muerte, pasar  n cuarenta a  os hasta que la Universidad de Oviedo inicie sus actividades el 21 de Septiembre de 1608. Es la d  cima Universidad m  s antigua de Espa  a, y la \'unica creada en el siglo XVII.

Los estudios que se impart  an inicialmente en sus aulas estaban encuadrados en las Facultades de Artes, Teolog  a, C  nones y Leyes, no lleg  ndose en aquellos a  os al centenar de estudiantes.

Las primitivas Facultades de Artes, Teolog  a y C  nones reciben nuevas denominaciones tras la etapa absolutista de Fernando VII que transform   la Universidad espa  ola, pasando a denominarse: Filosof  a, Derecho Civil y Can  nico, suprimi  ndose los estudios de Teolog  a en 1852. Los estudios tradicionales que se ofrec  an en la Universidad de Oviedo se relacionaban con lo que se podr  a considerar humanidades. En 1857 se establece la Facultad de Filosof  a y Letras, cuyas secciones se han convertido recientemente en Facultades: Filolog  a, Filosof  a y Ciencias de la Educaci  n, Geograf  a e Historia. A la actual Facultad de Derecho se han sumado la de Ciencias Econ  micas y la Escuela de Empresariales, creadas en 1972.

En cuanto a los estudios de Ciencias de la Salud, existi   una C  tedra de Medicina en la Universidad que desapareci   en 1809. Hasta 1968 no aparecen estudios de Medicina como tales con la creaci  n de la Facultad de Medicina, y posteriormente las Escuelas de Enfermer  a en 1977. En 1991 comienza la Diplomatura en Fisioterapia.

Respecto a estudios de ciencias, existi   brevemente una Facultad de Ciencias de 1858 a 1860, que no se restablece hasta el a  o 1862. En la actualidad, las secciones de Qu  mica, Biolog  a y Geolog  a constituyen

Facultades propias, y posteriormente se creó una Facultad de Ciencias Matemáticas y Físicas en el año 1991.

Los estudios técnicos como tales, no aparecen en la Universidad de Oviedo hasta los años 50 del s. XX con la incorporación a la Universidad de la Escuela Técnica Superior de Ingenieros de Minas (creada en 1957) y otras Escuelas Universitarias (Minas, Industriales), y la creación de la Escuela Técnica Superior de Ingenieros Industriales en 1979. A estos estudios se añaden los estudios de Informática, que aparecen en el año 1982 con la creación de las Escuelas Universitarias de Informática de Oviedo y Gijón. El ciclo superior en Informática se crea en el año 1991 y se integra en la Escuela Técnica Superior de Ingenieros Industriales e Informáticos.

La Universidad de Oviedo cuenta en la actualidad con centros docentes en siete Campus, en las ciudades de Oviedo (El Cristo, Llamasquique, Los Catalanes, Oviedo Centro y el Milán), Gijón (Campus de Gijón) y Mieres (Campus de Mieres).

A efectos docentes y de investigación, la Universidad de Oviedo se organiza en 38 Departamentos, 11 Facultades, 6 Escuelas, 3 Centros adscritos y 9 Institutos Universitarios que se describirán brevemente en las siguientes secciones.

14.3.2 La docencia en la Universidad de Oviedo

La estructura de la Universidad de Oviedo se estructura en torno a departamentos y centros. Los Departamentos son los órganos básicos encargados de coordinar las enseñanzas de una o varias áreas de conocimiento (actualmente existen 149 áreas de conocimiento), en uno o varios centros, de acuerdo con el plan de ordenación docente de la universidad, y de apoyar las actividades e iniciativas docentes e investigadoras del profesorado. Las Facultades y Escuelas son los centros encargados de la organización de las enseñanzas y de los procesos académicos, administrativos y de gestión conducentes a la obtención de títulos de carácter oficial y validez en todo el territorio nacional. Los Institutos Universitarios son centros dedicados a la investigación, a la creación, al desarrollo y a la innovación en campos específicos o interdisciplinares de la Ciencia, la Tecnología, las Artes y las Humanidades.

La [oferta formativa](#) del curso 2021-22 en lo que respecta a estudios de grado asciende a [61 grados](#), repartidos en áreas de conocimiento como sigue:

- 9 en Artes y Humanidades,
- 7 en Ciencias (uno de ellos es un doble grado en Matemáticas y Física),
- 8 en Ciencias de la Salud (dos de ellos en centros adscritos),
- 17 en Ciencias sociales y Jurídicas (4 de ellos en centros adscritos privados, uno online, uno doble) y
- 20 en Ingenierías y Arquitectura (uno de ellos un doble grado en Ingeniería Civil e Ingeniería de los Recursos Mineros y Energéticos).

En 20 de estos grados existe la posibilidad de seguir itinerarios bilingües; 11 de ellos pueden ser cursados íntegramente en la Universidad de Oviedo, entre los que se encuentra el Grado en Ingeniería Informática del Software. En los 9 restantes, la formación se debe completar mediante una estancia Programa Erasmus / Erasmus Prácticas en universidades o instituciones extranjeras que hayan firmado un convenio de movilidad con la Universidad de Oviedo y en la que se cursen asignaturas en inglés, o bien en otra universidad española cursando igualmente asignaturas en inglés en el marco del programa de movilidad SICUE.

Por otro lado, la [oferta de estudios de máster](#) consta de [62 títulos](#), que se distribuyen de la siguiente forma:

- 6 Erasmus Mundus,
- 8 internacionales,
- 7 interuniversitarios,
- 41 académico-profesionalizantes: 3 en la rama de Artes y humanidades, 6 en la rama de Ciencias, 7 en Ciencias de la salud, 10 en Ciencias sociales y jurídicas y 15 en la rama de Ingeniería y arquitectura.

En cuanto a los estudios de doctorado, la Universidad de Oviedo cuenta con 26 Programas de Doctorado, de los cuales 9 son de la rama de Ingeniería y arquitectura.

Por otra parte, la Universidad de Oviedo ofrece 2 títulos propios de grado, 34 de máster, 6 de especialista, 17 de experto y dos de formación continua.

En su apuesta por la formación más allá de la Universidad, y por el aprendizaje a lo largo de la vida, la Universidad de Oviedo cuenta también con un volumen importante de estudios de extensión universitaria, cursos de formación en idiomas y un programa para mayores (PUMUO).

La Extensión Universitaria es una institución con gran arraigo en la Universidad de Oviedo. La actividad de difusión del conocimiento se extiende fuera de los campus por toda la geografía del Principado de Asturias. Se trata de una contribución al desarrollo social, cultural y económico de Asturias por medio de la difusión

de la Ciencia, la Cultura y el Arte, y se realiza a través de cursos, conferencias y todo tipo de actividades culturales y formativas, con medios propios o en colaboración con entidades públicas o privadas. Como norma general, estos cursos y actividades son de libre acceso.

La Casa de las Lenguas imparte la enseñanza instrumental no reglada de idiomas en la Universidad Oviedo²⁵, prestando apoyo a sus objetivos de internacionalización. Su oferta actual incluye 57 cursos de 14 lenguas, así como cursos de español para extranjeros y cursos de capacitación lingüística para los profesores universitarios.

El PUMUO abre un espacio de encuentro y participación destinado a un sector de la población que ha superado los cincuenta años y que dispone de tiempo libre e inquietud por ampliar sus conocimientos. Con dos centenares de estudiantes cada curso, el programa tiene una duración de cinco años académicos y consta de una parte obligatoria, formada por asignaturas comunes para todo el alumnado, una parte optativa con el formato de talleres y que atiende intereses más concretos de cada matriculado, y una oferta de libre elección en el marco de la actividad cultural desarrollada por la Universidad. PUMUO tiene sedes en Oviedo, Gijón y Avilés.

Normativa propia de organización docente

El Grado en el que se imparte la asignatura [Arquitectura del Software](#) en la Universidad de Oviedo, además de al Real Decreto 1393/2007, se adhiere a la normativa general para la organización de los estudios de Grado en el proceso de transformación de las titulaciones al [EEES](#) de la Universidad de Oviedo, aprobada en Consejo de Gobierno de 23 de julio de 2008, que establece:

- Un crédito ECTS equivale a 25 horas de trabajo del estudiante. Cada curso académico constará de 60 ECTS, lo que equivale a unas 1.500 horas de trabajo del estudiante por curso.
- La organización del conjunto de las asignaturas será semestral, anual o mixta (exceptuando, en su caso, el trabajo de fin de grado y las prácticas externas).
- El número mínimo de ECTS de una asignatura será de 6 y siempre múltiplo de 3, excepto en el caso de asignaturas de formación básica que deberá ser múltiplo de 6.
- La duración del curso académico será como mínimo de 38 semanas y como máximo de 40 semanas.
- Los porcentajes máximos de presencialidad serán de:
 - Asignatura con teoría y práctica: 40%.
 - Asignatura exclusivamente práctica: 60%.
 - Trabajo fin de grado: 10% (40% en caso de que el trabajo fin de grado implique la realización de actividades que requieran la supervisión presencial del tutor).
 - Prácticas externas y asignaturas exclusivamente de práctica hospitalaria: 80% (en este caso la presencialidad se refiere al tiempo que el estudiante tiene que permanecer en el lugar donde realiza las prácticas).
- El número de créditos mínimo correspondiente a asignaturas o actividades de carácter obligatorio será, en general, de 210 ECTS. Se permitirán 180 ECTS en los casos en los que la organización modular permita un mejor aprovechamiento de los recursos humanos y materiales.
- En el caso de que existan menciones o intensificaciones deberán estar definidas con un mínimo de 30 ECTS.
- Para aquellas titulaciones que opten por un mínimo de 210 ECTS obligatorios la oferta máxima de optativas será de 90 ECTS. Para las que opten por un mínimo de 180 ECTS obligatorios la oferta máxima de optativas será de 120 ECTS.
- Al menos 48 ECTS de las materias de formación básica serán de la misma rama de conocimiento a la que se pretenda adscribir el título y se impartirán en el primer curso.
- La implantación de las nuevas titulaciones se realizará curso a curso.
- Se procurará que las asignaturas que se imparten en inglés se concentren en el mismo semestre.
- En el caso de títulos con directrices propias se ajustarán, además, a lo estipulado en la normativa correspondiente.

²⁵<https://lacasadelaslenguas.uniovi.es/cursosidiomas>

Directrices propias para enseñanzas técnicas

En la Universidad de Oviedo se constituyó una comisión de enseñanzas técnicas encargada de establecer unos criterios comunes para todas las enseñanzas técnicas más específicos que la normativa general expuesta anteriormente de la cual formaba parte el autor de esta memoria como Director de la Escuela de Ingeniería Informática. Se definieron las siguientes directrices:

1. Las asignaturas básicas comunes a todas o parte de las ingenierías, consistirán en cinco asignaturas de seis créditos cada una, incluida una asignatura de Fundamentos de Informática.
2. Las asignaturas comunes a los dos grados de Informática que se propone impartir en la Universidad de Oviedo, además de las cinco asignaturas comunes a las ingenierías, tienen otras quince asignaturas que completan los cursos de primero y segundo, denominadas Básicas de Informática y serán comunes de Informática y de carácter obligatorio.
3. El número de créditos de estas asignaturas es de 6 créditos ECTS.

De esta forma las asignaturas de los dos primeros cursos del grado en Ingeniería Informática del Software, que se impartirá en la Escuela de Ingeniería Informática de Oviedo y del Grado en Ingeniería Informática en Tecnologías de la Información, que se impartirá en la Escuela Politécnica de Ingenieros de Gijón, son las mismas, con el fin de facilitar la posible movilidad de estudiantes que comiencen en una de las titulaciones y quieran cambiar a la otra.

14.3.3 La investigación en la Universidad de Oviedo

La Universidad de Oviedo ha sido una de las primeras nueve universidades españolas en obtener el sello de Campus de Excelencia Internacional y su producción científica acredita la excelencia y la calidad reconocida internacionalmente. Los Grupos de Investigación son las unidades sobre las que se organiza la actividad investigadora. Para constituirse como tal, los grupos de investigación deben superar una evaluación externa de sus actividades durante los últimos 5 años y de su proyecto de trabajo para los siguientes 5. Existen un total de 166 grupos²⁶ que han superado la evaluación, y por tanto han sido reconocidos por el Consejo de Gobierno conforme al Reglamento de la Universidad de Oviedo (40 en Ciencias, 24 en Ciencias de la Salud, 38 en Ciencias Sociales y Jurídicas, 28 en Artes y Humanidades y 36 en la rama de Ingeniería y Arquitectura).

El repositorio de la Universidad de Oviedo tiene registradas 28426 publicaciones en revistas de investigadores de la Universidad de Oviedo desde 1966, 4992 capítulos de libros, 98 patentes, 81 pósteres, 3243 ponencias, discursos y conferencias y 6533 tesis (fecha de consulta, 15 de julio de 2021) defendidas en la Universidad de Oviedo.

Según Scopus, la producción científica de la Universidad de Oviedo asciende a 34970 publicaciones (consultado el 15 de julio de 2021), cuya distribución por ramas (de Scopus) se muestra en la Figura 14.1.

²⁶<https://www.uniovi.es/investigacion/centros/grupos>

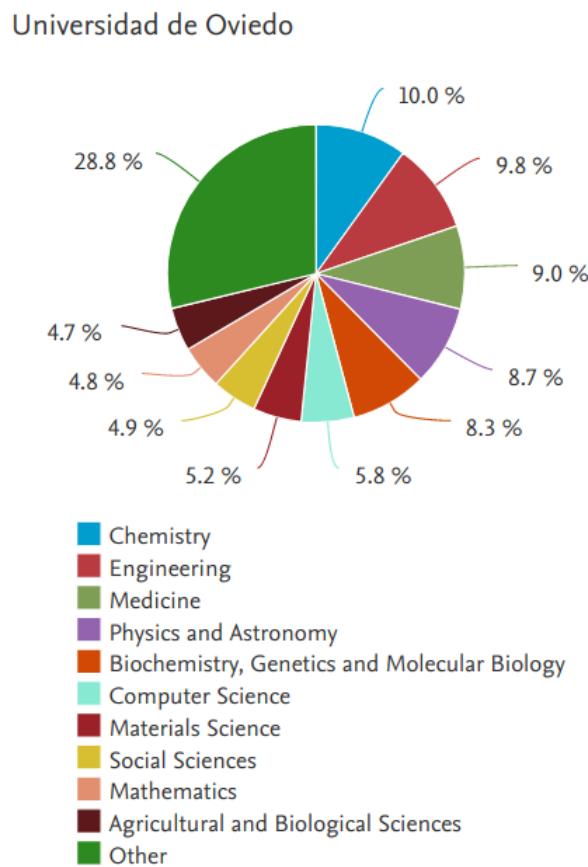


Figura 14.1: Producción científica de la Universidad de Oviedo, según Scopus (15 Julio 2021)

Como se puede observar, un 5,8 % de las publicaciones almacenadas en Scopus (unas 3376) pertenecen a la rama *Computer Science*.

En la Figura 14.2, se pueden observar las publicaciones de personal de la Universidad de Oviedo en el campo *Computer Science* desde 1984 (consultado el 15 de julio de 2021).

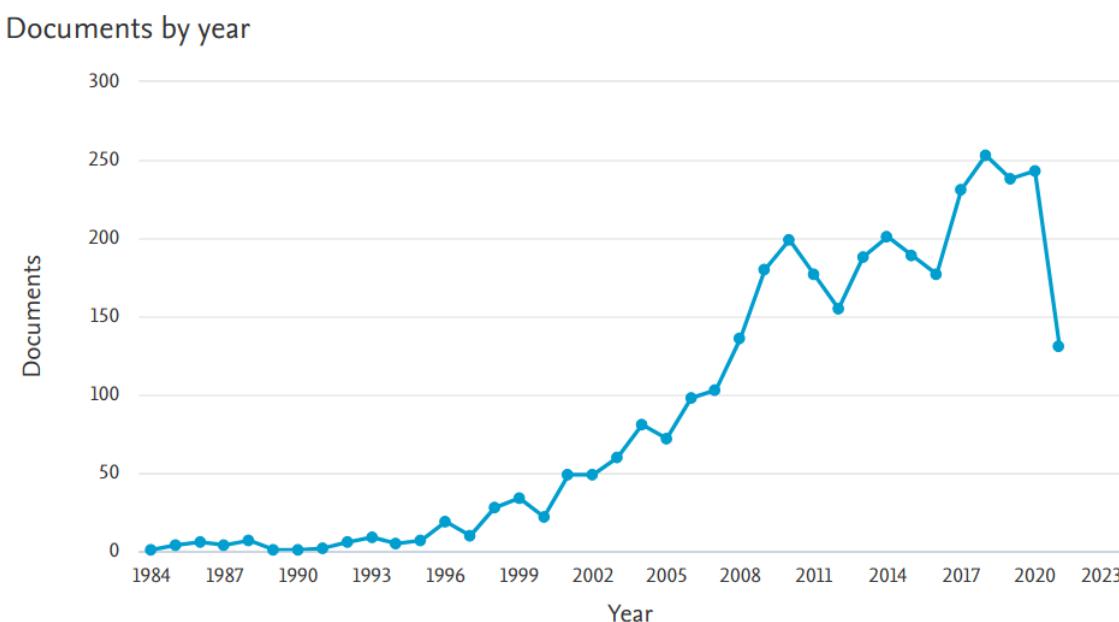


Figura 14.2: Producción científica en la rama Informática de la Universidad de Oviedo, según Scopus (15 Julio 2021)

Según Scimago²⁷, la Universidad de Oviedo ocupa el puesto general 541 en el año 2021, el puesto 300 en el ranking de investigación y el puesto 443 en el ranking de innovación.

Una vez descrita brevemente la Universidad de Oviedo, se describirán las características de los centros y del departamento que tienen relación con este proyecto docente.

14.3.4 Escuela de Ingeniería Informática de Oviedo

La Escuela Universitaria de Informática de Oviedo inició su actividad en el curso 1982/83, siendo su objetivo básico la formación académica y profesional de Diplomados Universitarios de Informática, en sus dos especialidades: Gestión y Sistemas. Los estudios comenzaron con las limitaciones propias del arranque de centros nuevos, sin aulas, ni laboratorios, y con escasez de profesorado. Los primeros pasos se realizan de la mano de la Escuela Universitaria de Empresariales de Oviedo y de la Escuela Técnica Superior de Ingenieros de Minas con las que comparte instalaciones.

La primera dirección de la Escuela de Ingeniería Informática fue llevada a cabo por Florentino Braña, nombrado director comisario y que compartía su cargo con el de director de la Escuela de Empresariales. En el año 1987, Dña. Andra Huerga Alonso será nombrada directora del Centro, cargo que ocupará hasta el año 1996.

A partir del curso 1989-90, el centro se ubica en las instalaciones del remodelado edificio de Ciencias (anteriormente albergaba la Facultad de Química) compartiéndolo con las licenciaturas de Físicas y Matemáticas y el Centro de Cálculo de la Universidad.

En el curso 1992-93 entra en vigor un nuevo Plan de Estudios y comienzan a impartirse las titulaciones de Ingeniero Técnico en Informática de Gestión e Ingeniero Técnico en Informática de Sistemas, pasando la Escuela a denominarse Escuela Universitaria de Ingeniería Técnica Informática de Oviedo.

En el año 1996, D. Juan Manuel Cueva Lovelle es elegido Director de la Escuela, en cuyo primer mandato, Jose Emilio Labra Gayo ocupa el cargo de Secretario Académico de la misma. Una de las tareas llevadas a cabo en dicho mandato fue la modificación del Plan de Estudios, que comenzará a impartirse en el curso 2002-03.

En el año 2004 pasa a ocupar el cargo de Director de la Escuela el autor de esta memoria, D. Jose Emilio Labra Gayo, que estará en el mismo durante dos mandatos de cuatro años. Debido al insuficiente número de

²⁷<https://www.scimagoir.com/institution.php?idp=35770>

aulas del centro, se venían utilizando aulas de otros edificios hasta que en el año 2004 se realizó el traslado al actual edificio Valdés Salas en el Campus de los Catalanes.

En el año 2007 se comienza a impartir el Máster Oficial en Ingeniería Web y en el año 2010 se realiza la adaptación del plan de estudios al Espacio Europeo de Educación Superior, creándose el Grado en Ingeniería Informática del Software y pasando la Escuela a denominarse Escuela de Ingeniería Informática.

En el año 2012, D. Benjamín López Pérez ocupa el cargo de director hasta que D. Fernando Álvarez García lo sustituye en el año 2020.

Grado en Ingeniería Informática del Software

El plan de estudios del Grado en Ingeniería Informática del Software, detallado en su memoria de verificación, obtiene la verificación del Consejo de Universidades (3 de Mayo de 2010), previo informe favorable de [ANECA](#) de fecha 24 de marzo de 2010, y se publica mediante la Resolución de 16 de marzo de 2011, de la Universidad de Oviedo, por la que se publica el plan de estudios de Graduado en Ingeniería Informática del Software (Boletín Oficial del Estado de 11 de julio de 2011). El grado tiene el código 2502374 en Registro de Universidades, Centros y Títulos²⁸.

La Resolución del Consejo de Universidades, de 3 de mayo de 2016, a través de su Comisión de Verificación, reunida en sesión del día 28 de abril de 2016, y a la vista del informe de carácter favorable emitido por [ANECA](#), renueva la acreditación del título de Graduado o Graduada en Ingeniería Informática del Software por la Universidad de Oviedo. El proceso de renovación de la acreditación del título se realiza mediante el Programa ACREDITA PLUS de [ANECA](#), con lo que además de la renovación de la acreditación se solicita el sello EURO-INF para el título.

El sello EURO-INF es un certificado concedido a una universidad respecto de un título universitario en el ámbito de la informática de grado o máster evaluado de acuerdo con una serie de estándares establecidos según los principios de calidad, relevancia, transparencia, reconocimiento y movilidad contemplados en el [EEES](#).

El sello EURO-INF se creó en el año 2006 con el apoyo de la Comisión Europea. Desde 2009, la European Quality Assurance Network for Informatics Education (EQANIE) es la asociación europea que regula su expedición, bien directamente o bien a través de agencias autorizadas. Es un sello reconocido internacionalmente y que facilita tanto la movilidad académica como la profesional, ya que permite identificar programas de informática que se adhieren a los citados estándares. El sello EURO-INF es un sello genérico, que es aplicable:

- En la diversidad de contextos europeos en los que se desarrolla la educación necesaria para el inicio de la profesión de informático.
- En las diversas ramas o especialidades de la informática.
- Para las diferentes competencias de los titulados provenientes de grado o máster, de acuerdo con el Marco Europeo de Cualificaciones en la Educación Superior.
- Para la diversidad de contextos normativos nacionales para el ejercicio de la profesión de informático en Europa.

Los criterios de evaluación del sello tratan de comprobar si los resultados de aprendizaje del título cumplen los criterios de calidad para las competencias, habilidades y conocimientos que constituyen la formación básica para la práctica profesional (grado) o para los estudios de posgrado (máster y doctorado). En particular, la evaluación trata de comprobar si el título proporciona los resultados de aprendizaje esperados por el sello en las siguientes cuatro categorías:

- Conceptos básicos de informática.
- Análisis, diseño e implementación.
- Competencias tecnológicas y metodológicas.
- Otras competencias profesionales.

La Comisión de Acreditación EURO-INF de [ANECA](#) resolvió la concesión del Sello EURO-INF durante el proceso de renovación de la acreditación del título.

Tras todos los trámites pertinentes en la Universidad de Oviedo y la [ANECA](#), en el BOE de 11 de julio de 2011 se publica el plan de estudios de Graduado en Ingeniería Informática en Informática del Software (GIISOFT), que consta de 240 créditos ECTS, distribuidos de materias de Formación Básica (FB), Obligatorias (OB), Optativas (OP), y de Trabajo Fin de Grado (TFG) de la forma siguiente:

²⁸La información oficial puede consultarse en <https://www.educacion.gob.es/ruct/estudiouniversidad.action?codigoCiclo=SC&codigoEstudio=2502374>

Tipo de materia	Créditos ECTS
Formación Básica (FB)	60
Obligatorias (OB)	144
Optativas (OP)	18
Prácticas externas obligatorias (PE)	6
Trabajo Fin de Grado (TFG)	12

El plan de estudios se estructura en nueve bloques:

1. Fundamentos Matemáticos
2. Fundamentos Informáticos
3. Soporte Físico y Hardware
4. Programación
5. Sistemas Informáticos
6. Interacción y Multimedia
7. Sistemas de Información y Persistencia
8. Ingeniería del Software
9. Práctica Profesional

y las materias y asignaturas relacionadas, junto con su carácter, duración y curso, se incluye en las tablas que se encuentran en la siguiente página.

En la tabla 14.1 se presenta el plan de estudios del Grado de Ingeniería Informática del Software que también se presenta en un formato más visual (obtenido del folleto de la titulación y que muestra las asignaturas por semestre)en la figura 14.3, en dicha figura se representan las asignaturas de formación básica (FB) en rojo, las obligatorias (OB) en verde: OB y los huecos para asignaturas optativas (OP) en amarillo.

Máster Universitario en Ingeniería Web

El Máster Universitario en Ingeniería Web consta de 120 créditos ECTS. Se pueden seguir dos itinerarios, profesional o investigador. La distribución de los créditos es la siguiente:

1. Módulo de materias básicas (15 ECTS). Pensado para desarrollar habilidades y capacidades que constituyen la base de un trabajo más específico posterior.
2. Módulo de materias obligatorias (50 ECTS). Constituye el núcleo central de adquisición de competencias.
3. Módulo de materias de especialidad profesional (21 ECTS). Se trata de un bloque constituido por asignaturas que capacitarán al alumno en destrezas de carácter más profesional, acercándolo en mayor medida a las habilidades más específicas de la profesión.
4. Módulo de materias de especialidad de Investigación (17,5 ECTS). Pensado para desarrollar en el alumno la capacidad investigadora, introduciendo tecnologías que aún se están desarrollando e incluyendo como objetivos de aprendizaje habilidades para el ejercicio de la investigación, de cara a la posterior elaboración de una tesis doctoral.
5. Trabajo Fin de Máster (22 ECTS).

Tanto si se cursa el módulo de materias de especialidad profesional como el módulo de materias de especialidad de investigación es preciso realizar el trabajo fin de máster.

A continuación se detallan las asignaturas correspondientes a estos módulos:

Materias	Asignaturas	Carácter	Créditos	Curso
Fundamentos matemáticos	Álgebra lineal	FB	6	1
	Cálculo	FB	6	1
	Estadística	FB	6	1
	Computación numérica	OB	6	2
Fundamentos informáticos	Fundamentos de Informática	FB	6	1
	Autómatas y matemáticas discretas	FB	6	1
	Computabilidad	OB	6	2
	Sistemas inteligentes	OB	6	1
Soporte físico y hardware	Ondas y electromagnetismo	FB	6	1
	Fundamentos de computadores y redes	FB	6	1
	Tecnología electrónica de computadores	OB	6	2
	Arquitectura de computadores	OB	6	2
Programación	Introducción a la programación	FB	6	1
	Metodología de la programación	FB	6	1
	Estructuras de datos	OB	6	2
	Tecnología y Paradigmas de programación	OB	6	21
	Algoritmia	OB	6	2
	Diseño de lenguajes de programación	OB	6	3
	Software de entretenimiento y videojuegos	OP	6	3,4
	Software para dispositivos móviles	OP	6	3,4
	Software para robots	OP	6	3,4
Sistemas informáticos	Sistemas operativos	OB	6	2
	Sistemas distribuidos e internet	OB	6	3
	Administración de sistemas y redes	OB	6	3
	Seguridad de sistemas informáticos	OB	6	3
Interacción y multimedia	Comunicación persona-máquina	OB	6	3
	Software y estándares para la Web	OB	6	3
	Realidad y accesibilidad aumentadas	OP	6	3,4
	Informática audiovisual	OP	6	3,4
Sistemas de información y persistencia	Bases de datos	OB	6	2
	Repositorios de información	OB	6	3
	Sistemas de información para la Web	OP	6	3,4
Ingeniería del Software	Ingeniería del proceso software	OB	6	3
	Diseño del software	OB	6	3
	Arquitectura del software	OB	6	3
	Ingeniería de requisitos	OB	6	4
	Calidad, validación y verificación del software	OB	6	4
	Modelos de Ingeniería del Software	OP	6	3,4
Práctica profesional	Empresa	FB	6	1
	Aspectos sociales, legales, éticos y profesionales de la informática	OB	6	4
	Dirección y planificación de proyectos informáticos	OB	6	4
	Prácticas externas	OB	6	4
	Integración de aplicaciones empresariales	OP	6	3,4
	Informática forense y auditoría	OP	6	3,4

Cuadro 14.1: Plan de Estudios del Grado en Ingeniería Informática del Software

PLAN DE ESTUDIOS

ASIGNATURAS PRIMER SEMESTRE	CRÉDITOS	ASIGNATURAS SEGUNDO SEMESTRE	CRÉDITOS
PRIMER CURSO			
Álgebra Lineal	6	Ondas y Electromagnetismo	6
Cálculo	6	Estadística	6
Empresa	6	Fundamentos de Computadores y Redes	6
Fundamentos de Informática	6	Autómatas y Matemáticas Discretas	6
Introducción a la Programación	6	Metodología de la Programación	6
SEGUNDO CURSO			
Tecnología Electrónica de Computadores	6	Sistemas Operativos	6
Arquitectura de Computadores	6	Comunicación Persona-Máquina	6
Estructuras de Datos	6	Bases de Datos	6
Tecnología y Paradigmas de la Programación	6	Computación Numérica	6
Computabilidad	6	Algoritmia	6
TERCER CURSO			
Repositorios de Información	6	Sistemas Distribuidos e Internet	6
Software y Estándares para la Web	6	Administración de Sistemas y Redes	6
Ingeniería del Proceso Software	6	Seguridad de Sistemas Informáticos	6
Diseño del Software	6	Arquitectura del Software	6
Optativa I	6	Diseño de Lenguajes de Programación	6
CUARTO CURSO			
Sistemas Inteligentes	6	Dirección y Planificación de Proyectos Informáticos	6
Ingeniería de Requisitos	6	Aspectos Sociales, Legales, Éticos y Profesionales de la Informática	6
Calidad, Validación y Verificación del Software	6	Prácticas Externas	6
Optativa II	6	Trabajo Fin de Grado	12
Optativa III	6		

El alumno elegirá a lo largo de la carrera, tres de las asignaturas que se citan a continuación:

- Informática Audiovisual
- Integración de Aplicaciones Empresariales
- Realidad y Accesibilidad Aumentadas
- Software de Entretenimiento y Videojuegos
- Software para Robots
- Informática Forense y Auditoría
- Modelos en Ingeniería del Software
- Sistemas de Información para la Web
- Software para Dispositivos Móviles

Figura 14.3: Resumen del plan de estudios

Módulo	Asignatura	Orientación	ECTS
Materias Fundamentales	Administración de Sistemas Operativos	-	3
	Lenguajes y Estándares de la Web	-	3
	Programación Orientada a Objetos	-	5
	Diseño y Programación de Interfaces de Usuario	-	3
	Análisis y Diseño Orientado a Objetos	-	3,5
	Sistemas de Persistencia de Objetos	-	3
Materias Obligatorias	Usabilidad, accesibilidad y adaptabilidad de sitios web	-	3
	Arquitecturas y diseño de sitios web	-	6,5
	Gestores de contenidos web	-	3
	Servicios web	-	5
	Administración de sistemas de persistencia de objetos	-	3
	Sistemas de seguridad en la web	-	3
	Modelos de negocio y comercio electrónico en la web	-	3
	Aspectos éticos y legales de la web	-	3
	Difusión de contenidos multimedia vía web	-	3
	Desarrollo de software para dispositivos móviles	-	3
	Sistemas de información geográfica y servicios web	-	3
	Administración de servidores web	-	3
Especialidad Profesional	Desarrollo e integración de aplicaciones internet	-	3
	Dirección y gestión de proyectos web	-	12
	Web Semántica	Profesional	4
	Arquitecturas software y procesamiento de lenguajes	Profesional	4
	Modelado de software web adaptable dirigido por modelos	Profesional	3
Especialidad Investigadora	Nuevos estándares en la web	Profesional	4
	Prácticas de empresa	Profesional	6
	Nuevos Avances en Web Semántica	Investigadora	4
	Lenguajes y plataformas de programación	Investigadora	5,5
	Diseño y construcción de MDA	Investigadora	3
Arquitectura y desarrollo de sistemas de e-learning		Investigadora	4
Arquitectura de la información en la web		Investigadora	4,5
Trabajo Fin de Máster		-	22

14.3.5 Centro Internacional de Postgrado de la Universidad de Oviedo

Conforme al artículo 7 de La Ley Orgánica de Universidades y 10 de los Estatutos de la Universidad de Oviedo, y dentro de la figura de *otros centros*, el Consejo de Gobierno, en sesión de 25 de noviembre de 2010, acordó la creación del Centro Internacional de Postgrado, que fue objeto de aprobación definitiva por el Consejo Social el 15 de diciembre de 2010.

El Centro Internacional de Postgrado se constituye inicialmente como una estructura de gestión universitaria encargada de los procesos académicos, administrativos y de gestión, conducentes a la obtención de los títulos de Máster Universitario, Doctorado, Títulos Propios y Formación Continua que tenga encomendados. Actualmente coordina la actividad de doctorado, ya que la mayoría de los estudios de Máster se han descentralizado, gestionándose desde los distintos Centros. Entre sus competencias figuran las siguientes:

1. Seleccionar y coordinar las propuestas de Máster Universitario, Programas de Doctorado, Títulos Propios y Formación Continua.
2. Organizar, gestionar y administrar las enseñanzas conducentes a la obtención de títulos de postgrado de la Universidad de Oviedo, sin perjuicio de las tareas organizativas que, por su propia naturaleza, deban atribuirse a los Departamentos o Centros donde se imparte docencia.
3. Difundir los estudios de postgrado ofertados por la Universidad de Oviedo, impulsando la internacionalización de las enseñanzas y potenciando la celebración de convenios o acuerdos con otras universidades, instituciones y empresas.
4. Colaborar y promover cuantas actividades satisfagan los fines estatutarios de la Universidad en materia de enseñanzas de Másteres Universitarios, Doctorado, Títulos Propios y de Formación Continua.

Doctorado en Informática

El doctorado en Informática está adscrito al Centro Internacional de Postgrado de esta Universidad. El Programa de Doctorado en Informática por la Universidad de Oviedo tiene como objetivo básico la formación de doctores en las tecnologías, aplicaciones, servicios y contenidos de la Sociedad de la Información, fomentando el avance del conocimiento, el desarrollo y la innovación en las tecnologías de la información y en las tecnologías para la creación de los contenidos digitales y de los servicios relacionados con estos, mediante la investigación científica y el desarrollo tecnológico de excelencia. Está diseñado como el resto de títulos de EES en base a Competencias, Capacidades y Destrezas. Las Competencias se Generales son las siguientes:

- CB11 Comprensión sistemática de un campo de estudio y dominio de las habilidades y métodos de investigación relacionados con dicho campo.
- CB12 Capacidad de concebir, diseñar o crear, poner en práctica y adoptar un proceso sustancial de investigación o creación.
- CB13 Capacidad para contribuir a la ampliación de las fronteras del conocimiento a través de una investigación original.
- CB14 Capacidad de realizar un análisis crítico y de evaluación y síntesis de ideas nuevas y complejas.
- CB15 Capacidad de comunicación con la comunidad académica y científica y con la sociedad en general acerca de sus ámbitos de conocimiento en los modos e idiomas de uso habitual en su comunidad científica internacional.
- CB16 Capacidad de fomentar, en contextos académicos y profesionales, el avance científico, tecnológico, social, artístico o cultural dentro de una sociedad basada en el conocimiento.

Además de la adquisición de competencias de carácter técnico, propias de la disciplina, se pretende que los alumnos adquieran o mejoren ciertas capacidades y destrezas generales o transversales, importantes en la formación como doctores en Informática y que también pretenden mejorar su calidad como investigadores. Las de mayor relevancia son:

- CA01 Desenvolverse en contextos en los que hay poca información específica.
- CA02 Encontrar las preguntas claves que hay que responder para resolver un problema complejo.
- CA03 Diseñar, crear, desarrollar y emprender proyectos novedosos e innovadores en su ámbito de conocimiento.
- CA04 Trabajar tanto en equipo como de manera autónoma en un contexto internacional o multidisciplinar.
- CA05 Integrar conocimientos, enfrentarse a la complejidad y formular juicios con información limitada.
- CA06 La crítica y defensa intelectual de soluciones

El programa consta de 4 líneas de investigación:

- Ingeniería del Software
 - Sistemas Inteligentes
 - Tecnologías Audiovisuales y de Red
 - Arquitecturas para Sistemas de Altas Prestaciones

La información completa del programa - información general, acceso y admisión, actividades formativas, Líneas de investigación y datos de contacto se encuentra en la página web de la Universidad de Oviedo²⁹.

14.4 Departamento y área de conocimiento

14.4.1 El Departamento de Informática

El Departamento de Informática se creó por decisión de Consejo de Gobierno de la Universidad de Oviedo de 18 de diciembre de 1996. Está compuesto por las siguientes áreas de conocimiento:

²⁹<http://www.uniovi.es/estudios/doctorado/>

- Arquitectura y Tecnología de los Computadores (ATC)
- Ciencia de la Computación e Inteligencia Artificial (CCIA)
- Ingeniería Telemática (IT)
- Lenguajes y Sistemas Informáticos (LSI)

Antes de la creación del departamento, las dos primeras áreas estaban asociadas al Departamento de Matemáticas y la tercera al de Ingeniería Eléctrica, Electrónica de Computadores y Sistemas. El área de Ingeniería Telemática nace en el seno del propio Departamento como consecuencia de la puesta en marcha de la Ingeniería de Telecomunicación.

A continuación se describen los recursos humanos de que dispone el Departamento de Informática en el curso 2020/2021. En la siguiente lista, CU se refiere a Catedrático de Universidad, TU a Profesor Titular de Universidad, CEU a Catedrático de Escuela universitaria, TEU a profesor Titular de Escuela Universitaria, TUI Titular de Universidad Interino, CD a contratado doctor, CDI contratado doctor interino, AYD ayudante doctor, ASO profesor asociado, PLI profesor laboral interino y BI Becario de investigación):

- 3 miembros del personal de administración y servicios (P.A.S.)
- 146 profesores, divididos de la siguiente forma:
 - Área de Arquitectura y Tecnología de Computadores: 14 (2 CU, 8 TU, 1 TEU, 2 AYD, 1 ASO)
 - Área de Ciencia de la Computación e Inteligencia Artificial: 50 (7 CU, 18 TU, 2 CEU, 2 TEU, 3 CD, 7 AYD, 2 ASO, 4 PLI, 5 BI)
 - Área de Ingeniería Telemática: 13 (5 TU, 1 CD, 2 CDI, 3 AYD, 2 ASO)
 - Área de Lenguajes y Sistemas Informáticos: 69 (3 CU, 21 TU, 4 CEU, 3 TEU, 9 CD, 6 PCOL, 7 AYD, 9 ASO, 2 PLI, 1 BI)

La docencia del Departamento de Informática se encuentra repartida por todos los campus de la Universidad de Oviedo. Los centros y las titulaciones en las que existen asignaturas cuyo responsable docente es el Departamento de Informática son los siguientes:

- Escuela de Ingeniería Informática
 - Grado en Ingeniería Informática del Software
 - Máster Universitario en Ingeniería Web
- Escuela de Ingeniería de Minas, Energía y Materiales de Oviedo
 - Grado en Ingeniería de Tecnologías Mineras
- Escuela Politécnica de Ingeniería de Gijón
 - Grado en Ingeniería Informática en Tecnologías de la Información
 - Grado en Ciencia e Ingeniería de datos
 - Grado en Ingeniería Eléctrica
 - Grado en Ingeniería Electrónica Industrial y Automática
 - Grado en Ingeniería Mecánica
 - Grado en Ingeniería Química Industrial
 - Grado en Ingeniería en Tecnologías Industriales
 - Grado en Ingeniería en Tecnologías y Servicios de Telecomunicación
 - Grado en Ingeniería en Organización Industrial
 - Máster Universitario en Ingeniería Informática
 - Máster Universitario en Ingeniería de Telecomunicación
- Escuela Politécnica de Mieres
 - Grado en Ingeniería en Geomática y Topografía
 - Grado en Ingeniería Forestal y del Medio Natural
 - Grado en Ingeniería de los Recursos Mineros y Energéticos
 - Grado en Ingeniería Civil
 - Máster en Geotecnología y Desarrollo de Proyectos SIG
- Escuela Superior de la Marina Civil
 - Grado en Ingeniería Marina
 - Grado en Ingeniería Náutica y Transporte Marino
- Facultad de Ciencias
 - Grado en Matemáticas
 - Máster Universitario en Modelización e Investigación Matemática, Estadística y Computación
 - Máster en Análisis de Datos para la Inteligencia de Negocios

- Facultad de Química
 - Grado en Ingeniería Química
- Facultad de Biología
 - Máster en Biotecnología Aplicada a la Conservación y Gestión
- Facultad de Formación del Profesorado y Educación
 - Máster en Formación del Profesorado
- Facultad de Medicina y Ciencias de la Salud
 - Grado en Enfermería
- Facultad de Filosofía y Letras
 - Grado en Historia del Arte

14.4.2 El área de Lenguajes y Sistemas Informáticos

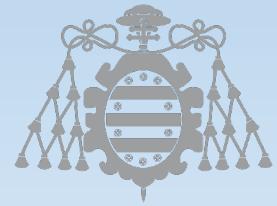
El concepto de área de conocimiento, introducido en la LRU, se define a efectos de las denominaciones de las plazas de la relación de puestos de trabajo de profesorado funcionario de cuerpos docentes universitarios en el Artículo 71 de la LOU así como en el Artículo 2 del Real Decreto 1888/1984, de 26 de septiembre, por el que se regulan los concursos para la provisión de plazas de los cuerpos docentes universitarios (Boletín Oficial del Estado de 16 de octubre de 1984), como aquellos campos del saber caracterizados por la homogeneidad de su objeto de conocimiento, una tradición histórica y la existencia de comunidades de personal docente e investigador, nacionales e internacionales.

El área de Lenguajes y Sistemas Informáticos abarca los conocimientos relacionados con los lenguajes de programación, el desarrollo de software, los sistemas operativos, las bases de datos, diseño de interfaces de usuario, ingeniería del software, etc. En la Universidad de Oviedo, el área de Lenguajes y Sistemas Informáticas ha estado ligada al área de Ciencias de la Computación e Inteligencia Artificial y de hecho, una gran mayoría de las asignaturas aparecen adscritas a ambas áreas de conocimiento.

Investigación

En el área LSI de la Universidad de Oviedo existen los siguientes grupos de investigación, que han obtenido evaluación externa y han sido aprobados por el Consejo de Gobierno de la Universidad de Oviedo.

- Computational Reflection, cuyo investigador principal es Francisco Ortín Soler. Sus líneas de investigación prioritarias son el desarrollo de sistemas adaptables, análisis de lenguajes de programación, sistemas de tipos, lenguajes dinámicos, máquinas abstractas, etc.
- Ingeniería del Software (GIIS), cuyo investigador principal es Javier Tuya González. Sus principales objetivos son las metodologías y herramientas para la calidad del software, especializado principalmente en las pruebas tanto del software como de las bases de datos.
- Ingeniería dirigida por modelos (MDE-RG), cuyo investigador principal es Juan Manuel Cuevas Lovelle y que tiene como principales líneas de investigación la ingeniería Dirigida por Modelos, el desarrollo de Software Dirigido por Modelos, la Internet de los Objetos, Modelado de Software Adaptable en Ingeniería Web, Lenguajes de Dominio Específico (DSL), modelado de Procesos de Negocio (BPM), arquitecturas dirigidas por modelos (MDA), modelos e Ingeniería Web, usabilidad, accesibilidad y auditoría Web, modelos para Ingeniería y modelado de aplicaciones para dispositivos móviles y ubicuos, así como para redes sociales y social media.
- Web Semántica Oviedo (WESO), cuyo investigador principal es el autor de esta memoria, Jose Emilio Labra Gayo, y cuyas líneas prioritarias son la aplicación de tecnologías de web semántica en diferentes dominios, la validación y representación del conocimiento, los servicios semánticos colaborativos, los datos abiertos y enlazados, y el análisis de datos sociales. El grupo WESO será descrito en más detalle en 18.2



Universidad de Oviedo

15. Concepto y Referencias Curriculares

En este capítulo se comenzará describiendo qué se entiende por *arquitectura del software*, repasando algunas de las definiciones que se han propuesto para el mismo, así como una breve introducción a la evolución de la disciplina.

Posteriormente, se describirán las principales referencias curriculares que se han tenido en cuenta para la elaboración de la propuesta docente, así como una revisión de cómo se imparte la asignatura en otras universidades internacionales y nacionales.

15.1 ¿Qué es la Arquitectura del Software?

La definición de qué es la arquitectura del software ha sido objeto de varios debates y controversias [4, 82, 85]. En [45], Gregor Hohpe indica:

Definir qué es la arquitectura, no es una tarea fácil, parece que hay casi tantas definiciones de arquitectura en el campo de tecnologías de la información, como arquitectos que la practican.

De hecho, el *Software Engineering Institute* mantiene una página de referencia sobre definiciones de arquitectura del software¹.

A continuación se incluyen algunas definiciones representativas:

El documento ISO/IEC/IEEE 42010:2011 *Systems and software engineering — Architecture description*² contiene las siguientes definiciones:

Arquitectura: conceptos o propiedades fundamentales de un sistema en su entorno, representado por sus elementos, las relaciones entre ellos, y los principios de su diseño y evolución.

En dicha definición pueden destacarse los siguientes aspectos:

- La arquitectura es una parte fundamental de un sistema de software
- Un sistema software está situado en un entorno y su arquitectura tiene en consideración el entorno en el que debe operar
- Una arquitectura es representada mediante elementos y relaciones entre los mismos, con el fin de comunicar a las personas interesadas cómo dicha arquitectura alcanza las necesidades del sistema.

¹<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=513807>

²<https://www.iso.org/standard/50508.html>

- La arquitectura también representa los principios de diseño y la evolución de los mismos, que suele ser realizado mediante la documentación de las principales decisiones tomadas.

Dicho documento también incluye la definición de *architecting* que podríamos traducir como *realizar una arquitectura*:

Realizar una arquitectura o architecting: proceso de concebir, definir, expresar, documentar, comunicar, certificar una implementación apropiada, mantener y mejorar una arquitectura a través del ciclo de vida de un sistema

De esta forma, también puede destacarse que la realización de una arquitectura no es algo que sea propio únicamente de las etapas iniciales de un sistema, sino que abarca el ciclo de vida del mismo, y que las tareas que realiza un arquitecto del software abarcan desde la concepción y definición de la arquitectura, hasta la certificación de la implementación y mantenimiento de la arquitectura, pasando por la documentación y comunicación de la misma. Este aspecto de comunicación y mantenimiento de la arquitectura puede ser crucial para la adopción de la misma, y conlleva habilidades no solo técnicas, sino también comunicativas y de gestión de equipos.

Una cita bastante popular sobre qué es la arquitectura del software y en qué se diferencia del diseño del software fue realizada por Grady Booch:³:

Toda arquitectura es diseño pero no todo diseño es arquitectura. La arquitectura representa las decisiones de diseño significativas que dan forma al sistema, donde significativo se mide por el coste de cambio.

Esta definición captura una idea importante respecto al trabajo que debe realizar un arquitecto del software consistente en la toma de decisiones significativas desde el punto de vista arquitectónico, así como presentar el aspecto del cambio y el coste de cambiar dichas decisiones.

En el libro *Software architecture in practice* [6] se destacan los siguientes aspectos de la arquitectura del software:

- La arquitectura es un conjunto de estructuras de software. La arquitectura se refiere a un conjunto de elementos unidos mediante una serie de relaciones. En el libro se identifican tres tipos de estructuras:
 - Estructura modular, cuyos elementos, conocidos como módulos, tienen sentido durante la implementación del sistema.
 - Estructura dinámica, que se refiere a los componentes que se pueden identificar en tiempo de ejecución
 - Disposición (*allocation*) que se refiere a cómo se pueden relacionar las estructuras de software con el resto del entorno, teniendo en cuenta aspectos organizativos y de desarrollo así como entornos de instalación y ejecución del software.
- La arquitectura es una abstracción. Puesto que la arquitectura trabaja con elementos y relaciones entre dichos elementos, es necesario omitir ciertos detalles de dichos elementos que no sean útiles para poder razonar sobre los mismos. De esta forma, la arquitectura es una abstracción que debe gestionar la complejidad de los sistemas mediante la omisión de detalles que no sean significativos. Habitualmente, la arquitectura trabaja sobre los interfaces de los módulos y componentes, omitiendo aspectos de la implementación interna de los mismos para poder gestionar la complejidad del sistema.
- Todo sistema software tiene una arquitectura del software. Todo sistema está compuesto de elementos y relaciones entre los mismos que soportan un cierto nivel de razonamiento. No obstante, aunque todo sistema tenga una arquitectura, esto no implica que la arquitectura sea conocida por todo el mundo. Es posible que las personas que diseñaran la arquitectura ya no formen parte del equipo, que la documentación ya no esté disponible, o nunca lo haya estado, que el código fuente se haya desviado de la arquitectura pretendida, etc. Puesto que una arquitectura puede existir de forma independiente a su descripción o especificación, es importante afrontar técnicas de gestión de la documentación de la arquitectura.
- La arquitectura incluye aspectos del comportamiento, que define cómo los elementos interactúan entre sí. La especificación de este comportamiento forma parte de la arquitectura e implica que presentar una arquitectura como un simple diagrama de cajas y líneas no es en sí una arquitectura. La cual debe

³Publicado en un post en su blog del año 2006 disponible en <https://web.archive.org/web/20110604175726/http://www.handbookofsoftwarearchitecture.com/index.jsp?page=Blog&part=All>

reflejar las interacciones y el comportamiento de los elementos y sus interacciones. Esto no implica que la descripción de una arquitectura deba contener todos los detalles, puesto que algunos de estos detalles pueden no ser significativos, pero en el momento que el comportamiento de un elemento sea significativo para que el sistema pueda ser aceptado, la arquitectura debe reflejarlo.

- No todas las arquitecturas son buenas arquitecturas. Las definiciones de arquitectura presentadas son indiferentes de la cualificación de que sean *buenas o malas*. Una arquitectura está formada por un conjunto de decisiones significativas que permitirán mejorar o no ciertos aspectos del sistema. Por ese motivo, un aspecto relevante de la disciplina es disponer de técnicas que permitan tomar dichas decisiones y evaluar si las decisiones tomadas mejoran o no ciertos aspectos de la calidad deseada de un sistema. Estos aspectos que están más relacionados con la calidad del sistema que con su funcionalidad se conocen como *atributos de calidad* y su gestión es uno de los aspectos más representativos de la arquitectura del software.

15.2 Evolución de la Arquitectura del Software

El concepto de arquitectura del software es una metáfora tomada a partir de la arquitectura tradicional de edificios, llevada al mundo del software. Aunque como toda metáfora, hay aspectos que no pueden tomarse literalmente, otros aspectos pueden servir de inspiración para entender mejor la disciplina.

Etimológicamente, la palabra arquitectura deriva del griego *ἀρχιτέκτων* donde *ἀρχι* – podría definirse como *jefe* y *τέκτων* como *creador*, de forma que arquitectura se refiere a las actividades realizadas por el creador jefe de una obra. La edición inglesa de la Wikipedia define arquitectura como *el proceso y producto de planificar, diseñar y construir edificios u otras estructuras*, lo cual puede extrapolarse a la arquitectura del software entendiendo el software como una de esas *otras estructuras* que son planificadas, diseñadas y construidas. Al cuestión de si la arquitectura es un arte o una ciencia ha sido planteada desde la época del renacimiento y la respuesta actual suele ser que es ambas cosas [64] unidas gracias a la ingeniería.

Una de las primeras referencias escritas sobre la arquitectura se debe a la obra de Vitruvio titulada *De Arquitectura*, escrita entre los años 30 a. C. y 15 a. C. y que define los 3 pilares de una arquitectura:

- *Utilitas* o Utilidad: Ser útil y funcionar bien para las personas que lo van a usar
- *Firmitas* o durabilidad: Mantenerse de forma robusta y en buena condición
- *Venustas* o elegancia: Ser agradable a las personas

Es interesante observar que dichos pilares podrían aplicarse de igual forma tanto a los edificios como al software.

Una de las primeras referencias al término *arquitectura del software* parece que se realiza en el marco de una Conferencia sobre técnicas de ingeniería del Software organizada por la OTAN en el año 1968. Algunas de las personalidades más destacadas del campo, como T. Hoare, E. Dijkstra, A. Perlis, F. Bauer, N. Wirth participaron en dicha conferencia. En los proceedings de dicha conferencia se atribuye a P. Naur la frase:

Los diseñadores del software están en una posición similar a la de los arquitectos o ingenieros civiles, particularmente aquellos que afrontan el diseño de construcciones grandes y heterogéneas, como ciudades y plantas industriales. Parece natural, por tanto, que observemos a dichas disciplinas por ideas sobre cómo atacar el problema del diseño. A modo de ejemplo da tal fuente de ideas, me gustaría mencionar a Christopher Alexander: Notes on the Synthesis of Form (Harvard Univ. Press)

Dicha afirmación no solamente es importante por considerar relevante la arquitectura como fuente de inspiración para el desarrollo del software, sino que también menciona el trabajo de Christopher Alexander que posteriormente servirá de inspiración para las propuestas basadas en patrones de diseño. No obstante, durante los años posteriores, el término arquitectura en informática solía asociarse a los aspectos físicos de la arquitectura de los sistemas computacionales.

Durante los años 70 se producen numerosos hitos que servirán para formar las bases de la disciplina, entre los que se pueden destacar:

- 1968 M. Conway enuncia lo que se conocerá posteriormente como Ley de Conway sobre la estructura de las organizaciones y los diseños que éstas producen.
- 1972 D. L. Parnas artículo sobre descomposición modular y ocultación de la información
- 1974 B. Liskov introduce el concepto de tipos abstractos de datos
- 1975 F. Brooks publica el libro *The Mythical Man Month*

- 1976 P. Chen introduce el concepto de modelos entidad-relación.

Un aspecto relevante en la historia de la disciplina durante los años 70 es la discusión sobre si la programación de ordenadores era un arte o una ciencia. Uno de los principales defensores de la programación como un arte es D. Knuth cuando propone que [56]:

La programación de ordenadores es un arte, porque aplica conocimiento acumulado al mundo, porque requiere habilidad e ingenio y especialmente, porque produce objetos de belleza.

Por otro lado, a finales de los años 70 comienzan a popularizarse metodologías para el análisis y diseño de software estructurado por D. Ross, L. Constantine, E. Yourdon, M. Jackson y T. DeMarco.

En los años 80 comienzan a introducirse los computadores personales y la necesidad de creación de software fiable que funcione en los mismos se acrecienta.

A finales de los años 80 se populariza el concepto de ingeniería del software basado en componentes y empiezan a aparecer trabajos que analizan la economía de los artefactos desarrollados en ingeniería del software así como los modelos de calidad [8]. Durante esa década, G. Booch, J. Rumbaugh e I. Jacobson habían propuesto diversas notaciones para el modelado de software que se unirán en lo que se conocerá como lenguaje unificado de modelado ó UML, el cual que tendrá una gran repercusión posterior para el modelado de sistemas de software. A principios de los 80 aparecieron las primeras herramientas CASE (Computer Aided Software Engineering) que alcanzarían bastante popularidad durante esa década y la siguiente [24].

Aunque el concepto de programación orientada a objetos ya había sido introducido en los años 60 con el lenguaje Simula y posteriormente con Smalltalk, será en la década de los 80 cuando realmente empiece a despegar y a ser adoptado industrialmente con la aparición de C++ en el año 1985, Eiffel en el año 1988 y posteriormente Java en 1991.

En 1995 se publica el Libro *Design patterns* [34] que tendrá una gran influencia, especialmente desde finales de los años 90, con numerosos libros inspirados en la idea de patrones, como la serie de 4 libros sobre arquitectura del software basada en patrones [13, 14, 55, 78], el libro sobre patrones de arquitectura empresarial [33], patrones de integración empresarial [46], patrones organizativos [21], e incluso de arquitectura del software basada en patrones para dummies [42].

En 1996, M. Shaw and D. Garlan publican un primer libro dedicado íntegramente a la arquitectura del software [80] que sienta las bases para la disciplina.

La etapa que va desde 1995 a 2000 podría considerarse como la etapa dorada de la arquitectura del software [19], en dicha etapa se plantean los catálogos de estilos arquitectónicos [70], se proponen lenguajes de descripción de arquitecturas [63], y se plantea el modelo de vistas de arquitectura [58]. En esa etapa el Instituto de Ingeniería del Software de Carnegie-Mellon propone el método ATAM (Architecture Tradeoff Analysis Method) [51] para la creación de arquitecturas del software basadas en el análisis de atributos de calidad y se publican varios libros que tendrán gran relevancia como [6] (la primera edición es de 1998), [44] y [36].

Por otro lado, a finales de los 90 se estaba creando una corriente de opinión en el desarrollo del software preocupada por la calidad del mismo y la necesidad de aumentar la velocidad de creación de nuevos productos software. En 1999 se publican dos libros que tendrán gran influencia: *Extreme programming explained* de Kent Beck [53] plantea una serie de prácticas de desarrollo como el desarrollo basado en pruebas o *test-driven development*, la programación en parejas, etc. Ese mismo año, Martin Fowler también publica el libro *Refactoring* [32] que propone la creación de técnicas que permitan mejorar la calidad de un sistema manteniendo su comportamiento.

En el año 2001 se organizó una reunión en el complejo hotelero The Lodge en Snowbird, Utah en la que participaron varias personas relevantes de la industria del desarrollo del software como Martin Fowler, Kent Beck, Ward Cunningham, Alistair Cockburn, Robert Martin, etc. A modo de conclusiones, acuerdan la creación de un documento firmado por todos ellos, que se bautizó como el *manifesto ágil*⁴.

En algunos contextos, dichas directivas fueron tomadas como una reacción a las propuestas de desarrollo de software basadas en modelos en cascada, que se entendía que proponían grandes documentos de especificación de requisitos y descripciones arquitectónicas en lo que se llegó a denominar de forma despectiva como *big design up-front*.

La comunidad de desarrollo software fue adoptando las prácticas ágiles y van apareciendo libros que promueven dichas prácticas y adquieren gran popularidad como [59, 79].

⁴<https://agilemanifesto.org/>

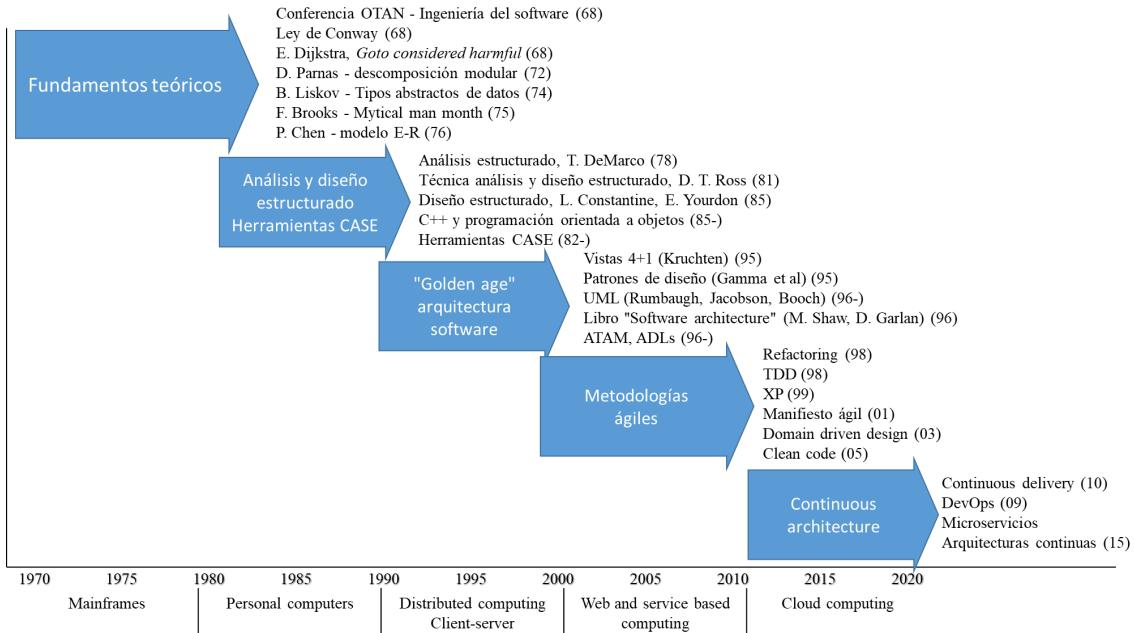


Figura 15.1: Principales etapas en la evolución de la arquitectura del Software

En 2010 se comienza a plantear la posibilidad de que realmente puedan coexistir la arquitectura del software con las metodologías ágiles [1] y empiezan a aparecer libros que dan una visión de la arquitectura del software compatible con las metodologías ágiles [12, 28].

Un aspecto relevante en la industria informática en la década de 2010 será la aparición del concepto de computación en la nube, popularizado por grandes corporaciones como Amazon, Google o Microsoft. También surgen plataformas que facilitan la gestión ágil de proyectos software con control de versiones, integración continua, revisión de código, etc. como *github*⁵ que facilitan la adopción de las metodologías ágiles de desarrollo y popularizan técnicas como el despliegue continuo de software [49] y las propuestas conocidas como *DevOps* [5, 54].

La utilización de la web como plataforma popularizó la creación de aplicaciones basadas en servicios web, que se englobarían en lo que se denominaría arquitecturas orientadas a servicios. Una evolución de dicho concepto se comienza a utilizar en el año 2012 con el nombre de microservicios⁶, que son aplicaciones basadas en servicios que se despliegan de forma independiente y permiten un control descentralizado de los datos y lenguajes con los que están implementadas.

La popularidad del estilo arquitectónico conocido como microservicios ha dado lugar a la aparición de un nuevo interés en la industria por aspectos de gestión y organización del software con la aparición de numerosos libros especializados [66, 67, 73] en microservicios que necesariamente explican conceptos de arquitectura del software.

En la línea de creación de sistemas informáticos en la nube, están adquiriendo cada vez más importancia aspectos como la seguridad, la fiabilidad y la estabilidad de los sistemas. El libro *Release it!* publicado inicialmente en 2007, ha publicado una segunda edición en 2018 [68] que añade temas como la ingeniería del caos [16]. Estos aspectos también se afrontan en la colección de libros desarrollada por ingenieros de Google [2, 7, 86].

Desde mediados de 2015 comienzan a aparecer libros sobre arquitectura del software que ofrecen una visión contemporánea de la arquitectura del software compatible con las metodologías ágiles de desarrollo y con las últimas tendencias basadas en microservicios, que se pueden bautizar como arquitecturas continuas o evolutivas, entre los que se pueden destacar [25, 31, 45, 65, 71].

En la figura 15.1 se han destacado algunos hitos que se han considerado relevantes para la evolución

⁵<http://github.com/>

⁶<https://martinfowler.com/articles/microservices.html>

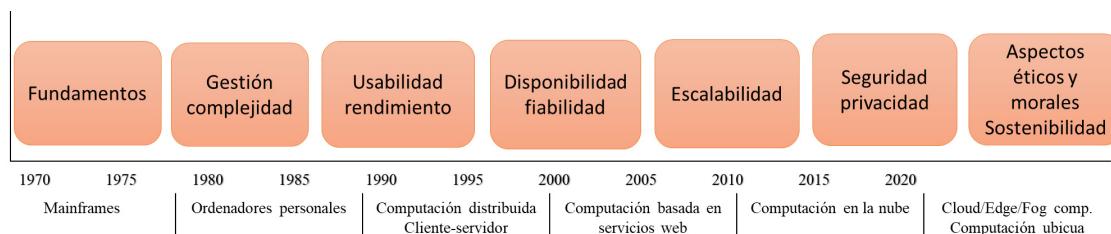


Figura 15.2: Aspectos afrontados por la ingeniería del software

de la disciplina de la arquitectura del software, junto con las fechas correspondientes. Los años deben considerarse simples aproximaciones basadas en algunos datos como fechas de publicación de los libros o de alguna herramienta representativa. Hay que tener en cuenta que muchas ideas pueden tardar un tiempo en popularizarse desde el momento en que se publican o pueden haber surgido antes de que se hayan publicado. En la figura también se ha incluido una línea inferior con los hitos tecnológicos más relevantes en cada año que permite establecer un cierto paralelismo entre los avances en la arquitectura del software y los avances tecnológicos correspondientes.

Respecto a cuál puede ser el futuro de la arquitectura del software, aunque puede ser difícil realizar estimaciones, parece claro que existe un cierto péndulo en cuanto a propuestas que pasa desde la computación en grandes computadores a la computación en pequeños sistemas, dicha evolución puede verse en cómo se pasó de la etapa de los grandes computadores (*mainframes*) a la etapa de computadores personales, pasando de nuevo a la etapa de computación cliente-servidor con servidores más grandes, que regresa a la computación basada en servicios web y vuelve a la computación en la nube donde se podría decir que es la etapa en la que se está en el año 2020. No obstante, ya han surgido propuestas que promueven la vuelta a la computación en los dispositivos finales (*edge computing*) e incluso sistemas híbridos conocidos como *fog computing*.

La propuesta de avanzar hacia arquitecturas evolutivas, que se adaptan a los inevitables cambios que se dan a lo largo de la vida de un sistema consideramos que se irá haciendo cada vez más popular en la industria.

En los últimos años, parece haber cierta coincidencia en revisar la figura del arquitecto del software, unida más a un rol que a una persona concreta, que se integra en equipos que siguen metodologías ágiles y que es capaz de participar en aspectos técnicos del proyecto, tomar decisiones y comunicarse con las personas interesadas. En ese sentido, se considera cada vez más relevante la adquisición de habilidades comunicativas y de negociación, aspectos que los libros más recientes sobre la disciplina están incorporando.

En un vídeo reciente sobre la evolución de la ingeniería del Software⁷, Grady Booch reflexiona sobre los aspectos que han sido afrontados en la disciplina a lo largo del tiempo, relacionados con las etapas tecnológicas. Así, durante las primeras etapas computacionales se resolvían aspectos de fundamentos algorítmicos y de teoría de la complejidad. Con la llegada de los ordenadores personales, hubo un gran interés en avanzar técnicas de interacción persona-máquina. Posteriormente, con la aparición de sistemas distribuidos y conectados, el interés ha pasado a aspectos que permitan gestionar la computación a gran escala. En el futuro la tendencia será hacia una computación ubicua basada en sistemas informáticos dispuestos en el ambiente que nos rodea. En esta etapa, los aspectos éticos y morales de la disciplina cobrarán una mayor relevancia. A modo de ejemplo, empiezan a aparecer trabajos que analizan la sostenibilidad de sistemas software [15] o la toma de decisiones arquitectónicas basadas en aspectos éticos [3].

15.3 Recomendaciones curriculares

Las recomendaciones curriculares han sido elaboradas a través de un proceso de revisión, redefinición y asentamiento de la disciplina de la Informática como respuesta al rápido avance científico y tecnológico al que se ve sometida.

En este apartado se revisarán las siguientes propuestas:

⁷<https://www.youtube.com/watch?v=QUz10Z1AfLc>

- Informe conjunto de ACM (Association for Computing Machinery) e IEEE (Institute of Electrical and Electronics Engineers).
- Propuesta SWBoK del IEEE.
- Propuesta curricular de la UNESCO (United Nations Educational Scientific and Cultural Organization) e IFIP (International Federation for Information Processing): [Informatics Curriculum Framework 2000](#).

15.3.1 Informe conjunto de ACM e IEEE

ACM (Association for Computing Machinery), es la mayor y más antigua organización educacional y científica sobre computación existente. Desde 1947 ACM ha proporcionado un foro de intercambio de información, ideas y descubrimientos.

IEEE (Institute of Electrical and Electronics Engineers) promueve el proceso de ingeniería de creación, desarrollo, integración, compartición y aplicación del conocimiento sobre ciencias y tecnologías de la información para el beneficio de la profesión.

A continuación se resumen brevemente las propuestas curriculares propuestas por ambos organismos:

Curriculum 1968 de ACM. ACM fue la primera organización que realizó una definición rigurosa de los estudios de informática, por medio del informe conocido como Curriculum 68. En él se proponían los estudios básicos y los contenidos troncales, agrupados en asignaturas, para una carrera de Informática de 4 años, correspondiente a un primer ciclo en los países anglosajones.

Curriculum 1978 de ACM. El Curriculum 68 tuvo una notable influencia en los planes de estudios de numerosas universidades de todo el mundo. Más adelante se inició una revisión del mismo que culminó con la publicación del Curriculum 78.

Curriculum 1984 de ACM. En 1984, la ACM publicó una leve modificación de la propuesta anterior con el objeto de reflejar en ella los avances de la Informática en esos años. La crítica al Curriculum 84 se basa generalmente en el fuerte peso que tienen las asignaturas relacionadas con la programación, en contraposición con las dedicadas a la tecnología de computadores.

La sociedad de Informática de IEEE-CS (IEEE Computer Society) ha realizado propuestas con una perspectiva algo diferente a la de ACM. Se han realizado tres recomendaciones, una en 1977, otra en 1983, y a partir de 1991, han realizado propuestas junto con ACM. Si las recomendaciones de ACM fueron criticadas por dar excesivo peso a la programación y la teoría, el modelo de IEEE-CS ha recibido críticas en el sentido contrario, es decir, por estar más orientado a la Tecnología de Computadores.

Curriculum 1991 de ACM/IEEE. A finales de los 80, ambas entidades, la ACM y la IEEE-CS, unieron sus fuerzas para llevar a cabo una revisión más profunda de los Curricula, resultando de ello el Computing Curricula 1991 (CC91). Unifica, por tanto, las recomendaciones de las dos sociedades más importantes de la informática. A diferencia del Curriculum 78 y el de 83 de la IEEE-CS, que se centraban en ofrecer temarios para cursos individuales, el CC91 divide el campo de conocimiento asociado a la informática en áreas de conocimiento individuales. Cada área corresponde a una materia que debe cubrir el currículum en algún momento del programa que lo desarrolle, aunque se deja una flexibilidad considerable para que cada institución establezca la estructura de los cursos que han de desarrollar las áreas de conocimiento de acuerdo a sus propias necesidades. El propio apéndice del CC91 incluye 11 ejemplos de distintas formas de combinar las áreas para formar cursos según las necesidades de distintas instituciones.

Curriculum 2001 de ACM/IEEE (CC2001). En 1998 ACM e IEEE-CS establecieron de nuevo un equipo conjunto, denominado Computing Curricula 2001 (CC2001), cuyo objetivo era revisar el CC91 y desarrollar una versión del mismo que considerase los avances de las tecnologías de computación en la última década, y con el fin de que perduren a lo largo de la siguiente década.

Curriculum 2005 de ACM/IEEE (CC2005). En CC2001 realizaron un conjunto de informes sobre las grandes familias de las disciplinas en computación. A partir de estos informes en Computing Curricula 2005 se establecieron cinco especialidades de la Informática:

- Computer Science (Ciencias de la Computación)
- Computer Engineering (Ingeniería de Computadores)
- Information Systems (Sistemas de Información)
- Information Technology (Tecnología de la Información)
- Software Engineering (Ingeniería del Software), el primer informe en 2004 (SE2004)

Curriculum 2014 de ACM/IEEE en Ingeniería del Software (SE2014). Se realiza una revisión detallada de SE2004, actualizando secciones, incorporando por ejemplo los nuevos ciclos de vida del proceso de desarrollo de software. Se revisará en detalle a continuación.

Área/Unidad	Título	Horas
DES	Software Design	48
DES.con	Design concepts	3
DES.str	Design strategies	6
DES.ar	Architectural design	12
DES.hci	Human-computer interaction design	10
DES.dd	Detailed design	14
DES.ev	Design evaluation	3

Cuadro 15.1: Unidades del área de diseño del software

Curriculum 2020 de ACM/IEEE (CC2020). Se trata de un informe actualizado que añade la disciplina de Ciberseguridad.

Software Engineering Computing Curricula

En cuanto a la Ingeniería del Software el estado y evolución de los informes realizados ha sido:

- SE2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering publicado en Febrero de 2015 ⁸
- GSWE2009: Curriculum Guidelines for Graduate Degree Programs in Software Engineering incluye las guías para las titulaciones de nivel de máster.
- SE 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering

El *Software Engineering Education Knowledge* (SEEK o SE2014), se organiza en tres niveles. El nivel más alto de la jerarquía lo constituye el área, que representa una especialidad o disciplina particular en Ingeniería del Software. Las áreas son divididas en unidades, que representan módulos temáticos individuales dentro de éstas, y finalmente, cada unidad es dividida en un conjunto de temas.

Identifica un núcleo, constituido por un conjunto de unidades sobre las que hay un consenso generalizado en cuanto a su necesidad en la formación de un titulado. Aquellas unidades que no sean núcleo, se definen como optativas.

El núcleo no representa un currículum completo, por lo que además de definir el núcleo define implementaciones detalladas de cursos: agrupación de unidades, impartidas a lo largo de 15 semanas, 3 veces a la semana, con una duración de 50 a 60 minutos cada clase.

A continuación se describe la disciplina de Arquitectura del Software, identificando en primer lugar el área donde se encuentra enclavada, para a continuación identificar las unidades que constituyen el núcleo de la misma. Finalmente, se describen los cursos propuestos que se encuentran relacionados con la disciplina.

Áreas

El cuerpo de conocimiento correspondiente a Ingeniería del Software (SEEK), se encuentra organizado en torno a 10 áreas, entre las que se puede destacar **Diseño de software (DES)**, que es la que puede estar más relacionada con **Arquitectura del Software**.

- Computing essentials (CMP)
- Mathematical and engineering fundamentals (FND)
- Professional practice (PRF)
- Software modeling and analysis (MAA)
- Requirements analysis and specification (REQ)
- Software design (DES)
- Software verification & validation (VAV)
- Software process (PRO)
- Software quality (QUA)
- Security (SEC).

En la tabla 15.1 se resumen las unidades de conocimiento del área de Software Design (DES) y se enumera el número mínimo de horas recomendadas para cada área y unidad.

En SE2014 se especifican los temas dentro de la unidad concreta de Architectural design (DES.ar) indicando el nivel de habilidad cognitiva para cada uno y su relevancia siguiendo la siguiente pauta.

⁸Disponible en: <https://www.acm.org/binaries/content/assets/education/se2014.pdf>

- K - Conocimiento: recordar el material previamente aprendido, la observación de prueba y recuerdo de la información; es decir, *traer a la mente la información apropiada* (por ejemplo, fechas, eventos, lugares, el conocimiento de las principales ideas y dominio de la materia).
- A - Aplicación (a): El uso de material aprendido en situaciones nuevas y concretas. Por ejemplo, el uso de información, los métodos, conceptos y teorías para resolver problemas que requieren las habilidades o conocimientos presentados.

La relevancia de un tema para el núcleo se designa de una manera similar:

- E - Esencial (E): El tema es parte del núcleo.
- D - Deseable (D): El tema no es parte del núcleo, pero debe ser incluido en el núcleo de un programa en particular, si es posible; de lo contrario, se debe considerar parte del electiva materiales.

Esta especificación para cada uno de los temas que componen la unidad de Architectural design (DES.ar), que es la más relacionada con el perfil de la plaza se muestra en la tabla 15.2:

Referencia		k,c,a	E,D
DES.ar	Architectural Design		
DES.ar1	Architectural styles, patterns, and frameworks	a	E
DES.ar2	Architectural trade-offs among various attributes	a	E
DES.ar3	Hardware and systems engineering issues in software architecture	k	E
DES.ar4	Requirements traceability in architecture	k	E
DES.ar5	Service-oriented architectures	k	E
DES.ar6	Architectures for network, mobile, and embedded systems	k	E
DES.ar7	Relationship between product architecture and the structure of development organization and market	k	E

Cuadro 15.2: Habilidad cognitiva y carácter de los temas de la unidad Architectural design (DES.ar)

Como se ve muestra, SE2014 considera que todos los contenidos de la unidad de Architectural design (DES.ar) forman parte del núcleo y a nivel cognitivo los clasifica como de conocimiento y aplicación.

15.3.2 Software Engineering Body of Knowledge (SWBoK)

El Software Engineering Institute (SEI) es un instituto federal estadounidense de investigación y desarrollo, fundado por Congreso de los Estados Unidos en 1984 para desarrollar modelos de evaluación y mejora en el desarrollo de software, está Financiado por el Departamento de Defensa de los Estados Unidos y administrado por la Universidad Carnegie Mellon.

El *Software Engineering Coordinating Committee*, promovido por la *IEEE Computer Society* ha definido desde 1998 el *Software Engineering Body of Knowledge* (SWEBoK)⁹ [9] cuyo objetivo es establecer los conjuntos de criterios apropiados y las normas para la práctica profesional de la ingeniería del software, sobre las que se puedan basar las decisiones industriales, la certificación profesional y los currículos.

El propósito del SWEBoK es identificar y describir el conocimiento generalmente aceptado en la ingeniería del software, pero no establece un currículum para los estudios del ámbito de la Ingeniería del software, como así lo remarcan sus autores. En este apartado se estudia el SWEBoK porque a partir de él se puede definir la profesión y diferentes currícula para los estudios.

SWEBoK identifica quince áreas de conocimiento que constituyen el núcleo de conocimiento de la ingeniería del software:

1. Requisitos de Software
2. Diseño de Software
3. Construcción de Software
4. Pruebas de Software

⁹<https://www.computer.org/education/bodies-of-knowledge/software-engineering/>

5. Mantenimiento de Software
6. Gestión de la configuración
7. Gestión de la Ingeniería de Software
8. Proceso de Ingeniería de Software
9. Herramientas y métodos de la Ingeniería de Software
10. Calidad del Software
11. Práctica Profesional de la Ingeniería de Software
12. Economía de la Ingeniería de Software
13. Fundamentos de Computación
14. Fundamentos Matemáticos
15. Fundamentos de Ingeniería

El SEI, basado en las primeras versiones del SWEBoK, publicó en 1999 las *Guidelines for Software Engineering Education* que definen un currículum completo para la disciplina de la Ingeniería del Software. Se establece que todos los currículos del ámbito de la ingeniería del software deben contemplar seis áreas, entre las que destaca el área de Ingeniería del Software, que incluye:

- Ingeniería de requisitos Software
- Diseño de Software
- Calidad del Software
- Arquitecturas del Software
- Construcción de Software
- Evolución del Software
- Métodos Formales
- HCI (Human-Computer Interaction)
- Organización de proyectos, planificación y seguimiento
- Procesos Software
- Aspectos profesionales y éticos de la Ingeniería del Software

En el área de conocimiento 2, dedicado al diseño del software se incluyen los siguientes capítulos:

- Fundamentos de diseño de software
- Temas clave en Diseño del Software
- Estructura y arquitectura del software
- Diseño de interfaces de usuario
- Análisis y evaluación de la calidad del diseño de software
- Notaciones de diseño de software
- Métodos y estrategias de diseño de software
- Herramientas de diseño de software

En el capítulo 3 sobre estructura y arquitectura del software se mencionan los apartados que están muy relacionados con la arquitectura del software:

- Estructura y puntos de vista arquitectónicos
- Estilos arquitectónicos
- Patrones de diseño
- Familias de programas y *frameworks*

Aunque en otros capítulos se incluyen otros temas que también se pueden relacionar con la arquitectura del software como en el capítulo 5, que se incluyen apartados sobre atributos de calidad o el capítulo 6, sobre notaciones de diseño que podrán utilizarse para la documentación de arquitecturas.

15.3.3 Propuesta curricular UNESCO-IFIP

La UNESCO (United Nations Educational, Scientific and Cultural Organization) es una organización constituida en la Conferencia de Londres en noviembre de 1945, en la que participan, en su inicio 20 estados miembros. Actualmente cuenta con 186 países miembros.

El principal objetivo de la UNESCO es el de contribuir a la paz y la seguridad en el mundo, promoviendo la colaboración entre naciones a través de la educación, la ciencia, la cultura y la comunicación con el objeto de establecer un respeto universal por la justicia, los derechos humanos y las libertades fundamentales.

En el año 2000, UNESCO en colaboración con IFIP (International Federation for Information Processing) crea el ICF-2000 [ICF, 2000]: *Informatics Curriculum Framework*, que puede ser visto como el sucesor de la propuesta previa de UNESCO de 1994 [UNESCO, 1994], pero con un ámbito más amplio que el anterior y, además, se beneficia de los esfuerzos realizados por otros currículos.

El objetivo de ICF-2000 es abordar la situación de constante cambio a la que se enfrenta la informática. Como su nombre indica se trata de un marco a partir del cual se pueden construir diferentes implementaciones de currícula.

Perfiles y categorías

ICF-2000 propone ocho categorías profesionales diferentes, y cuatro perfiles de graduados para satisfacer de forma eficiente las necesidades educacionales de estas categorías profesionales. Dichas categorías se representan en la tabla 15.3.

Perfil	Categorías
BIP (Basic Instrumental I-Profile)	Usuarios instrumentales
BCP (Basic Conceptual I-Profile)	Usuarios de aplicaciones conceptuales
MIP (Minor I-Profile)	Usuarios de aplicaciones de interfaz Usuarios de aplicaciones de investigación Usuarios de aplicaciones de dirección
MAP (Major I-Profile)	Trabajadores operacionales Trabajadores de ingeniería Trabajadores de investigación

Cuadro 15.3: Perfiles asociados a las categorías profesionales ICF-2020

De entre las ocho categorías profesionales, las más relacionadas con la Titulación de Ingeniero en Informática son las siguientes:

- Trabajadores Operacionales. Requieren una formación informática especializada en el área de explotación, control y mantenimiento de tecnología y aplicaciones informáticas. Dos ejemplos son administradores de bases de datos u operadores de ordenadores.
- Trabajadores de Ingeniería. Deben tener una comprensión general y unas habilidades bien desarrolladas en informática como una disciplina amplia, especialmente en el área de análisis, diseño e implementación. A esta categoría pertenecen los analistas de sistemas de información, ingenieros de software o programadores científicos.
- Trabajadores de Investigación. Requieren una comprensión general y unas habilidades bien desarrolladas en informática, así como una disciplina amplia, especialmente, como un campo de investigación. Se supone que son quienes desarrollarán la disciplina informática y sus conceptos, tanto de forma individual como relacionándola con otras disciplinas, colaborando con investigadores de otras áreas.

Tal y como señala el propio marco, en el caso de una universidad con un departamento de informática consolidado, el perfil más interesante es el MAP, ya que incluye estas tres categorías.

Unidades

Cada uno de los perfiles está constituido por una serie de unidades, que son bloques de tamaño variable y que, entre otras cosas, hacen referencia a otros currícula informáticos. En el caso del MAP, el perfil consta de las 16 unidades que se indican en la tabla 15.4

De entre las unidades anteriores, se pueden destacar las unidades: MAP-03 (arquitectura de sistemas software) y MAP-13 (Desarrollo del Software) por su relación con la Arquitectura del Software.

La unidad MAP-03 indica dos competencias:

- Capacidad de especificar y diseñar programas informáticos (o componentes de programas más grandes) de tamaño pequeño o moderadamente complejo
- La actitud de dominar el proceso de diseño de programas informáticos, desde un punto de vista tanto conceptual como pragmático.

La unidad MAP-13 indica 2 competencias:

- Capacidad de contribuir sustancialmente al diseño y la ingeniería de software seleccionando un enfoque, metodología o herramienta concreta
- La actitud de dominar el proceso de ingeniería de software.

1. Contexto de las aplicaciones informáticas	9. Sistemas y procesamiento distribuido
2. Arquitectura de los sistemas de información	10. Calidad y Seguridad
3. Arquitectura de los sistemas Software	11. Actuando como un profesional
4. Arquitectura de los Sistemas de Computadores	12. Modelización y Desarrollo de Sistemas
5. Redes y Comunicaciones	13. Desarrollo de software
6. Formalismo en el procesamiento de la información	14. Miscelánea / Estado del Arte
7. Interacción y Presentación	15. Optativo entre a) Aspectos Operacionales b) Principios de Ingeniería y c) Metodología de Investigación
8. Inteligencia Artificial	16. Proyecto final

Cuadro 15.4: Unidades del perfil MAP en ICF-2020

15.4 Otras referencias curriculares

15.4.1 IASA Global

IASA Global¹⁰ es una asociación internacional de arquitectos de tecnologías de la información establecida inicialmente en el año 2002 como la *International Association for Software Architects* ha ido aumentando su ámbito para abarcar arquitectos empresariales, de negocios, etc. Cuenta con más de 50 capítulos locales, incluido un capítulo en español¹¹. Proporciona una certificación para arquitectos con 4 niveles: Fundación, asociado, especialista y profesional. En IASA se ha creado un documento denominado ITABOK 3.0 (*Information Technology Architecture Body of Knowledge*) con cuatro especialidades (arquitectura de negocios, arquitectura de información, arquitectura de infraestructuras y arquitectura del software).

El cuerpo de conocimiento incluye los siguientes 5 pilares:

- Estrategia de negocios de tecnología: fundamentos empresariales, desarrollo y racionalización de estrategias, análisis industrial, gestión de riesgos, etc.
- Dinámicas humanas: relaciones con clientes, liderazgo y gestión, habilidades de colaboración, negociación, presentación y escritura.
- Diseño: modelado de requisitos, descripciones de arquitectura, descomposición y reutilización, patrones y estilos de diseño, vistas y puntos de vistas, etc.
- Entorno TI: gestión de proyectos técnicos, gestión de activos, gestión de cambio, infraestructuras, desarrollo de aplicaciones, gestión de conocimiento, soporte a las decisiones, etc.
- Atributos de calidad: balanceo y optimización de atributos de calidad, monitorización y gestión, gestionabilidad, rendimiento, fiabilidad, escalabilidad, seguridad, usabilidad, etc.

La especialidad de arquitectura del software incluye los siguientes apartados:

- Procesos, metodologías y desarrollo
- Herramientas de arquitectura del software
- Ingeniería del Software para arquitectos
- Servicios, flujo de trabajo y mensajería
- Atributos de calidad avanzados
- Gestión de *stakeholders* avanzada
- Patrones de arquitectura del software
- Plataformas y *frameworks* tecnológicos
- Gestión de datos, información y conocimiento

15.4.2 iSAQB

iSAQB¹² (International Software Architecture Qualification Board) es una asociación de expertos en arquitectura de software de la industria, consultoría, capacitación y otras organizaciones.

¹⁰<https://iasaglobal.org/>

¹¹<https://es.iasaglobal.org/>

¹²<https://www.isaqb.org/>

El iSAQB es un organismo internacional que tiene como objetivo crear planes de estudio y de formación uniformes para arquitectos de software. Ofrece dos tipos de certificado profesionales: CPSA Foundational y CPSA Advanced (Certificado Profesional de Arquitectura de software fundacional y avanzado). Aunque la organización es internacional, está establecida fundamentalmente en Alemania y Austria.

En el año 2019 se publicó el libro *Software architecture fundamentals: a study guide for the Certified Professional for Software Architecture* [39] que ofrece una guía de estudio que abarca los principales temas de dicha certificación. La tabla de contenidos de dicho libro es:

- Fundamentos de arquitectura del software
- Diseñando arquitecturas del software
- Descripción y comunicación de arquitecturas del software
- Arquitecturas del software y calidad
- Herramientas para arquitectos de software

15.5 Arquitectura del software en otras universidades

Además de las propuestas curriculares que se han fijado como base para el desarrollo de la asignatura, se ha considerado también el estudio de referencias curriculares en otras universidades.

A la hora de encontrar cursos que imparten contenidos relacionados con la arquitectura del software, se han observado 3 tendencias principales:

- Dentro de una asignatura de *ingeniería del software*. Normalmente, la asignatura tiene una gran carga lectiva para poder dar cabida a otros conceptos de ingeniería del software que se combinan con los temas específicos de arquitectura del software.
- Dentro de una asignatura de *arquitectura y diseño del software*. En estos casos, suele ser una asignatura que unifica los conceptos de diseño y arquitectura del software.
- Como una asignatura independiente y específica de *arquitectura del software*. Este es el modelo que se ha seguido en el grado de Ingeniería Informática del Software de la Universidad de Oviedo y permite separar claramente los conceptos de diseño o de ingeniería del software de los conceptos de arquitectura propiamente dichos.

A través de la Web se ha podido acceder a la información sobre asignaturas que siguen los modelos indicados intentando incluir un ejemplo de cada una de ellas.

15.5.1 Arquitectura del software en universidades internacionales

Carnegie Mellon University

En la Universidad Carnegie Mellon se imparte el curso 18-653SV: *Software Architecture and Design*¹³ que incluye los siguientes tópicos:

- Repaso de arquitectura de software
- Patrones micro-arquitectónicos (patrones de diseño)
- Patrones macro-arquitectónicos (patrones modernos o arquitectónicos)
- Arquitectura Orientada a Servicios
- Modelado arquitectónico
- Puntos de vista y perspectivas
- Técnicas de análisis arquitectónico
- Tácticas arquitectónicas (Calidad de Servicios)
- Arquitecturas ágiles
- Otros tópicos avanzados

El *Software Engineering Institute* ofrece además un Certificado Profesional en Arquitectura del Software junto con varios cursos que permiten alcanzar dicha certificación, entre los que se encuentra el de *Software Architecture: Principles and Practices*¹⁴, que tiene los siguientes contenidos:

- Definición y repaso de arquitectura del software
- El ciclo de influencia de la arquitectura: qué influencia a los arquitectos y arquitectos del software
- Comprendiendo y alcanzando los atributos de calidad
- Diseño de arquitectura utilizando patrones y tácticas
- Documentando la arquitectura del software

¹³<https://courses.ece.cmu.edu/18653SV>

¹⁴<https://www.sei.cmu.edu/education-outreach/courses/course.cfm?coursecode=P35>

- Evaluando la arquitectura del software
- Reutilización de arquitectura mediante líneas de productos
- Arquitectura en proyectos ágiles

Rochester Institute of Technology: Software Engineering

En el *Rochester Institute of Technology* se ofrece un grado en Ingeniería del Software que contiene 2 asignaturas relacionadas con la arquitectura del software: *Software System Requirements and Architecture* y *Software architecture* (optativa).

La primera incluye los siguientes descriptores:

- Principios y prácticas relacionados con la identificación de *stakeholders*
- Desglosando requisitos funcionales y de calidad
- Trasladando requisitos en estructuras arquitectónicas
- Análisis de arquitecturas candidatas respecto a los requisitos

La asignatura optativa de arquitectura del software incluye:

- Definición de estructuras arquitectónicas
- Análisis de arquitecturas en términos de compromisos entre restricciones en conflicto
- Documentación de arquitecturas para su uso a lo largo del ciclo de vida
- Rol del arquitecto durante las actividades de codificación

Universidad de Alberta: Software architecture

La Universidad de Alberta ha creado un curso de arquitectura del software que se ofrece a través de la plataforma Coursera¹⁵. El curso forma parte de la especialización de Análisis y Diseño de Software. Los contenidos del curso son:

- Diagramas de arquitectura UML
- Estilos arquitectónicos
- Arquitectura en la práctica
- Proyecto final

15.5.2 Georgia Tech: Software architecture and design

En el College of Computing de Georgia Tech se imparte la asignatura *CS630: Software Architecture and Design* a nivel de Máster siguiendo el modelo de unificar arquitectura y diseño del software. Los objetivos y resultados de aprendizaje son:

- Expresar el análisis y diseño de una aplicación mediante UML
- Especificar la semántica funcional de una aplicación utilizando OCL
- Especificar y evaluar arquitecturas de software
- Seleccionar y utilizar estilos arquitectónicos apropiados
- Comprender y aplicar técnicas de diseño orientado a objetos
- Seleccionar y aplicar patrones de diseño apropiados
- Comprender y realizar una revisión de diseño

15.5.3 Arquitectura del software en universidades nacionales

A continuación se presenta una muestra de cómo se imparten los conceptos de arquitectura del software en algunas universidades nacionales. Se ha intentado presentar ejemplos de las 3 variantes: asignatura independiente de arquitectura del software, asignatura que integra arquitectura y diseño del software, y dentro de asignaturas de ingeniería del software.

Universidad Politécnica de Madrid: Arquitectura y Diseño Software

En el Grado en Ingeniería del Software de la Escuela Técnica Superior de Ingeniería de Sistemas Informáticos de la Universidad Politécnica de Madrid se imparte esta asignatura en tercer curso (segundo semestre), coordinada por Fernando Arroyo Montoro¹⁶.

En la asignatura se recomiendan como conocimientos previos las asignaturas:

- Ingeniería del Proceso Software y Construcción

¹⁵<https://www.coursera.org/learn/software-architecture>

¹⁶Guía docente: https://www.upm.es/comun_gauss/publico/guias/2020-21/2S/GA_61IW_615000253_2S_2020-21.pdf (Julio 2021)

- Fundamentos de Ingeniería del Software
- Ingeniería de requisitos y modelado

Las competencias de la asignatura son:

- CC8 - Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- CE2 - 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
- CE4 - 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.
- CT2 - Resolución de problemas: Identificar, analizar y definir los elementos significativos que constituyen un problema para resolverlo con criterio y de forma efectiva
- CT3 - Comunicación oral: Expresar con claridad y oportunidad las ideas, conocimientos y reflexiones propios a través de la palabra, adaptándose a las características de la situación y la audiencia para lograr su comprensión.

Y como resultados de aprendizaje:

- RA106 - Conoce y aplica las teorías, modelos y técnicas actuales para la identificación de los problemas, el análisis, el diseño del software, el desarrollo, la implementación, la verificación y la documentación.
- RA39 - Modela y Diseña soluciones atendiendo a los compromisos de eficiencia, modularidad
- RA105 - Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de teorías, modelos y técnicas actuales.
- RA83 - Realiza una tarea compleja de manera autónoma, seleccionando las estrategias más convenientes para abordar el estudio, en base al análisis de las condiciones y la meta propuesta. Analiza e interpreta la información, maneja las tecnologías de la información y comunicación (TICs), muestra habilidades de comunicación e interacción para un aprendizaje colaborativo. Valora la efectividad de la planificación de las tareas y toma las decisiones oportunas para lograr su propósito.
- RA47 - Es capaz de trabajar como miembro de un equipo con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos y teniendo en cuenta los recursos disponibles

El temario propuesto en la asignatura es:

1. Arquitectura software y el ciclo de vida
 - a) Introducción y concepto de arquitectura
 - b) Evolución histórica del concepto de arquitectura
 - c) El papel del arquitecto software
2. Diseño de arquitecturas
 - a) Atributos de calidad de una arquitectura
 - b) Diseño dirigido por atributos
 - c) Flujo de diseño de una arquitectura
 - d) Aspectos de diseño, opciones y decisiones
 - e) Tipos de decisiones
3. Vistas arquitectónicas
 - a) Introducción a las vistas arquitectónicas
 - b) Modelo IEEE 1471
 - c) Especificación de puntos de vista
 - d) Modelo 4+1 de Kruchten
4. Documentación de arquitecturas
 - a) Estándares de documentación
 - b) Nomenclaturas
 - c) Combinación de vistas
5. Patrones arquitectónicos
 - a) Introducción a los patrones arquitectónicos
 - b) Descripción de patrones arquitectónicos
6. Evaluación de arquitecturas y refactorización

- a) Introducción a la evaluación de arquitecturas
 - b) Técnicas de análisis de arquitecturas
 - c) Modelo ATAM de evaluación de arquitecturas
 - d) Refactorización
7. Arquitecturas de referencia
 - a) Concepto de arquitectura de referencia
 - b) Ejemplos de arquitecturas de referencia
 8. Líneas de producto
 - a) Ejemplos de líneas de producto
 - b) Modelos de características

Universidad Politécnica de Cataluña: Arquitectura del Software

En la Facultad de Informática de Barcelona, se imparte la asignatura Arquitectura del Software, coordinada por Cristina Gómez Seoane, de 6 créditos ECTS ¹⁷.

Los contenidos:

1. Introducción a la arquitectura y diseño del software: Metodologías de desarrollo de software. Arquitectura y diseño de software. Arquitectura lógica y física de un sistema software. Rol de los patrones de diseño en el diseño del software. Arquitectura y diseño del software en metodologías clásicas y ágiles.
2. Arquitectura en capas y orientada a objetos Patrón arquitectónico en capas. Aplicación del patrón arquitectónico en capas a los sistemas y servicios software. Principios de diseño de la arquitectura en capas. Patrón arquitectónico orientación a objetos. Aplicación del patrón arquitectónico orientación a objetos a los sistemas software. Principios de diseño de la arquitectura orientada a objetos.
3. Diseño del software utilizando metodologías clásicas. Diseño de la capa de dominio. Patrones Domain Model y Transaction Script. Patrones de diseño de la capa de dominio. Uso de servicios software. Diseño de la capa de presentación. Diseño externo de un sistema software. Diseño interno de la capa de presentación. Diseño de la capa de datos. Tecnología de bases de datos relacionales. Estrategias de gestión de la persistencia. Generación automática de la persistencia. Diseño directo de la persistencia.
4. Diseño de software utilizando metodologías ágiles. Extreme Programming. TDD. Tests en TDD. Coding en TDD. Pair Programming. Simple Design and Incremental design and architecture. Code Smells and Refactorings.

Universidad de Sevilla: Arquitectura e Integración de Sistemas Software

En la Universidad de Sevilla, se imparte la asignatura "Arquitectura e Integración de Sistemas Software", impartida por el Departamento de Lenguajes y Sistemas Informáticos, en el segundo semestre de segundo curso ¹⁸.

Se incluyen los siguientes bloques temáticos:

- BLOQUE I. Arquitectura del Software.
- BLOQUE II. Integración de Sistemas Software.
- BLOQUE III. Desarrollo de Soluciones de Integración.

Universidad Castilla - La Mancha: Ingeniería del Software

La Universidad Castilla - La Mancha imparte el Grado en Ingeniería Informática con 4 tecnologías específicas: Ingeniería de Computadores, Ingeniería del Software, Computación y Tecnologías de la Información. En el plan de estudios ¹⁹ aparecen dos asignaturas de Ingeniería del Software obligatorias en segundo y tercer curso, y una asignatura de Diseño del Software, pero no contiene una asignatura específica de arquitectura del software.

La asignatura de diseño del software tiene el siguiente temario en la Escuela Superior de Informática (Ciudad Real):

- Tema 1: Presentación y estudio de un proyecto cliente-servidor
- Tema 2: Diseño arquitectónico
- Tema 3: Patrones de diseño

¹⁷Guía docente: <https://www.fib.upc.edu/es/estudios/grados/grado-en-ingeneria-informatica/plan-de-estudios/asignaturas/AS> (Julio 2021)

¹⁸Página asignatura:http://www.lsi.us.es/docencia/pagina_asignatura.php?id=97 (Julio 2021)

¹⁹<https://esi.uclm.es/categories/plan-de-estudios>

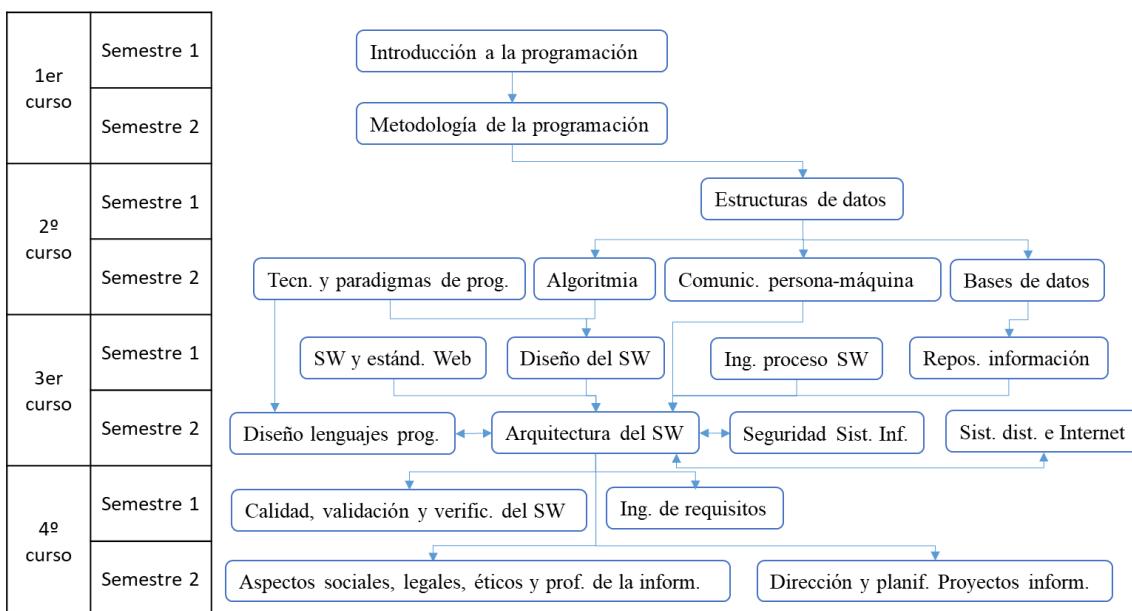


Figura 15.3: Relación de arquitectura del software con otras asignaturas del plan de estudios

- Tema 4: Diseño de software seguro
- Tema 5: Generación automática de código

Con lo que la parte dedicada a la arquitectura del software se cubriría en el tema 2 de diseño arquitectónico.

En la Escuela Superior de Ingeniería Informática (Albacete) el temario es:

- Tema 1: Fundamentos del diseño de software
- Tema 2: Diseño orientado a objetos
- Tema 3: Patrones de diseño software
- Tema 4: Patrones de creación
- Tema 5: Patrones estructurales
- Tema 6: Patrones de comportamiento
- Tema 7: Manejo del patrón MVC
- Tema 8: Arquitecturas software
- Tema 9: Análisis y evaluación del diseño OO
- Tema 10: Utilizar patrones de diseño en la práctica
- Tema 11: Trabajo de la asignatura

15.6 Relación con otras materias

La arquitectura del software forma una parte fundamental de la ingeniería del software, por lo que está relacionada fuertemente con todas las materias de esta disciplina. Además, hay otras materias del ámbito de la ingeniería informática más general como puede ser programación, bases de datos, algoritmia, arquitectura de computadores, administración de sistemas, etc. con los que también guarda una gran relación.

Con el fin de acotar las materias con las que se podría relacionar, en esta sección se incluyen principales las asignaturas del Grado en Ingeniería Informática del Software de la Universidad de Oviedo con las que se considera que hay una mayor relación.

La figura 15.3 presenta un esquema de las principales asignaturas del grado que afectan o se ven afectadas por la asignatura de arquitectura del software. A continuación se hace un repaso de las mismas destacando la relación existente e incluyendo los contenidos de las mismas tal y como aparecen en las propuestas de guías docentes del curso 2020-21.

15.6.1 Asignaturas de programación

Se trata de tres asignaturas que se imparten en primer y segundo curso en las que se tratan los fundamentos de la programación. Las dos primeras asignaturas: *introducción a la programación y metodología de la programación* se imparten en primer curso y permiten a los estudiantes conocer el paradigma orientado a objetos. Por su parte la asignatura de *tecnologías y paradigmas de la programación* ofrece una introducción a otros paradigmas, como la programación funcional, que pueden ser muy útiles como base para el desarrollo del prototipo en las prácticas de arquitectura del software.

Los contenidos que se proponen en introducción a la programación son:

1. El Modelo de objetos
 - Objetos, Propiedades, Métodos, Clases
2. Elementos fundamentales de un lenguaje de programación orientado a objetos
 - Sintaxis y semántica, Tipos de datos básicos, Constantes y Variables, Asignación y expresiones, Estructuras de control, Entrada/Salida.
3. Estructuras de datos fundamentales
 - Arrays, cadenas y procesamiento de cadenas.
4. Elementos avanzados de la programación orientada a objetos
 - Encapsulación y ocultación de información, jerarquías de clases.
5. Metodologías y desarrollo de software de calidad
 - Conceptos fundamentales de análisis y diseño, uso de lenguaje de modelado de software, Documentación, Depuración, pruebas y validación.

Los contenidos propuestos para la asignatura *Metodología de la programación* que se imparte en el segundo semestre de primer curso son:

1. Revisión del paradigma de Orientación a Objetos.
 - Fundamentos del paradigma de objetos, Herencia, Polimorfismo, Interfaces
2. Introducción a la Metodología de la programación.
 - Análisis y diseño, Fundamentos de notación UML, Pautas de diseño, implementación, pruebas y documentación
3. Estructuras de datos básicas.
 - Colecciones (listas, pilas, colas), Algoritmos básicos de búsqueda
4. Tratamiento de errores en tiempo de ejecución
 - Errores en ejecución, Técnicas básicas para el tratamiento de errores de ejecución, Excepciones
5. Persistencia, serialización, entrada/salida
 - Sintaxis y clases de E/S
6. Introducción a la programación multihilo.

Como puede verse, en la primer asignatura ya se realiza una introducción básica al desarrollo de software de calidad, así como a los lenguajes de modelado de software, documentación y pruebas del software, los cuales son fundamentales para la arquitectura del software. Estos conceptos se verán reforzados en la asignatura de metodología de la programación, incidiendo en el lenguaje UML.

Los contenidos impartidos en la asignatura *Tecnología y Paradigmas de programación* son:

1. Lenguajes y paradigmas de programación
 - Lenguaje de Programación; Características de los Lenguajes de Programación; Paradigmas de Programación; Tecnología de Programación
2. Paradigma orientado a objetos
 - Abstracción; Encapsulamiento; Modularidad; Sobrecarga; Herencia; Clases Abstractas e Interfaces; Excepciones; Asertos; Diseño por Contratos; Genericidad; Inferencia de Tipos
3. Fundamentos del paradigma funcional
 - Cálculo Lambda; Isomorfismo Curry-Howard; Funciones como Entidades Primer Orden; Cláusulas; Currificación; Aplicación Parcial; Continuaciones; Evaluación Perezosa; Transparencia Referencial; Pattern Matching; Funciones de Orden Superior; Listas por comprensión
4. Fundamentos de la programación concurrente y paralela
 - Programación Concurrente y Paralela; Proceso e Hilo; Paralelización de Algoritmos; Paso Asíncrono de Mensajes; Creación Explícita de Hilos; Condición de Carrera; Context Switching & Thread Pooling; Foreground & Background Threads; Sincronización de Hilos y Procesos; Interbloqueo; Estructuras de Datos Thread-Safe; Paralelización mediante tareas y Task Parallel Library; Paradigma Funcional en la Paralelización de Algoritmos
5. Tipado Dinámico y Meta-programación

- Tipado Dinámico; Duck Typing; Multiple Dispatch; Reflexión Computacional; Anotaciones o Atributos; Generación Dinámica de Código; MetaProgramación

La introducción del paradigma funcional y de la programación concurrente y paralela en esta asignatura son muy interesantes para la asignatura de arquitectura del software.

15.6.2 Estructuras de datos y algoritmia

Se trata de dos asignaturas impartidas en segundo curso en el primer semestre (*estructuras de datos*) y en el segundo semestre (*algoritmia*). El conocimiento de ambas se considera importante para la arquitectura del software y se considera adecuado que estas asignaturas se hayan cursado previamente a la misma.

Los contenidos impartidos en estructuras de datos de acuerdo a la guía docente son:

1. Conceptos Fundamentales
 - Tipos de datos predefinidos; Tipos de datos estructurados; Genericidad en java, contenedores de tipo <T>; Introducción a la algoritmia; Estimación de tiempos de ejecución.
2. Estructuras de datos en Red
 - Grafos genéricos; Conceptos básicos de Teoría de Grafos; Matrices de Adyacencia; Listas de Adyacencia; Caminos de coste mínimo; El algoritmo de Dijkstra.+; El algoritmo de Floyd-Warshall; Búsqueda en profundidad; Árboles de recubrimiento de coste mínimo; Algoritmo de Prim versus Algoritmo de Kruskal.
3. Estructuras de datos Jerárquicas
 - Árboles genéricos; Conceptos básicos sobre árboles; Métricas de rendimiento; Árboles de búsqueda.
 - Árboles binarios; Árboles de búsqueda binarios.
 - Árboles de Adelson-Velski; Árboles multicamino; Árboles de Bayer & McCreight
4. Colas de Prioridad
 - Colas de prioridad genéricas; Montículos binarios genéricos; Operaciones especiales utilizando montículos binarios.
5. Estructuras Diccionario
 - Transformación de claves (hashing); Tablas hash genéricas; Funciones de dispersión genéricas; Tablas hash abiertas; Tablas hash cerradas; Exploración lineal; Agrupamientos (clustering); Borrado Perezoso (Lazy deletion); Exploración cuadrática; Dispersión doble; Redispersión.

Por otro lado, los contenidos impartidos en la asignatura de algoritmia son:

1. Principios de Algoritmia.
 - Conceptos; Complejidad analítica; Tiempos de ejecución empíricos
2. Ordenación.
 - Algoritmos básicos de ordenación; Algoritmos rápidos de ordenación; Análisis y comparación de algoritmos de ordenación.
3. Técnicas de diseño de algoritmos
 - Divide y Vencerás (Recursividad); Algoritmos Voraces; Programación dinámica; Vuelta atrás; Algoritmos de ramificación y poda.
4. Algoritmos paralelos y avanzados.
 - Divide y vencerás paralelo; Paralelización de otras técnicas; Algoritmos avanzados (Árboles de juegos y otras técnicas)

15.6.3 Diseño de Software

Esta asignatura se imparte en el grado en Ingeniería Informática del Software, en primer semestre de tercer curso y es una asignatura que tiene una fuerte relación con la arquitectura del software. De hecho, en algunas propuestas de planes de estudios, se imparte una única asignatura de diseño del software que incluye aspectos de arquitectura del software. En la propuesta del grado de Ingeniería Informática del Software se decidió separar ambas asignaturas en dos, de forma que la arquitectura del software es una continuación natural del diseño, e incluso podría contemplarse la posibilidad de que la asignatura de diseño fuese un pre-requisito. No obstante, durante el diseño del plan de estudios se decidió evitar la creación de pre-requisitos para dar a los estudiantes una mayor flexibilidad en la elección de docencia y se presenta como una recomendación en la guía docente.

Los contenidos impartidos actualmente en la asignatura de diseño de software son:

1. Técnicas y principios de calidad para el diseño disciplinado de software

- Introducción al diseño
 - Representación del diseño
 - Criterios de diseño
 - Paso del diseño a implementación
2. Patrones de diseño
 - Introducción a los patrones
 - Estructura de un patrón
 - Patrones por familia: de creación, estructurales y comportamiento
 - Catálogo de patrones
 3. Evolución disciplinada de diseños y reingeniería
 - Los patrones como herramienta de comunicación
 - El papel del diseño en el proceso de desarrollo de software
 - Introducción a la reutilización de diseños mediante frameworks
 - El diseño en metodologías ágiles

15.6.4 Ingeniería del proceso software

Esta asignatura se imparte actualmente en el primer semestre de tercer curso y se centra en aspectos de metodologías y técnicas de desarrollo de software, entre los que se destacan las metodologías ágiles que se utilizan en la asignatura de arquitectura del software. Se considera adecuada la secuenciación cronológica establecida entre ambas asignaturas y de hecho, la utilización de aspectos de desarrollo ágil en la asignatura de arquitectura del software como la utilización de desarrollo basado en pruebas, integración continua, revisión y propiedad colectiva del código, se vuelven a incidir durante la realización de la práctica de arquitectura del software.

Los contenidos que se imparten en la asignatura según la guía docente son:

1. Conceptos generales de la Ingeniería del Software:
 - Problemática general y características del software.
 - Procesos software, metodologías y técnicas.
 - Estándares y herramientas.
2. Procesos y Metodologías más comunes:
 - Metodologías Ágiles.
 - Metodologías estándar en la Administración Pública.
 - Proceso Unificado.
 - Relación entre procesos de Análisis y Diseño.
 - Otros procesos y estándares.
3. Técnicas:
 - Análisis y técnicas: Historias de usuario y análisis de casos de uso, Análisis de clases del dominio, Prototipado de la interfaz.
 - Otras técnicas (V&V, pruebas, gestión).

15.6.5 Software y estándares para la Web

Esta asignatura se imparte en el primer semestre de tercer curso. En la asignatura se imparten los fundamentos de las tecnologías relacionadas con la Web y dada la gran importancia que tiene la Web en el desarrollo actual del software, los contenidos de esta asignatura sirven de base para la arquitectura del software.

De hecho, en los últimos tres cursos, en los que en la asignatura de arquitectura del software se utilizó una práctica basada en arquitecturas web descentralizadas, los prototipos desarrollados por los estudiantes han utilizado el lenguaje Javascript, cuya introducción se realiza en esta asignatura. Por lo que la dependencia de estos conocimientos se ha acrecentado.

Los contenidos impartidos en Software y estándares para la Web según la guía docente son:

1. Desarrollo de sitios web mediante estándares.
2. Lenguajes de marcado.
 - HTML5; Usabilidad Web; Accesibilidad Web; Tecnologías XML, DTDs, Schemas y XSLT; Procesamiento y generación de XML
3. Lenguajes de presentación.
 - Fundamentos de CSS; Módulos de CSS; Adaptabilidad web

4. Computación Web
 - Protocolo HTTP y HTTPS; Computación en la nube
5. Computación en el cliente.
 - Lenguajes de Script; Lenguajes ECMAScript y JavaScript; Tecnologías y recursos relacionados con JavaScript; jQuery; AJAX; JSON; Consumo de servicios Web; APIs de HTML5
6. Computación en el servidor.
 - Lenguaje PHP; Almacenamiento en el Servidor: archivos y sistemas de gestión de bases de datos

15.6.6 Bases de Datos y Repositorios de información

Se trata de dos asignaturas, la asignatura de *bases de datos*, que se imparte en el segundo semestre de segundo curso, y la asignatura *repositorios de información* que se imparte en el primer semestre de tercer curso, es decir, ambas asignaturas se imparten antes que la asignatura de arquitectura del software, lo cual permite asumir que muchos de los estudiantes ya las han cursado. Por supuesto, la mayoría de las arquitecturas del software requieren algún mecanismo de persistencia de datos, por lo que las bases de datos y los repositorios de información son fundamentales.

Los contenidos impartidos en bases de datos según la guía docente son:

1. Introducción y estructura de los Sistemas de Gestión de Bases de Datos (SGBD)
 - a) Conceptos fundamentales de bases de datos
 - b) Ventajas de los SGBD
 - c) Estructura de un SGBD
 - d) Introducción a los modelos de datos
 - e) Ciclo de vida de bases de datos
2. Diseño conceptual (usando el modelo Entidad-Relación)
 - a) Elementos principales del modelo Entidad-Relación (E-R)
 - b) Restricciones de integridad en el modelo E-R: cardinalidad, superclaves, claves primarias, claves candidato
 - c) Elementos adicionales del modelo E-R extendido
3. Modelo relacional
 - a) Elementos principales del modelo relacional
 - b) Claves primarias y claves externas
 - c) Lenguajes formales de manejo de datos
4. Lenguaje relacional de datos SQL y desarrollo de aplicaciones
 - a) Elementos principales del lenguaje SQL como lenguaje de manejo de datos
 - b) Consultas avanzadas en SQL
 - c) SQL como lenguaje de definición de datos
 - d) Desarrollo de aplicaciones en SQL (PL/SQL y mediante bibliotecas).
 - e) Definición de restricciones de integridad en SQL (asertos y disparadores)
5. Restricciones de integridad
 - Aplicación de las restricciones de integridad
 - Dominios, claves, integridad referencial
 - Dependencias funcionales
6. Diseño lógico (usando el modelo relacional)
 - Fundamentos de la teoría de la normalización
 - Formas normales y algoritmos de normalización
7. Introducción a otros modelos de datos
 - Modelo orientado a objetos, objeto-relacional, XML, lógico, NOSQL

Los contenidos impartidos en la asignatura de repositorios de información de acuerdo a la guía docente de 2020-21 son:

1. Introducción y conceptos fundamentales.
2. Desarrollo de aplicaciones relacionales con SQL avanzado, bibliotecas de acceso a bases de datos y transacciones.
3. Patrones y estándares de acceso a datos.
4. Desarrollo de aplicaciones orientadas a objetos con software intermedio de mapeo objeto-relacional.
5. Modelos de datos alternativos: Sistemas de indexación de documentos.
6. Modelos de datos alternativos: Bases de datos NoSQL.

15.6.7 Comunicación persona-máquina

Se trata de una asignatura que se imparte en el primer semestre de segundo curso y, por tanto, es anterior a la asignatura de arquitectura del software. En la asignatura se tratan temas de gran importancia para la arquitectura del software y que guardan relación con atributos de calidad como la usabilidad, la accesibilidad, etc. por lo que se considera que la asignatura de arquitectura del software se ve influida por esta asignatura, lo cual permite reforzar estos conocimientos poniéndolos en valor al evaluar los prototipos desarrollados por los estudiantes de arquitectura del software teniendo en cuenta los atributos de calidad de usabilidad y accesibilidad.

Según la guía docente del curso 2020-21, los contenidos impartidos son:

1. Fundamentos de la Comunicación Persona-Máquina
 - Definición y conceptos
 - Principios básicos relacionados con la interacción
2. Introducción al desarrollo de interfaces gráficas de usuario en Java.
 - Librerías para el desarrollo de interfaces gráficas de usuario en java.
 - Características generales de una interfaz gráfica de usuario.
3. Diseño centrado en el usuario
 - Origen y definición
 - Modelo iterativo
 - Fases. Técnicas aplicadas en cada fase
4. Modelado de usuarios
 - El factor humano
 - Modelos de usuario
 - Técnicas de modelado
5. Técnicas de diseño
 - Cardsorting
 - Mockups, wireframes, prototipos

15.6.8 Sistemas distribuidos e Internet

Esta asignatura se imparte en el segundo semestre de tercer curso en paralelo a la asignatura de arquitectura del software.

Los contenidos propuestos para esta asignatura en la guía docente son:

1. Introducción a las tecnologías y aplicaciones Web
2. Ingeniería para el desarrollo Web
3. Desarrollo clásico de aplicaciones Web
4. Desarrollo ágil de aplicaciones Web
5. Desarrollo de aplicaciones Web basado en Servicios
6. Seguridad en las aplicaciones Web

Como puede verse, los contenidos de esta asignatura están orientados a la creación de aplicaciones Web, lo cual tiene una gran importancia en la arquitectura del software actual. Cuando se inició la docencia de esta asignatura se detectaron solapamientos de contenidos entre ambas que con el tiempo se han ido solucionando gracias a una mayor coordinación. El enfoque actual de la asignatura de arquitectura del software se orienta más a mostrar diferentes estilos y patrones arquitectónicos, entre los que se ha incluido un tema que repasa los patrones utilizados para sistemas distribuidos. Para la realización del prototipo, se ha buscado deliberadamente una propuesta arquitectónica diferente a la de las aplicaciones Web tradicionales, basada en la descentralización del almacenamiento de datos, con el fin de presentar a los estudiantes modelos alternativos.

15.6.9 Seguridad de Sistemas Informáticos

Esta asignatura se imparte en el segundo semestre de tercer curso en paralelo a la asignatura de arquitectura del software, por lo que no puede plantearse como pre-requisito de la misma. No obstante, el atributo de calidad de seguridad está cobrando una importancia cada vez mayor en la arquitectura del software y se considera que lo ideal es que la asignatura de arquitectura del software estuviese impartida después de la de seguridad, para poder solicitar a los estudiantes que tuviesen en cuenta dicho atributo en sus prácticas. Puesto que con el plan de estudios actual esta situación no es posible, lo que se hace es indicar a los estudiantes que se tendrán en cuenta los aspectos de seguridad en sus prototipos, aceptando que los

mismos están siendo impartidos por esta asignatura.

Una opción a considerar sería plantear una mayor coordinación con esta asignatura para que la temática de la práctica a realizar fuese compatible con las actividades que se realizan en la asignatura de seguridad.

A continuación se indican los contenidos de esta asignatura según la guía docente de 2020-21:

1. Introducción
 - Visión general de la seguridad.
 - Conceptos básicos.
 - Anatomía de un ataque.
2. Aplicaciones de la criptografía
 - Conceptos generales de criptografía.
 - Aplicaciones de la criptografía en la seguridad de los sistemas informáticos.
3. Seguridad de los Sistemas Operativos
 - Introducción.
 - Medidas generales de protección del sistema.
 - Control de acceso al sistema y a los recursos.
 - Fortalecimiento de la seguridad del sistema.
4. Políticas de seguridad
 - Modelos de estrategias de defensa y políticas de seguridad.
 - Sistemas de Gestión de la Seguridad de la Información.
 - Normas relacionadas.
 - Aplicaciones prácticas de las políticas de seguridad.
5. Seguridad perimetral y de red
 - Principios generales de diseño de una red segura.
 - Auditoría de seguridad de servicios en red.
6. Seguridad de aplicaciones
 - Problemas de seguridad comunes en aplicaciones
 - Autenticación segura en aplicaciones
 - Desarrollo de software seguro.
 - Uso de software de terceros
7. Introducción a las técnicas de explotación y post-explotación
 - Técnicas comunes de exploración de sistemas
 - Obtención de shells
 - Metasploit Framework
 - Operaciones típicas de post-explotación

15.6.10 Diseño de lenguajes de programación

Esta asignatura se imparte en segundo semestre de tercer curso en paralelo a la asignatura de arquitectura del software, por lo que en principio podrían ser asignaturas independientes. No obstante, la asignatura describe técnicas de implementación de lenguajes de programación basadas en una descomposición modular tradicional en análisis léxico, sintáctico, semántico y generación de código que puede servir como un modelo de arquitectura de referencia basada en pipes & filters así como para justificar el estilo arquitectónico *intérpretes y lenguajes de dominio específico* descrito en la asignatura.

Los contenidos impartidos en la asignatura de diseño de lenguajes de programación según la guía docente son:

1. Conceptos básicos del diseño de lenguajes de programación
2. Análisis léxico
3. Análisis sintáctico
4. Análisis semántico
5. Lenguajes y representaciones intermedias
6. Generación de código

15.6.11 Ingeniería de requisitos

Se trata de una asignatura que se imparte en el primer semestre de cuarto curso, es decir, después de la asignatura de arquitectura del software. En realidad, las competencias que se pueden adquirir en estas asignaturas podrían ser muy útiles para cursar la asignatura de arquitectura del software, por lo que la

posibilidad de que esta asignatura se impartiese antes que la de arquitectura del software sería interesante. No obstante, dado que en el plan de estudios se ha establecido que sea anterior arquitectura del software a ingeniería de requisitos, es necesario adaptar el enfoque para asumir que los estudiantes de arquitectura del software no tienen conocimientos de ingeniería de requisitos y para tratar de proporcionar una serie de competencias o inquietudes sobre la captura de requisitos que aumenten la motivación de los estudiantes en el momento que cursen la asignatura de ingeniería de requisitos.

1. Fundamentos de la Ingeniería de Requisitos
2. Proceso de la Ingeniería de Requisitos
3. Técnicas de obtención de requisitos
4. Análisis de Requisitos
5. Especificación de Requisitos Software
6. Validación de Requisitos
7. Gestión de Requisitos
8. La Ingeniería de Requisitos en los Métodos Ágiles

15.6.12 Dirección y planificación de proyectos informáticos

Se trata de una asignatura impartida en el segundo semestre de cuarto curso, por lo que muchos estudiantes que cursen esta asignatura habrán cursado la asignatura de arquitectura del software de tercer curso. La arquitectura del software tiene una gran relación con la dirección y planificación de proyectos, dado que las decisiones arquitectónicas influyen claramente en el éxito del proyecto, y pueden estar influidas por la propia dirección y gestión del proyecto, tanto como atributos de calidad requeridos o como restricciones a tener en cuenta. Al impartirse ambas asignaturas en paralelo, en el mismo semestre, pero en cursos diferentes, se puede considerar la coordinación entre las asignaturas. En varias ocasiones, las prácticas de la asignatura de dirección y planificación de proyectos tomaron como referencia proyectos propuestos el curso anterior en la asignatura de arquitectura del software. De esta forma, los estudiantes ya conocían el dominio de ambos proyectos y podrían centrarse en otros aspectos del proyecto como la organización de recursos, presupuestos, costes, riesgos, etc. Aunque se podrían estudiar otro tipo de coordinaciones entre las asignaturas más avanzadas, como que los estudiantes de la asignatura de dirección de proyectos actuasen como directores de los proyectos que realizan los estudiantes de arquitectura del software, las dificultades logísticas y organizativas de este planteamiento han hecho que de momento se haya desestimado.

De acuerdo a la guía docente, los contenidos impartidos en la asignatura de dirección y planificación de proyectos son:

1. Introducción a la Dirección de Proyectos Informáticos
2. Medición y estimación
3. Organización de los recursos y del proyecto
4. Gestión de Recursos Humanos
5. Gestión de Costes
6. Gestión de riesgos
7. Gestión de calidad
8. Documentación y comunicación
9. Seguimiento y Control

15.6.13 Calidad, Validación y verificación del software

Esta asignatura se imparte actualmente en el primer semestre de cuarto de la titulación. Por tanto, es una asignatura que se imparte de forma posterior a la asignatura de arquitectura del software. Esta situación organizativa hace que en la propuesta actual de la asignatura de arquitectura del software se introduzcan conceptos relacionados con las pruebas del software que si se impartiese esta asignatura con anterioridad podrían obviarse. A pesar de esa posible disparidad cronológica, también se percibe que esta situación puede permitir a los estudiantes que cursen la asignatura de calidad, validación y verificación del software en cuarto curso tener una visión más integral de la necesidad del análisis de la calidad del software y de la realización de un buen plan de pruebas, lo cual puede ser positivo.

De acuerdo a la guía docente, los contenidos impartidos son:

1. Conceptos generales
 - Conceptos generales de calidad en el software.
 - Principios de validación, verificación y pruebas de software.

2. Técnicas de prueba:
 - Basadas en la especificación.
 - Basadas en la estructura.
 - Otras técnicas de pruebas dinámicas.
 - Pruebas estáticas. Inspecciones y revisiones del software.
3. Procesos:
 - Preparación y ejecución de pruebas.
 - Las pruebas en el contexto de proyectos ágiles
 - Tipos y niveles de prueba.
 - Planes de calidad y de pruebas.
 - Medición, análisis y reporte de defectos.
 - Modelos de procesos, evaluación y certificación.
 - Metodologías, herramientas y estándares aplicables.

15.6.14 Aspectos Sociales, Legales, Éticos y Profesionales de la Informática

Se trata de una asignatura impartida en el segundo semestre de cuarto curso y, por tanto, en el curso siguiente al que se imparte la asignatura de arquitectura del software. Aunque en la asignatura de arquitectura del software no se incide directamente en los aspectos sociales, legales, éticos y profesionales por considerar que serán planteados en el siguiente curso, sí que se considera que dichos aspectos son cada vez más importantes y así se les indica a los estudiantes. De hecho, la utilización de una práctica basada en Solid en la asignatura en los últimos tres cursos, que incide directamente sobre la privacidad de los datos, sirve para mostrar indirectamente a los estudiantes la importancia de estas consideraciones y que vayan tomando conciencia de las mismas.

Según la guía docente, se imparten los siguientes contenidos:

1. Aspectos sociales
 - Impacto social y ambiental de la Informática.
2. Aspectos éticos
 - Deontología y responsabilidad profesional.
3. Aspectos Legales
 - Protección de datos.
 - Propiedad intelectual, licencias y contratos.
 - Sociedad de la Información y Comercio Electrónico.
 - Delito Informático.
4. Aspectos profesionales
 - Actividad profesional: peritajes y auditoría de protección de datos.

15.7 Conclusiones de la revisión de referencias curriculares

El análisis de las propuestas curriculares anteriores y los diseños curriculares implantados, junto con algunos otros que no se han incluido en esta documentación, proporciona una idea de las perspectivas existentes, y ha sido de gran ayuda en la elaboración del diseño curricular de este proyecto docente.

A continuación, se exponen algunas de las conclusiones sacadas de todo ello.

Respecto a las propuestas curriculares, el diseño conjunto de IEEE y ACM constituye una propuesta curricular sólida y de prestigio. Una prueba de ello es que tanto la mayoría de universidades extranjeras como españolas tienen esta propuesta como referente fundamental.

El cuerpo de conocimiento SWBoK de 2014 ofrece también una propuesta que incluye los contenidos de arquitectura del software como los puntos de vista, los estilos arquitectónicos y las decisiones de diseño arquitectónico, así como aspectos del análisis de atributos de calidad. La principal referencia son los libros [6] y [18].

La propuesta UNESCO-IFIP del año 2000 supuso un cambio importante de concepción respecto a versiones anteriores, consistente en la creación de un marco flexible de desarrollo para diversos perfiles curriculares, y se apoya en los currícula de ACM/IEEE para los detalles de implantación. No obstante este modelo parece no seguir activo y de hecho, la versión en PDF del mismo, que antes estaba disponible en la Web, parece que ha sido borrada.

Las implantaciones curriculares en las Universidades Nacionales y Extranjeras no difieren en gran

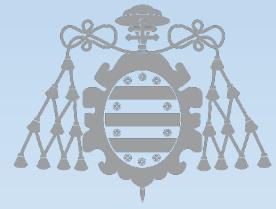
medida respecto a sus contenidos, tanto teóricos como prácticos. En general, suelen emplear los contenidos descritos en la propuesta de IEEE y ACM.

Las mayores diferencias radican en cómo organizar los contenidos de arquitectura del software dentro de las titulaciones que incluyen una especialización de ingeniería del software, encontrándose 3 modelos:

- Incluir los contenidos dentro de una o varias asignaturas sobre ingeniería del software
- Incluir los contenidos en una asignatura de diseño de software
- Incluir los contenidos en una asignatura específica de arquitectura del software

Se considera que el último modelo es más adecuado dado que permite preservar la esencia de la arquitectura del software sin que ésta se diluya en conceptos de otras asignaturas, que pueden dar más importancia a aspectos de diseño o a aspectos generales de la ingeniería del software.

En cuanto a los conceptos previos necesarios para cursar la materia, muchas universidades imponen la superación de ciertas asignaturas como pre-requisito previo. En las universidades españolas es más común ver estas indicaciones como meras recomendaciones. Los requisitos más referentes a la parte teórica son el haber cursado las asignaturas relacionadas Programación, Diseño del software e Ingeniería del Proceso Software.



Universidad de Oviedo

16. Metodología docente

En esta sección se repasan algunos aspectos a tener en cuenta en el proceso de enseñanza-aprendizaje en el ámbito universitario. Para la elaboración del método docente que se propone llevar a cabo en la asignatura, se realiza previamente un análisis de algunos métodos existentes, y se contrasta con las características del grupo destinatario. Se realizará lo mismo con el método de evaluación elegido para la valoración del rendimiento del estudiante.

16.1 Introducción

La educación universitaria supone el nivel superior del sistema educativo, que tiene por objetivo es desarrollar la capacidad del individuo para desenvolverse en determinadas situaciones. Para ello la educación deberá proporcionar los elementos necesarios para que el individuo se desenvuelva en la sociedad, en general, y en un campo profesional o de conocimiento en particular (en este caso, la ingeniería informática).

La Universidad tiene asignadas tradicionalmente con este objeto tres misiones: adquisición del conocimiento (investigación), transmisión del conocimiento (docencia) y aplicación del mismo (socialización).

En el caso de un proyecto docente, la parte más directamente relacionada con el mismo son las actividades de docencia. El propósito de la misma es dotar al estudiante de unos conocimientos (saber), habilidades (saber hacer), y actitudes acordes con las principales características del entorno organizativo y/o social (saber estar) y motivaciones responsables de que la persona quiera o no realizar los comportamientos propios de la competencia (querer hacer).

El docente debe poner en contacto dos elementos diferenciados: el objeto de estudio que obliga al docente a reflexionar sobre la naturaleza de la disciplina a impartir y los alumnos, cuya diversidad de capacidades, preparación e intereses necesita una reflexión sobre la definición de un modelo o método racional de enseñanza.

16.2 Los estudiantes

Aunque en cada curso escolar las características del alumnado de la asignatura varían, existe generalmente un perfil de referencia que puede utilizarse, obtenido a partir de los datos de matrícula de cada curso y la experiencia personal acumulada por el candidato durante los más de treinta años dedicados a la docencia universitaria y los últimos 9 años dedicados a la docencia de esta asignatura en concreto.

La edad común del estudiante es de 21-23 años, aunque existe un porcentaje bajo de estudiantes que, al compaginar sus estudios con un trabajo, poseen una edad más avanzada. Este segundo tipo de estudiante

suele, en la mayoría de los casos, abstenerse de asistir a las clases teóricas, acudiendo únicamente a las prácticas de laboratorio.

En los últimos tiempos, la enseñanza universitaria se está enfrentando a varios retos que requieren una atención adecuada para que puedan ser afrontados con éxito:

- Los estudiantes actuales están acostumbrados a utilizar de forma constante dispositivos como teléfonos móviles, *tablets* u ordenadores portátiles, y en muchas ocasiones acuden a las clases equipados con todo un repertorio de nuevas tecnologías que suelen utilizar de forma indiscriminada.
- Existe una gran cantidad de información disponible en la Web sobre cualquier disciplina que se quiera enseñar. En muchas ocasiones, esta información no solamente apuntes estáticos en formato PDF, sino que pueden ser tutoriales en Youtube o incluso cursos completos sobre partes de una disciplina como los cursos MOOC (*Massive Open Online Courses*).
- La aparición de la pandemia en el año 2020 ha permitido mostrar al público en general la viabilidad de la enseñanza no presencial. Aunque antes de la pandemia ya existían una gran cantidad de ofertas con modalidad de enseñanza no presencial, su utilización era minoritaria comparada con las opciones presenciales.

Los tres factores anteriores suponen una serie de retos para las instituciones universitarias tradicionales y, sobre todo, para los profesores que deben impartir docencia adaptando sus usos y costumbres a esta nueva realidad de estudiantes hiper-conectados tecnológicamente. Un docente que intente impartir una lección magistral al estilo clásico se encontrará con una audiencia que probablemente no estará pendiente de lo que dice porque sabe que muchas de las cosas que cuenta las puede encontrar en la Web, con lo que es necesario llevar a cabo un replanteamiento considerable de las actividades presenciales.

16.3 Modalidades de enseñanza-aprendizaje

En el pasado, la enseñanza se basaba en la figura del maestro, que se limitaba a transmitir una serie de conocimientos, de validez irrefutable, que los alumnos debían asimilar tal y como se los contaban. Hoy en día se da importancia también al proceso del aprendizaje, al estudiante, sin olvidarse del otro elemento, el profesor. Es por eso que siguen siendo importantes las técnicas necesarias para hacer más eficaz el trabajo del profesor.

Ahondando más en este último aspecto, el avance de la Ciencia y de la Tecnología, especialmente la Informática, es vertiginoso. Cuando los estudiantes se incorporen al mundo profesional, muchas de las técnicas e instrumentos que utilicen serán sensiblemente diferentes de las que se le hayan presentado durante su periodo universitario.

Por tanto, no hay que limitarse a conseguir que el estudiante posea unos conocimientos determinados, sino que hay que capacitarle para descubrir, elaborar y asimilar información por sí mismo. La labor del profesor, además de la transmisión de conocimientos, incluye la de ser un guía y facilitador del proceso de aprendizaje.

Para todo ello es imprescindible la integración de:

- El conocimiento de la materia, que el profesor debe dominar, para ser capaz de transmitir sus conocimientos acerca de la misma. Para este punto, es importante la labor investigadora del profesor, ya que implica una continua ampliación y actualización del conocimiento y permite enfocar la enseñanza desde una perspectiva amplia y actual.
- El conocimiento de los procesos de enseñanza y aprendizaje, para hacer más efectivo el proceso educativo, mediante técnicas pedagógicas que incrementen el rendimiento del alumno.
- El conocimiento de la práctica docente, que permite seleccionar las técnicas más adecuadas de enseñanza para la situación concreta del grupo destinatario. En esta faceta es muy importante la experiencia práctica acumulada durante los años de docencia.

El profesor debe, por una parte, transmitir unos conocimientos y, fundamentalmente, fomentar la capacidad crítica y auto-formadora del estudiante; por otro lado, debe tener una preparación específica y un conocimiento suficiente de la didáctica para ser capaz de llevar a cabo esta labor de un modo apropiado.

En [50] se propone una clasificación de en qué se centran los profesores cuando piensan sobre el proceso de enseñanza.

- Nivel 1. Enfocarse en qué es el estudiante, consiste en considerar únicamente la capacidad o motivación de los estudiantes como principal factor, echándoles a ellos la culpa cuando se producen malos resultados.

- Nivel 2. Enfocarse en qué hace el profesor, consiste en centrarse en las actividades y conocimientos del profesor, considerando que en caso de fracaso, el problema es porque el profesor no ha sabido motivar o explicar bien una determinada disciplina.
- Nivel 3. Enfocarse en qué hace el estudiante. En este nivel, el profesor se centra en organizar las actividades que el estudiante realiza durante el proceso de enseñanza-aprendizaje, las cuales, si están bien elegidas, le permitirán adquirir las destrezas necesarias.

El método docente que se propone en dicho libro se denomina *alineación constructiva* y consiste en alinear los objetivos de aprendizaje, con las actividades de aprendizaje que se realizarán y las técnicas de evaluación. En la presente propuesta se seguirá dicha metodología.

16.3.1 Clases expositivas

A pesar de que en los últimos tiempos las clases expositivas están siendo muy criticadas, ha sido seguramente el método más utilizado en la enseñanza universitaria tradicional y este tipo de clases también tiene algunas ventajas, además de que los sistemas administrativos suelen mantener la clase expositiva a grupos grandes como una actividad necesaria e incorporada en las planificaciones.

En [72] se relacionan algunas ventajas e inconvenientes de las clases expositivas y también se proponen algunas técnicas que pueden utilizarse para sacar un mayor provecho de las mismas. Algunas de las consideraciones que se proponen son:

- Es un método eficaz para asegurar una experiencia de aprendizaje compartida, donde toda la clase se junta regularmente.
- Puede ser útil para captar la atención y motivar a los estudiantes, que pueden salir de las clases con ganas de aprender.
- Ofrecen una posibilidad para que los estudiantes resuelvan dudas o aclaren ideas de temas que ya han intentado aprender o conocían.
- Permite a los estudiantes *aprender haciendo* obteniendo realimentación en directo del profesor.
- Permiten añadir cambios en el tono de voz, énfasis, expresiones faciales o lenguaje corporal, al material impreso o audiovisual, ofreciéndoles a los estudiantes una visión de qué aspectos son importantes y cuáles no.
- Puede proporcionar material para discusiones o exploraciones posteriores.
- Permite retar las creencias de los estudiantes y tratar de cambiar sus perspectivas y actitudes de forma global a todo el grupo..
- Puede crear ocasiones en las que algunos estudiantes puedan comprender ciertas ideas o consolidar conceptos como una experiencia común con otros compañeros.
- Permiten explicar a grandes grupos de estudiantes tareas relacionadas con las evaluaciones que tendrán que llevar a cabo posteriormente.
- Es un *método económico*, puesto que se puede organizar de una forma rápida, sin requerir grandes inversiones materiales, y repetir varias veces a un gran número de alumnos.
- Las clases expositivas pueden ser grabadas, lo que permitiría que sean visualizadas posteriormente por los estudiantes en el momento que ellos lo deseen.

No obstante, también es posible destacar algunos problemas:

- Puede provocar *pasividad* en el estudiante. En la actualidad, con la proliferación de clases basadas en la utilización de proyectores, las clases pueden acabar convirtiéndose en simples actuaciones de los profesores que recitan los contenidos mientras los estudiantes hacen como si estuviesen atentos.
- Favorece la enseñanza de contenidos alejados de la experiencia real de los alumnos. En el mejor de los casos, es eficaz sólo para alumnos con un desarrollo cognitivo adecuado y familiarizados con la materia que se explica.
- El estudiante percibe sólo la elaboración final, no la génesis ni la estructura del discurso.
- Puede ser muy difícil captar la atención continuada de unos estudiantes que están expuestos a múltiples elementos que les pueden dispersar. En la actualidad, los estudiantes universitarios acuden a las clases provistos de múltiples dispositivos, que muchas veces utilizan durante las clases. Intentar captar su atención puede ser un gran reto al alcance únicamente de profesores con un gran carisma o capacidad de presentación. La posibilidad de prohibir la utilización de dicho material puede ser poco efectiva en el ámbito universitario. En general, se considera mucho más productivo apoyarse en dichas tecnologías y adaptar la técnica expositiva para hacer uso de las mismas.

Algunas de las principales técnicas de enseñanza expositiva podrían ser:

Explicación Oral. Dependiendo de las circunstancias, la explicación puede hacerla verbalmente el profesor o bien ser completada mediante un texto escrito o medios audiovisuales. La técnica de exposición ideal debería contener los siguientes elementos:

- Si se considera necesario, realizar un breve repaso de los contenidos previos requeridos para comprender la exposición a fin de situar a los alumnos en el contexto específico. Conviene tener en cuenta que, normalmente, los estudiantes alcanzan la mayor capacidad de concentración a los 20 ó 30 minutos del comienzo de la explicación. Además, la disposición horaria habitual requiere para los estudiantes participar en varias asignaturas seguidas en un mismo día. Comenzar con un breve repaso permitirá a los estudiantes centrarse en la asignatura.
- Comenzar con el planteamiento de *preguntas-problema* que ofrezcan al estudiante una mayor motivación sobre lo que va a explicarse. En general, es difícil seleccionar este tipo de preguntas-problema y no suele ser una técnica habitual. La posibilidad de acostumbrar a los alumnos a estas introducciones permitirá un mayor aprovechamiento a largo plazo del tiempo de explicación, evitando recurrir a técnicas motivadoras basadas en la intimidación.
- Introducir un *organizador previo*. Se puede realizar mediante la utilización de medios audiovisuales o un pequeño esquema.
- Elaborar la explicación mediante *generalizaciones y ejemplos*. Es conveniente intentar mantener el equilibrio entre los conceptos y terminología teórica introducida (especialmente en disciplinas técnicas o matemáticas) y sus aplicaciones prácticas. Una buena estrategia es utilizar una serie de ejemplos y pasar a presentar una generalización de los mismos mediante la teoría correspondiente. En otras ocasiones, puede exponerse un determinado concepto general y a continuación una serie de ejemplos de dicho concepto.
- Terminar con una *síntesis final* que se relacione con el organizador previo y la pregunta-problema inicial. Conviene mantener una clara organización de la explicación, teniendo especial cuidado con la planificación temporal, que permita disponer de tiempo para la elaboración del resumen final de lo explicado.
- La longitud de la exposición no debe ser elevada, ya que la atención del alumno durante períodos largos de tiempo disminuye. Si se requiere la utilización de más tiempo. Es conveniente la *utilización de descansos* o variaciones temáticas que mantengan el interés de los estudiantes.

Los principales medios didácticos para la explicación oral son:

Pizarra La pizarra forma parte indiscutible del entorno tradicional universitario, a pesar de haber sido denostada por inducir indirectamente en técnicas de enseñanza expositiva demasiado convencionales. Es necesario valorar las ventajas de su aplicación respecto a otros métodos.

- Permite un desarrollo dinámico de la exposición.
- No es necesario apagar las luces de la clase. Favoreciendo el contacto visual entre profesor y estudiantes.
- Permite borrar y escribir nuevos contenidos de forma sencilla, tanto por el profesor como por los estudiantes.

Trasparencias/proyector. Actualmente, la mayoría de las aulas universitarias cuentan con un proyector que permite visualizar las presentaciones realizadas por ordenador. Esta técnica se ha convertido en una de las herramientas más utilizadas de apoyo a las explicaciones orales.

Las principales características podrían ser:

- Permiten al profesor disponer de un *guión de los contenidos* que imparte en cada momento.
- *No ofrecen interactividad*. No es posible modificar sus contenidos de forma sencilla.
- Generalmente, requieren unas condiciones luminosas que obligan a mantener la clase con menos iluminación. En un principio estas condiciones pueden favorecer la concentración, pero también la somnolencia y desinterés de los estudiantes.
- Si las presentaciones se ponen a disposición de los estudiantes, éstos pueden tomar una actitud pasiva durante la clase que puede favorecer su falta de concentración.
- Permite un *desarrollo más interactivo*. Se pueden incorporar animaciones, vídeos o incluso la posibilidad de explicaciones alternativas.
- Permite *enlazar con otras aplicaciones informáticas*. En estos momentos, quizás sea la herramienta ideal para presentaciones informáticas que requieren la combinación entre una presentación y una aplicación concreta.
- Permite incorporar vídeos, animaciones, sonidos, etc. Aparte de los recursos de la máquina, el único límite es la *capacidad creativa* del profesor.

- Puede resultar complicado disponer de las condiciones necesarias para su utilización. Escasez de proyectores, aulas con mala calidad visual, problemas de última hora con software o hardware, etc.

Vídeo. La utilización de vídeos permite combinar imágenes en movimiento y sonidos. Aunque la realización de vídeos no está al alcance de cualquier profesor, hoy en día existen múltiples vídeos de casi cualquier temática disponibles en la Web. Hacer un uso puntual de ciertos vídeos, preferiblemente de breve duración, puede mejorar la atención de los estudiantes. No obstante, también hay que tener en cuenta que impiden la comunicación interactiva y obligan a bajar la iluminación de la sala, facilitando la desconcentración de los estudiantes.

Estudio Dirigido. Mediante esta técnica el profesor va guiando al estudiante por una serie de pasos intermedios hasta que alcance los objetivos finales. Este tipo de estrategia será eficaz cuando:

- La información es moderadamente compleja, y los estudiantes pueden aprenderla significativamente, mediante la repetición de un ciclo que consiste en la realización individualizada de actividades convenientemente programadas.
- Debe tratarse de temas con cierto interés para el alumno.
- Se adapta a las nuevas tecnologías informáticas, pudiendo utilizarse para la enseñanza a distancia. El profesor puede dirigir el estudio de un reducido número de estudiantes indicando las materias que cada estudiante debe ir aprendiendo por su cuenta.

Clases invertidas o Flipped classroom. es un método que ha demostrado su eficacia en el ámbito universitario [62]. Mediante esta técnica, los profesores proporcionan el material didáctico a los estudiantes antes de las clases presenciales para que ellos lo puedan estudiar con antelación. El tiempo dedicado a las clases presenciales se utiliza entonces para comprobar los resultados de su estudio mediante cuestionarios en línea, o actividades de evaluación formativa o discusión en clase. De esta forma, al no tener que dedicar tanto tiempo en las clases a las explicaciones, se dispone de más tiempo para que los estudiantes sean los protagonistas de su propio aprendizaje, con actividades de aplicación o discusión, que les permiten construir conocimiento y extraer sentido por sí mismos de los conceptos de una disciplina.

16.3.2 Enseñanza por Descubrimiento

La enseñanza por descubrimiento aparece como una interesante alternativa respecto a las técnicas tradicionales, las principales ventajas de este tipo de enseñanza son:

- El conocimiento adquirido por descubrimiento suele retenerse más tiempo.
- Favorece una mayor motivación, provocando cambios conceptuales complejos y estableciendo una mayor cercanía con situaciones prácticas reales.
- En algunas disciplinas prácticas, el descubrimiento de heurísticos o algoritmos es un objetivo educativo en sí mismo.

Sin embargo, este tipo de enseñanza no está libre de críticas. Se hace necesaria una valoración rigurosa de la situación educativa particular antes de lanzarse a este tipo de enseñanza. Los aspectos negativos de la enseñanza por descubrimiento podrían ser:

- No es cierto que sólo se entienda lo que uno descubre: la mayor parte de lo que una persona sabe, ha sido descubierto y transmitido por otras personas.
- No es la única alternativa válida a la memorización. Con frecuencia se confunden procesos con productos de aprendizaje. La alternativa al aprendizaje memorístico es el aprendizaje significativo. El descubrimiento no deja de ser un proceso que puede llevar al aprendizaje significativo y no está demostrado que lo produzca de una manera más eficaz y duradera que la explicación.
- Puede ser más lento al considerar que la mera aplicación de una metodología permite acceder a concepciones científicas avanzadas, sin disponer de conocimientos específicos previos.
- No está al alcance de todos los alumnos.
- Motiva y da confianza sólo si concluye con éxito; por lo que puede ocasionar disfunciones a causa de la personalidad del profesor, la falta de motivación inicial del alumno, o por su experiencia con descubrimientos triviales anteriores que podría haber aprendido de otra manera.
- Sólo se desarrollan capacidades potenciales de forma espontánea, además, puesto que el deseo de aprender no suele ser espontáneo, se hace necesaria una estimulación externa.
- Su aplicación es arriesgada y difícil con grupos grandes de estudiantes.

El descubrimiento puede hacerse de forma autónoma, lo que escapa a situaciones académicas normales, o de manera más o menos guiada. Existen dos tipos de procesos fundamentales que pueden adquirirse por

descubrimiento.

- *Procesos algorítmicos*: Son una serie de operaciones mentales que llevan, en un número finito de paso, a la solución de un problema. Aparecen en una gran cantidad de disciplinas y aunque su utilización puede convertirse en una rutina, su aprendizaje puede conllevar un proceso de descubrimiento de las relaciones y conexiones existentes entre elementos aparentemente dispares.
- *Procesos Heurísticos*: Operaciones de búsqueda no elementales, que no tienen por qué resolver íntegramente el problema. El estudiante desarrolla la habilidad de utilizar una serie de reglas generales que le permiten afrontar la búsqueda de soluciones. Estas habilidades serán aplicables para resolver otro tipo de problemas similares o no al problema original.

16.3.3 Aprendizaje basado en problemas/proyectos

En el aprendizaje basado en problemas el docente se convierte en el tutor de un grupo pequeño de alumnos que tienen que resolver un problema específico relacionado con la disciplina de estudio. Es una propuesta educativa que ya había sido planteada por John Dewey a principios del s. XX [23] y que se ha venido utilizando en múltiples instituciones universitarias [26].

Algunas de las características del aprendizaje basado en problemas son:

- El aprendizaje está centrado en el estudiante
- El aprendizaje puede realizarse en pequeños grupos, lo cual puede mejorar otras habilidades como la capacidad de trabajo en equipo, comunicativas, etc.
- El profesor actúa como un facilitador o guía del proceso.
- Los problemas son un vehículo para desarrollar habilidades de resolución de problemas
- La información se adquiere mediante aprendizaje autodirigido.

El aprendizaje basado en problemas está muy relacionado con el *aprendizaje autorregulado*, mediante el cual el estudiante debe tener una actitud activa para controlar el tiempo y esfuerzo que dedica al aprendizaje, y el *aprendizaje colaborativo* en el que los problemas son resueltos por equipos de estudiantes que tienen que colaborar entre sí.

Esta metodología ofrece varios retos que deben tenerse en cuenta cuando se plantea:

- Este tipo de metodologías ofrece una cierta incertidumbre tanto para los estudiantes que no saben si serán capaces de resolver los problemas, como para los profesores, que pueden no saber cómo van a afrontar los estudiantes los problemas planteados. Dependiendo del tipo de problema, puede ocurrir que los profesores tampoco conozcan todas las soluciones o variantes que se puedan producir.
- Es importante acotar el tiempo y recursos que los estudiantes van a destinar a la resolución del proyecto o problema. Pueden darse ocasiones de estudiantes excesivamente motivados que dejen de lado otras materias de un curso para centrarse únicamente en el proyecto planteado. La autorregulación es necesaria tanto para que los estudiantes trabajen en el problema, como para que sepan poner freno al esfuerzo destinado al mismo.
- El profesor debe responsabilizarse de elegir los problemas a resolver, que pueden variar entre los extremos de problemas abiertos, sin una solución conocida, o cerrados, con una determinada solución que el profesor normalmente conoce y espera que los estudiantes encuentren.
- En trabajos realizados en grupos, es necesario decidir el tamaño y la composición de los mismos. Aspectos que pueden determinar el éxito o fracaso de la experiencia para un estudiante concreto.
- Algunos estudiantes con un rendimiento elevado en actividades individuales, pueden encontrar problemas para integrarse en el trabajo en grupo o preferir trabajar de forma individual al entender que otros estudiantes pueden empeorar su rendimiento.
- La motivación y dedicación de cada estudiante al trabajo en grupo puede ser dispar, creándose posibles problemas entre los equipos que los profesores deben tratar de detectar y resolver.

16.3.4 Prácticas de aula y Seminarios

Se trata de actividades organizadas en aulas similares a las clases expositivas, pero con grupos más reducidos, lo cual puede permitir la organización de algunas actividades de aprendizaje alternativas a la clase expositiva, como puede ser el planteamiento y resolución de problemas, o incluso las presentaciones, exposiciones o debates de trabajos individuales o en grupo realizados por los estudiantes en una audiencia más manejable.

16.3.5 Clases de prácticas de laboratorio

Las clases prácticas de laboratorio se realizan en grupos reducidos de alumnos en un laboratorio, que en la Universidad de Oviedo oscila entre 12 y 18 estudiantes. Permiten una mayor interacción entre el estudiante y el profesor favoreciendo una realimentación beneficiosa para ambos.

En las carreras técnicas, las prácticas de laboratorio forman un núcleo esencial en la formación de los alumnos.

Una clasificación de los diversos métodos utilizados para llevar a cabo este tipo de actividades sería:

Clases abiertas. Los alumnos disponen de libertad para resolver los problemas planteados por el profesor.

Los alumnos se enfrentan por sí mismos a la resolución de los problemas y sólo cuentan con las ayudas proporcionadas por el entorno de desarrollo o de sus conocimientos.

Clases abiertas tuteladas. Los alumnos deben resolver una serie de problemas, pero cuentan con la presencia del profesor.

El profesor estará disponible para resolver pequeñas dudas de los alumnos como errores de programación o de concepto leves. El profesor intentará no resolver él mismo los problemas planteados, puesto que se perderá la esencia de las prácticas

Clases abiertas con guión. El profesor plantea pequeños ejercicios que los alumnos van resolviendo en cortos espacios temporales. Se intenta que los alumnos no se queden atascados y que vayan resolviendo los diversos ejercicios por su cuenta. El profesor cuenta con un guión de ejercicios que va planteando secuencialmente a los alumnos. En algunos casos, el profesor puede proporcionar ejercicios resueltos similares a los ejercicios planteados que les servirán a los alumnos para hallar las soluciones por comparación.

Clases guiadas. El profesor explica pequeños conceptos y los alumnos repiten las acciones realizadas por el profesor.

Además de la clasificación anterior, la metodología didáctica a utilizar en las clases de laboratorio requiere unas atenciones especiales. La docencia adecuada de prácticas de laboratorio exige un esfuerzo y dedicación superior, en muchas ocasiones, a la docencia de clases teóricas. Se debe tener en cuenta:

- Es necesario comprobar el correcto funcionamiento de los equipos requeridos para la docencia. Las conocidas *leyes de Murphy* aparecen con demasiada frecuencia en este tipo de actividades. Además, la instalación y utilización del *software* necesario puede suponer una tarea compleja que no se debe subestimar.
- También hay que asegurarse que los alumnos conocen cómo utilizar los equipos de prácticas, así como el objetivo y metodología de las prácticas a realizar.
- Conviene comprobar que el tiempo requerido para la realización de las prácticas deja margen suficiente para cualquier eventualidad y permite que los alumnos menos hábiles puedan desarrollar su trabajo.
- El profesor estará preparado para resolver las dudas claramente y en un tiempo compatible con el trabajo a realizar. La utilización de algún soporte docente, como cañones retroproyectores, podrá facilitar la resolución de dudas comunes.
- Es conveniente no entorpecer ni atosigar al alumno con continuas indicaciones o correcciones. Dejándole que tropiece con los problemas e intente solucionarlos por sí mismo, aprendiendo de sus propios errores.
- Finalmente, es necesario comprobar que el nivel de las prácticas está en consonancia con el nivel de teoría. De lo contrario, las prácticas pueden desmotivar a los alumnos.

Un aspecto interesante de las clases de laboratorio es que pueden utilizarse para afrontar otras actividades alternativas, como el aprendizaje basado en problemas o proyectos, que permitirá a los estudiantes resolver de forma individual o en grupo los problemas o proyectos propuestos por los profesores, lo cuales pasarán a tener un rol de guías en el proceso.

16.3.6 Tutorías

El *New College* fundado en Oxford en 1379, puede considerarse la cuna del sistema tutorial, que consistía en que cada postgraduado, residente en el College, se encargaba de dirigir a uno o varios alumnos en sus estudios de forma individualizada. El sistema prosperó en Oxford y en Cambridge donde la enseñanza se basaba en la relación estrecha tutor-alumno: el tutor orienta al alumno en su labor académica y en general en su vida universitaria.

Este sistema adquirió gran prestigio y algunas universidades de todo el mundo han intentado recoger lo más significativo de él, es decir, el contacto individual profesor-alumno.

Las consultas permiten que cualquier alumno pueda exponer las dudas que se le han planteado al estudiar

los temas o profundizar en aspectos concretos, o solicitar orientación bibliográfica específica o más avanzada.

La labor del profesor, como tutor, no debe ser la de resolver solamente los problemas al alumno, sino la de indicar posibles caminos de solución, invitando al trabajo continuado para que el sistema sea realmente efectivo.

Las tutorías resultan beneficiosas, no sólo para el alumno que consulta, sino para la docencia general de la asignatura, puesto que el profesor puede notar si existen ciertas deficiencias conceptuales o de base, y así poder subsanarlas en próximas clases.

16.3.7 Actividades complementarias

Como se ha mencionado, además de las tareas reconocidas oficialmente, un profesor universitario suele encargarse de otro tipo de actividades complementarias que mejoran la calidad de la enseñanza, entre las que se podrían destacar las siguientes:

Preparación de material didáctico

La confección de material didáctico para una asignatura, bien sea mediante apuntes, software, etc. es una actividad docente de gran importancia. En algunas ocasiones es conveniente valorar el tipo de material a realizar. Cuando los estudiantes disponen de apuntes oficiales de una asignatura pueden adoptar actitudes más pasivas, evitando tomar notas en clase o no buscando bibliografía de consulta en la biblioteca. Para evitarlo, el profesor deberá incentivar la búsqueda de fuentes de consulta alternativas o poner ejemplos diferentes de los que aparecen en los apuntes, a fin de ofrecer una mayor variedad de perspectivas sobre el tema.

Utilización de Software de soporte

Los estudiantes universitarios actuales, que suelen estar enmarcados en lo que se conoce como *nativos digitales* están acostumbrados a utilizar material informático de soporte. Esto se da todavía más en estudiantes de titulaciones informáticas. Por ese motivo, se considera importante aportar material digital complementario a la enseñanza, así como participar en los diferentes entornos de soporte a la enseñanza proporcionados por la institución.

Actualmente, la Universidad de Oviedo cuenta con licencias de uso para dos tipos de plataformas:

- Plataforma Moodle. Se trata de un sistema de gestión de aprendizaje abierto, que comenzó a desarrollarse en el año 2002 y que en la Universidad de Oviedo se incorpora en lo que se denomina Campus Virtual¹. Proporciona numerosas utilidades y herramientas para la enseñanza como son la posibilidad de utilizar foros de discusión, chats, wikis, incluir material lectivo, etc.
- Plataforma Microsoft Teams. Se trata de una herramienta inicialmente creada por Microsoft para la gestión y comunicación empresarial, que ha ido adquiriendo cada vez más aceptación para facilitar la enseñanza no-presencial gracias a la posibilidad de creación de reuniones virtuales. Este software pasó a ser de gran importancia durante la pandemia, en el que se convirtió en el elemento imprescindible para impartir la docencia no presencial.

Además de dichas plataformas, existen múltiples tecnologías que pueden facilitar la enseñanza y el trabajo colaborativo como Zoom², Google Meet³, Slack⁴, Discord⁵, etc.

Organización de debates, seminarios y conferencias

La realización de debates o discusiones puede favorecer la creación de un clima científico entre los estudiantes, motivándoles para tomar partido por determinados puntos de vista y aumentando su capacidad crítica. Estos debates pueden realizarse dentro del ámbito de una asignatura o de forma pública con asistencia de otras personas.

Otra posibilidad es la realización de seminarios o talleres, que pueden servir como complemento en la formación del alumnado y para el propio perfeccionamiento del profesorado en áreas específicas. El objetivo de los seminarios es ampliar las perspectivas académicas y profesionales de los alumnos, así como completar los contenidos del Plan de Estudios que, por la dinámica de la Informática, pueda quedarse desfasado en algunas materias.

¹<http://campusvirtual.uniovi.es/>

²<https://zoom.us/>

³<https://meet.google.com/>

⁴<https://slack.com/>

⁵<https://discord.com/>

Estas actividades complementarias, pueden tener los siguientes objetivos:

- Cubrir áreas de interés de tipo aplicado, no previstas en el Plan de Estudios, a fin de aumentar la enseñanza de tipo práctico o aplicado o incluso preparando el camino de los estudiantes de grado al máster o a la vida profesional.
- Introducir áreas o temas informáticos específicos novedosos, que no se expongan ya en otras asignaturas del Plan de Estudios, por carencias de mismo, o por falta del tiempo necesario, y que sean de interés para los alumnos.
- Profundizar en temas ya vistos, desde un punto de vista teórico o aplicado, a fin de cubrir inquietudes o deseos de especialización en campos concretos.

Tecnologías de Enseñanza no presencial

Las tecnologías multimedia, la aparición de Internet y la reciente pandemia han propiciado la aparición de toda una serie de técnicas que facilitan la enseñanza no presencial. A pesar de que la Universidad de Oviedo mantiene una estructura de formación presencial, la utilización de material didáctico no presencial puede ser un complemento útil que tal y como se ha visto durante la pandemia, evita desplazamientos y ofrece una mayor independencia de horarios al estudiante.

Entre las tecnologías de este tipo, se pueden destacar:

- *Página Web*. La creación de páginas Web con contenido de interés para los estudiantes de una asignatura es un complemento importante de la docencia. El material que se incluya en la página puede incluir apuntes, exámenes, enunciados de prácticas, ejercicios, enlaces relacionados, etc. El autor de esta memoria ha realizado la correspondiente página Web de la asignatura que está disponible en <http://arquisoft.github.io/>. Esta página se utiliza como una especie de *landing page* con información abierta de la asignatura, que ofrece mayor visibilidad del contenido impartido a lo largo del tiempo y es visible por cualquier persona interesada.
- *Correo electrónico*. Ofrece un mecanismo de comunicación asíncrono entre profesor y estudiante. El mayor problema de este sistema es la saturación por parte del profesor, que puede tener dificultades para atender a todos los correos que le puedan llegar de sus estudiantes.
- *Listas de correo*. Ofrecen un sistema de comunicación restringido al ámbito de los estudiantes matriculados en la asignatura, en el que las preguntas y respuestas son recibidas por todos los que pertenecen a la lista. Tienen varias ventajas sobre el correo electrónico para su uso en la docencia de clases masificadas ya que el hecho de que todos los miembros de la lista reciban las respuestas evita una saturación por parte del profesor. Además, si se fomentan actitudes colaborativas, pueden ser los propios estudiantes los que respondan y ayuden a otros compañeros. En ocasiones el problema de saturación aparece cuando la lista es demasiado grande o se tratan temas que no son de interés para todos los miembros de la lista.
- *Foros de discusión*. Permiten a los miembros del foro intercambiar opiniones sobre temas concretos en un sistema que normalmente almacena dichas opiniones para su posterior consulta. A diferencia de las listas de correo, los foros de discusión evitan la saturación al clasificar los temas por hilos de discusión y ser accedidos por los miembros a través de páginas Web cuando lo desean.
- *Sistemas colaborativos de edición*. Los sistemas de tipo *wiki* son servidores de información que permiten a sus usuarios crear y editar los contenidos. El ejemplo más popular es *wikipedia*⁶. La utilización de estos sistemas en la enseñanza parece ofrecer nuevas posibilidades interesantes [41].
- *Chat*. Los sistemas de conversación ofrecen una tecnología de comunicación síncrona con alto grado de interacción. Algunos de estos sistemas incorporan herramientas que permiten intercambiar dibujos, voz e imágenes. Aunque esta tecnología ofrece una mayor sensación de interacción, puede requerir un gran esfuerzo por parte del profesor si intenta resolver las dudas de todos los estudiantes en tiempo real.
- *Vídeoconferencia*. Esta tecnología ofrece una gran sensación de cercanía al permitir intercambiar voz e imágenes. Durante la pandemia se ha llevado a cabo su utilización de forma general y se ha demostrado que puede ser una tecnología viable. Una ventaja adicional es que se pueden grabar las clases, lo que permite al estudiante volver a ver las grabaciones en el momento que lo considere oportuno.

⁶<http://www.wikipedia.org/>

Trabajos Fin de Grado/Máster

Pese a que la realización del *Trabajo Fin de Grado/Máster* no forma parte del contexto académico de la asignatura *Arquitectura del Software*, merece una especial mención por los conocimientos y aprendizaje que aporta al estudiante y al profesor.

La evolución científica y tecnológica de las asignaturas no siempre puede estar recogida con profundidad en el programa de la misma. Los trabajos fin de grado son un buen método también para evolucionar en una línea temática, adquiriendo conocimientos nuevos y más especializados. La realización de trabajos fin de carrera en los que pueden participar los estudiantes de la asignatura, bien como usuarios de los mismos, o colaborando en algunos aspectos del desarrollo, puede tener aspectos motivadores, tanto para los autores de los trabajos que se encuentran con usuarios *de verdad*, como para los estudiantes de la asignatura que ven lo que pueden llegar a hacer al finalizar la carrera.

Contacto con titulados

A través de su paso por la Escuela, los titulados adquieren unos conocimientos de base amplios. No obstante, en ciertos temas específicos, deben ampliarse, perfeccionarse y sobre todo, actualizarse. En el ámbito de la asignatura es posible organizar conferencias, o incluso cursos de extensión y posgrado, que permitan realizar la labor de actualización mencionada y, por otra parte, permitan a los profesores universitarios estar más en contacto con la realidad social y profesional de los egresados.

Además, algunos egresados, cuando adquieren experiencia profesional acaban ejerciendo de arquitectos de software en sus empresas. Se considera que poder contar con la experiencia y asesoramiento de antiguos estudiantes que están ejerciendo de arquitectos de software es una oportunidad única para los profesores universitarios, que tiene múltiples ventajas, como el acercamiento entre la Universidad y la Industria, el establecimiento de canales de comunicación y colaboración, la actualización de contenidos a impartir, etc.

Movilidad estudiantil

Además de impartir docencia de su asignatura, el profesor universitario debe facilitar a los estudiantes la realización de experiencias que aumenten su formación integral, como puede ser la realización de estancias en otros centros universitarios. En opinión del autor de esta memoria, debe fomentarse la realización de acuerdos de intercambio con otras instituciones. Este tipo de acciones requiere una dedicación y esfuerzos que muchas veces no suelen estar suficientemente reconocidos. Entre los programas de intercambio, se encuentran las *becas Erasmus* (intercambio de estudiantes dentro de la Comunidad Europea), *becas SICUE Séneca* (intercambio de estudiantes entre universidades nacionales) o las *becas FARO* para facilitar la experiencia laboral de estudiantes en otros países.

16.4 Evaluación de la enseñanza

A continuación se describen brevemente algunas de las principales técnicas de evaluación de conocimientos. Posteriormente se indica cuál es el tipo de evaluación que se utilizará en la asignatura y finalmente se indican aspectos de evaluación de la propia actividad docente.

16.4.1 Técnicas de evaluación

Tal y como se indica en [43]: La evaluación del aprendizaje de los estudiantes es una función fundamental de la educación superior. Es el medio mediante el cual se aseguran y expresan los estándares académicos y tiene una importancia vital sobre el comportamiento del estudiante, el tiempo del profesorado, la reputación de la Universidad y, sobre todo, sobre las vidas futuras de los estudiantes.

Tradicionalmente, la evaluación de los conocimientos o del rendimiento académico se ocupaba únicamente de los resultados o productos finales del proceso enseñanza-aprendizaje. El elevado tiempo, esfuerzo, medios y dramatismo puestos en juego en esta fase del proceso por parte del profesor y los estudiantes, no tienen, frecuentemente, relación con los resultados que se pretenden conseguir. Actualmente, la evaluación en el campo educativo toma como objeto de estudio tanto los resultados, como el propio proceso seguido para conseguirlos.

La evaluación sirve para:

- Orientar al estudiante y profesor sobre los aprendizajes y competencias adquiridas.
- Motivar (aunque también puede convertirse en motivación negativa)
- Valorar los programas y métodos docentes.
- Calificar, con unas implicaciones legales importantes.

En [72] se indica que uno de los problemas de la evaluación es que casi todos los profesores que la realizan, han sobrevivido a alguna forma de evaluación que se les ha hecho. Lo cual puede hacer que se resistan a plantearse si la evaluación que ellos recibieron fue válida, justa o transparente, y puede explicar por qué muchas formas de evaluación tradicional todavía se siguen realizando en la educación universitaria. Además, la aplicación de nuevas técnicas de evaluación puede venir acompañada de una mayor carga de trabajo que normalmente no se ve reconocida.

Entre las características deseables de una buena evaluación, se destacan:

- *Validez*, el formato de la evaluación debe estar acorde a los resultados de aprendizaje
- *Fiabilidad*, diferentes evaluadores deberían estar de acuerdo en las marcas que se asignan. También se relaciona con la percepción que puedan tener los estudiantes de que la calificación es justa.
- *Autenticidad*, mide hasta qué punto se correlaciona con las cosas que los estudiantes necesitarán ser capaces de realizar al dejar la institución educativa. Se refiere a la relevancia en el mundo real de la actividad de evaluación.
- *Transparencia*, se refiere a que los estudiantes puedan ver cómo funciona la evaluación y el proceso de calificación.
- *Inclusividad*, se refiere a que la evaluación pueda ser llevada a cabo por un rango de estudiantes con necesidades de aprendizaje especiales, con el fin de minimizar la posible discriminación de candidatos con necesidades particulares.
- *Manejabilidad*, la actividad de evaluación debe ser manejable en la práctica, es decir, debe poder ser afrontada con los límites de recursos temporales existentes, tanto por parte de los profesores, como de los estudiantes.

Existen múltiples técnicas de evaluación [40, 72] entre las que se pueden destacar:

1. *Pruebas tradicionales abiertas*. Consisten en proponer al alumno un número reducido de cuestiones, cuya amplitud sea relativamente extensa, de forma que éste las desarrolle con total libertad. En estas pruebas se pueden comprobar los conocimientos adquiridos por el alumno. Las pruebas abiertas conceden más libertad al estudiante para que exponga sus ideas con cierta creatividad, permitiendo organizar la respuesta según su estilo personal de aprender. Entre los inconvenientes pueden citarse la dificultad de objetivar la calificación o la confusión que a veces se genera entre la capacidad expresiva del alumno y el verdadero dominio de la materia. Los exámenes pueden promover un tipo de aprendizaje *superficial*, en el que los estudiantes dejan para los días previos al examen el estudio de la disciplina, aprendiendo unos contenidos de forma rápida que se olvidan a la misma velocidad que se aprendieron. La evaluación de los exámenes de pregunta abierta también puede verse influida por factores subjetivos del evaluador.
2. *Pruebas abiertas con apuntes o libro*. Es una variante de las pruebas anteriores, en las que se permite la utilización de libros de texto o apuntes. Una ventaja de esta posibilidad es que se da menos importancia al aprendizaje memorístico consistente en *reproducir cosas*, enfatizando la capacidad de relacionar conceptos y la creatividad. Una desventaja es que la creación de este tipo de pruebas puede ser más compleja, además de las dificultades logísticas de gestionar cuántos materiales extra pueden acarrear los estudiantes. Otra variante de este método, es permitir el uso de ordenadores portátiles con acceso (o no) a Internet, que puede facilitar la logística, pero abriría nuevos retos para gestionar qué tipo de equipamiento se permite que garantice la igualdad entre estudiantes.
3. *Pruebas estructuradas u objetivas* Una *prueba objetiva* es un tipo de examen estructurado, donde a cada alumno se le presentan exactamente las mismas preguntas o preguntas equivalentes. El ejemplo más común de pruebas objetivas son los exámenes con preguntas de elección múltiple. Están compuestas por un conjunto de preguntas o ítems que se traducen en fórmulas cerradas en las cuales el alumno no elabora la respuesta, sino que únicamente tiene que indicar la respuesta entre una serie de alternativas.

La determinación de las respuestas aceptables o refutables ha de ser unívoca y exacta, lo cual da lugar a que se puedan calificar objetivamente, y significa que, independientemente de quien las valore, se llegarán a los mismos resultados.

Las principales ventajas de las pruebas objetivas son:

- Permiten medir una amplia muestra del contenido y operaciones mentales sobre las que se desea obtener información, gracias al elevado número de preguntas con que pueden contar.
- Su calificación es objetiva y por tanto, es fácil, rápida y consistente.
- Si están bien construidas, permiten medir una buena parte de los aprendizajes o destrezas adquiridos por los alumnos.

- Facilidad de aplicación y corrección. La corrección de pruebas objetivas se puede realizar mediante plantillas o de forma mecanizada.
 - Facilitan la elaboración y análisis de los resultados.
 - Pueden abarcar un mayor espectro del temario en tiempo limitado.
 - Los exámenes de años anteriores pueden estar disponibles para años siguientes, creando una batería de pruebas que puede ser utilizada incluso como material de entrenamiento por los estudiantes.
4. *Pruebas orales.* Tienen importantes ventajas pues son útiles para valorar la calidad de ciertos aprendizajes, las estrategias en la resolución de problemas, el análisis de dificultades individuales, el uso del lenguaje específico de la materia, las habilidades adquiridas para la comunicación oral, la capacidad de intervenir en discusiones de grupo, etc. También tienen inconvenientes como la subjetividad, dificultad en la calificación, gran inversión de tiempo en asignaturas excesivamente masificadas o el posible bloqueo de algunos estudiantes.
5. *Pruebas prácticas o de ejecución.* En este tipo de pruebas se pide al estudiante una actividad real para comprobar el dominio de una materia. La comprobación toma en cuenta tanto los procesos como los productos. La información que aporta una prueba de ejecución es mucho más amplia que cualquier otro tipo de prueba.
- Dada la naturaleza de las asignaturas en enseñanzas técnicas, es evidente la importancia de este tipo de pruebas en la evaluación global del alumno. Un ejemplo serían los exámenes prácticos en asignaturas de programación.
6. *Ensayos o trabajos.* Consiste en solicitar a los estudiantes la realización de un ensayo o trabajo que profundice sobre algún tema. Este tipo de evaluación puede permitir una mayor individualidad y fomentar la creatividad de los estudiantes, reflejando un aprendizaje en mayor profundidad que los exámenes. Un problema de esta técnica puede ser la fiabilidad de las evaluaciones, dado que diferentes profesores pueden asignar calificaciones dispares, por lo que es necesario establecer criterios de valoración homogéneos. En ocasiones, con esta técnica se acaban valorando más las capacidades de expresión y aspectos formales de presentación, que el propio contenido. Una evaluación adecuada de estos trabajos también puede consumir un gran tiempo por parte de los profesores, además de que las subjetividades pueden influir considerablemente en las evaluaciones.
7. *Presentaciones:* consiste en la realización de presentaciones por parte de los estudiantes sobre un determinado tema a una audiencia (normalmente los compañeros más el profesor o profesores del curso). Una de las principales ventajas es que en las presentaciones no hay dudas de quién está siendo evaluado y que las habilidades requeridas para hablar en público pueden ser muy importantes en las futuras carreras de los estudiantes. Al enfrentarse a realizar presentaciones, se requiere una reflexión y auto-evaluación por parte de los estudiantes que puede incidir en un aprendizaje menos superficial. Si las presentaciones van seguidas de preguntas y respuestas, los estudiantes también se pueden acostumbrar a preparar entrevistas o exámenes orales. Entre las desventajas, este tipo de actividad puede requerir una gran cantidad de tiempo y algunos estudiantes pueden encontrar esta actividad estresante. Otro aspecto que hay que considerar es que en general, las presentaciones no pueden realizarse en paralelo, por lo que los estudiantes que presentan más tarde, pueden aprender aspectos de las presentaciones iniciales. Con lo que las evaluaciones pueden también verse influidas.
8. *Portafolios* consisten en colecciones de evidencias realizadas por los estudiantes sobre actividades que han llevado a cabo durante un curso. Normalmente, los portafolios dicen mucho más sobre el trabajo de los estudiantes que un examen final. Una desventaja de esta técnica es que su revisión puede consumir una gran cantidad de tiempo y su evaluación objetiva puede ser complicada.
9. *Muestras o exhibiciones de pósteres* es una técnica que consiste en pedir a los estudiantes que sinteticen sus conocimientos de un determinado tema en un póster auto-explicativo que puede exhibido. Puede complementarse con un acto de exhibición pública de los pósters en la que los estudiantes estén presentes y se les pueda realizar preguntas sobre los mismos. Esta técnica tiene varias ventajas al inducir en los estudiantes la capacidad de síntesis y fomentar un aprendizaje en profundidad para que puedan resolver las cuestiones que se les plantean. Aunque esta técnica tiene numerosos aspectos positivos, puede ser difícil encontrar evaluaciones fiables de los mismos, influyendo múltiples aspectos en la evaluación. A veces incluso puede ser complicado identificar la verdadera autoría de los mismos.
10. *Evaluación basado en problemas.* Como se ha indicado en la sección 16.3.3, el aprendizaje basado en problemas o proyectos es una técnica que ha alcanzado una gran popularidad y va acompañada de la correspondiente evaluación a partir de las acciones llevadas a cabo durante el mismo. Estas acciones

pueden consistir en presentar una posible solución al problema planteado. Este tipo de planteamientos puede ofrecer una experiencia de mayor autenticidad al reflejar mejor lo que se pueden encontrar los estudiantes en el mundo real. Cuando los problemas planteados son abiertos y no tienen una única solución, la evaluación de los posibles candidatos a solución puede ser compleja.

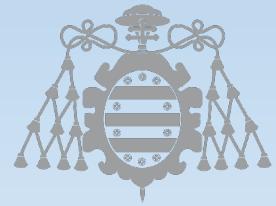
En ocasiones, el aprendizaje basado en problemas se realiza mediante trabajos en equipos, por lo que la evaluación de los mismos requiere identificar también los aspectos de trabajo en grupos y los aspectos de trabajo individual.

16.4.2 Evaluación del Docente

Por último, se considera que el docente debe también evaluar su trabajo, asumiendo las críticas de forma constructiva y analizando algunos aspectos importantes: el cumplimiento de la programación del contenido de la disciplina y desarrollo del curso, su actitud hacia la asignatura y ante los estudiantes, la creación de un clima de participación adecuado (teniendo en cuenta las limitaciones existentes), el rendimiento de los estudiantes y el análisis de las encuestas que normalmente se efectúan a los estudiantes. Además, el profesor debe comprometerse en un proceso de perfeccionamiento y reciclaje de conocimientos permanente.

Actualmente, son muchas las universidades que realizan encuestas sobre los profesores, en las que se estudian aspectos relacionados con el conocimiento de la asignatura, la claridad en las explicaciones, la habilidad para mantener la atención en clase, la puntualidad, etc.

Estas encuestas, unidas a la auto-evaluación de los propios profesores—para detectar factores circunstanciales que no queden reflejados en las encuestas—pueden constituir una forma de evaluación docente del profesorado que facilite una mejor calidad en su docencia.



Universidad de Oviedo

17. Propuesta docente

En este bloque se establece el núcleo del diseño curricular para la asignatura de Arquitectura del Software y se describe la guía docente propuesta para la asignatura.

En primer lugar, se indican los *Resultados de aprendizaje* y las *competencias generales* de la asignatura. Posteriormente se establecen los contenidos mediante unidades didácticas.

La especificación del contenido se realiza inicialmente de forma general, tal y como aparece en la guía docente, y posteriormente de forma detallada identificando las principales unidades didácticas que se corresponden con actividades de enseñanza de unas 2 horas. En cada unidad didáctica se describen los principales temas que se tratarán, junto con unas recomendaciones bibliográficas que indican dónde se puede profundizar en dichos temas o en los que están particularmente bien descritos.

Una vez definida en detalle la información relativa a la materia teórica, se describen los objetivos de las prácticas, tanto de aula (seminarios) como de laboratorio, así como la propuesta de evaluación que se realizará, seguido de una bibliografía comentada. Finalmente, se incluye en un anexo la guía docente de la asignatura que ha sido propuesta por el autor de esta memoria, que es coordinador actual de la misma, al departamento de informática de la Universidad de Oviedo.

17.1 Competencias y Resultados de aprendizaje

Los competencias y resultados de aprendizaje proporcionan criterios sobre qué es lo que se debe enseñar y cómo hacerlo, ayudan a clarificar los tipos de actividades que deben ser desarrolladas y sirven de guía de evaluación del aprendizaje, estableciendo una relación entre lo que se enseña y lo que se evalúa.

17.1.1 Competencias

De acuerdo con la Memoria de Verificación del título de Grado en Ingeniería del Software de la Universidad de Oviedo, las competencias generales que el alumno debe adquirir al cursar la asignatura de Arquitectura del Software son las siguientes.

Competencias Generales Instrumentales: Habilidades metodológicas

- CG-1: *Competencia para el diseño de soluciones a problemas complejos humanos*

Competencias específicas comunes a la rama informática

- Com.1: *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.*
- Com.8: *Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.*
- Com.11: *Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.*

Competencias específicas-Tecnología específica de Ingeniería del Software

- ISW.1: *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.*
- ISW.3: *Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.*
- ISW.4: *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.*

Otras competencias relacionadas con la arquitectura del software

Aunque en la memoria de verificación del grado no se asociaron directamente otras competencias a la asignatura, hay otras competencias en la memoria que se consideran importantes y que también se alcanzan indirectamente a través de algunas de las actividades propuestas, como son:

- CG-9: *Habilidades de Comunicación oral (bien directa, bien apoyada por medios audiovisuales)*
- CG-10: *Destreza en la comunicación oral y escrita en inglés.*
- CG-11: *Competencia para el trabajo en equipo.*
- CG-12: *Liderazgo*
- CG-13: *Negociación.*
- CG-15: *Capacidad para la gestión humana, técnica, funcional y económica de sistemas complejos*
- CG-18: *Sentido de la responsabilidad*
- CG-22: *Capacidad de planificación y organización*
- CG-24: *Toma de decisiones*
- CG-25: *Razonamiento crítico*
- CG-28: *Motivación por la calidad*

17.1.2 Resultados de aprendizaje

En el siguiente cuadro se indican los resultados de aprendizaje establecidos para la materia de Ingeniería del Software, asociados a las competencias con las que se establecen en la memoria de verificación del título [20]. De esos resultados de aprendizaje, el RA.IS-2 y el RA.IS-8 no se incluyen en la guía docente porque las competencias que proporcionan no están incluidas en las competencias planteadas para la asignatura en la memoria. No obstante, se incluyen porque aunque no tengan una relación directa según la memoria, algunas de las actividades realizadas en la asignatura sí que pueden influir en dichos resultados de aprendizaje.

- RA.IS-1. Realizar Proyectos de Ingeniería del Software complejos que den solución a problemas reales y solucionarlos mediante técnicas y tecnologías relacionadas con los procesos de fabricación de software incluyendo frameworks, patrones arquitectónicos, patrones de diseño y de integración persiguiendo el desarrollo de software de calidad: [Com1], [Com2], [Com.8], [Com10], [Com13], [ISw.1], [ISw.4], [ISw.6], [Bas5] , [CG1], [CG3],[CG4],[CG5],[CG10], [CG22].
- RA.IS-2. Dirigir/Integrarse en equipos de trabajo para el desarrollo de aplicaciones software en cualquiera de los grados de implicación técnica [CG9], [CG12], [CG13], [CG18], [CG19], [CG22], [CG24], [Bas5], [Bas6]
- RA.IS-3. Aplicar distintas técnicas de construcción de software en el diseño de bajo nivel: [Com.8], [ISw.1], [ISw.4], [CG1].
- RA.IS-4. Desarrollar diseños y programación orientados a objetos con un elevado nivel de competencia: [Com.8], [ISw.4], [CG1], [CG4].
- RA.IS-5. Evolucionar y refactorizar diseños existentes ante cambios en los requisitos: [ISw.1], [ISw.6].
- RA.IS-6. Determinar el grado de mantenibilidad, fiabilidad y eficiencia de diseños software: [Com.8], [ISw.1], [ISw.4], [CG1].
- RA.IS-7. Diseñar e implementar software aplicando diferentes tecnologías *middleware*: [Com11], [ISw3].
- RA.IS-8. Extraer, Organizar, Negociar, Priorizar y Validar requisitos funcionales y no funcionales utilizando diferentes técnicas [CG7], [CG9], [ISW2]
- RA.IS-9. Diseñar y llevar a cabo controles e inspecciones eficientes y efectivas de validación, verificación y calidad, planes de pruebas: [Com1], [Com16], [ISw1], [ISw4], [CG2], [CG4], [CG11], [CG28].
- RA.IS-10. Analizar estadísticamente la densidad de defectos y probabilidad de fallos: [Com16], [ISw4], [ISw5], [CG25].
- RA.IS-11. Evaluar la calidad de un proceso de software desde el punto de vista de la calidad del producto: [Com1], [Com18], [ISw1], [CG2], [CG11], [CG28].

17.2 Contenidos

La selección de los contenidos teóricos de esta asignatura está basada en diversas consideraciones. Además de las directrices generales que se establecen en los estudios del Grado en Ingeniería Informática del Software, también se tienen en cuenta las principales recomendaciones curriculares descritas en el capítulo 15. Por otra parte, la información recogida de programas de asignaturas de universidades nacionales e internacionales relacionadas con la disciplina de Arquitectura del Software, así como la propia experiencia del candidato impartiendo esta asignatura desde el curso 2012–2013, ha servido para acabar de perfilar los contenidos que se pueden impartir en esta asignatura.

Es necesario tener en cuenta también criterios relativos a número de horas de que dispone la asignatura para ajustar el contenido al tiempo disponible. El análisis de los conocimientos de partida de los alumnos, adquiridos en otras materias del grado, es también un aspecto a tener en cuenta para ajustar el programa, evitando lagunas que puedan perjudicar el aprendizaje o solapamientos innecesarios.

El establecimiento del nivel de detalle relativo a los contenidos concretos es fruto del estudio de las principales fuentes bibliográficas, junto con la experiencia adquirida impartiendo esta asignatura durante 8 cursos. Se ha seleccionado los aspectos que mejor se adaptan a la asignatura en el contexto del grado en el que se imparte. Los contenidos propuestos se distribuyen en las siguientes unidades didácticas:

- **Conceptos de Arquitectura de Software**
 1. Definiciones de Arquitectura de Software
 2. Comunicando la Arquitectura del Software
 3. Papel del Arquitecto del Software y *Stakeholders*
 4. Atributos de calidad
 5. Creación y validación de Arquitecturas
 6. Técnicas de arquitectura del software: Tácticas, estilos y patrones arquitectónicos
- **Taxonomías de Arquitectura del Software**
 7. Construcción, Gestión de configuraciones

8. Modularidad
9. Comportamiento: monolito
10. Comportamiento: sistemas distribuidos y Big data
11. Disposición
12. Arquitectura del software y entorno empresarial

A continuación se desglosa con un poco más de detalle los aspectos que se cubren en las diferentes unidades didácticas con una breve indicación de las fuentes bibliográficas que se consideran recomendables. Dado que el candidato ya ha impartido esta asignatura, puede consultarse las presentaciones e incluso vídeos del curso 2020-21 en la página web del curso¹.

17.2.1 Unidad 1. Definiciones de Arquitectura del Software

En esta unidad se define qué es la Arquitectura del Software y se presenta una breve evolución de los principales hitos en Arquitectura del Software. También se indica cuál sería la diferencia entre Arquitectura y Diseño del software, así como la relación con otras disciplinas de arquitectura como puede ser la arquitectura empresarial, la arquitectura de sistemas, la arquitectura de la información, etc. Se comentan los beneficios y los retos de la Arquitectura del Software, así como la relación con las metodologías de desarrollo ágil, y las 2 principales leyes de la Arquitectura del Software de acuerdo con [65]:

- Todo en arquitectura del software es una solución de compromiso
- El porqué es más importante que el cómo.

Recomendaciones bibliográficas: Un buen libro de referencia del material incluido en esta lección es: [65]. El reciente libro de G. Hohpe [45] ofrece una visión actualizada del rol del arquitecto de software que puede ser muy útil.

17.2.2 Unidad 2. Comunicando la Arquitectura del Software

En esta sección se hablará de cómo comunicar una arquitectura del software, mediante la creación de documentación o mediante la realización de presentaciones con los diagramas habituales. Se describirá que el objetivo fundamental de una buena documentación es *comunicar* una arquitectura del software para las diferentes personas interesadas, se introducirá el concepto de vistas y puntos de vistas, y se hablará del rol de la documentación en proyectos con metodologías ágiles.

En la lección se dedica un apartado especial a describir diferentes herramientas para la construcción y generación de diagramas de arquitectura, como pueden ser:

- Bocetos dibujados a mano alzada
- Herramientas de dibujo tipo Draw.io² o Powerpoint.
- Herramientas de generación de diagramas a partir de descripciones textuales, tipo PlantUML³.
- Herramientas de modelado como Enterprise Architect⁴, Visual Paradigm⁵ o Modelio⁶.
- Herramientas de generación de modelos mediante ingeniería inversa como puede ser Structure101⁷ ó Sonargraph⁸
- Lenguajes de descripción de arquitecturas que definen formalmente la arquitectura del sistema. Ejemplos podrían ser: xArch/xADL⁹ ACME¹⁰ ó AADL¹¹

En la lección se hablará también de algunos de los diferentes enfoques de documentación existentes como el enfoque 4+1 de Kruchten [58], el enfoque vistas y más (*Views and beyond*) de Clements et al [18] o el modelo C4 de Simon Brown¹².

Se describirá con más detalle las plantillas Arc42¹³ que son las que se utilizan para la documentación

¹<https://arquisoft.github.io/course2021.html#lectureNotes>

²<https://draw.io/>

³<https://plantuml.com/>

⁴<https://sparxsystems.us/>

⁵<https://www.visual-paradigm.com/>

⁶<https://www.modelio.org/>

⁷<https://structure101.com/>

⁸<https://www.hello2morrow.com/products/sonargraph>

⁹<http://isr.uci.edu/projects/xarchuci/>

¹⁰<http://www.cs.cmu.edu/~able/>

¹¹<http://www.aadl.info/>

¹²<https://c4model.com/>

¹³<https://arc42.org/>

del prototipo solicitado en la práctica. El principal motivo de elegir este modelo es que tiene plantillas para diferentes sistemas (Markdown, Asciidoc, Word, etc.) y en varios idiomas (inglés, español, alemán y ruso), además de que son unas plantillas que cubren perfectamente los diferentes aspectos de una descripción de arquitectura.

Finalmente, se recordará la importancia de mantener sincronizada la documentación con la implementación, planteando el uso del estilo de codificación arquitectónicamente evidente propuesto en [28].

Recomendaciones bibliográficas: Una de las referencias más relevantes sobre documentación de arquitecturas del software es [18], aunque el segundo volumen del libro de Simon Brown [12] también tiene una revisión más actualizada de técnicas de documentación de arquitectura. El libro *Arc42 by example* [83], junto con la propia página Web de Arc42 ofrece una buena justificación sobre las plantillas Arc42 que se utilizan en la asignatura.

17.2.3 Unidad 3. Papel del arquitecto del software y Stakeholders

En esta unidad se discute el rol de Arquitecto de Software en los equipos de desarrollo de software.

Se insiste en el término *rol* para separarlo de un cargo personal porque se considera que en los equipos modernos de desarrollo de software no siempre es la misma persona la que actúa como arquitecto, sino que es un rol que en ocasiones incluso va rotando entre los miembros del equipo.

Se presentan las expectativas que se espera que cumpla un arquitecto de software, como son: toma de decisiones arquitectónicas, análisis continuo de la evolución de la arquitectura, conocimiento de tendencias tecnológicas e industriales, asegurar cumplimiento de decisiones arquitectónicas tomadas, conocimiento del dominio del problema, amplitud técnica para valorar diferentes alternativas, habilidades interpersonales para la defensa de decisiones tomadas y la negociación con las diferentes personas involucradas, comprensión de la política empresarial.

Se comentan aspectos relativos al **trabajo en equipo** como el factor del autobús, la pérdida por proceso, la ley de Brook, las situaciones de ignorancia colectiva o de difusión de responsabilidad.

Se describen también diferentes topologías de equipos así como la Ley de Conway y la maniobra inversa de Conway.

La segunda parte de esta unidad didáctica se dedica a definir qué son los *stakeholders* ó personas interesadas, así como a técnicas para identificar stakeholders como el mapa de stakeholders.

Recomendaciones bibliográficas: Los aspectos de trabajo en equipo y el factor del autobús están muy bien descritos en [86]. El libro [81] también presenta una buena explicación de la ley de Conway y la maniobra inversa de Conway. El libro [65] contiene buenas explicaciones de la pérdida por proceso y la ley de Brook.

17.2.4 Unidad 4. Atributos de calidad

Esta unidad didáctica se centrará en los atributos de calidad, que en ciertos contextos se conocen como requisitos no funcionales o características arquitectónicas.

La unidad comienza identificando los diferentes tipos de requisitos, identificando los requisitos funcionales, los no funcionales y las restricciones. En la lección se insiste en varios aspectos importantes:

- Que los atributos de calidad no son *booleanos*, sino que se identifican mediante un cierto grado de cumplimiento
- Que los atributos de calidad no son independientes entre sí, sino que pueden tirar del sistema en líneas opuestas y hay que encontrar una solución de compromiso entre las diferentes posibilidades.

En la lección se define qué son los *escenarios de calidad* y los principales componentes de un escenario de calidad: *estímulo, respuesta, artefacto, medida de respuesta* y entorno. Se hablará también de tipos de escenarios de calidad y de cómo priorizar los diferentes escenarios de calidad a partir de la percepción de importancia para el éxito por parte del cliente, y la percepción de dificultad para alcanzarlo, por parte del arquitecto.

En la lección se mencionará la lista de atributos de calidad ISO 25010 *Software Quality Model* [48] y se explicarán técnicas para identificar atributos de calidad como los talleres de atributos de calidad y los árboles de atributos de calidad.

La segunda parte de la unidad didáctica se enfoca en cómo medir el cumplimiento de un atributo de calidad, se describen diferentes tipos de métricas como:

- Métricas operacionales como por ejemplo la disponibilidad, calculada como tiempo medio entre fallos (MTBF) dividido por el tiempo medio de recuperación (MTTR),
- Métricas estructurales (por ejemplo la complejidad ciclomática de McCabe),

- Métricas de procesos (estabilidad, desplegabilidad, ...)

La unidad se finalizará haciendo mención a las arquitecturas evolutivas, que manejan el concepto de función de encaje (*fitness function*) para resaltar la evolución de la arquitectura a lo largo del tiempo.

Recomendaciones bibliográficas: Un buen libro que describe los atributos de calidad es [6]. El libro [65] utiliza el concepto de características de arquitectura para referirse a los atributos de calidad e incluye también una descripción de los diferentes tipos de métricas. El libro [31] introduce el concepto de arquitecturas evolutivas y de función de encaje.

17.2.5 Unidad 5. Creación y validación de arquitecturas

Esta unidad didáctica plantea la temática de cómo alcanzar la arquitectura del software. Se describe cómo integrar el proceso de creación de la arquitectura del software con el resto del proyecto, mencionando la importancia de buscar un equilibrio entre comenzar a desarrollar sin haber dedicado suficiente tiempo al planteamiento de la arquitectura, lo cual puede conllevar un mayor tiempo de reconstrucción de lo implementado, respecto al otro extremo que sería estar demasiado tiempo planteando la arquitectura, sin dejar espacio para la implementación.

Se hablará de la técnica ADD (Attribute Driven Design), presentando la variante ADD 3.0 [47].

En esta unidad didáctica se dedica un apartado especial a las decisiones arquitectónicas y al registro de decisiones arquitectónicas, un aspecto muy importante para comprender la naturaleza dinámica de la arquitectura del software.

Se realiza también un repaso a las posibles incidencias arquitectónicas a tener en cuenta cuando se lleva a cabo una arquitectura de software, como son:

- Riesgos. Cosas malas que podrían ocurrir pero que todavía no han ocurrido, pueden ser mitigados o aceptados. Se menciona cómo crear una tabla de valoración de riesgos o los talleres de *risk storming*.
- Desconocidos. Aspectos de los que no se tiene suficiente información y que puede ser importante identificar para convertirlos en riesgos.
- Problemas. Cosas malas que ya han pasado en un determinado desarrollo o en la implantación de una determinada arquitectura del software. Estos problemas pueden ser arreglados o asumirse.
- Deuda técnica. Decisiones equivocadas que se toman consciente o inconscientemente. Dada la reciente importancia que está adquiriendo la deuda técnica en la creación de software, en la unidad se entrará en más detalle sobre tipos de deuda técnica y otros aspectos relevantes.
- Diferencias de comprensión entre lo que piensan los *stakeholders* sobre cómo es la arquitectura. Las arquitecturas pueden evolucionar rápidamente de forma que las diferencias aparecen de forma repentina e inesperada, por lo que se insiste en la importancia de que los arquitectos del software establezcan buenos canales de comunicación y formación con los *stakeholders*.
- Erosión de la arquitectura. Es la diferencia entre la arquitectura diseñada y el sistema implementado. En general, el sistema implementado casi nunca es igual al sistema diseñado. Se insistirá en lo importante que es establecer mecanismos que permitan observar la evolución de la implementación o incluso participar en la misma. También se explicará el concepto de *código arquitectónicamente evidente*, que impulsa la creación de código que refleje las decisiones arquitectónicas.
- La propia *evolución del contexto* que se produce cuando aspectos clave del contexto cambian después de haber tomado una decisión. Se insiste en la importancia de disponer de un registro de decisiones arquitectónicas dinámico y accesible para que puede ser compartido y consultado por los diferentes *stakeholders* facilitando la comprensión de las motivaciones que dan lugar a las diferentes decisiones tomadas en la arquitectura, las cuales podrán ser revisadas o modificadas a lo largo del proyecto.

En la última parte de la unidad, se describen métodos de evaluación de arquitecturas, como son ATAM (*Architecture Trade-off Analysis Method*) o CBAM (*Cost Benefit Analysis Method*).

Recomendaciones bibliográficas: La variante ADD 3.0 se introduce en [47]. El libro [65] ofrece una buena introducción al registro de decisiones arquitectónicas. Los conceptos de *código arquitectónicamente evidente* y *architectural drift* son introducidos en el libro [28]. Los métodos ATAM y CBAM son descritos en [6].

17.2.6 Unidad 6. Técnicas de Arquitectura del Software

En esta Unidad se introducen los conceptos de tácticas, estilos y patrones arquitectónicos. Se indica la importancia de que el arquitecto esté al tanto de las nuevas tendencias en la disciplina así como en la experiencia acumulada por otros arquitectos. Este conocimiento se captura en tácticas, estilos y patrones:

- Una táctica es una técnica de diseño que pretende alcanzar una respuesta a algunos atributos de

calidad. Las tácticas se enfocan por tanto en la respuesta a un atributo de calidad, pudiendo chocar con otros atributos de calidad. Las tácticas suelen definirse como posibles respuestas a un estímulo y suelen presentarse como un catálogo de posibles tácticas ante determinados atributos de calidad. La rápida evolución tecnológica hace que algunos de estos catálogos se queden obsoletos con cierta rapidez.

- *Estilo arquitectónico.* Los estilos definen la forma general de un sistema. Las definiciones de estilos arquitectónicos suelen incluir los elementos principales, las relaciones entre estos elementos, las restricciones que se imponen en su combinación y una lista de atributos (ventajas, inconvenientes,...) asociados al estilo. En general no hay estilos puros y se mencionará que el estilo final de un sistema complejo suele estar formado de combinaciones de varios estilos.
- *Patrón arquitectónico.* Un patrón es una solución general y reutilizable a algún problema recurrente que aparece en un contexto. Se mencionarán los diferentes tipos de patrones arquitectónicos: estructurales, de comportamiento, de despliegue, etc. También se destacará la diferencia entre estilo y patrón arquitectónico, así como la diferencia entre un catálogo de patrones y un lenguaje de patrones.

En la unidad didáctica también se discutirá la importancia de decidir qué aspectos es necesario construir y qué aspectos pueden reutilizarse, para describir el concepto de *arquitecturas de referencia*, *arquitecturas de software en dominios específicos*, familias o pilas tecnológicas (*technology stacks*), componentes desarrollados externamente como COTS (*Comercial Off-the-shelf*) y FOSS (*Free Open Source Software*), marcos de aplicaciones y plataformas.

Recomendaciones bibliográficas: El libro [6] ofrece un catálogo de tácticas arquitectónicas. Existen numerosos libros que recogen patrones arquitectónicos, como [18, 28]

En [18] se realiza una taxonomía de los patrones arquitectónicos en: modularidad, componentes y conectores, disposición, etc.

17.2.7 Unidad 7. Taxonomías: Construcción, mantenimiento y gestión de configuraciones

En esta unidad didáctica se describen aspectos sobre la construcción, e implementación del software, así como el mantenimiento del mismo. Se introduce la noción de software como un elemento vivo que evoluciona con el tiempo y requiere constante mantenimiento. Se define el concepto de *gestión de configuraciones* y los aspectos relacionados de línea base (*baseline*), ítems de configuración, control y auditoría de configuraciones, sistemas de control de versiones, gestión y automatización de la construcción, técnicas y herramientas de trabajo en equipo y colaboración, seguimiento de incidencias, etc. En la unidad se comienza realizando un repaso a algunas metodologías de construcción de software, describiendo con más detalle aspectos de la metodología ágil que se utilizarán en las prácticas de laboratorio como la utilización de desarrollo basado en pruebas ó TDD (*Test-driven development*), integración continua, propiedad colectiva del código, etc.

Se repasarán los sistemas de control de versiones, prestando especial atención al sistema *git* que se utiliza en las prácticas. También se definirá qué es una versión y la diferencia entre versión y *release*, así como diferentes convenciones de nombrado de versiones, como el versionado semántico. Se describe el concepto de entrega continua (*continuous delivery*) y su relación con la integración continua, así como el concepto de *devops*.

Se realizará un repaso de herramientas de construcción de software, como los lenguajes de programación, compiladores, herramientas de análisis estático y dinámico de código, profilers, detectores de cobertura de código y fragmentadores de código (*program slicers*). Se hablará de herramientas de gestión de dependencias y del grafo de dependencias, así como del problema de incluir dependencias cíclicas y la métrica CCD (Cummulative component dependency).

En la unidad también se describirán las herramientas de automatización de la construcción como Maven, npm, etc.

Se recomienda que esta unidad didáctica comience a impartirse lo antes posible para que sus conocimientos puedan ser llevados a la práctica durante la realización del prototipo.

Recomendaciones bibliográficas: Un libro que cubre los aspectos tratados en esta unidad didáctica es [49].

17.2.8 Unidad 8. Taxonomías: Modularidad

Esta unidad didáctica presenta aspectos para la descomposición de un proyecto de software en módulos en tiempo de construcción. Se distinguen los módulos como elementos que tienen relevancia en tiempo de construcción de los componentes, que tienen relevancia en tiempo de ejecución. Se describirán varios estilos

de modularidad, comenzando con el conocido como *big ball of mud* [30], que aunque puede considerarse un antipatrón, realmente ocurre en muchas más ocasiones de las deseables. En la lección se repasan las definiciones básicas de modularidad: interfaz/cuerpo de un módulo, ocultación de información, etc. así como algunas recomendaciones y principios sobre modularidad como:

- Principios SOLID, propuestos por Robert Martin [60]. Estos principios se describirán brevemente puesto que también son introducidos en la asignatura *Diseño de software*.
- Cohesividad. Se define qué es la cohesividad así como la recomendación de incrementar la cohesividad de los módulos y se presenta la métrica LCOM (*Lack of cohesion of methods*) de Chidamber y Kemerer [17].
- También se presentan los principios REP (*Reuse/Release Equivalence Principle*), CCP (*Common Closure Principle*) y CRP (*Common Reuse Principle*) de Robert Martin [60].
- Acoplamiento. Se presenta una definición de acoplamiento como el grado de interdependencia entre dos módulos de software, y se presentan los principios de acoplamiento: ADP (*Ayclic Dependencies Principle*), SDP (*Stable Dependencies Principle*) y SAP (*Stable Abstractions Principle*) propuestos por Robert Martin [60].
- Conocimiento. Esta métrica permite aglutinar en un único valor los conceptos de cohesividad y acoplamiento. Se define conocimiento como aquellas cosas que nacen y crecen juntas, de forma que un cambio en una de ellas requiere cambio en las otras para que el sistema siga siendo correcto. El conocimiento conlleva 3 propiedades: grado, localidad y fuerza, que indican las direcciones sobre las que conviene refactorizar el código.
- El principio de robustez ó ley de Postel fue enunciado a principios de los años 80 por Jon Postel para indicar aspectos que debería tener el interfaz de un módulo, que debe ser liberal en lo que acepta de terceras partes, pero conservador en lo que envía. Aunque este principio se concibió para describir el protocolo TCP/IP, la idea puede extrapolarse para el diseño de cualquier API.
- Ley de Demeter o principio de menor conocimiento se describe como un aspecto a tener en cuenta para el diseño de módulos con menor acoplamiento al incidir sobre aquellos casos en los que se invocan métodos o se acceden a atributos de módulos de terceras partes.
- Interfaces fluidos. Es una recomendación para la definición de interfaces públicos que mejorar la legibilidad del código de los clientes que los utilizan facilitando el encadenamiento de invocaciones y el autocompletado por parte de los editores.
- Otras recomendaciones relacionadas con la modularidad como los patrones GRASP (*General Responsibility Assignment Software Patterns*), o los lemas YAGNI (*You ain't gonna need it*), DRY (*Don't repeat yourself*) y KISS (*Keep it simple stupid*).

En la unidad se describirán estilos de descomposición modular, como son:

- Arquitectura en capas. Es uno de los estilos de descomposición clásicos. Se describirá en qué consiste, cuáles son las principales restricciones del mismo, así como las consecuencias de su uso y las variantes del mismo: descomposición en capas estricta o cerrada vs laxa o abierta. También se distinguirán las diferencias entre capas y módulos, y entre capas (*layers*) y niveles (*tiers*).
- Aspectos. La programación orientada a aspectos adquirió una gran popularidad a principios de los años 2000 al promover una mayor modularidad de los sistemas mediante la separación de incumbencias. Un aspecto es un módulo que implementa una característica transversal (*crosscutting concern*), como puede ser un aspecto de *logging*, seguridad, monitorización, etc. que afecta a una gran cantidad de módulos de un sistema. El objetivo es aglutinar los aspectos en módulos concretos, facilitando que desarrolladores especializados implementen dichos aspectos. Una vez implementados los aspectos, su código será insertado en la aplicación final de forma automática mediante el compilador de aspectos. Se describirá brevemente esta técnica, así como una valoración de las consecuencias de su uso.
- Basados en dominio. Existen varias descomposiciones modulares que promueven dar prioridad al dominio para la descomposición modular, en lugar de las capacidades técnicas. Se describirán las siguientes propuestas:
 - Domain-driven design. Se trata de una combinación de estilo arquitectónico y metodología de desarrollo propuesta por Eric Evans [27], que ha adquirido una gran influencia en desarrollo de software. Promueve la creación de un lenguaje ubicuo, involucrando a los expertos del dominio junto con los desarrolladores y estableciendo límites contextuales (*boundary contexts*). Esta propuesta define conceptos como agregados, entidades, objetos valor, repositorios, factorías y servicios que forman un vocabulario muy útil para el diseño de aplicaciones.
 - Estilo hexagonal y arquitectura limpia. El estilo hexagonal fue propuesto por Alistair Cock-

burn¹⁴ como una propuesta que promueve centrar el sistema en un modelo de dominio, independiente de otros aspectos considerados externos como infraestructuras o marcos de aplicaciones. En la misma línea, Robert Martin propuso el concepto de arquitectura limpia [60] en el que el sistema define un modelo de dominio central mediante entidades, seguido de unos casos de uso que dependen de dichas entidades, y dejando que las bases de datos, los dispositivos, interfaces externos, etc. estén fuera del núcleo del sistema. Las dependencias de esta forma aparecen mediante adaptadores de dichos elementos externos que dependerán del núcleo del sistema, en lugar de hacerlo a la inversa.

- Modelos centrados en datos. Consiste en plantear aplicaciones como sistemas sencillos a partir de una base de datos central utilizando las operaciones CRUD (Create-Retrieve-Update-Delete) básicas. Estas aplicaciones incluyen en ocasiones la posibilidad de generación automática (*scaffolding*) y ofrecen como principal ventaja la velocidad de desarrollo *time-to-market*. En la unidad se describe brevemente este estilo debido a que en múltiples ocasiones puede ser una solución eficaz
- Naked objects. Se trata de una propuesta que promueve la generación automática de aplicaciones a partir de un modelo de objetos [69]. Aunque la idea ha quedado un tanto obsoleta, se describirá brevemente debido a su relevancia y la posibilidad de que surjan ideas similares en el futuro.

Un estilo que podría considerarse en este apartado es el estilo de microservicios que realmente también promueve la descomposición basada en el dominio. No obstante, dado que dicho estilo también encaja como una evolución de los estilos basados en servicios, se incluye dentro de los mismos porque se considera que los estudiantes podrán comprender mejor los motivos de su aparición en dicho contexto.

Recomendaciones bibliográficas: Los principios SOLID, así como los aspectos de cohesividad y acoplamiento están bien explicados en el libro *Clean architecture* de Robert Martin [60]. El principio de Conocimiento ha sido incluido en [65].

17.2.9 Unidad 9. Taxonomías: Comportamiento. Sistemas básicos y monolitos

Esta unidad didáctica se dedicará a presentar los estilos arquitectónicos clasificados en función de los componentes y los conectores del software en tiempo de ejecución, es decir observando el comportamiento del software.

El tema se divide en dos partes: sistemas básicos y monolitos, tratados en esta unidad y los sistemas distribuidos y big data, tratados en la siguiente.

Se presentarán los siguientes estilos:

- Flujo de datos
 - Batch ó secuencial
 - Pipes & Filters
 - Variante: Pipes & Filters con interfaz uniforme
- Organización del trabajo
 - Master-slave
- Sistemas interactivos
 - Modelo vista-controlador (MVC)
 - Variantes: PAC (Presentación-abstracción-control), DCI (Datos-contexto-interacción)
- Repositorio
 - Datos compartidos
 - Blackboard
 - Basados en reglas
- Invocación
 - Call-return
 - Cliente-servidor con las variantes:
 - Cliente-servidor sin estado
 - Cliente-servidor con servidor replicado
 - Cliente-servidor con caché
 - Arquitecturas basadas en eventos
 - Publish-subscribe
 - Modelos de actores

¹⁴<http://alistair.cockburn.us/Hexagonal+architecture>

- CQRS (*Command-query responsibility segregation*)
 - *Event sourcing*
- Sistemas adaptables
 - *Plugins*
 - *Microkernel*
 - Reflection
 - Intérpretes y Lenguajes de dominio específico (DSLs)
 - Código móvil
 - ◊ Código bajo demanda
 - ◊ Evaluación remota
 - ◊ Agentes móviles

Recomendaciones bibliográficas: La clasificación inicial de los estilos arquitectónicos basado en componentes y conectores se ha tomado de [6]. Otros libros que incluyen catálogos de estilos y patrones arquitectónicos, y que pueden ser lecturas recomendadas son [28, 42].

17.2.10 Unidad 10. Taxonomías: Comportamiento. Sistemas distribuidos y big data

Continuando con la unidad anterior sobre estilos arquitectónicos basados en comportamiento, esta unidad realiza una clasificación de los estilos en función de los componentes y conectores para sistemas distribuidos y *big data*.

Se presentarán los siguientes estilos:

- Estilos de integración. Se describirán los estilos tradicionales de integración de aplicaciones como son: Transferencia de ficheros, Base de datos compartida, Invocación procedimiento remoto, Mensajería, y mediante log de eventos.
- Topologías de integración: se mencionarán las topologías Hub & spoke y Bus de mensajería, que han tenido una gran influencia en las arquitecturas del software basadas en servicios.
- Patrón broker. Se trata de un patrón que tuvo una gran relevancia histórica de la mano de tecnologías como CORBA [87].
- Peer-to-peer. Las configuraciones Peer-to-peer adquirieron una gran importancia de la mano de proyectos como Napster, BitTorrent o Gnutella. En la actualidad, siguen siendo relevantes con la aparición de nuevos proyectos como Blockchain. Aunque por motivos de tiempo, no se podrá profundizar en este tipo de arquitectura en las clases de teoría, sí se podría considerar la inclusión de algunos temas de seminario o en las clases de laboratorio aspectos relacionados con estas tecnologías.
- Arquitecturas Orientadas a Servicios. Se definirá el concepto de arquitectura orientada a servicios de forma general y posteriormente se describirán los modelos:
 - WS-* basado en las tecnologías propuestas por el W3C como eran SOAP, WSDL y UDDI, así como otras propuestas y especificaciones sobre diferentes temas como seguridad, transacciones, orquestación, etc. Aunque actualmente dichas tecnologías se han ido abandonando, todavía existen proyectos y empresas que hacen uso de las mismas y se considera interesante para la formación de los estudiantes que conozcan al menos la existencia de estas propuestas y quizás incluso los motivos por los que ya no se utilizan tanto.
 - REST (Representational State Transfer). Esta propuesta de Roy T. Fielding [29] se inspiraba en la propia arquitectura de la Web y ha ido reemplazando a las propuestas WS-*. Se explicará en más detalle qué son los servicios REST y el modelo de desarrollo basado en recursos. Un aspecto interesante es que este estilo podría considerarse una combinación de varios estilos como arquitectura en capas, cliente servidor sin estado, con caché y con servidor replicado, pipes and filters con interfaz uniforme y código bajo demanda.
 - Arquitectura basada en servicios. Aunque este término no ha adquirido popularidad como tal, sí que ha sido utilizado mayoritariamente en múltiples proyectos. Fue incluido en [65] para definir un estilo de aplicaciones pragmático que se utiliza para la creación de aplicaciones empresariales. Se trata de un estilo en el que se definen varios servicios que acceden a una base de datos común y que son invocados generalmente desde un mismo interfaz de usuario¹⁵.
- Microservicios. Los microservicios han adquirido una gran popularidad como una solución útil para afrontar la maniobra inversa de Conway, promoviendo la descomposición de los equipos siguiendo la

¹⁵En la práctica, las implementaciones prácticas de este estilo pueden tener múltiples variaciones, como varios interfaces de usuario, más de una base de datos o servicios compuestos a partir de otros servicios

descomposición modular del sistema. Se trata de descomponer las aplicaciones en microservicios, de forma que cada microservicio es una unidad de construcción y despliegue independiente, generalmente con su propio sistema de persistencia. De esta forma los microservicios promueven un mayor desacoplamiento y cohesividad al estar enfocados a la realización de unas pocas tareas de forma eficiente. Dada la reciente popularidad de los microservicios, este estilo será descrito con más detalle y se profundizará en analizar sus características y consecuencias con el fin de que los estudiantes puedan adquirir una mejor visión de las ventajas de los mismos, pero también de los problemas que su utilización puede acarrear como el incremento en complejidad de desarrollo, la posible aparición de monolitos distribuidos o el deterioro estructural.

- Serverless, también conocido como FaaS (*Function as a Service*) o BaaS (*Backend as a Service*), consiste en promover la utilización de funciones que se ejecutan en sistemas de terceras partes como puede ser AWS Lambda, Google Cloud Functions ó Ms Azure Functions, etc. de forma que el desarrollador delega la administración de los servicios de esas terceras partes a la empresa correspondiente (Amazon, Google, Microsoft, ...), olvidándose de alguna forma de la existencia de un servidor. Estas empresas suelen ofrecer contratos que incluyen escalabilidad automática y suministro basado en la carga del servicio, con un modelo de costes basado en la utilización del servicio que puede ser interesante para algunas start-ups o en situaciones que se requiere acceder a grandes recursos computacionales a un coste razonable. Este término también está adquiriendo una cierta popularidad dado que algunas de estas grandes empresas también promueven su uso. Por ese motivo se describirán también con más detalle las posibles ventajas e inconvenientes de su uso.

La unidad didáctica incluye además un tratamiento de los estilos arquitectónicos relacionados con la escalabilidad y el tratamiento de grandes cantidades de datos (*big data*). Entre ellos, se menciona:

- *MapReduce* es un algoritmo de computación distribuida desarrollado por Google y publicado en el año 2004 [22]. Tras su publicación, se realizaron numerosas implementaciones, algunas de código abierto como Hadoop¹⁶ que tendrían una gran repercusión en la adopción de soluciones de procesamiento de grandes cantidades de datos. *MapReduce* podría considerarse como una combinación de varios patrones arquitectónicos como *batch* y *master-slave* adaptados para su computación en granjas de computadores.
- Arquitectura Lambda. Fue propuesta por Nathan Marz en un post de su blog en el año 2011¹⁷ con el fin de combinar las capacidades de computación de grandes cantidades de datos de *MapReduce* con la necesidad de computación en tiempo real. En la propuesta arquitectónica también se puede observar la combinación de patrones arquitectónicos como la utilización de 3 capas (*batch*, *serving* y *speed*).
- Arquitectura Kappa es una propuesta realizada por Jay Kreps en 2014 como alternativa a la arquitectura lambda¹⁸ que propone la utilización de un único sistema de procesamiento de eventos basado en el uso de sistemas de almacenamiento de *logs* distribuidos como Apache Kafka. Esta arquitectura también puede considerarse una combinación del patrón *Event Sourcing* y *CQRS*.

Recomendaciones bibliográficas: Dada la reciente popularidad de los microservicios, se han publicado varios libros sobre el tema, entre los que destaca especialmente [66, 67]. La arquitectura lambda se describe en detalle en Nathan Marz sobre *big data* [61]. A su vez, Jay Kreps escribió un sencillo libro en el que describe y justifica la utilización de *logs* distribuidos [57].

17.2.11 Unidad 11. Taxonomías: Disposición

En esta unidad didáctica se cubrirá el tema de la disposición (*allocation*) del software. La disposición se entiende en este contexto como la relación del software con su entorno, que puede ser tanto cuando el software ya está operativo y en producción (cómo desplegar el software), como cuando se está desarrollando (equipos y distribución de trabajo). En la unidad se tratarán los aspectos de empaquetamiento, distribución y despliegue del software, así como los patrones y tácticas para mantener el software en producción.

Se describirán diferentes opciones de ejecución del software: *on premises*, *cloud computing*, *edge computing* y *fog computing* y se describirán técnicas de empaquetamiento y envío del software, describiendo la diferencia entre entornos de desarrollo, pruebas y producción. También se hablará de los canales de distribución de software, así como de los mercados de aplicaciones.

Respecto a los entornos de ejecución, se describirá brevemente la diferencia entre máquinas físicas,

¹⁶<https://hadoop.apache.org/>

¹⁷<http://nathanmarz.com/blog/how-to-beat-the-cap-theorem.html>

¹⁸<https://www.oreilly.com/radar/questioning-the-lambda-architecture/>

máquinas virtuales y contenedores. Respecto a los contenedores, se entrará en más detalle en su descripción debido a su creciente relevancia para la definición de entornos de software reproducibles. Además, la propia arquitectura de los sistemas de contenedores como Docker aporta varias lecciones interesantes desde el punto de vista de arquitectura del software.

En cuanto al despliegue, se definirá el concepto de canal de despliegue ó *deployment pipeline*, que permite automatizar el proceso de construcción, pruebas, despliegue y publicación del software con el fin de alcanzar sistemas de despliegue y entrega continua (*continuous delivery*). Se describen en este contexto la lista de 12 factores¹⁹ a tener en cuenta para alcanzar estos objetivos.

Un aspecto cada vez más relevante es el software en producción, y las tácticas, estilos y patrones que permiten una mayor resiliencia del software para que pueda superar las diferentes incidencias que puedan ocurrir. Se describirán atributos de calidad que pueden identificarse en este contexto como es la resiliencia, configurabilidad, observabilidad, disponibilidad, etc. Para alcanzar estos atributos de calidad es necesario incorporar técnicas de configuración, *logging*, monitorización, gestión de datos, etc. Se caracterizarán tipos de problemas de los sistemas, como son las faltas (*faults*), errores y fallos (*failures*), indicando la importancia de que los sistemas continúen funcionando incluso en presencia de faltas. En la unidad se describirán patrones para software en producción como son²⁰: balanceo de carga, *timeout*, *circuit breaker*, *Bulkheads*, *Steady state*, *Fail fast*, *handshaking*, *Backpressure*, *Governor* y *test harnesses*, mencionando lo que se ha denominado como teoría del caos. También se describen técnicas de prueba del software en producción y despliegue progresivo, como *Canary releases*, *feature toggles*, pruebas A/B, *Multi-armed bandits* y pruebas de carga. Finalmente, se hablará de qué acciones realizar tras los incidentes y posibles caídas, como los informes *post-mortem*.

Recomendaciones bibliográficas: Uno de los libros clásicos sobre cómo mantener el software en producción es *Release it!* [68], cuya segunda edición contiene una revisión actualizada de dichos temas y mantiene su vigencia. El libro *Site reliability engineering* [7] también es una referencia muy recomendable.

17.2.12 Unidad 12. Arquitectura del software y Entorno empresarial

Esta unidad didáctica se centra en la relación de la arquitectura del software con el entorno empresarial. Se comenzará describiendo el rol del arquitecto del software en la empresa. Para ello, se presentará una breve clasificación sobre diferentes tipos de empresas y sobre varios tipos de arquitectos: arquitectos empresariales, de soluciones, de negocio, etc. Se describirá brevemente qué es la arquitectura empresarial, mencionando el marco Zachman y el marco TOGAF (*The Open Group Architecture Framework*). También se hablará del ciclo de sobreexpectativas de Gartner con el fin de fomentar el pensamiento crítico de los estudiantes ante los planteamientos que van surgiendo en el contexto de tecnologías empresariales.

Se presentará una taxonomía de software empresarial como:

- Software ERP (*Enterprise Resource Planning*) en el que se hablará brevemente de sistemas de gestión empresarial como SAP
- Software CRM (*Customer Relationship Management*) que gestiona la relación entre la empresa y el cliente.
- Software SCM (*Supply Chain Management*) que gestiona temas de logística, mercancías, proveedores, etc.
- WMS (*Warehouse Management Systems*), sistemas de gestión de almacenes, que cobran cierta importancia local en el ámbito asturiano en el que está instalada la empresa que desarrolla el software Mecalux EasyWMS.
- Software PLM (*Product Lifecycle Management*), que gestiona el ciclo de vida de los productos.
- Software ECM (*Enterprise Content Management*), que englobaría aspectos de gestión de documentos de la empresa y en general, de gestión de conocimiento empresarial.
- Software BPM (*Business Process Management*), que facilita la gestión de procesos de negocios, dando soporte a la toma de decisiones y a lo que se conoce como *business intelligence*. En este apartado se hablará de BPMN (*Business Process Modeling and Notation*), una notación visual orientada a personal de negocio de uso bastante extendido que puede ser útil también en la documentación de la arquitectura del software.
- Software B2B (*Business to Business*) que facilita el comercio electrónico entre empresas.

¹⁹<https://12factor.net/>

²⁰Se incluyen los nombres en inglés en aquellos patrones cuya traducción al español no está suficientemente clara para facilitar una nomenclatura común

Temas	Horas totales	Presencial					No presencial		
		Clase expositiva	Seminario	Práctica de laboratorio	Tutorías grupales	Sesiones de evaluación	Total	Trabajo grupo	Trabajo autónomo
1 (Conceptos)	40	7	2	4	1	1	15	15	10
2 (Taxonomías)	110	14	5	24	1	1	45	45	20
Total	150	21	7	28	2	2	60	60	30

Cuadro 17.1: Reparto de horas de trabajo presencial y no presencial

- EAI (Enterprise Application Integration), software para la integración de aplicaciones empresariales.

En la unidad didáctica se describirán también las líneas de producto software, que pretenden desarrollar productos software que comparten un conjunto de funcionalidades para satisfacer un determinado segmento de mercado reduciendo el esfuerzo de desarrollo y mejorando la productividad mediante la reutilización de software o generación automática de sistemas.

Se incluye un apartado dedicado al software como servicio en la empresa, describiendo la transición de software como producto, a software como servicio, que afecta a la propia arquitectura de dicho software. Se describirán diferentes indicadores de calidad de servicio y qué son los acuerdos a nivel de servicio, así como otros aspectos relativos al gobierno y gestión de servicios, y a la propia evolución del software. Respecto a la evolución del software, se mencionan las leyes de Lehman sobre la evolución del software, como la ley de cambio continuo, o la ley de complejidad creciente. También se describirá qué es la factorización de software, con técnicas para afrontar la deuda técnica y para trabajar con proyectos heredados o *legacy*. Un aspecto relevante que se describirá son las herramientas de análisis de comportamiento del código (*Behavioural Code Analysis*), que permiten detectar puntos calientes ó *hotspots*, que pueden ser buenos candidatos a refactorizar. En general, con esta sección, se pretende transmitir la necesidad de observar el software como un sistema complejo en evolución, que hace muy importante afrontar la propia arquitectura como una arquitectura evolutiva.

Recomendaciones bibliográficas: Un libro que contiene una buena descripción del reciente papel del arquitecto de software en el entorno empresarial es [45]. El análisis de código basado en comportamiento se introduce en el libro [84].

17.3 Actividades de aprendizaje

A continuación se describe la organización de la asignatura, tanto desde el punto de vista de la distribución temporal, como de la coordinación entre las diferentes actividades.

En cuanto al reparto de horas de trabajo, la propuesta se ciñe a las restricciones dadas por la organización docente de la Universidad. La asignatura tiene 6 créditos ECTS, que suponen un total de 150 horas de trabajo, de las cuales 60 horas serán presenciales y 90 horas serán no presenciales.

El reparto de horas totales se representa en la tabla 17.1, en la que se indican las horas para los 2 temas principales de la asignatura, el tema 1 (Conceptos de arquitectura del software) y tema 2 (Taxonomías de arquitectura del software). Las horas presenciales se dividen en 27 horas de clases expositivas, 7 horas de prácticas de aula/seminario y 28 horas de prácticas de laboratorio. Las clases no presenciales se dividen en 60 horas de trabajo en grupo y 30 horas de trabajo autónomo e individual.

Para el primer tema se proponen 15 horas presenciales y 25 horas no presenciales, mientras que al segundo tema se proponen 110 horas presenciales y 90 horas no presenciales.

Cabe destacar que la asignatura dispone de una página Web en la que se puede seguir la evolución de la asignatura en los últimos 3 años: <https://arquisoft.github.io/> (véase figura 17.1)

La asignatura también hace uso del Campus Virtual²¹ proporcionado por la Universidad de Oviedo, que

²¹<http://campusvirtual.uniovi.es/>

Software Architecture Course

Jose Emilio Labra Gayo, University of Oviedo, Spain

§ Context

This course is part of the [Software Engineering degree](#) at the [School of Computer Science Engineering](#). It is also part of the specific training module titled Software engineering. Other related subjects are [Software Process Engineering](#); [Software Design](#); [Software Quality, validation and Verifying](#); [Requirements Engineering](#) and Software Engineering Modelling.

The subject is compulsory and is taught in Spanish and English during the second semester of the third year of the Degree in Software Engineering.

It has 6 ECTS credits, which represent a total of 150 hours, 60 on-campus activities hours and 90 self-study hours.

From the organizational point of view, the course will have two hours of lectures a week (for a total of 21 hours), one hour seminar (for a total of 7 hours), two hours of laboratory practice (for a total 28 hours), and 2 group tutorials hours.

Figura 17.1: Página Web de la asignatura

está basado en la plataforma Moodle²². La figura 17.2 presenta una pantalla capturada de la página web de la asignatura en el campus virtual.

El uso dado para el campus virtual ha ido variando a lo largo de los cursos. Inicialmente era el portal de referencia principal de la asignatura, a través del cual se alojaba material docente y se utilizaban otras herramientas del mismo, como el foro de discusión, el tablón de noticias, el Wiki, etc. Sin embargo, el material alojado en dicho entorno tenía las limitaciones de estar únicamente disponible para los estudiantes durante el tiempo en el que estuviesen matriculados de la misma, y no ofrecía información histórica sobre la misma, por lo que se decidió crear una página Web pública y abierta, específica para la asignatura, dejando el Campus Virtual para material específico del curso: notificaciones a los estudiantes, realización de exámenes y respuestas de los mismos, cuestionarios, etc. Durante el 2013-14, se creó una página en OpenCourseWare²³ para la asignatura. Aunque se mantiene activa, se considera que la página Web actual ofrece una mayor versatilidad y visibilidad para los contenidos.

En el curso 2019-20, durante la pandemia, comenzó a utilizarse el software de Microsoft Teams para impartir clases virtuales. Actualmente se plantea seguir utilizándolo para las comunicaciones asíncronas y en caso de que sea necesario realizar actividades no presenciales.

17.3.1 Clases expositivas

La distribución temporal asignada para las clases de teoría consiste en dos horas seguidas de clase durante 10 semanas, seguidas de una hora de clase la última semana.

La distribución temporal se realiza siempre de forma flexible para amoldarse a las diferentes circunstancias que puedan surgir en cada curso.

La tabla 17.2 presenta la distribución temporal de las clases expositivas del pasado curso 2020-21. Debido a la situación de pandemia, en el curso 2020-21 se grabaron en vídeo las clases y se subieron los

²²<https://moodle.org/>

²³Página en OpenCourseware: <http://ocw.uniovi.es/course/view.php?id=177>

Arquitectura del Software (Grupo Español) (Grado en Ingeniería Informática del Software)

NOTA: Además de la información disponible en el campus virtual, se ha creado la página Web: <http://arquisoft.github.io> que contiene material didáctico que se considera de interés general y se deja abierto.

La Arquitectura del Software forma parte de los procesos de ingeniería de construcción de aplicaciones y es una materia básica para un Ingeniero en Informática.

Esta asignatura se enmarca dentro de la materia de **Ingeniería del Software** y de forma transversal al módulo de **Formación Específica (Ingeniería del Software)**. Dentro de este también están las asignaturas de **Inginería del Proceso Software, Diseño del Software, Calidad, Validación y Verificación del Software, Ingeniería de Requerimientos y Modelado** en **Ingeniería del Software**.

Accesos Directos:

- Cuestionarios
- Encuestas
- Foros
- Glosarios
- Paquetes SCORM
- Recursos
- Tareas
- Wikis del curso

Calendario:

Lun	Mar	Miér	Jue	Vie	Sáb	Dom
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Clave de eventos: >

Actividad reciente: Actividad desde miércoles, 6 de octubre de 2021,

Figura 17.2: Página de la asignatura en el campus virtual

Equipos

- ASW_ES1
 - General
 - ASW_ES1A
 - ASW_ES1B
- ASW_ES2
- ASW_EN2
- ASW_ES3
- ASW_ES6
- ASW_EN3
- ASW_ES5

General

Publicaciones Archivos Bloc de notas de clase 3 más + Equipo Reunirse ...

Clase 12 ha finalizado: 27 s

JOSE EMILIO LABRA GAYO 18:20 En el siguiente enlace podréis acceder a más información sobre la asignatura: <http://arquisoft.github.io>

JOSE EMILIO LABRA GAYO 18:20 La lista de alumnos está en el campus virtual: <https://www.campusvirtual.uniovi.es/course/view.php?id=3835>

JOSE EMILIO LABRA GAYO 18:22 Más información sobre el enunciado de la práctica: <https://arquisoft.github.io/course2021.html>

Nueva conversación

Figura 17.3: Captura de pantalla del equipo de arquitectura del Software en Microsoft Teams

contenidos a un canal Youtube de la propia asignatura. En la tabla se han incluido enlaces a las presentaciones y los vídeos de dichas grabaciones.

Fecha	Título	Presentación	Vídeo (inglés)	Vídeo (español)
03/Feb (1h)	Presentación de la asignatura	Ingles Español		
03/Feb (1h)	Definiciones básicas	Ingles Español	Ingles (I) Ingles (II)	Español (I) Español (II)
10/Feb (1h)	Comunicando la arquitectura	Ingles Español	Ingles	Español
10/Feb (1h)	Conferencia: Open decentralization: Empathy and SOLID for new developers	Ingles	Ingles	Español
17/Feb (1h)	Papel del arquitecto de software y <i>stakeholders</i>	Ingles Español	Ingles	Español
17/Feb (1h)	Atributos de calidad	Ingles Español	Ingles	Español
24/Feb (0.5h)	Alcanzando la arquitectura del software	Ingles Español	Ingles	Español
24/Feb (0.5h)	Técnicas de arquitectura del software	Ingles Español	Ingles	Español
24/Feb (1h)	Taxonomías arquitectura del software: construcción	Ingles Español	Ingles (I) Ingles (II)	Español (I) Español (II)
05/Mar (1h)	Taxonomías arquitectura del software: modularidad	Ingles Español	Ingles (I) Ingles (II)	Español (I) Español (II)
10/Mar (1h)	Taxonomías arquitectura del software: comportamiento (monolito)	Ingles Español	Ingles (I) Ingles (II) Ingles (III) Ingles (IV)	Español (I) Español (II)
17–24/Mar (4h)	Taxonomías arquitectura del software: comportamiento (distribuidos y gran escala)	Ingles Español	Ingles (I) Ingles (II) Ingles (III) Ingles (IV)	Español (I) Español (II) Español (III) Español (IV)
07–14/Abr (3h)	Taxonomías arquitectura del software: Disposición	Ingles Español	Ingles (I)	Español (I)
14/Abr (1h)	Conferencia: Fighting the merchants of complexity	Ingles	Ingles (II)	Español (II)
23/Abr (1h)	Taxonomías arquitectura del software: negocio	Ingles Español	Ingles	Español

Cuadro 17.2: Distribución temporal clases teóricas – Curso 2020-21

Conferencias externas

Se considera que una parte esencial de un arquitecto del software es estar al tanto de las nuevas tendencias tecnológicas en la industria del software. Con el fin de acercar el entorno empresarial a la Universidad, en las clases expositivas se incluyen de forma regular entre 2 ó 3 conferencias externas llevadas a cabo por

profesionales del sector.

En la tabla 17.3 se presenta una lista de las conferencias impartidas desde el curso 2012-13.

Fecha	Ponente	Título
11/Abr/2021	Jorge Manrubia Díez Programador Senior, Basecamp	Fighting the merchants of complexity
10/Feb/2021	Ángel Maldonado, Pablo Cañal, Gema Rico CEO y desarrolladores en Empathy.co	Open Decentralization: SOLID and Empathy for new developers
01/Abr/2020	Diego Berrueta Principal Engineer, Atlassian	Scaling towards a thousand microservices
19/Feb/2020	Alejandro Juan García Project Manager, Empathy.co	Solid Search
12/Jun/2019	Mitzi Laszlo Solid Manager en Inrupt	Solid Presentation
20/Mar/2019	Pablo Bermejo Digital Platform Engineering Manager at DXC Technology	Software Architecture #10YearsChallenge
20/Ene/2019	Sergio Fernández Software Development Engineer, Amazon	Trends in Software Architecture: A professional (dis)orientation
09/Abr/2018	Luis Fernández Palacio Software architect, CapGemini	Software architecture based on microservices
20/Abr/2017	Manuel Jesús Fernández Quality department manager, Mecalux	Arquitectura del software del producto Mecalux EasyWMS
21/Mar/2017	Miguel Fernández Senior Platform Engineer, Github	Refactoring systems at scale
14- 15/Mar/2017	Iván Arrizabalaga Getino CTO, Zapiens	Zapiens: from legacy to production, and more

Cuadro 17.3: Relación de conferencias impartidas en la asignatura desde el curso 2016–17

17.3.2 Prácticas de aula/seminario

Las prácticas de aula o seminarios son actividades organizadas en aulas con grupos más reducidos. Los seminarios permiten un mayor grado de interactividad con los estudiantes y permiten afrontar actividades de aprendizaje diferentes a las clases expositivas tradicionales. Inicialmente, en la asignatura se habían utilizado los seminarios para profundizar en determinados temas, realizar algunos talleres o como complemento de las clases teóricas. Sin embargo, tras participar en una actividad sobre docencia universitaria, en el curso 2016-17 se decidió explorar la alternativa de aprovechar las clases de seminarios para que los estudiantes realizasen presentaciones de ciertos temas que ellos prepararían con una mayor profundidad. Las presentaciones serían realizadas por equipos formados por 2 ó 3 estudiantes con el fin de fomentar también el trabajo en grupo. El primer año los temas elegidos eran estilos arquitectónicos que los estudiantes podían explicar con más detalle. Los estilos arquitectónicos que se solicitaron fueron:

- Pipes and filters
- Layers
- Publish-subscribe
- MVC
- Peer to peer

- Microkernel
- Plugins
- Modelos de actores
- MapReduce
- Stream processing
- Intérpretes y lenguajes de dominio específico
- Arquitectura lambda

Los dos siguientes años se consideró que podría ser interesante que los temas fuesen sobre proyectos reales, de código abierto.

La inspiración de esta actividad venía dada por los trabajos realizados en una asignatura de arquitectura del software de la Universidad de Delft, que solicitaba a grupos de estudiantes de nivel de máster que documentasen la arquitectura de determinados proyectos open source. Los resultados se publicaron en formato libro en las ediciones de 2015 (<http://delftswa.github.io/index.html>), 2016 (<https://delftswa.gitbooks.io/desosa2016/>), 2017 (<https://delftswa.gitbooks.io/desosa-2017/>) y 2018 (<https://delftswa.gitbooks.io/desosa2018/>). Dicha asignatura se había inspirado en el libro [10] que también exploraba las arquitecturas de proyectos de código abierto.

Siguiendo esa línea, se decidió solicitar a los estudiantes que explorasen las arquitecturas de varios proyectos de código abierto y se les indicaba que tenían que explorar los siguientes aspectos:

- Objetivos del proyecto
- *Stakeholders* involucrados
- Atributos de calidad
- Estilo/patrones arquitectónicos
- Aspectos de desarrollo y gestión de incidencias
- Documentación
- Módulos y componentes
- Construcción y compilación del proyecto (se valoraba que los estudiantes fuesen capaces de compilar e incluso modificar el proyecto)

Los proyectos elegidos en el curso 2017-18 fueron:

- Maven: <https://maven.apache.org/>
- Spring boot: <https://spring.io/projects/spring-boot>
- Android Studio: <https://developer.android.com/studio>
- Ionic: <https://ionicframework.com/>
- Angular: <https://angular.io/>
- Apache Zookeeper: <https://zookeeper.apache.org/>
- Akka: <https://akka.io/>
- Gatling: <https://gatling.io/>
- Apache Mesos: <http://mesos.apache.org/>
- Docker: <https://www.docker.com/>
- Kubernetes: <https://kubernetes.io/>
- Apache Spark: <https://spark.apache.org/>
- Apache Flink: <https://flink.apache.org/>
- Lagom Framework: <https://www.lagomframework.com/>

Los proyectos elegidos en el curso 2018-19 fueron:

- Asciidoctor: <https://asciidoctor.org/>
- pandoc: <https://pandoc.org/>
- Dokie: <https://dokie.li/>
- Mocha: <https://mochajs.org/>
- WebRTC: <https://webrtc.org/>
- WebPack: <https://webpack.js.org/>
- Varnish: <https://varnish-cache.org/>
- Memcached: <https://memcached.org/>
- IPFS: <https://ipfs.io/>
- gitter: <https://gitter.im/>

En el curso 2019-20 se solicitaron los temas:

- PlantUML: <https://plantuml.com/>
- Modelio: <https://www.modelio.org/>

- Yarn: <https://yarnpkg.com/>
- npx: <https://github.com/npm/npnx>
- Jest: <https://jestjs.io/>
- Fast-check: <https://dubzzz.github.io/fast-check>
- React Native: <https://facebook.github.io/react-native/>
- Redux: <https://redux.js.org/>
- Redis: <https://redis.io/>
- gRPC: <https://www.grpc.io/>

La experiencia de solicitar trabajos sobre proyectos de código abierto se consideró muy positiva porque permitía a los estudiantes analizar arquitecturas de proyectos reales, dándoles la posibilidad de observar aspectos de los mismos que seguramente no habían tenido en cuenta. No obstante, los trabajos realizados en ocasiones tendían a ser un poco repetitivos, por lo que en el curso 2020-21 se exploró la alternativa de plantear trabajos sobre temas o ideas relacionadas con la arquitectura del software, que muchas veces se publicaban en artículos de blogs o similares, con el fin de que los estudiantes pudiesen leer sobre temas de actualidad relacionados con la arquitectura del software y profundizar sobre los mismos.

En el curso 2020-21 se han elegido los siguientes temas:

- Registros de decisiones arquitectónicas
- Arquitecturas ágiles
- Entrega continua (*continuous delivery*)
- Patrones de ramificación
- Arquitectura limpia
- *Event sourcing*
- Compromisos de los microservicios
- Patrón: *Circuit breaker*
- Ingeniería del Caos
- *Serverless*

Los estudiantes debían realizar un pequeño artículo tipo entrada de *blog* y una presentación de unos quince minutos sobre el tema. Debido a la situación de pandemia, las presentaciones de los estudiantes fueron llevadas a cabo de forma remota mediante Microsoft Teams, grabadas y publicadas bajo su consentimiento, estando disponibles en: <https://arquisoft.github.io/course2021.html#seminarPresentations>.

17.3.3 Prácticas de laboratorio

Se considera que la arquitectura del software es una disciplina con un carácter aplicado importante, que los estudiantes comprenderán mejor cuando se enfrenten a proyectos reales. Por eso mismo, se propone seguir un aprendizaje basado en proyectos para la parte práctica de la asignatura en el que se solicitará a los estudiantes la realización de un trabajo práctico a lo largo de todo el curso.

Las sesiones presenciales dedicadas a las prácticas de laboratorio se realizan durante dos horas seguidas cada semana, con un total de 28 horas repartidas en 14 semanas. Se propone un modelo que combina una presentación de una tecnología por parte del profesor durante la primer hora, seguida de una sesión en la que los estudiantes pueden aprovechar para hacer una reunión de trabajo para avanzar en la realización de la práctica.

El objetivo que se persigue con este esquema es introducir conceptos necesarios para la realización de la práctica, combinado con la posibilidad de facilitar a los estudiantes la realización de reuniones de sus respectivos equipos con la presencia del profesor que actuará como tutor de los grupos y les guiará en la resolución de las dudas que puedan tener sobre la práctica.

Además de las sesiones anteriores, también se realizarán 4 sesiones de control destinadas a revisar la evolución del trabajo de prácticas realizado por los estudiantes. El motivo de estas sesiones es llevar a cabo controles intermedios de la evolución del trabajo de los diferentes grupos durante el curso, para detectar y corregir en la medida de lo posible los problemas que puedan ir surgiendo, permitiendo a los profesores de prácticas dar realimentación sobre el trabajo realizado hasta el momento.

Las sesiones de control tienen el siguiente esquema:

- La primera sesión de control se realiza en la quinta semana de prácticas y revisar la primera versión de la documentación. El objetivo es comenzar con un entregable más sencillo desde un punto de vista técnico que les permita familiarizarse con los sistemas de control de versiones y la herramienta

github. La documentación se solicita en formato *Asciidoc*²⁴ siguiendo la filosofía de documentación como código [38]. De esta forma, los equipos pueden empezar a realizar contribuciones al proyecto que se reflejan en *commits* e incidencias desde el primer momento. Esta sesión de control puede ser útil para detectar casos de estudiantes que abandonan la asignatura de forma temprana y que no están contribuyendo al equipo. En ocasiones, los estudiantes se matriculan del curso, pero deciden abandonar. Dado que los equipos se forman con los estudiantes que hay matriculados y asignados a cada grupo de prácticas, no es posible detectar estos casos de abandono hasta que haya transcurrido un cierto tiempo de docencia. La quinta semana de clases, suele ser un buen momento para detectar estos casos y comprobar si ha habido un problema puntual o es realmente un caso de abandono.

- La segunda sesión de control consiste en la creación de una versión inicial del prototipo que tenga una funcionalidad básica. Esta sesión se realiza en la semana octava y normalmente tiene una mayor complejidad técnica porque es necesario implementar algo que funcione y *tocar* código. En este entregable suelen producirse también casos de abandono o de estudiantes que se ven superados por la carga docente combinada entre esta asignatura y las otras asignaturas de las que se han matriculado. Al ser un entregable que requiere un trabajo más técnico, es posible detectar qué estudiantes se están comprometiendo con la asignatura llevando acabo un aprendizaje autónomo, y que estudiantes están *dejándose llevar*. Puede ser conveniente realizar avisos a estos últimos estudiantes para revisar con detalle sus contribuciones en la tercera sesión de control. En esta sesión de control se solicita a los equipos que el entregable contenga además de una funcionalidad básica, una serie de pruebas unitarias que se ejecuten mediante el sistema de integración continua.
- La tercera sesión de control se realiza en la undécima semana y se solicita una primera *release* o versión pública del prototipo junto con una actualización de la documentación para sincronizar lo que pusieron inicialmente en la documentación con lo que realmente están realizando. Este entregable suele ser clave para detectar qué equipos llevan la asignatura al día. Además, es importante llevar a cabo el control de la sincronización entre la documentación y el prototipo. Durante las clases de teoría se insiste en lo importante que es desarrollar código en el que la arquitectura se vea reflejada, es decir, que los módulos y componentes que se describen en la documentación estén realmente visibles en el código. Esta sesión de control permite insistir en la importancia de dicha sincronización. Durante esta sesión de control también se insiste en la creación de pruebas de aceptación automatizadas a partir de historias de usuario con sistemas como Cucumber²⁵.
- La última sesión de prácticas consistirá en la presentación pública de la práctica. Idealmente, esta sesión se realizaba en una única jornada en el Salón de Actos de la Escuela y como una convocatoria pública abierta. Sin embargo, las restricciones organizativas del curso han impedido este tipo de jornada, por lo que en los últimos cursos se ha realizado dentro de la sesión de prácticas de laboratorio de cada equipo. Lo cual permite dedicar un espacio de casi una hora por equipo. La jornada consiste en una presentación realizada por cada equipo de unos 30 minutos, seguida de un período de preguntas y respuestas realizado por los profesores o cualquier otra persona del público. La presentación será realizada por los integrantes del equipo, pero los profesores pueden elegir cuál de dicho integrantes presenta en cada momento (en la práctica, se suelen ir alternando para dar la oportunidad de presentar a todos los estudiantes). En los dos cursos pasados, debido a la situación de pandemia, esta jornada se realizó de forma *online* a través de Teams. En la presentación, los estudiantes deben realizar una demostración práctica del prototipo, además de explicar los diferentes aspectos de la arquitectura propuesta.

El esquema de sesiones prácticas utilizado en el curso 2020-21 ha sido el siguiente:

1. Introducción a la práctica, organización de equipos
2. Control de versiones: git y *github*
3. Diagramas UML de arquitectura (diagramas de clase, componentes, despliegue, secuencia). Herramientas para creación de diagramas UML. Introducción a documentación y plantillas arc42
4. Introducción a tecnologías para la práctica: React y Solid²⁶.
5. Revisión del primer entregable: documentación
6. Automatización de la construcción: Maven, Gradle, npm, etc.

²⁴<https://asciidocdoctor.org/>

²⁵<https://cucumber.io/>

²⁶React y Solid son las tecnologías que se utilizaron el curso 2020–21, otros cursos podrían introducirse otras tecnologías

7. Distribución y despliegue: páginas github, docker
8. Revisión del segundo entregable: primer prototipo
9. Desarrollo basado en pruebas (TDD) e integración continua
10. Pruebas de aceptación y desarrollo basado en comportamiento (BDD)
11. Revisión del tercer entregable: Primera *release* con documentación actualizada
12. Pruebas de carga y otras pruebas: usabilidad, seguridad, escalabilidad, etc.
13. Monitorización y evaluación de rendimiento (*profiling*)
14. Día de la presentación y *release* final

17.3.4 Tutorías grupales

Estas sesiones se reservan para resolver dudas de los estudiantes. Un aspecto que ha funcionado bastante bien en algunos cursos ha sido dedicar la última sesión de tutorías grupales para dar unas nociones sobre cómo realizar presentaciones en público, lo cual puede ser bastante útil las semanas anteriores a la presentación del trabajo práctico.

17.4 Evaluación de la asignatura

Para la evaluación de la asignatura se propone un esquema híbrido adaptado a las actividades de aprendizaje realizadas y alineado con los resultados de aprendizaje. El objetivo es alcanzar una alineación constructiva entre los resultados de aprendizaje y las tareas de evaluación [50] que permita compensar las posibles desviaciones que tendrían esquemas de evaluación centrados en un único aspecto.

La evaluación se divide en dos notas, la nota de teoría y la nota de prácticas, dándole más importancia a la nota de prácticas (60 % del total) que a la nota de teoría (40 %). El motivo es transmitir a los estudiantes la importancia del trabajo de práctica lo cual encaja con los resultados de aprendizaje descritos en 17.1.2.

17.4.1 Evaluación de teoría

La evaluación de la nota de teoría se propone como la combinación de la nota del examen teórico con las notas de diversas pruebas parciales que se realizarán a lo largo del curso a modo de evaluación continua. Las pruebas parciales pueden incluir presentaciones de diversos temas en los seminarios tal y como se indica en la sección 17.3.2 además de las respuestas que se realicen a preguntas planteadas durante el curso. Los cuestionarios podrán realizarse mediante tecnologías tipo Zapiens²⁷ o Kahoot²⁸, aunque preferiblemente se utilizarán técnicas integradas con las herramientas que ofrece la Universidad como Teams²⁹ o Moodle³⁰. El objetivo es realizar preguntas a los estudiantes durante el curso que les permita repasar de forma activa los contenidos, evitando dejar el estudio del material para última hora y facilitando el aprendizaje de los contenidos a largo plazo [11, 77]. Aunque se podría plantear la posibilidad de sustituir el examen final por las pruebas continuas, se considera que una prueba final puede ser útil para validar los conocimientos de forma unificada.

17.4.2 Evaluación de práctica

Para la evaluación de los contenidos prácticos, se utiliza también un esquema híbrido que tenga en cuenta el trabajo individual de los estudiantes, junto con el trabajo realizado en grupo. Se asigna un 70 % a la nota del trabajo en grupo con el fin de fomentar que los estudiantes participen en sus equipos.

La evaluación del grupo se realiza de forma estructurada mediante una rúbrica que incluye los siguientes aspectos:

- Presentación del proyecto
 - Calidad de la presentación
 - Capacidad para responder a las preguntas planteadas
- Documentación técnica
 - Introducción y objetivos
 - Restricciones de arquitectura

²⁷<https://zapiens.ai/>

²⁸<https://kahoot.com/>

²⁹<http://teams.microsoft.com/>

³⁰<https://moodle.com/>

- Contexto y ámbito del sistema
- Estrategia de la solución
- Vista de Bloques de construcción
- Vista en tiempo de ejecución
- Vista de despliegue
- Preocupaciones transversales
- Decisiones de diseño
- Requisitos de calidad
- Riesgos y Deuda técnica
- Requisitos funcionales (aquí se incluye una lista que depende del proyecto de cada año)
- Requisitos funcionalesopcionales (depende del proyecto de cada año)
- Desarrollo
 - Gestión del proyecto (incidencias, actas, etc.)
 - Integración continua
 - Pruebas de integración y unitarias (informe de cobertura)
 - Calidad de código
 - Pruebas de aceptación
 - Empaquetamiento y despliegue
 - Pruebas de carga
 - Pruebas de usabilidad
 - Otras pruebas
- Otros atributos de calidad
 - Usabilidad y apariencia
 - Seguridad
 - Privacidad

Las rúbricas son cubiertas por todos los profesores de la asignatura que actuán como un tribunal de evaluación. En algunos cursos, cuando las presentaciones se realizaban en una única sesión, las rúbricas también eran cubiertas por el resto de equipos llevando a cabo una evaluación por pares. Sin embargo, en los últimos años, por necesidades organizativas, la jornada de presentación se realiza en la última sesión de laboratorio, con lo cual no se puede garantizar que el resto de equipos puedan participar, y se ha desestimado esta opción.

Como se ha indicado en la sección 17.3.3, en las prácticas de laboratorio se realizan 4 sesiones de control a lo largo del curso. Las tres primeras sesiones permiten observar la evolución de los grupos, donde el profesor de prácticas pueda dar realimentación positiva a cada equipo y detectar posibles conflictos o situaciones anómalas. Actualmente, no se califican con el fin de que los estudiantes no se sientan presionados durante los mismos, pero se puede considerar la opción de realizar una calificación parcial que pueda servir de motivación y realimentación a los equipos.

Para la asignación de la nota individual, cada profesor de prácticas analizará las contribuciones individuales de cada estudiante a través de la plataforma github³¹. Dado que se solicita a los estudiantes que todas las actividades *oficiales* del proyecto se realicen a través de github, es posible realizar un seguimiento de sus contribuciones, que pueden ser de diferentes tipos: *commits*, *pull requests*, participación en *issues*, actas de reuniones, etc. La herramienta github proporciona resúmenes de la actividad en un repositorio a través de la opción *Contributors*³². Además del uso de github, el programa *gitinspector*³³ ofrece un cuadro de mandos muy útil con información que permite ver la evolución de los diferentes integrantes de cada equipo a lo largo de las semanas del curso, pudiendo incluso identificar quiénes son los que trabajan en unos ficheros o en otros.

Un aspecto importante que se lleva a cabo es solicitar a cada estudiante la creación de un *issue* individual en el que realice un auto-informe describiendo brevemente las contribuciones individuales que ha realizado. Este *issue* debe ser realizado antes de cada una de las sesiones de control, y puede contener enlaces a los *issues* concretos en los que ha estado trabajando. Este *issue* puede ser anotado por el resto de compañeros del equipo indicando si están o no de acuerdo con su contenido. Esta actividad ha demostrado ser muy valiosa

³¹<https://github.com/>

³²A modo de ejemplo, pueden verse las contribuciones de un proyecto en: https://github.com/Arquisoft/radarin_en3b/graphs/contributors

³³<https://github.com/ejwa/gitinspector>

porque permite a los estudiantes reflexionar sobre sus propias contribuciones y a los profesores detectar mejor las verdaderas contribuciones de los estudiantes. En este sentido, desde el comienzo del curso se insiste a los estudiantes que las únicas contribuciones que se tendrán en cuenta son aquellas que se reflejan en github. Cualquier otra contribución, como mensajes en otras plataformas tipo *whatsapp*, documentos en carpetas personales de sus ordenadores o similares no serán tenidos en cuenta. De esta forma, se considera que los estudiantes también aprenden a identificar las actividades que llevan a cabo en un ambiente de trabajo, en el que la trazabilidad de las actividades realizadas es importante. Una ventaja de esta forma de trabajar basada en evidencias a través de *github* es su compatibilidad con el trabajo remoto, algo que se ha demostrado especialmente útil durante la pandemia y que ha permitido adaptar las actividades de la asignatura a esta modalidad de forma sencilla.

17.4.3 Evaluaciones alternativas

Aunque la mayoría de los estudiantes realizarán la evaluación siguiendo el cauce ordinario, es necesario tener en cuenta diferentes alternativas que pueden ocurrir ante determinadas situaciones como pueden ser estudiantes que trabajan y no pueden acudir a las clases, estudiantes que abandonan la asignatura, etc.

En principio, la asistencia a las actividades de aprendizaje presenciales es obligatoria y se tiene en cuenta. Durante la pandemia, dichas actividades se sustituyeron por actividades síncronas no presenciales a través de Teams, cuya asistencia también era obligatoria y se tenía en cuenta. En concreto, para que los estudiantes puedan participar en la evaluación ordinaria se solicita que asistan a un mínimo de un 80 % de las actividades presenciales o síncronas. La asistencia a la jornada de presentación de prácticas también se considera obligatoria y si un estudiante no participa en la misma se considerará como un no presentado.

En caso de no superarse la asignatura por evaluación continua, no se podrá aprobar la misma en la convocatoria ordinaria. En las convocatorias extraordinarias se realizará un examen (40 % de la nota final) y un proyecto individual (60 % de la nota final) similar a las actividades individuales o en grupo que se hayan propuesto a lo largo del semestre.

En algunos casos concretos, los estudiantes pueden acogerse a la modalidad de *evaluación diferenciada*. En el caso de teoría, el estudiante realizará las pruebas teóricas igual que el resto de estudiantes puesto que la mayoría de dichas pruebas pueden realizarse de forma no presencial. En el caso de presentaciones de trabajos de seminarios, los estudiantes deberán al menos realizar la presentación de su trabajo el día que les corresponda.

En el caso de los trabajos de laboratorio, si el estudiante puede adaptarse al trabajo en grupo con el resto de los compañeros, entonces se evaluará de la misma forma y solamente tendrá que acudir al 20 % de las sesiones presenciales. En caso de no poder adaptarse, deberá ponerlo en conocimiento de los profesores antes de que finalice el primer mes de docencia de la asignatura para que se le comunique un proyecto individual de prácticas.

17.4.4 Resumen de la evaluación

La nota total N de la asignatura se compone de la nota de teoría (N_T) y de la nota de prácticas (N_P) y se calcula como:

$$N = N_T \times 0,4 + N_P \times 0,6$$

La nota de teoría se calcula a partir de la nota del examen final N_E más la nota de las pruebas parciales N_{PP} con el siguiente porcentaje:

$$N_T = N_E \times 0,7 + N_{PP} \times 0,3$$

La nota de prácticas se calcula a partir de la nota del grupo N_G y la nota individual N_I como:

$$N_P = N_G \times 0,7 + N_I \times 0,3$$

17.4.5 Premios y concursos

Durante la conferencia Internacional de Web Semántica del año 2018³⁴ el autor de esta memoria participó en el Workshop on Decentralizing the Semantic Web³⁵ organizado por Ruben Verborgh, Tobias Kuhn y Tim Berners-Lee en el que se realizó una presentación sobre el proyecto Solid³⁶, un proyecto que pretende fomentar la creación de aplicaciones Web que respeten la privacidad de los datos de los usuarios y actúen de forma descentralizada. Durante la presentación, Tim Berners-Lee indicaba que en dicho proyecto sería interesante contar con desarrolladores interesados en realizar nuevas aplicaciones que cumpliesen con dichos principios. Tras acabar la presentación se le planteó la posibilidad de que en la práctica de asignatura de arquitectura del software se realizasen prototipos siguiendo dichos principios, la idea fue bien recibida e incluso se sugirió la posibilidad de organizar una especie de concurso apoyado por la empresa Inrupt, uno de cuyos fundadores es Tim Berners-Lee, que está dedicada fundamentalmente al proyecto Solid³⁷.

De esa forma, en el curso 2018-19, el enunciado de la práctica de la asignatura consistía en la realización de una aplicación conversacional compatible con los principios Solid. Al acabar el curso, los equipos participantes podían presentarse voluntariamente a lo que se denominó *Solid Challenge* que sería un concurso cuyo premio consistiría en un certificado firmado por Tim Berners-Lee. El jurado del concurso estaría formado por el profesor de la asignatura y dos miembros de la comunidad Solid. La entrega de se realizó en un acto llevado a cabo en el Salón de Actos al que acudió la directora del proyecto Solid de aquel momento, Mitzi Laszlo.

Durante la jornada de entrega del premio, Mitzi sugirió la involucración de empresas asturianas en este evento y la empresa Empathy.co³⁸ decidió apoyarlo desde entonces, ofreciendo a los ganadores del concurso una beca de formación además de una cantidad de dinero. Desde entonces, el jurado está compuesto por dos representantes de Solid, dos representantes de Empathy, y un profesor de la asignatura. Los criterios que se han tenido en cuenta para seleccionar al equipo ganador han sido:

- Estética y facilidad de uso por personas no técnicas.
- Seguridad y respeto a la privacidad de los usuarios
- Capacidad de respuesta de la aplicación
- Originalidad de la solución.
- Calidad de la documentación técnica
- Calidad del código.

Debido a la pandemia, en las dos últimas ediciones no se ha realizado un acto de entrega presencial. No obstante, el acto de entrega se hizo coincidir con el evento Solid World, en el que participó Tim Berners-Lee. Dicho acto ha sido grabado y los vídeos están accesibles en: <https://vimeo.com/425919959> y <https://vimeo.com/572726511>.

De esta forma, en los últimos tres cursos académicos, la práctica de la asignatura cuenta con un aliciente más en forma de concurso, que permite dar una mayor visibilidad al trabajo de los estudiantes y aumentar su motivación.

17.5 Justificación y posibles variaciones

La descomposición de la asignatura en dos componentes: teoría y prácticas se considera adecuada para dar una importancia suficiente a la parte práctica de la asignatura que encaja con las competencias y resultados de aprendizaje propuestos.

Respecto a la teoría, se considera importante evitar que los estudiantes se centren únicamente en estudiar la parte teórica de la asignatura para un examen final y por eso se plantean diversas actividades a lo largo del curso que llevan intercalados aspectos de evaluación. De esa forma, se ha conseguido que los estudiantes se enfrenten al examen final como una prueba sencilla que les permite certificar los conocimientos que han adquirido.

La parte de prácticas de la asignatura fomenta el trabajo colaborativo y en equipo, mediante aprendizaje basado en un proyecto de software complejo. Con el esquema planteado, el profesor de prácticas pasa a ser un guía tutor de los equipos, que apoya su trabajo. Se considera que este formato de prácticas facilita el

³⁴<http://iswc2018.semanticweb.org/>

³⁵<https://iswc2018.desemweb.org/>

³⁶<https://solidproject.org/>

³⁷<https://inrupt.com/>

³⁸<https://empathy.co/>

	1. Definiciones	2. Comunicando arquitectura	3. Rol arquitecto y Stakeholders	4. Atributos de calidad	5. Creación y validación	6. Técnicas arquitectura	7. Construcción	8. Modularidad	9. Componentes: monolito	10. Componentes: distribuidos y big data	11. Disposición	12. Entorno empresarial
RA.IS-1	✓											
RA.IS-2		✓										
RA.IS-3			✓				✓					
RA.IS-4				✓			✓					
RA.IS-5		✓		✓			✓					
RA.IS-6	✓		✓	✓			✓					
RA.IS-7	✓		✓	✓			✓					
RA.IS-8	✓		✓	✓			✓					
RA.IS-9	✓		✓	✓			✓					
RA.IS-10	✓		✓				✓			✓		
RA.IS-11	✓		✓	✓	✓						✓	✓

Cuadro 17.4: Relación entre Resultados de aprendizaje y unidades didácticas: ✓ indica que hay una fuerte relación

aprendizaje autorregulado y colaborativo, aspectos que les puede ser muy útil para su posterior integración en equipos en su vida laboral. Los estudiantes aprenden a responsabilizarse de sus acciones y a cumplir con plazos y fechas de entrega, así como a practicar capacidades de estimación de esfuerzos.

En la tabla 17.4 se relacionan los resultados de aprendizaje descritos en la sección 17.1.2 con las unidades temáticas planteadas en la sección 17.2.

Se considera que la propuesta docente aquí presentada encaja con el contexto en el que se presenta, tanto por el tipo de estudios, como por el tipo de estudiantes y los resultados de aprendizaje considerados.

En ese sentido, además de las competencias específicas establecidas para la asignatura concreta, la realización de la práctica en equipo y las presentaciones realizadas por estudiantes en las clases de seminarios también fomenta otras competencias generales como la CG-9. *Habilidades de comunicación oral*, la CG11. *Competencia para el trabajo en equipo*, CG-12: *liderazgo*, CG-13: *negociación*, etc. Dichas competencias se han incluido en la sección 17.1.1.

Las sesiones de control realizadas durante las prácticas se llevan a cabo como un mecanismo que facilita la detección temprana de problemas en las dinámicas de trabajo en equipo. En las dos primeras ediciones de la asignatura, se realizó una única sesión de entrega de prácticas y en dicha sesión aparecieron equipos con problemas entre sus componentes, que si se hubiesen detectado a tiempo, podrían haberse encaminado mejor. Desde que se utilizan las sesiones de control, cuando se detectan este tipo de problemas, se pueden tomar acciones que permiten mejorar la dinámica. Algunas decisiones que se tomaron han sido intercambiar a miembros de equipos que no se llevaban bien con sus compañeros, detectar estudiantes que no realizan contribuciones, analizar las causas y actuar en consecuencia, detectar equipos que no han entendido el enunciado del problema, etc.

Se considera que la organización del concurso Solid en paralelo a la asignatura ha servido para una visibilidad al trabajo de los estudiantes, así como para que puedan empezar a ver un horizonte más allá del contexto académico de la Universidad, colaborando e intercambiando mensajes con otros desarrolladores

internacionales, y observando proyectos en evolución, que no siempre tienen una gran documentación ni soporte comercial.

A continuación se presentan algunas posibles variaciones que se podrían tener en cuenta:

- **Mantenimiento de software existente:** Un aspecto fundamental en el software actual es la necesidad de mantener software existente o de trabajar con software *legacy* o heredado. De hecho, la mayoría de los estudiantes se enfrentarán a proyectos de mantenimiento de software heredado que no ha sido creado por ellos mismos en su vida laboral. Este tipo de proyectos no suele recibir suficiente atención y se considera que su tratamiento recibe poco énfasis en general. Una posible forma de que los estudiantes valoren la importancia de estos proyectos de mantenimiento podría ser pedirles como práctica el mantenimiento de software existente. Existen varias posibilidades que se han estado valorando o incluso se han realizado de forma experimental:

- Solicitar que mantengan software existente en algún proyecto de código abierto o de alguna empresa. Esta posibilidad sería la más interesante, pero también es arriesgado porque podría implicar que manos poco experimentadas se pusiesen a mantener código real con la consiguiente distorsión, así como las dificultades inherentes al proceso de evaluación.
- Otra posibilidad sería que los estudiantes mantengan proyectos de software desarrollados por cursos anteriores añadiendo nuevas funcionalidades. Un problema de esta posibilidad es que los estudiantes no se enfrentan al problema de plantear o diseñar una arquitectura nueva para un software, y si se parte ya de una determinada descomposición modular, entonces una parte de la arquitectura ya les vendría dada, entrando la asignatura en aspectos que se saldrían de la arquitectura del software y entrarían más en el puro mantenimiento del mismo.
- Otra posibilidad intermedia sería dividir la práctica en dos partes, una primera parte de mantenimiento y evolución de una práctica de cursos anteriores y una segunda, de creación de un componente nuevo, que les obligue a diseñar una nueva arquitectura. Esta posibilidad se realizó durante varios años, en los que la práctica se descomponía en 2 módulos. Un primer módulo consistía en un pequeño microservicio que los estudiantes del año siguiente tenían que adaptar a las nuevas necesidades. Esta opción se abandonó cuando la práctica se centró en las tecnologías Solid, que requerían un replanteamiento considerable de la arquitectura. Además, se consideró que era más adecuado arrancar la asignatura explicando conceptos de control de versiones y trabajo colaborativo con *github* sobre la documentación, que sobre código fuente directamente.
- Otra posibilidad que se llevó a cabo durante un curso fue que durante una parte del curso los estudiantes trabajasen con una parte de la práctica, y que en mitad del curso, tuviesen que trabajar con partes de la práctica realizadas por sus compañeros. En concreto, se descompuso el prototipo en 2 módulos y cada equipo de trabajo en 2 sub-equipos, de forma que durante la primera mitad del curso, un sub-equipo trabajaba en un módulo y en la segunda mitad se intercambiaban los módulos. La opción se descartó en años posteriores porque conllevaba una cierta complejidad organizativa cuando los sub-equipos eran muy pequeños y algunos estudiantes abandonaban la asignatura, dejando al otro sub-equipo sin la otra parte del código. Esta opción se podría volver a considerar para próximos años.

- **Integración de la práctica de esta asignatura con la de otras asignaturas del mismo semestre:** Un aspecto que se ha valorado durante estos años es una mayor integración con otras asignaturas, que podría consistir por ejemplo, en solicitar una práctica común entre varias de las asignaturas con las que se comparte semestre. Esta posibilidad de momento no se ha llevado a cabo debido a las problemáticas prácticas que acarrea, como es el caso de que no todos los estudiantes se matriculan de forma completa de todas las asignaturas de un semestre, o estudiantes que podrían abandonar una de las asignaturas en mitad del semestre.
- **Coordinación con la asignatura de dirección y planificación de proyectos de cuarto curso:** Los estudiantes de dicha asignatura podrían ejercer como directores de proyectos de las prácticas que realizarían los estudiantes de la asignatura de arquitectura del software. De momento, esta posibilidad tampoco se ha llevado a cabo por las complejidades organizativas que podrían suponer al coordinar los horarios de prácticas de laboratorio de las dos asignaturas así como los problemas indicados anteriormente.
- **Cambio de semestre:** Aunque esta propuesta supondría una modificación del plan de estudios y en principio, no sería posible. Podría darse el caso de que se realizase una modificación del mismo en el futuro. En principio, se considera que esta asignatura debería ir después de la asignatura de *Diseño del Software*, la cual, a su vez debería impartirse después de las asignaturas de introducción

a la programación y metodología de la programación. De esa forma, los posibles movimientos de semestre de esta asignatura podrían ser pasarlo a cuarto curso. Dicho cambio no sería problemático y de hecho, podría ser beneficioso que los estudiantes tuvieran un mayor nivel de maduración en los aspectos de desarrollo de software, lo cual redundaría en una mejor comprensión de la problemática asociada con la creación de software de calidad.

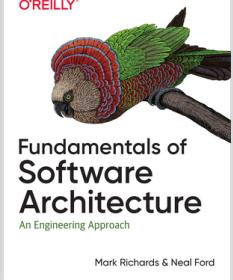
- **Enseñanza no presencial:** Aunque la docencia de esta asignatura se plantea dentro de un esquema de enseñanza presencial, no hay excesivos problemas para adaptar dicha enseñanza a un modelo no presencial. De hecho, durante la mitad del curso 2019-20 y durante todo el curso 2020-21, la asignatura se impartió de forma no presencial y no hubo excesivos problemas. Durante el pasado curso, de hecho, se impartían las clases expositivas a través del software Microsoft Teams, y las clases eran grabadas y publicadas en un Youtube³⁹. Las clases de prácticas se impartían también de forma no presencial y dado que se utiliza *github* para la gestión de los proyectos, un software adaptado para el trabajo remoto, los estudiantes pudieron adaptarse al trabajo no presencial sin mayor problema.

³⁹Los vídeos están disponibles en: <https://www.youtube.com/playlist?list=PLX272f9N-3pwACzAgnSMBBz23sfPNx7bF>

17.6 Bibliografía básica comentada

En esta sección se realiza una breve revisión de algunos de los libros que se consideran relevantes para la asignatura. De forma complementaria, se ha creado una lista denominada *Software architecture* en GoodReads⁴⁰ que contiene los libros aquí comentados.

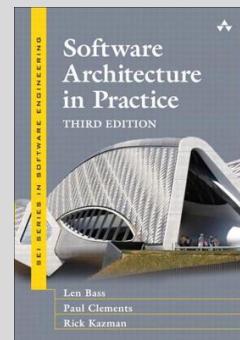
Para cada libro, se incluye la tabla de contenidos (sólo los ítems principales) y una serie de comentarios sobre la adecuación del libro para la asignatura. Cabe decir que los comentarios realizados son impresiones personales del autor de este documento y que algunas de las apreciaciones (especialmente las negativas) deben ser tomadas en el contexto de la adecuación del libro para la asignatura.

Fundamentals of Software Architecture
Contenidos: <ul style="list-style-type: none"> ■ Chapter 1: Introduction ■ Chapter 2: Architectural Thinking ■ Chapter 3: Modularity ■ Chapter 4: Architecture Characteristics Defined ■ Chapter 5: Identifying Architecture Characteristics ■ Chapter 6: Measuring and Governing Architecture Characteristics ■ Chapter 7: Scope of Architecture Characteristics ■ Chapter 8: Component-Based Thinking ■ Chapter 9: Architecture Styles ■ Chapter 10: Layered Architecture Style ■ Chapter 11: Pipeline Architecture ■ Chapter 12: Microkernel Architecture ■ Chapter 13: Service-Based Architecture ■ Chapter 14: Event-Driven Architecture Style ■ Chapter 15: Space-Based Architecture ■ Chapter 16: Orchestration-Driven Service-Oriented Architecture ■ Chapter 17: Microservices Architecture ■ Chapter 18: Choosing the Appropriate Architecture Style ■ Chapter 19: Architecture Decisions ■ Chapter 20: Analyzing Architecture Risk ■ Chapter 21: Diagramming and Presenting Architecture ■ Chapter 22: Making Teams Effective ■ Chapter 23: Negotiation and Leadership Skills ■ Chapter 24: Developing a Career Path <p>Comentarios: Este libro supone una aproximación pragmática a la arquitectura del software que combina la experiencia del autor en la aplicación práctica de la disciplina en la industria. El autor publica además una serie de videos de corta duración en Youtube titulados Software Architecture Monday que son muy recomendables. Los últimos capítulos del libro incluyen temas sobre gestión de equipos, negociación y habilidades de liderazgo que consideramos muy importantes para los arquitectos del software.</p> <p>Referencia: Mark Richards Neal Ford. <i>Fundamentals of Software Architecture</i>. O'Reilly, 6 de mar. de 2020. ISBN: 1492043451</p> 

⁴⁰La lista está disponible en: https://www.goodreads.com/list/show/160472.Software_architecture#44144493

Software Architecture in Practice**Contenidos:**

- Part 1: Introduction
 - 1. What is Software Architecture?
 - 2. Why is Software architecture important?
 - 3. Too many Contexts of Software Architecture
- Part 2: Quality Attributes
 - 4. Understanding Quality Attributes
 - 5. Availability
 - 6. Interoperability
 - 7. Modifiability
 - 8. Performance
 - 9. Security
 - 10. Testability
 - 11. Usability
 - 12. Other quality attributes
 - 13. Architectural Tactics and Patterns
 - 14. Quality Attribute Modeling and Analysis
- Architecture in the Life Cycle
 - 15. Architecture in Agile Projects
 - 16. Architecture and requirements
 - 17. Designing an architecture
 - 18. Documenting Software Architectures
 - 19. Architecture, implementation and testing
 - 20. Architecture reconstruction and Conformance
 - 21. Architecture Evaluation
 - 22. Management and Governance
- Part 4: Architecture and Business
 - 23 Economic Analysis of Architectures
 - 24 Architecture Competence
 - 25 Architecture and Software Product Lines software
- Part 5. The Brave New World
 - Architecture in the Cloud
 - Architecture for the Edge
 - Epilogue



Comentarios: Es probablemente el libro que mejor cubre todos los aspectos de la disciplina de arquitectura. El libro va actualmente por la tercera edición y los autores están preparando una cuarta que todavía no ha sido publicada. Quizá el hecho de que la primera edición del libro sea de 1997 hace que el libro tenga un enfoque un tanto tradicional a pesar de que los autores han incluido nuevos capítulos sobre cómo afrontar la arquitectura del software en entornos ágiles.

El libro hace un repaso de tácticas para afrontar determinados atributos de calidad, entre los que destaca: disponibilidad, interoperabilidad, modificabilidad, rendimiento, seguridad, testabilidad y usabilidad.

Referencia: Len Bass, Rick Kazman y Paul Clements. *Software Architecture in Practice: 3rd edition*. SEI Series in Software Engineering. Addison Wesley, 1 de oct. de 2012. ISBN: 0321815734

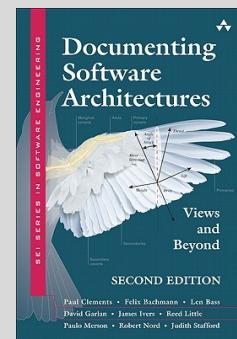
Documenting Software Architectures: Views and Beyond

Contenidos:

- Part I: A collection of Software Architecture Styles
 - 1. Module Views
 - 2. A Tour of some Module Styles
 - 3. Component and Connector Views
 - 4. A Tour of some Component and Connector Views
 - 5. Allocation Views and a Tour of some Allocation Styles
- Part II: Beyond Structure: Completing the Documentation
 - 6. Beyond the Basics
 - 7. Documenting Software Interfaces
 - 8. Documenting Behavior
- Part III: Building the Architecture Documentation
 - Choosing the Views
 - Building the Documentation Package
 - Reviewing an Architecture Document
- Epilogue: Using Views and Beyond with Other Approaches

Comentarios: Es uno de los libros clásicos acerca de la documentación de arquitecturas del software que sigue el modelo del *Software Engineering Institute* basado en proponer vistas y puntos de vista. Establece la clasificación de estilos arquitectónicos en módulos, componentes y conectores; y disposición, que se utiliza en la asignatura. Este libro puede considerarse una de las referencias clásicas sobre la arquitectura del software. En su segunda edición, los autores han realizado un esfuerzo por actualizar los contenidos incluyendo referencias a cómo documentar arquitecturas del software en ambientes ágiles.

Referencia: Paul Clements, Felix Bachmann y Len Bass. *Documenting Software Architectures: Views and Beyond*. Addison Wesley, 1 de oct. de 2010. 537 páginas. ISBN: 0321552687



Clean Architecture

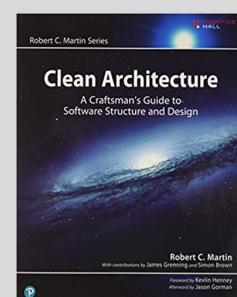
Contenidos:

- Part I: Introduction
- Part II: Starting with the bricks: Programming Paradigms
- Part III: Design Principles
- Part IV. Component Principles
- Part V. Architecture
- Part VI. Details

Comentarios: Presenta una propuesta de arquitectura limpia o *Clean architecture*, que se centra en los aspectos importantes del modelo y las reglas de negocio, estableciendo que las bases de datos, la Web o los *frameworks* son detalles con los que la arquitectura debe interactuar.

También realiza una descripción de los principios SOLID y de una serie de principios de diseño de componentes que se utilizan a la hora de presentar técnicas de descomposición modular en arquitecturas.

Referencia: Robert C. Martin. *Clean Architecture*. Prentice Hall, 10 de sep. de 2017. 432 páginas. ISBN: 0134494164



Building Evolutionary Architectures

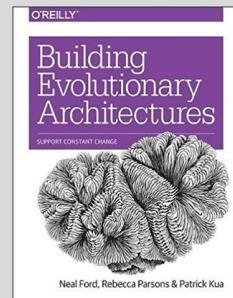
Contenidos:

- 1: Software Architecture
- 2. Fitness Functions
- 3. Engineering Incremental Change
- 4. Architectural Coupling
- 5. Evolutionary Data
- 6. Building Evolvable Architectures
- 7. Evolutionary Architecture Pitfalls and Antipatterns
- 8. Putting Evolutionary Architecture into Practice

Comentarios: Presenta una nueva propuesta para la arquitectura del software que denomina arquitectura evolutiva inspirada en la forma en que la propia naturaleza evoluciona. Indica que un aspecto importante en el software contemporáneo es la respuesta al cambio y la evolución continua, de forma que un arquitecto debe establecer una serie de criterios de calidad que se pueden ir comprobando mediante lo que denomina función de encaje (*fitness function*).

Aunque se trata de un libro corto, la propuesta es muy interesante y encaja con una nueva forma de entender el rol del arquitecto del software.

Referencia: Neal Ford, Rebecca Parsons y Patrick Kua. *Building Evolutionary Architectures*. O'Reilly UK Ltd., 8 de oct. de 2017. ISBN: 1491986360



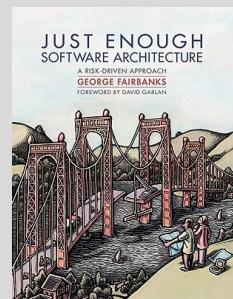
Just Enough Software Architecture: A Risk-Driven Approach

Contenidos:

- 1: Risk-driven Software Architecture
- 2: Software Architecture
- 3: Risk-driven model
- 4: Example: Home Media Player
- 5: Modeling Advice
- 6: Engineers use Models
- 7: Conceptual Model of Software Architecture
- 8: The Domain Model
- 9: The Design Model
- 10: The Code Model
- 11: Encapsulation and Partitioning
- 12: Model elements
- 13: Model relationships
- 14: Architectural Styles
- 15: Using Architecture Models
- 16: Conclusion

Comentarios: Es un libro que comienza a ofrecer una nueva visión de la arquitectura del software, más adaptada al desarrollo ágil. Propone establecer la cantidad justa de arquitectura del software teniendo en cuenta los posibles riesgos a subsanar. También plantea la idea de que el código debe ser evidente desde el punto de vista arquitectónico, expresando la intención en el código.

Referencia: George Fairbanks. *Just Enough Software Architecture: A Risk-Driven Approach*. Marshall y Brainerd, 1 de sep. de 2010. 360 páginas. ISBN: 0984618104



The Software Architect Elevator

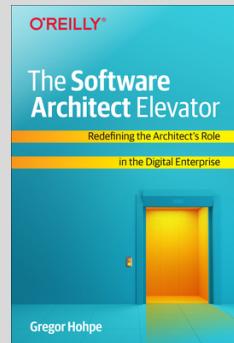
Contenidos:

- I. Architects
- II. Architecture
- III. Communication
- IV. Organizations
- V. Transformation
- VI. Epilogue: Architecting IT Transformation

Comentarios: Se trata de un libro reciente basado en capítulos breves en los que el autor expresa su opinión sobre diferentes aspectos de la arquitectura del software.

En general, las reflexiones del autor están basadas en la experiencia y se consideran muy valiosas. Aunque el formato del libro no hace posible su adopción como libro de texto, es una lectura muy recomendable para los estudiantes que puedan estar interesados en esta disciplina y algunos capítulos pueden servir de base para la organización de debates o similares en las clases.

Referencia: Gregor Hohpe. *The Software Architect Elevator: Transforming Enterprises with Technology and Business Architecture*. O'Reilly Media, Inc, USA, 24 de abr. de 2020. 350 páginas. ISBN: 1492077542



Release It!: Design and Deploy Production-Ready Software

Contenidos:

- 1: Living in Production
- 2: Case Study: The Exception that Grounded an Airplane
- 3: Stabilize your System
- 4: Stability Antipatterns
- 5: Stability Patterns
- 6: Case Study: Phenomenal Cosmic Powers. Itty-Bitty Living Space
- 7: Foundations
- 8: Processes on Machines
- 9: Interconnect
- 10: Control Plane
- 11: Security
- 12 Case Study: Waiting for Godot
- 13: Design for Deployment
- 14: Handling versions
- 15: Case Study: Trampled by your own Customers
- 16: Adaptation
- 17: Chaos Engineering

Comentarios: Se trata de la segunda edición de un libro publicado inicialmente en 2007. El libro describe una serie de patrones y antipatrones a tener en cuenta a la hora de lanzar un software a producción. El libro condensa la experiencia práctica del autor mediante una serie de casos de estudio que hacen interesante su lectura.

La segunda edición ha añadido un nuevo capítulo sobre la ingeniería del caos.

Aunque el libro no está pensado para ser un libro de texto, el material de la unidad didáctica 17.2.11 se ha basado fundamentalmente en este libro, que se considera una lectura muy recomendable.

Referencia: Michael Nygard. *Release It!* The pragmatic programmers. O'Reilly UK Ltd., 1 de feb. de 2018. 350 páginas. ISBN: 1680502395



Continuous Architecture in Practice

Contenidos:

- Chapter 1: Chapter 1. Why Software Architecture Is More Important than Ever
- Chapter 2. Architecture in Practice: Essential Activities
- Chapter 3. Data Architecture
- Chapter 4. Security as an Architectural Concern
- Chapter 5. Scalability as an Architectural Concern
- Chapter 6. Performance as an Architectural Concern
- Chapter 7. Resilience as an Architectural Concern
- Chapter 8. Software Architecture and Emerging Technologies
- Chapter 9. Conclusion

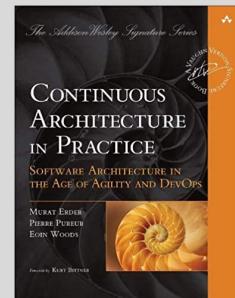
Comentarios: Es un libro de publicación muy reciente que en realidad es una continuación de un libro de los mismos autores titulado *Continuous Architecture: Sustainable Architecture in an Agile and Cloud-Centric World* y publicado en 2015.

En este libro se profundiza aún más en los conceptos de arquitecturas evolutivas y se ofrece una visión actualizada del rol de la arquitectura del software en la industria.

A lo largo del libro se utiliza como ejemplo un sistema de intercambio financiero que proporciona una plataforma para subir y procesar letras de crédito.

El libro incluye capítulos específicos para la arquitectura de datos, seguridad, escalabilidad, rendimiento y resiliencia que encaja con algunas de las preocupaciones más importante de los sistemas actuales. También incluye un último capítulo en el que afronta técnicas de aprendizaje automático y *blockchain* desde el punto de vista arquitectónico que es bastante interesante.

Referencia: Murat Erder, Eoin Woods y Pierre Pureur. *Continuous Architecture in Practice: Software Architecture in the Age of Agility and Devops*. Addison Wesley Pub., 21 de mayo de 2021. 336 páginas. ISBN: 0136523560



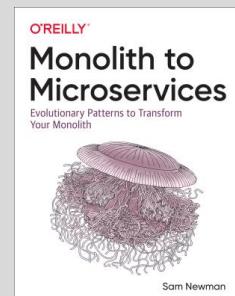
Monolith to Microservices

Contenidos:

- Chapter 1: Just Enough Microservices
- Chapter 2: Planning a migration
- Chapter 3: Splitting the monolith
- Chapter 4: Decomposing the database
- Chapter 5: Growing pains
- Chapter 6: Closing words

Comentarios: Es un libro que presenta cómo llevar a cabo la transformación de arquitecturas tipo monolito a microservicios, justificando las decisiones en aspectos arquitectónicos. El autor había escrito también otro libro titulado *Building Microservices* que ofrece una perspectiva más práctica sobre cómo desarrollar microservicios, que también puede ser recomendable.

Referencia: Sam Newman. *Monolith to Microservices: Sustaining Productivity While Detangling the System*. O'Reilly UK Ltd., 1 de dic. de 2019. ISBN: 1492047848



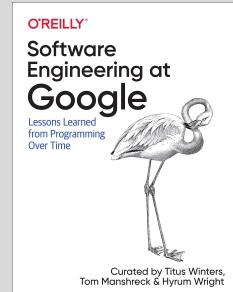
Software Engineering at Google

Contenidos:

- Part I: Thesis. What is Software Engineering
- Part II: Culture. How to work well with teams, Knowledge sharing, Engineering for equality, How to lead a team, leading at scale, measuring engineering productivity
- Part III: Processes. Style guides and rules, Code review, Documentation, testing, deprecation
- Part IV: Tools. Version control and branch management, Code search, build systems and build philosophy, code review, static analysis, dependency management, large scale changes, Continuous Integration, Continuous Delivery, Compute as a service
- Part V: Conclusion

Comentarios: Contiene una visión de la Ingeniería del Software desde la perspectiva de Google con una serie de recomendaciones sobre cómo trabajar en equipos, realizar documentación y aplicar técnicas de ingeniería del software en entornos industriales del tamaño de Google, cuya lectura se considera muy recomendable.

Referencia: Titus Winters, Tom Manshreck e Hyrum Wright. *Software Engineering at Google: Lessons Learned from Programming Over Time*. O'Reilly UK Ltd., 1 de mar. de 2020. 583 páginas. ISBN: 1492082791



Design it!

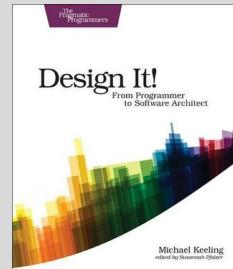
Contenidos:

- Introducing Software Architecture: become a software architect, design thinking fundamentals
- Architecture design fundamentals: devise a design strategy, empathize with stakeholders, Dig for Architecturally Significant Requirements, Choose an Architecture (Before It Chooses You), Create a Foundation with Patterns, Manage Complexity with Meaningful Models, Host an Architecture Design Studio, Visualize Design Decisions, Describe the Architecture, Give the Architecture a Report Card, Empower the Architects on Your Team
- Activities to Understand the Problem, Activities to Explore Potential Solutions, Activities to Make the Design Tangible, Activities to Evaluate Design Options

Comentarios: Se trata de un libro bastante práctico que aporta una visión moderna de la arquitectura del software y de las actividades que un arquitecto del software debe realizar.

La segunda parte del libro contiene una lista de 38 posibles actividades algunas de las cuales son muy interesantes.

Referencia: Michael Keeling. *Design It!* O'Reilly UK Ltd., 1 de nov. de 2017. ISBN: 1680502093



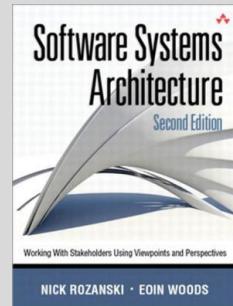
Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives

Contenidos:

- Part I: Architecture fundamentals
- Part II: The process of software architecture
- Part III: A viewpoint catalog
- Part IV: The perspective catalog
- Part V: Putting it all together

Comentarios: Se trata de la segunda edición de un libro del año 2005 que, aunque quizás se ha quedado un poco desfasado, puede ser útil como referencia. Incluye una serie de catálogos de siete puntos de vista (contexto, funcional, información, concurrencia, desarrollo, despliegue y operaciones) y una selección de perspectivas (seguridad, rendimiento, escalabilidad, disponibilidad, etc).

Referencia: Nick Rozanski y Eoin Woods. *Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley Professional, 2011. ISBN: 0132906120



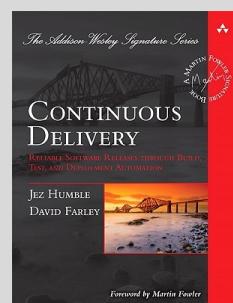
Continuous Delivery

Contenidos:

- Part I: Foundations: the problem of delivering software, configuration management, continuous integration, implementing a testing strategy
- Part II: The Deployment pipeline: anatomy of the deployment pipeline, build and deployment scripting, the commit stage, automated acceptance testing, testing nonfunctional requirements, deploying and releasing applications
- Part III: The Delivery ecosystem: managing infrastructure and environments, managing data, managing components and dependencies, advanced version control, managing continuous delivery

Comentarios: Se trata de uno de los mejores libros que presenta los conceptos de integración y entrega continua, tanto desde una forma teórica como con una justificación práctica. El libro se ha convertido en un clásico en este campo, especialmente en lo que se refiere al concepto de DevOps, y muchos de los conceptos que propone han sido incluidos en varias de las unidades didácticas del curso.

Referencia: David Farley Jez Humble. *Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation*. Addison Wesley, 1 de jul. de 2010. 512 páginas. ISBN: 0321601912



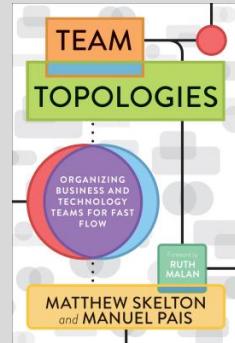
Team topologies

Contenidos:

- Part I. Teams as the Means of Delivery: The Problem with Org Charts, Conway's Law and Why It Matters, Team-First Thinking
- Part II. Team Topologies That Work For Flow: Static Team Topologies, The Four Fundamental Team Topologies, Choose Team-First Boundaries
- Part III. Evolving Team Interactions for Innovation and Rapid Delivery: Team Interaction Modes, Evolve Team Structures with Organizational Sensing

Comentarios: Es un libro que describe las diferentes técnicas de organización de trabajo en equipos. El libro no está enfocado únicamente en la arquitectura de software, pero se considera que su tratamiento de la ley de Conway es muy relevante para esta disciplina, especialmente cuando se presentan los microservicios.

Referencia: Matthew Skelton. *Team topologies : organizing business and technology teams for fast flow*. Portland, OR: IT Revolution, 2019. ISBN: 1942788827



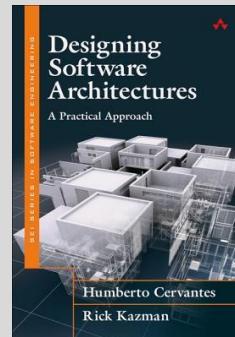
Designing Software Architectures: A Practical Approach

Contenidos:

- 1. Introduction
- 2. Architectural design
- 3. The architecture design process
- 4. Case study: FCAPS system
- 5. Case study: Big data system
- 6. Case study: Bank system
- 7. Other design methods
- 8. Analysis in the Design Process
- 9. The Architecture Design Process in the Organization
- 10. Final words

Comentarios: Es un libro que presenta la variante ADD 3.0, que incluye el concepto de *architectural drivers*. Incluye tres casos prácticos explicados en detalle siguiendo la metodología ADD 3.0 que son interesantes, aunque corren el riesgo de quedar un tanto desfasados.

Referencia: Rick Kazman Humberto Cervantes. *Designing Software Architectures: A Practical Approach*. Addison Wesley Publishing Company, 13 de mayo de 2016. 320 páginas. ISBN: 0134390784



Software architecture for developers

Contenidos:

- Volume 1. Part 1: Software architecture
- Volume 1. Part 2: The software architecture role
- Volume 1. Part 3: The process of architecting
- Volume 2. Part 1: Visualise
- Volume 2. Part 2: Document
- Volume 2. Part 3: Tooling: Sketches, diagrams, models and tooling

Comentarios: Es un libro publicado de forma independiente por el autor Simon Brown, que ofrece una visión de la arquitectura del software compatible con el desarrollo ágil. Se divide en 2 volúmenes, en el primer volumen describe conceptos básicos de la arquitectura del software, del rol del arquitecto del software y del proceso de creación de una arquitectura. En la segundo volumen describe técnicas para comunicar, visualizar y documentar arquitecturas del software. En el libro se propone el modelo C4, que aunque no es el elegido para la asignatura, sí está adquiriendo cierta importancia en los últimos años y podría utilizarse en próximas ediciones de la misma. La página web del libro <https://softwarearchitecturefordevelopers.com/> incluye material audiovisual como presentaciones y vídeos que también es interesante.

Referencia: Simon Brown. *Software architecture for developers*. Volumen 1 y 2. LeanPub, 2012. URL: <https://softwarearchitecturefordevelopers.com/>

Software Architecture
for DevelopersTechnical leadership and
the balance with agility

Simon Brown

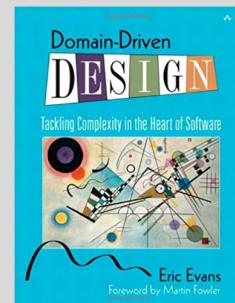
Software Architecture
for DevelopersVisualise, document and explore
your software architecture

Simon Brown

Domain-Driven Design: Tackling Complexity in the Heart of Software

Contenidos:

- Part I: Putting the Domain Model to Work
 - 1. Crunching knowledge
 - 2. Communication and the use of Language
 - 3. Binding Model and Implementation
- Part II: The Building Blocks of Model-Driven Design
 - 4. Isolating the Domain
 - 5. A Model Expressed in Software
 - 6. The Life-cycle of a Domain Object
 - 7. Using the language: an Extended Example
- Part III: Refactoring toward Deeper Insight
 - 8. Breakthrough
 - 9. Making Implicit Concepts Explicit
 - 10. Supple Design
 - 11. Applying Analysis Patterns
 - 12. Relating Design Patterns to the Model
 - 13. Refactoring toward Deeper Insight
- Part IV: Strategic Design
 - 14. Maintaining Model Integrity
 - 15. Distillation
 - 16. Large-Scale structure
 - 17. Bringing the Strategy together



Comentarios: Este libro del año 2003 se ha convertido en un clásico al introducir el concepto de *Domain-driven design*. Algunas de las ideas del libro han ido adquiriendo cada vez más importancia en el planteamiento actual de la arquitectura del software, como la creación de límites de contexto para los microservicios.

Referencia: Eric J. Evans. *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison Wesley, 1 de ago. de 2003. 529 páginas.
ISBN: 0321125215

17.7 Anexo: Guía docente

A continuación se incluye de la guía docente tal y como ha sido propuesta por el candidato para la asignatura de arquitectura del software en el curso 2021-22.



Guía docente

1. Identificación de la asignatura

NOMBRE	Arquitectura del Software		CÓDIGO	GIISOF01-3-008
TITULACIÓN	Grado en Ingeniería Informática del Software		CENTRO	Escuela de Ingeniería Informática
TIPO	Obligatoria	Nº TOTAL DE CRÉDITOS	6	
PERÍODO	Semestral	IDIOMA	Español	
COORDINADOR/ES		TELÉFONO /EMAIL		UBICACIÓN
Jose Emilio Labra Gayo		labra@uniovi.es /3394		Desp. 206 – 3 ^a Planta – Edif. Fac. Ciencias Oviedo
PROFESORADO		TELÉFONO /EMAIL		UBICACIÓN
Jose Emilio Labra Gayo		labra@uniovi.es /3394		Desp. 206 – 3 ^a Planta – Edif. Fac. Ciencias Oviedo

2. Contextualización (en el caso de asignaturas compartidas se contextualizará, si existen diferencias, para cada una de las titulaciones donde se comparte).

Esta asignatura se enmarca dentro de la materia de **Ingeniería del Software** y de forma transversal al módulo de **Formación Específica (Ingeniería del Software)**. Dentro de esta también están las asignaturas de *Ingeniería del Proceso Software*, *Diseño del Software*, *Calidad, Validación y Verificación del Software*, *Ingeniería de Requisitos* y *Modelado en Ingeniería del Software*.

La asignatura es Obligatoria y se imparte durante el segundo semestre del tercer curso del Grado en Ingeniería Informática del Software.

Cuenta con 6 créditos ECTS, que suponen un total de 150 horas de trabajo; 60 horas presenciales y 90 horas no presenciales.

Organizativamente, la asignatura tendrá 2 horas de clases expositivas a la semana (hasta un total de 21 horas), una hora de seminario (hasta un total de 7 horas), 2 horas de prácticas de laboratorio (hasta un total de 28 horas) y 2 horas totales de tutorías grupales.

3. Requisitos (en el caso de asignaturas compartidas, si existen diferencias, se señalarán los mismos para cada una de las titulaciones donde se comparte).

Para cursar esta asignatura se recomienda haber superado los objetivos de aprendizaje establecidos en las asignaturas *Ingeniería del Proceso Software* y *Diseño de Software* de primer semestre de tercer curso.



4. Competencias y resultados de aprendizaje (en el caso de asignaturas compartidas, si existen diferencias, se señalarán los mismos para cada una de las titulaciones donde se comparte).

De acuerdo con la Memoria de Verificación del título de Grado en Ingeniería del Software de la Universidad de Oviedo, las competencias generales que el alumno debe adquirir al cursar la asignatura de Arquitectura del Software son las siguientes (la primera columna de las tablas indica la notación de cada competencia en la Memoria de Verificación del Título):

Competencias Generales Instrumentales: Habilidades Metodológicas

CG-1	Competencia para el diseño de soluciones a problemas complejos humanos
------	--

Competencias Específicas – Común a la rama de Informática

Com.1	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.
Com.8	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
Com.11	Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

Competencias Específicas – Tecnología Específica Ingeniería del Software

ISW.1	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.
ISW.3	Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles
ISW.4	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.

Los resultados de aprendizaje que el alumno alcanzará tras cursar la asignatura según la Memoria de Verificación son los siguientes (la primera columna de la tabla indica la notación de cada resultado de aprendizaje en la Memoria de Verificación del Título):

RA.IS-1.	Realizar Proyectos de Ingeniería del Software complejos que den solución a problemas reales y solucionarlos mediante técnicas y tecnologías relacionadas con los procesos de fabricación de software incluyendo frameworks, patrones arquitectónicos, patrones de diseño y de integración persiguiendo el desarrollo de software de calidad [Com1], [Com2], [Com.8], [Com10], [Com13], [ISw.1], [ISw.4], [ISw.6], [Bas5] , [CG1], [CG3],[CG4],[CG5],[CG10], [CG22]
----------	--



RA.IS-3.	Aplicar distintas técnicas de construcción de software en el diseño de bajo nivel [Com.8], [ISw.1], [ISw.4], [CG1]
RA.IS-4.	Desarrollar diseños y programación orientados a objetos con un elevado nivel de competencia [Com.8], [ISw.4], [CG1], [CG4]
RA.IS-5.	Evolucionar y refactorizar diseños existentes ante cambios en los requisitos [ISw.1], [ISw.6]
RA.IS-6.	Determinar el grado de mantenibilidad, fiabilidad y eficiencia de diseños software [Com.8], [ISw.1], [ISw.4], [CG1]
RA.IS-7	Diseñar e implementar software aplicando diferentes tecnologías middleware [Com11], [ISw3]
RA.IS-9	Diseñar y llevar a cabo controles e inspecciones eficientes y efectivas de validación, verificación y calidad, planes de pruebas [Com1], [Com16], [ISw1], [ISw4], [CG2], [CG4], [CG11], [CG28].
RA.IS-10	Analizar estadísticamente la densidad de defectos y probabilidad de fallos [Com16], [ISw4], [ISw5], [CG25]
RA.IS-11	Evaluar la calidad de un proceso de software desde el punto de vista de la calidad del producto [Com1], [Com18], [ISw1], [CG2], [CG11], [CG28].

5. Contenidos.

1. Definición y conceptos básicos de Arquitectura de Software
 - a. Definiciones de Arquitectura de Software
 - b. Papel del arquitecto del Software: roles y personas interesadas
 - c. Documentación de arquitectura del Software
 - d. Atributos de calidad y conceptos de diseño
2. Taxonomías de Arquitectura del Software
 - a. Tácticas, estilos y patrones arquitectónicos
 - b. Construcción y mantenimiento
 - c. Modularidad
 - d. Comportamiento: componentes y conectores
 - e. Integración
 - f. Disposición: Despliegue y distribución
 - g. Negocio y entorno empresarial

6. Metodología y plan de trabajo.

El reparto de horas de trabajo de cada modalidad por temas y en global se refleja en las siguientes tablas:

TRABAJO PRESENCIAL	TRABAJO NO PRESENCIAL



Temas	Horas totales	Clase Expositiva	Prácticas de aula /Seminarios/ Talleres	Prácticas de laboratorio /campo /aula de informática/ aula de idiomas	Prácticas clínicas hospitalarias	Tutorías grupales	Prácticas Externas	Sesiones de Evaluación	Total	Trabajo grupo	Trabajo autónomo	Total
1	40	7	2	4		1		1	15	15	10	25
2	110	14	5	24		1		1	45	45	8	65
Total	150	21	7	28	0	2	0	2	60	60	30	90

Siguiendo la filosofía de los créditos europeos, en la asignatura se plantearán actividades presenciales y no presenciales en las que se llevará a cabo un seguimiento por parte de los profesores de la asignatura.

Las actividades presenciales seguirán cinco modalidades:

- Clases expositivas, en donde se plantearán los fundamentos de la materia y se guiará al alumno para su trabajo autónomo.
- Prácticas de aula/Seminarios, se plantea un aprendizaje activo y colaborativo integrando el trabajo en clase con el trabajo que se realice en el campus virtual. En los seminarios, los estudiantes podrán realizar presentaciones de diversos temas seleccionados por los profesores y que serán evaluadas.
- Prácticas de laboratorio, donde se realizarán diferentes proyectos en los que se diseñará e implementarán algoritmos para dar solución a diferentes problemas propuestos. Se realizarán tanto proyectos individuales como en grupo, estos proyectos requerirán del trabajo no presencial de los alumnos.
- Tutorías grupales, se realizará un seguimiento de los alumnos para detectar lagunas y orientarles para solucionarlas.
- Sesiones de evaluación, se realizarán exámenes tanto teóricos como prácticos con ordenador, para evaluar los conocimientos de los alumnos.

MODALIDADES		Horas	%	Totales
Presencial	Clases Expositivas	21	14,00%	60 Horas
	Práctica de aula / Seminarios / Talleres	7	4,67%	
	Prácticas de laboratorio / campo / aula de informática / aula de idiomas	28	18,67%	
	Prácticas clínicas hospitalarias	0		
	Tutorías grupales	2	1,33%	
	Prácticas Externas	0		
	Sesiones de evaluación	2	1,33%	
No presencial	Trabajo en Grupo	60	40,00%	90 horas
	Trabajo Individual	30	20,00%	
Total		150		



De forma excepcional, si las condiciones sanitarias lo requieren, se podrán incluir actividades de docencia no presencial. En cuyo caso, se informará al estudiantado de los cambios efectuados

7. Evaluación del aprendizaje de los estudiantes.

Evaluación continua

La evaluación continua contempla los siguientes aspectos:

1. **Nota de Teoría:** la nota se compondrá a partir de un examen final teórico obligatorio y de diversas pruebas parciales que se realizarán a lo largo del curso. Las pruebas parciales pueden incluir presentaciones de diversos temas al resto de estudiantes en los seminarios.

La nota teórica será: **70% Nota examen + 30% Nota pruebas parciales**

2. **Nota de Laboratorio:** El estudiante realizará proyectos en grupo y proyectos individuales.
La nota práctica será: **70% nota grupo + 30% nota individual.**

Para superar este proceso de evaluación continua se requiere:

- Asistencia mínima del 80% a las clases de teoría, seminarios y prácticas de laboratorio.
- Nota mínima de teoría y prácticas tanto grupo como individual superior a 5
- Asistencia a la jornada de presentación de la práctica

La nota se calculará como:

$$\text{Nota} = \text{Nota Teoría} * 0.40 + \text{Nota Prácticas} * 0.60$$

Los estudiantes que no superen el 5 en teoría o prácticas serán calificados como Suspensos. La calificación en este caso se obtiene mediante:

$$\text{Nota final Suspensa} = \text{Mínimo}(4, \text{Nota})$$

Evaluación extraordinaria

En caso de no superarse la asignatura por evaluación continua, no se podrá aprobar la misma en la convocatoria ordinaria. En las convocatorias extraordinarias se realizará un examen (40% de la nota final) y un proyecto individual (60% de la nota final) similar a las actividades individuales o en grupo que hayan propuesto a lo largo del semestre.

Se requiere que:

- Nota mínima examen y proyecto individual iguales o superiores a 5

Si cumplen los requisitos anteriores, la nota se calculará como:

$$\text{Nota} = \text{Nota prueba teórica} * 0.40 + \text{Nota del proyecto individual} * 0.60$$

Norma general:

Las evaluaciones no realizadas o no entregadas por inasistencia entrarán a contabilizar en el cómputo de la nota correspondiente con un cero.



Evaluación diferenciada

En el caso de teoría, el estudiante realizará las pruebas teóricas igual que el resto de estudiantes puesto que la mayoría de dichas pruebas pueden realizarse de forma no presencial. En el caso de presentaciones de trabajos de seminarios, los estudiantes deberán al menos realizar la presentación de su trabajo el día que les corresponda.

En el caso de los trabajos de laboratorio, si el estudiante puede adaptarse al trabajo en grupo con el resto de los compañeros, entonces se evaluará de la misma forma y solamente tendrá que acudir al 20% de las sesiones presenciales. En caso de no poder adaptarse, deberá ponerlo en conocimiento de los profesores antes de que finalice el primer mes de docencia de la asignatura para que se le comunique un proyecto individual de prácticas. Se requiere que:

- Nota mínima examen y proyecto individual sea igual o superior a 5

La fórmula de cálculo de la nota final es:

$$\text{Nota} = \text{Nota prueba teórica} * 0.40 + \text{Nota del proyecto individual} * 0.60$$

De forma excepcional, si las condiciones sanitarias lo requieren, se podrán incluir métodos de evaluación no presencial. En cuyo caso, se informará al estudiantado de los cambios efectuados

8. Recursos, bibliografía y documentación complementaria.

[Bass et al, 03] *Software Architecture in Practice*, Len Bass, P. Clements, R. Kazman, Addison-Wesley, 3^a Edición, 2012

[Buschmann et al, 96] *Pattern-Oriented Software Architecture Volume 1: A System of Patterns*, Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, Michael Stal, Ed. Wiley, 1996

[Buschmann et al 07a] *Pattern-Oriented Software Architecture Volume 4: A Pattern Language for Distributed Computing*, Frank Buschmann, Kevlin Henney, Douglas C. Schmidt, Ed. Wiley, 2007

[Buschmann et al 07b] *Pattern Oriented Software Architecture Volume 5: On Patterns and Pattern Languages*, Frank Buschmann, Kevlin Henney, Douglas C. Schmidt, Ed. Wiley, 2007

[Cervantes, Kazman, 16] *Designing Software Architectures*, Humberto Cervantes, Rick Kazman, Addison-Wesley, 2016

[Eeles, Cripps, 10] *The Process of Software Architecting*, Peter Eeles, Peter Cripps, Ed. Addison-Wesley, 2010

[Erl, 09] *SOA Design Patterns*, T. Erl, Prentice Hall, 2009

[Erl, 05] Software Oriented Architecture: Concepts, Technology and Design, T. Erl, Prentice Hall, 2005

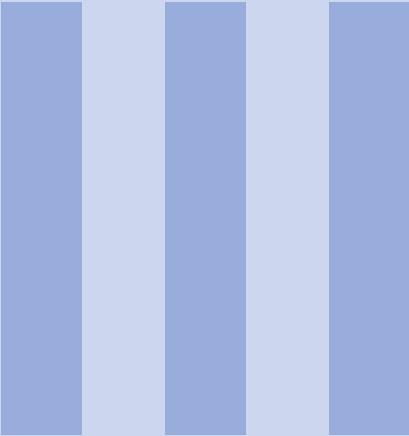
[Fairbanks, 12] *Just Enough Software Architecture*, George Fairbanks, Ed. Marshall & Brainerd, 2012 (3^a Ed.)

[Ford et al, 17] *Building evolutionary architectures*, N. Ford, R. Parsons, P. Kua, Ed. O'Reilly, 2017



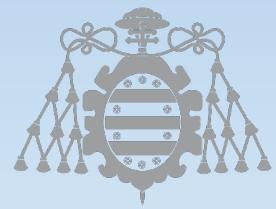
- [Fowler, 02] *Patterns of Enterprise Application Architecture*, Martin Fowler, Ed. Addison-Wesley, 2002
- [Humble, 10] *Continuous delivery*, J. Humble, Addison-Wesley, 2010
- [Hohpe, 03] *Enterprise Integration Patterns: Designing, Building and Deploying Messaging Solutions*, G. Hohpe, B. Woolf, Addison-Wesley, 2003
- [Keeling, 04] *Design it!, from programmer to software architect*, Michael Keeling, *The pragmatic programmers*, Pragmatic Bookshelf, 2017
- [Kleppmann, 17] *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*, Martin Kleppmann, O'Reilly, 2017
- [Kircher et al, 04] *Pattern-Oriented Software Architecture Volume 3: Patterns for Resource Management*, Michael Kircher, Prashant Jain, Ed. Wiley, 2004
- [Knoernschild, 12] *Java Application Architecture: Modularity Patterns with Examples Using OSGi (Agile Software Development)*, K. Knoernschild, Prentice Hall, 2012
- [Martin, 17] *Clean Architecture*, Robert C. Martin, Prentice-Hall, 2017
- [Newman, 15] *Building microservices*, Sam Newman, O'Reilly, 2015
- [Nygaard, 18] *Release It!*, Second edition, The pragmatic programmer, 2018
- [Rozanski et al, 11] *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives* (Ed. 2), Nick Rozanski, Eóin Woods, Addison-Wesley, 2011
- [Schmidt et al, 00] *Pattern-Oriented Software Architecture Volume 2: Patterns for Concurrent and Networked Objects*, Douglas Schmidt, Michael Stal, Hans Rohnert, Frank Buschmann, Ed. Wiley, 2000
- [Taylor et al, 00] *Software Architecture: Foundations, Theory and Practice.*, R. N. Taylor, N. Medvidovic, E. Dashofy, Ed. Wiley, 2009

Otros recursos, bibliografía y documentación complementaria se localizarán en la Página Web de la asignatura, dentro del Campus Virtual de la Universidad de Oviedo.



Proyecto Investigador

18	Contexto del proyecto	213
19	Proyecto ANGLIRU – PID2020-117912RB-C21	217
20	Referencias bibliográficas	271
	Índice	277



Universidad de Oviedo

18. Contexto del proyecto

18.1 Introducción

El proyecto de investigación que se presenta en este documento se enmarca en el ámbito de la utilización de datos enlazados y grafos de conocimiento para la integración y reutilización de datos.

Se trata de un proyecto de investigación presentado en la convocatoria del Plan Nacional de Investigación de 2020, que obtuvo financiación y se considera que puede ser el mejor representante de las líneas de investigación que serán realizadas por el candidato en los próximos años.

18.2 Marco de la investigación

La investigación que se propone se realizará en el grupo WESO (WEb Semantics Oviedo), fundado en el año 2004 por el candidato y del cual actúa como coordinador/investigador principal desde dicho año. WESO (Web Semantics Oviedo)¹ es un grupo de investigación que tiene como objetivo el avance de las tecnologías semánticas y su aplicación práctica a diferentes dominios. El grupo WESO está asociado al área de Lenguajes y Sistemas Informáticos del Departamento de Informática, que ha sido descrito en 14.4.1.

El grupo se creó a partir de una colaboración con la Oficina Española del W3C y la Fundación CTIC² para desarrollar un proyecto de aplicación de tecnologías semánticas para el acercamiento de la administración pública al ciudadano. Dicho proyecto tenía como objetivo la publicación semántica del Boletín Oficial del Principado de Asturias y fue una de las primeras aplicaciones de Web Semántica al ámbito legislativo. Aunque el grupo WESO nació en el seno del Departamento de Informática, se considera un grupo multidisciplinar al colaborar e incluso incorporar personas de otras áreas de conocimiento desde sus inicios. De hecho, en el proyecto mencionado se colaboraba con Enrique del Teso, del Departamento de Filología Española. Posteriormente, se incorpora al grupo Patricia Ordóñez de Pablos, del Departamento de Administración de Empresas, y más adelante también se realizarán colaboraciones con Cristina Valdés, del Departamento de Filología Inglesa.

Gracias a la experiencia adquirida en el ámbito legislativo, en el año 2009 comienza una colaboración con la empresa Gateway S.C.S³ que se plasmará en el proyecto Avanza *Prototipo de plataforma pan-europea de servicios de información para PYMES en licitación pública internacional (10DERS INFORMATION*

¹<http://www.weso.es/>

²<https://www.fundacionctic.org/>

³<http://gateway-scs.es/en/>

SERVICES), y en el año 2011 se inicia otra colaboración con la Biblioteca Nacional del Congreso de Chile para el proyecto *Historia de la Ley* que pretende modelar las leyes chilenas mediante datos enlazados.

Durante la etapa del año 2010-12 se colabora con Cristina Valdés en la Red Temática Internacional sobre *Multilingual Web* y se lidera el grupo comunitario W3C *Best Practices on Multilingual Linked Open Data*.

En el año 2012 se obtuvo la evaluación positiva por parte de la ANEP como Grupo Oficial de Investigación. Los 3 miembros doctores que formaban parte del grupo en aquella solicitud eran: Daniel Gayo Avello, Patricia Ordóñez de Pablos y Jose Emilio Labra, todos ellos se mantienen en el grupo hasta la fecha. Desde el año 2012-13, el doctor Aquilino Adolfo Juan Fuente comparte la asignatura Arquitectura del Software con el Dr. Jose Emilio Labra Gayo y comienzan a colaborar en diversas investigaciones que dan lugar a varias publicaciones conjuntas y a la incorporación de Aquilino Adolfo al grupo.

Durante el año 2012, el grupo WESO realiza diversas colaboraciones con la empresa Treelogic⁴ que se plasman en varios contratos de investigación. En ese año también comienza una colaboración con la Web Foundation⁵ para el proyecto WebIndex⁶. Dicho proyecto consistía en la representación de datos estadísticos de países mediante datos enlazados y su presentación fue realizada por Tim Berners-Lee. A partir de dicho proyecto, en el año 2013 se realiza un portal de datos para la IFAD *Fondo Internacional para el Desarrollo de la Agricultura* de las Naciones Unidas que se denominará LandPortal⁷.

Todas estas colaboraciones permiten realizar varias contrataciones y becas de formación para estudiantes, principalmente del Máster de Ingeniería Web, que permiten la adquisición y generación de conocimiento, y asentar aspectos de investigación. En el año 2012 se defiende la primer tesis doctoral realizada en el ámbito del grupo WESO y del año 2012 al 2014 el Ministerio de Ciencia e Innovación financió el proyecto ROCAS (*ReasOning in the Cloud by Applying Semantics*) que permite al grupo trabajar en aspectos más teóricos de la Web Semántica. En esa misma época, también se obtuvo la subvención del proyecto Avanza *TagFlow, sistema Avanzado de monitoreo y tagging de Flujos de datos multimedia* en colaboración con la empresa BMAT Licensing S.L. que permite trabajar en algoritmos de recomendación de material musical.

La experiencia práctica adquirida en la creación de portales de datos con RDF motivó la participación en el *Validating RDF Workshop*⁸ organizado por el W3C. En dicho Workshop, Jose Emilio Labra Gayo entra en contacto con Eric Prud'hommeaux y Harold Solbrig, y comienzan a colaborar en el desarrollo del lenguaje Shape Expressions (ShEx), que puede utilizarse para validar datos RDF. Tras el Workshop, se crea el grupo de trabajo *Data Shapes* de W3C, en el cual Jose Emilio Labra Gayo también participa y que dará lugar al lenguaje SHACL (Shapes Constraint Language) que será recomendación W3C en el año 2017.

En el año 2017 se concedió el proyecto del plan Nacional VARSSHA (*Validating RDF Streams using Shapes*) que tiene como objetivo aplicar lenguajes de validación de datos RDF como ShEx ó SHACL a grandes cantidades de datos.

En el año 2018 obtuvo la renovación de la evaluación positiva por parte del Ministerio de Ciencia, Innovación y Universidades con el código GR-2011-0067.

En el año 2019, Wikidata, el grafo de conocimiento que da soporte a Wikipedia, decide adoptar el lenguaje ShEx para la definición de esquemas de entidades, que puedan utilizarse para la validación de sus datos. Desde entonces, el grupo WESO ha realizado diversas colaboraciones con la Fundación Wikimedia o proyectos relacionados.

Por otro lado, durante estos años el grupo se fue consolidando con nuevas colaboraciones con las empresas Zapiens⁹ e Izertis¹⁰. Uno de los proyectos más relevantes ha sido la participación en la primera fase del proyecto Hércules de la Universidad de Murcia¹¹ consistente en la *Arquitectura Semántica e Infraestructura Ontológica* para un sistema de gestión de la investigación para las universidades españolas. Con este proyecto, el grupo WESO ha adquirido conocimiento sobre el dominio de la gestión de la investigación.

En el verano de 2020, Roberto García, profesor Titular de la Universidad de Lleida¹² realizó una estancia de investigación en el grupo WESO, a partir de la cual comenzó una colaboración entre el grupo WESO y el grupo de investigación de Roberto.

⁴<https://www.treelogic.com/>

⁵<https://webfoundation.org/>

⁶<https://thewebindex.org/>

⁷<https://landportal.org/>

⁸<https://www.w3.org/2012/12/rdf-val/>

⁹<https://zapiens.ai/>

¹⁰<https://www.izertis.com/>

¹¹<https://www.um.es/web/hercules/>

¹²<https://rhizomik.net/~roberto>

De esta forma, cuando apareció la convocatoria de 2020 del programa Retos de la Sociedad, Jose Emilio Labra y Roberto García decidieron plantear un proyecto coordinado con ambas instituciones que podría tener como temática la aplicación de grafos de conocimiento a la gestión de datos de investigación. Este dominio es de interés a ambos, al primero por la experiencia reciente en el proyecto Hércules, y al segundo, porque ocupa actualmente el cargo de Adjunto del Vicerrector de Investigación y Transferencia.

Además de los investigadores doctores, el grupo cuenta actualmente con 1 estudiante de doctorado contratados, 8 colaboradores externos y 2 investigadores contratados, aunque dicho número puede variar dependiendo de los proyectos o investigaciones que se estén llevando a cabo.

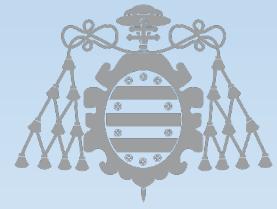
18.2.1 Actividades y cooperación con otros grupos de investigación e instituciones

El grupo WESO ha tenido desde sus inicios un afán de colaboración y cooperación con otros grupos de investigación e instituciones. A continuación se indican las principales empresas, instituciones y grupos de investigación con los que se han realizado colaboraciones:

- World Wide Web Consortium. Participación en los grupos de trabajo *Data Shapes* y *Web Of Things*. Participación en los grupos comunitarios *Shape Expressions* y *SHACL*. Chairman del *Best Practices on Multilingual Linked Open Data* W3C community group.
- Web Foundation. Participación en los proyecto *WebIndex*, *OpenData Barometer*, *A4AI – Alliance for Affordable Internet*.
- Biblioteca Nacional del Congreso de Chile. Participación y colaboración con el proyecto *Historia de la Ley*
- Empresa Micelio BVBA: Colaboración en proyecto para *Wikibase-SAF realisation phase in Luxembourg*
- Scholia: Colaboración en proyecto *Robustifying Scholia: paving the way for knowledge discovery and research assessment through Wikidata* a través de la Universidad de Virginia, financiado por la *Sloan Foundation*.
- Wikimedia Deutschland: Participación en eventos relacionados con la aplicación de Shape Expressions para la validación de datos.
- Universidad Autónoma San Luis de Potosí: Colaboración con el profesor Edgar Castillo Barrera en la utilización de tecnologías semánticas para el desarrollo basado en componentes.
- Empresa Professional Wiki: Colaboración en asesoramiento sobre tecnologías Wikibase para la Biblioteca Nacional de Alemania.
- Asociación Salud sin Bulos: Participación en dos últimos congresos y propuesta de colaboración para proyecto ANGLIRU sobre detección de datos incorrectos sobre salud.
- Helsinki Centre for Digital Humanities: Colaboración con Hyvönen Eero en portales de datos. Fruto de esta colaboración, Jose Emilio Labra Gayo ha sido invitado como tribunal de la tesis doctoral de Mikko Koho titulada: *Representing, Using and Maintaining Military Historical Linked Data on the Semantic Web*.
- INRIA Lille, colaboración con Iovka Boneva en el desarrollo del lenguaje ShEx. Como fruto de dicha colaboración el doctorando Herminio García González realizó una estancia de investigación en dicho grupo.
- Agile Knowledge Engineering and Semantic Web – AKSW de la Universidad de Leipzig. Colaboración con Sebastian Hellman en la investigación sobre grafos de conocimiento. Como fruto de dicha colaboración, el doctorando Daniel Fernández Álvarez realizó una estancia de investigación en dicho grupo.
- Grupo IDLab (Internet Technology and Data Science Lab) de la Universidad de Gante, Bélgica. Se mantiene una colaboración con Ruben Verborgh y Anastasia Dimou relacionada con la validación de la calidad de datos enlazados y la aplicación de tecnologías semánticas en el proyecto SOLID. En el año 2020 se había planificado la realización de una estancia de investigación del doctorando Paulino Álvarez de Ron Ondina en dicho laboratorio. La estancia fue aplazada debido a la situación de pandemia.
- Ministerio de Cultura del Gobierno de Luxemburgo. Colaboración en el proyecto SAF (*Shared Authority File*).
- Izertis: Colaboración en proyecto Hercules, LODLearning y WeDoctor
- Zapiens: Colaboración en proyecto para generación de preguntas y respuestas a partir de grafos de conocimiento.
- Empathy: Colaboración en proyecto para la creación de un prototipo de buscador sobre tecnologías

SOLID

- BMat Licensing. Participación en proyecto Avanza: *TagFlow, sistema Avanzado de monitoreo y tagging de Flujos de datos multimedia*
- Treelogic: Colaboración en proyecto *RETINAS – REal TIme video ANalysis for Security applications y Origin – ORganizaciones Inteligentes Globales INnovadoras*
- Gateway S.C.S.: Colaboración en proyecto: *Prototipo de plataforma pan-europea de servicios de información para PYMES en licitación pública internacional (10DERS INFORMATION SERVICES)*
- Fundación CTIC: Colaboración en proyectos para *Desarrollo de un Modelo de Servicios de Información para la Administración basado en la Web Semántica e Integración de web semántica.*



Universidad de Oviedo

19. Proyecto ANGLIRU – PID2020-117912RB-C21

El proyecto ANGLIRU (Applying kNowledge Graphs to research data interoperabiLlity and ReUsability) fue presentado en la convocatoria 2020 del Programa Estatal de I+D+i Orientada a los Retos de la Sociedad, Ministerio de Ciencia e Innovación.

Se trata de un proyecto coordinado en el que el Coordinador e Investigador Principal del Subproyecto 1 es el candidato, Jose Emilio Labra Gayo, y el Investigador Principal del subproyecto 2 es Roberto García González, de la Universidad de Lleida.

El proyecto ANGLIRU, de código PID2020-117912RB-C21, obtuvo financiación según la resolución de la concesión publicada el 28 de Septiembre de 2021¹.

A continuación se presenta la memoria científico-técnica del proyecto, tal y como se presentó. En la sección 19.2 se incluye la resolución de la concesión del mismo y en la sección 19.3 se incluye el informe de valoración obtenido.

19.1 Memoria del proyecto

A continuación se incluye la memoria del proyecto, tal y como se entregó, siguiendo por tanto el formato y la plantilla establecidas en la convocatoria. La memoria se presentó en idioma inglés debido a que en los proyectos coordinados se requería la utilización de dicho idioma.

¹http://www.aei.gob.es/stfls/MICINN/Ayudas/PE_2017_2020/PE_Orientada_Retos_Sociedad/FICHEROS/Proyectos_IDI_Retos_Investigacion/Resolucion_de_concesion_PID2020_fda.pdf

Convocatoria 2020 «Proyectos de I+D+i»

TÍTULO DEL PROYECTO COORDINADO (ACRÓNIMO): AplicaNdo Grafos de conocimiento para la interoperabiLIdad y ReUtilización de datos de investigación (ANGLIRU)

TITLE OF THE COORDINATED PROJECT (ACRONYM): Applying kNowledge Graphs to research data interoperabiLity and ReUsability (ANGLIRU)

SUBPROJECTS DATA

SUBPROJECT 1:

IP 1 COORDINADOR 1: Jose Emilio Labra Gayo

TITLE: Applying kNowledge Graphs to research data interoperabiLity (ANGLI)

SUBPROJECT 2:

IP 1: Roberto García González

TITLE: Applying kNowledge Graphs to research data ReUsability (ANGRU)

1. JUSTIFICATION OF COORDINATION

The ANGLIRU project has a set of ambitious goals that are better addressed by combining the expertise of both the Universidad de Oviedo and Universitat de Lleida teams. Each team has focused past research, development and innovation activities, detailed in Section2, in different parts of what has been identified as the main contributions of the project. One aspect is to improve research data interoperability and the other is its reusability. This has motivated dividing the project into 2 main subprojects:

- ANGLI, led by the WESO research team from Universidad de Oviedo, which focuses mainly on objectives related to interoperability. The WESO team has a lot of experience in the development of linked data portals and RDF data quality and validation, which in the last years has been applied to large knowledge graphs like Wikidata and in different domains like e-government, e-health, life sciences and bioinformatics. A lot of research developed by the WESO research group has indeed been applied to tackle the interoperability problem of RDF and knowledge graphs, and more recently they have participated in the HERCULES project to develop a research management system based on semantic technologies. This is materialised through its leadership in the work package devoted to interoperability and the "health and bioinformatics" validation use case.
- ANGRU, led by the GRIHO research team from Universitat de Lleida, which focuses mainly on facilitating data reusability. In this case, the GRIHO research group will contribute its experience leading the reusability work package, with the Rhizomer knowledge graph explorer or the application of semantic and blockchain technologies for reuse licenses representation and management. Moreover, GRIHO has participated in many projects in the agriculture and sustainability fields, which will facilitate applying project results through the Agriculture and Green Deal use case.

In addition to these complementary skills, both teams have ample experience in data management through knowledge graphs. This will facilitate their collaboration and coordination, which is materialised by the fact that all teams will participate in all work packages. Moreover, there will be a task dedicated to project management in the work plan, which will be led by a representation of both teams. In this task, teams will be organised and coordination risks will be tracked, including a contingency plan to take action when coordination issues are detected.

2. SCIENTIFIC PROPOSAL

The main challenge addressed by the project, from the list of challenges considered by the call, is **Challenge 7-Digital economy, society and culture**. The main objectives of the project are related to facilitating the construction of rich shared data spaces from research data, contributing thus to digitalisation and the application of data to solve societal and economic problems. In addition to contributing to creating an infrastructure that facilitates this construction, the project will also validate its proposal through the application of its results to specific use cases in the sustainable agriculture and bioinformatics for health domains. The project also envisions that its results will facilitate that scientific research results reach society, for instance, through the health use case and its application to fighting fake news. Consequently, it is also considered relevant as a target sub-challenge call's **Challenge 6-Social sciences and humanities, science with and for the society**.

One of the main contributions to date addressing the importance of creating data spaces come from research data management. They stress the importance of making research data available and reusable, especially in the scientific domain. This proposal was materialised in 2016 through the publication of the FAIR Guiding Principles for scientific data management and stewardship (Wilkinson et al., 2016). This paper sets the foundations for the FAIR principles, which have been adopted as the foundation to improve the infrastructures supporting the reuse of scholarly data. They have gained great relevance in the context of research data management and adoption by many organisations worldwide, like the European Open Science Cloud¹ or the Research Data Alliance².

Many tools have been developed to put these principles into practice, like Dataverse³, Zenodo⁴ or subject-specific repositories like Dryad⁵ or others available from www.re3data.org⁶. Though big collections of data have been made available thanks to these initiatives, there is still a long path to go to make them more than collections of data files, unrelated even if about the same topic or with limited reuse conditions alternatives. For instance, as shown in the assessment of Dataverse⁷, Dryad and Zenodo (Wilkinson et al., 2018), there is not full support for any of the interoperability principles from any of these tools, as shown in **Figure 1**. Moreover, the support they provide to reusability is usually limited to a set of predefined licenses, which limit the willingness of sharing data by providers that are not ready to make their data open and require the inclusion of specific reuse conditions.

¹ <https://www.eosc-portal.eu>

² <https://www.rd-alliance.org>

³ <https://dataverse.org/>

⁴ <https://zenodo.org/>

⁵ <https://datadryad.org/>

⁶ <https://www.re3data.org/>

⁷ <https://scholar.harvard.edu/files/mercecrosas/files/fair-dataverse-tromso.pdf>

FM [AID*]	Question	Dataverse Q'aire	Dataverse Optimized	Dryad Q'aire	Dryad Optimized	Zenodo Q'aire	Zenodo Optimized
Identifier type	1	DOI	DOI	DOI	DOI	DOI	DOI
F1A	2	Yellow		Yellow		Yellow	
F1B	Not tested in Q'aire						
F2A	4A						
F2A	4B			Red		Red	
F3	5B	Red	Red	Red			
F4	6A	Red		Red		Red	Red
F4	6B	Red		Red		Red	Red
A1.1	7A						
A1.2	8A						
A1.2	8B	N/A	N/A	N/A	N/A	N/A	N/A
A2	9						
I1	10	Yellow	Yellow	Red	Red	Yellow	Yellow
I2	11	Yellow	Yellow	Yellow	Yellow	Red	Red
I3	12	Red	Red	Red	Red	Red	Red
R1.1	13						
R1.2	14A						

Figure 1. Automatic evaluation of FAIR-compliance (Wilkinson et al., 2018). Green, yellow and red indicate full, partial or lack of support for the corresponding guideline based on either questionnaire-based or semi-automated assessment for each evaluated repository.

The lack of more support for the interoperability and reusability principles is especially relevant when considering that it is one of the main emphasis of the FAIR Principles as stated in their seminal paper (Wilkinson et al., 2016): “The FAIR Principles put specific emphasis on enhancing the ability of machines to automatically find and use the data, in addition to supporting its reuse by individuals.”

Hypothesis

We consider that to bring FAIR Principles into their full realisation, it is necessary to go beyond machine-actionable metadata that facilitates findability. The actual data in the dataset should be made also machine-actionable. This can be achieved through the generation of Knowledge Graphs (KGs) (Hogan et al., 20), a directed graph whose node and relation types have domain-specific semantics. KGs allow encoding the knowledge into a form that is human interpretable and amenable to automated analysis and inference.

Richer representations of the dataset data would not just benefit automated integration, but also findability. This way, it is possible to search datasets based on their content, not just their metadata. For instance, datasets containing measures of a particular dimension of a specific biological pathway⁸.

Then, it will be possible to automatically integrate the datasets. For instance, in the case of the datasets about a biological pathway, integrate them independently of the unit of measure used because they are automatically converted.

To ingest the data and provide rich representations that facilitate data integration, it is possible to develop assistants that guide users through the explicit definition of the data

⁸ <https://www.wikipathways.org>

model which can include mechanisms to assess the quality of the integrated data. Some quality mechanisms can be:

- Missing values, especially for fields labelled as mandatory
- Determine data types from most common values to detect inconsistent values (corresponding to other data types)
- Statistical analysis to spot potential outliers
- Compliance checks that verify if the processed data complies with predefined data shapes defined in either ShEx or SHACL
- etc.

Regarding facilitating reusability, it is also possible to represent in a machine-actionable way the licensing terms of the data and include provenance to track data, including blockchain-based mechanisms to provide trust through immutable and easily auditable records of data publishing and reuse events.

We structure the proposed solution as a set of services that extend and integrate existing FAIR platforms like Dataverse or Zenodo. Datasets will go through the existing platforms to satisfy the FAIR guidelines that are already well-covered by existing tools. Moreover, this way, existing data repositories can be enriched from their sources, without disrupting their current functionality and user base.

Previous Work

The vision of the semantic web was proposed in early 2000 by (Berners-lee et al., 01) as a vision for a web of data where machines can automatically process the information published on the web. In 2007, linked data was also proposed as a set of principles⁹ that could improve the adoption of the web of data (Bizer 11). A relevant aspect of linked data is the promotion of standards like RDF to return useful information to machines.

RDF can be seen as the *lingua franca* of the semantic web and linked data, a flexible graph-based data model that improves interoperability by promoting the use of URLs that can also be used as a knowledge representation language on top of which a set of technologies have been developed like RDF Schema¹⁰ and OWL¹¹ to ontologies and SPARQL¹² as a query language. Although RDF's flexibility is very powerful as an integration language, it can also be seen as a drawback for interoperability when the consumers of RDF data don't have an explicit data model available.

In 2012, Google popularized the concept of Knowledge Graphs with the announcement that they were using them to improve their search results¹³ and their adoption by different companies has been increasing in the last years. There are collaborative and global knowledge graphs like Wikidata¹⁴ or DBpedia¹⁵, but also domain-specific and commercial ones (Hogan et al., 20).

⁹ <https://www.w3.org/DesignIssues/LinkedData.html>

¹⁰ <https://www.w3.org/TR/rdf-schema/>

¹¹ <https://www.w3.org/TR/owl2-primer/>

¹² <https://www.w3.org/TR/sparql11-query/>

¹³ <https://www.blog.google/products/search/introducing-knowledge-graph-things-not/>

¹⁴ <https://www.wikidata.org/>

In 2014, Shape Expressions (ShEx) was proposed as a concise and human-readable language to describe and validate RDF data (Prud'hommeaux et al., 14), and in 2017 the W3C data shapes working group proposed SHACL¹⁶ as a recommendation. Since then, the adoption of both shapes (ShEx/SHACL) has increased (Lieber et al., 2020). Two remarkable projects have adopted shapes: Wikidata and SOLID. In the case of Wikidata, shapes are defined in a collaborative way forming an ecosystem of shapes which are interrelated¹⁷. The SOLID project¹⁸ consists of a set of specifications that enable a space where users are in control of their data and privacy, separating data from applications by services to fulfil users needs but respect their privacy. Solid is adopting shapes to achieve interoperability (Verborgh 19).

On the other hand, the FAIR principles proposed in (Wilkinson et al., 2016) have had a big impact on the research community and they are now considered an important aspect of research strategies (Collins et al 18) with some metrics been proposed (Wilkinson et al., 19) as well as guidelines about their implementations (Jacobsen et al., 20). Although FAIR principles are not the same as linked data (Mons et al., 17), they can be combined to improve interoperability (Wilkinson et al., 17).

In the following 2 subsections, we present a short overview of the previous work done by the two teams related to the current proposal.

Previous work by WESO Research team, University of Oviedo

WESO (WEb Semantics Oviedo) was established in 2004 as a research group to apply semantic technologies to practical problems. The group started as a collaboration with the CTIC Foundation for a project that tried to represent legal documents as RDF, being a precursor of what would later be called linked data (Berrueta 06).

Since them, the group was hired by several companies and institutions to help in the generation of linked data portals and knowledge graphs in different domains like the National Library of Congress (Cifuentes 11, Cifuentes 19), the World Wide Web Foundation (Labra et al., 14) or the National Fund for Agricultural Development, United Nations¹⁹, which led to their interest in RDF validation techniques (Labra et al., 13).

The collaboration with Eric Prud'hommeaux and Harold Solbrig led to the development of the Shape Expressions (ShEx) language (Prud'hommeaux 14). In 2014 a W3C data shapes working group was chartered taking as one of its inputs the ShEx language. During the working group, SHACL (Shapes Constraints Language) was defined and became a recommendation in 2017. Jose Emilio Labra Gayo participated in the development of both languages and appears as the editor of the ShEx semantic specification²⁰ and the SHACL test-suite²¹. He published a book on the topic (Labra et al., 18) as well as a book chapter about challenges in RDF validation (Labra et al., 19). Between 2017 and 2020, WESO

¹⁵ <https://wiki.dbpedia.org/>

¹⁶ <https://www.w3.org/TR/shacl/>

¹⁷ https://www.wikidata.org/wiki/Wikidata:Database_reports/EntitySchema_directory

¹⁸ <https://solidproject.org/>

¹⁹ <https://landportal.org/>

²⁰ <http://shex.io/shex-semantics/>

²¹ <https://w3c.github.io/data-shapes/test-suite/>

obtained funding for the Validating RDF Streams using Shapes project where they are working on new algorithms to validate RDF using shapes²².

One of the first applications for ShEx was the validation of clinical trials represented in HL7 FHIR RDF²³ although it was increasingly adopted to describe linked data portals and knowledge graphs. In 2019, Wikidata announced its adoption to describe entity schemas (Thornton et al., 19) which led to further collaborations between the WESO research group and Wikidata. In fact, Jose Emilio Labra Gayo was invited to participate in the Wikibase Summit in New York in 2018²⁴, the Wikicite conference, Berkeley, California, 2018²⁵, Wikimedia-hackathon, Prague, 2019²⁶, and the Wikidata Conference, Germany, 2019²⁷.

Jose Emilio Labra Gayo has been part of the knowledge graphs community and has been invited as the keynote speaker to the 1st International Conference on Knowledge Graphs and Semantic Web, June 2019²⁸ as well as participated in the summer and winter schools associated with that conference. He has also participated in the survey paper on knowledge graphs (Hogan et al., 20).

There has been an increasing interest to create ShEx/SHACL data models for RDF in different domains. More concretely, the health and bioinformatics community has a large number of projects which contain RDF data which in principle should be interoperable but in practice are quite complex to integrate. One solution to this problem is to leverage on open and collaborative knowledge graphs like Wikidata (Waagmeester et al., 20a).

In fact, in the report "Turning FAIR into reality" by the European Commission (Collins et al., 18), Wikidata appears as an example of FAIR data. In 2019, Jose Emilio Labra Gayo was invited to the Japan Biohackathon²⁹ which resulted in the preprint (García et al., 20). In April 2020 during COVID-19, Jose Emilio Labra Gayo participated in the virtual biohackathon COVID-19-bh20³⁰ and during that participation, we created a protocol to add COVID-19 information to Wikidata, which resulted in a publication that has recently been accepted in the BMC Biology journal (Waagmeester et al., 20b). In November, our proposal was also accepted in the Elixir virtual biohackathon and Jose Emilio Labra Gayo led the project 35 about Wikidata subsetting³¹.

Since 2019, WESO research group has been working on the Hercules project³², a project proposed that is trying to develop a University Research Data Semantics system for Spanish universities to develop semantic web technologies that gather new information and integrate multiple nodes with heterogeneous ontologies and vocabularies. The University of Murcia signed an agreement with the Spanish Ministry of Economy, Industry and Competitiveness

²² Project reference: TIN2017-88877-R (RETOS call, 2017)

²³ <https://www.hl7.org/fhir/>

²⁴ https://www.wikidata.org/wiki/Wikidata:WikiProject_Wikidata_for_research/Meetups/2018-09-19-21-New-York

²⁵ https://meta.wikimedia.org/wiki/WikiCite_2018

²⁶ https://www.mediawiki.org/wiki/Wikimedia_Hackathon_2019

²⁷ https://www.wikidata.org/wiki/Wikidata:WikidataCon_2019

²⁸ <http://kgswc.org/>

²⁹ <http://2019.biohackathon.org/participants>

³⁰ <https://github.com/virtual-biohackathons/covid-19-bh20>

³¹ <https://www.biohackathon-europe.org/>

³² <https://www.um.es/web/hercules/proyectos/asio>

(MINECO) in 2017 backing the HERCULES project with an 80% of co-financing from the European Regional Development Fund program (ERDF) within the 2014-2020 period. The purpose of this agreement was to establish collaboration amongst MINECO and the University of Murcia, directed towards the improvement of public services and business innovation through the Public Procurement of Innovation. Some goals of the system were to create a research management system with semantic capabilities and infrastructures and create support systems for the detection of synergies in R&D between universities. The HERCULES project was divided into three main subprojects: semantic architecture and ontological infrastructure, research management system; and data enrichment and methods of analysis. WESO research group in collaboration with Izertis was selected as one of the contractors for the first subproject: semantic architecture and ontological infrastructure. The partial results of this collaboration have been published in (Labra et al., 20)³³. Within the framework of this project, semantic web techniques and technologies are being developed. One of the deliverables of the project was a study about the FAIR metrics, which can be seen in the submitted analysis³⁴ and the proposed metrics³⁵. The collaboration of WESO as part of the HERCULES project led to the participation of Jose Emilio Labra Gayo in the Research Data Alliance³⁶ as part of the working groups: Research Metadata Schemas, FAIR Data Maturity Model and RDA in Spain.

In October 2020, WESO signed a contract with Micelio BV. for a project with the government of Luxembourg where we will create a Shared Authority File for all the museums in Luxembourg using Wikibase technologies and Shape Expressions as an RDF data modelling and validation tool.

Previous work by UdL team

Universitat de Lleida (UdL) has experience in the deployment of the research data repository Dataverse. There is already one instance of this repository running at UdL and it is led by one of the members of its research team, who is also the responsible for the university's Library and Documentation Users Services. This experience will be used to guide the integration of ANGLIRU and existing research data repositories. Moreover, this deployment will be used to test the results of the project.

Regarding the use of semantic technologies for knowledge graph generation and the provision of advanced data management services, Universitat de Lleida has been involved in previous Spanish projects like OMediaDis (Open Platform for Multichannel Content Distribution Management) and InDAGuS-UX (Improved User Experience for Sustainable Open Government Data Infrastructures). In both projects, semantic data was used to apply knowledge graphs technologies to the management of multimedia content and open government data.

Universitat de Lleida has also experience in the application of knowledge graphs to agricultural data through the coordination for the European Food Safety Agency (EFSA) of a project that created a global media monitoring system on aspects related to plant health threats. To this end, an ontology and a knowledge graph were developed for plant pests and

³³ <https://zbmed.github.io/damalos/>

³⁴ <https://bit.ly/39S80yB>

³⁵ <https://bit.ly/37PRzQZ>

³⁶ <https://www.rd-alliance.org/>

diseases considered risks for the European Union. The ontology guided the researchers responsible for creating the knowledge graph and facilitated its enrichment through its connection with other knowledge graphs such as UniProt or DBpedia. This facilitated creating a multilingual knowledge graph to expand the amount of media to be monitored to media worldwide.

In connection with licensing terms and the use of blockchain technologies to provide trusted services for their management, UdL has the experience gained in the context of the H2020 project InVID (In Video Veritas) to facilitate the verification of social media videos and their reuse by journalists. In this project, he led the work package related to rights management, including the process of negotiating the terms of reuse of content between providers and journalists, represented through Semantic Web technologies and the Copyright Ontology. Also the formalization of the agreements reached through blockchain technologies, which provided trust to the involved parties, and the management of the identities of all participants through a Self-Sovereign Identity system.

Universitat de Lleida has also experience transferring all these technologies to the industry, materialised in different contracts with organisations at the international level. First of all, with Sony to improve copyright management using ontologies and knowledge graphs to model licenses. More recently, with the startup JAAK, where the copyright ontology and blockchain technologies were combined to build a shared infrastructure to allow the music and media industries to collaborate on a global view of content ownership and rights. Members of the UdL team have also gained a lot of experience about the application of knowledge graphs in the industry as part-time consultants for one of the leading companies in the application of this kind of technologies to the pharmaceutical and financial domains, Cambridge Semantics, the company that coined the term Semantic Data Lake.

Objectives

The main objective of ANGLIRU is to contribute to the FAIR community of research data repositories, building on top of existing platforms and helping these systems achieve full compliance with the FAIR principles especially regarding the principles less covered by current systems, i.e. interoperability and reusability.

This will be achieved through the application of technologies and methodologies based on semantic data, knowledge representation, ontologies and especially knowledge graphs. The strategy will be multidisciplinary, based on the experience of a team that has applied these technologies to make contributions to very diverse domains, including those relevant to the use cases of the project: sustainable agriculture and bioinformatics for health. Moreover, there is the involvement of personnel from the Universitat de Lleida's Library User Services which is responsible for the deployment of the research data repository Dataverse in this institution. There will be additional institutional support through Marta Oliva Solé, who is appointed Coordinator of Digital Transformation by Universitat de Lleida's rector.

Specific objectives

The Universidad de Oviedo team will focus on the following specific objectives:

- SO1 Data transformation pipelines between heterogeneous sources of data and heterogeneous formats and data models like XML, JSON, CSV, etc. to RDF data which preserve the intended data models.
- SO2 Large scale validation of knowledge graphs.

- SO3 Extraction of data models and schemas from existing RDF data.
- SO4 RDF data shapes authoring, visualisation and mappings.
- SO5 Research about the interplay between ontologies, vocabularies, rules and shapes.
- SO6 Interoperability between research data and personal data stores.
- SO7 FAIR representation of clinical data represented as FHIR.
- SO8 Knowledge graph subsetting for specific domains like health and life sciences.
- SO9 Knowledge graph-based fact-checking about health information.

While the Universitat de Lleida's researchers will deal with these specific objectives:

- SO10: Integration of ANGLIRU with existing research data repositories
- SO11: Generation of Knowledge Graphs from data and metadata retrieved from research data repositories
- SO12: Enrichment of Knowledge Graphs through semi-automatic semantic annotation to improve data quality, discoverability and integration.
- SO13: Improve reusers' awareness about datasets structure and reuse opportunities.
- SO14: Contribute to dataset licensing opportunities widening the range of licenses through representations that facilitate its automated management and the trust of the involved parties.
- SO15 Facilitate data reuse through data extractions tailored to the tools used by researchers, including models to apply Artificial Intelligence methods.
- SO16 Apply Knowledge Graphs in the context of sustainable agriculture research data.

Methodology and Work Plan

The following work plan will be carried out using an agile methodology combining Scrum for the definition of sprints, retrospectives and review and Kanban for keeping track of the tasks to be carried out and their status. Planning and tracking of the work plan will be coordinated as part of Task 5.1 and GitHub will be used to facilitate it, especially the use of GitHub projects, issues, features, bugs, etc. This approach will not be just followed for software development, all other tasks ranging from non-software deliverables preparation to diffusion activities like scientific papers preparation will also benefit from the use of an agile methodology for their coordination and tracking.

The team responsible for the materialisation of the work plan will be composed of all the members of the research team plus the additional personnel we propose to hire as detailed later in the corresponding subsection. The following list of members for each institution includes the initials used later to specify their involvement in work packages and tasks.

Universidad de Oviedo's team:

- **Research Team:** Jose Emilio Labra Gayo (JELG), Daniel Gayo Avello (DGA), Patricia Ordóñez de Pablos (POP), Aquilino Adolfo Juan Fuente (AAJF)
- **Work team:** Daniel Fernández Álvarez (DFA)
- **Requested PhD Studentships:** PHD-UO-1, PHD-UO-2
- **Requested Developer Contracts:** DEV-UO-1, DEV-UO-2

Universitat de Lleida's team:

- **Research Team:** Roberto García González (RGG), Rosa Gil Iranzo (RGI), Marta Oliva Solé (MOS), Juan Manuel Gimeno Illa (JGI) and Eva Estupinyà Piñol (ESP).
- **Requested PhD Studentships:** PHD-UDL-1, PHD-UDL-2
- **Requested Developer Contracts:** DEV-UDL-1, DEV-UDL-2

The work plan is decomposed on a set of work packages shown in **Figure 2**, and the details about the participants, objectives and tasks for each work package are provided in the following tables.

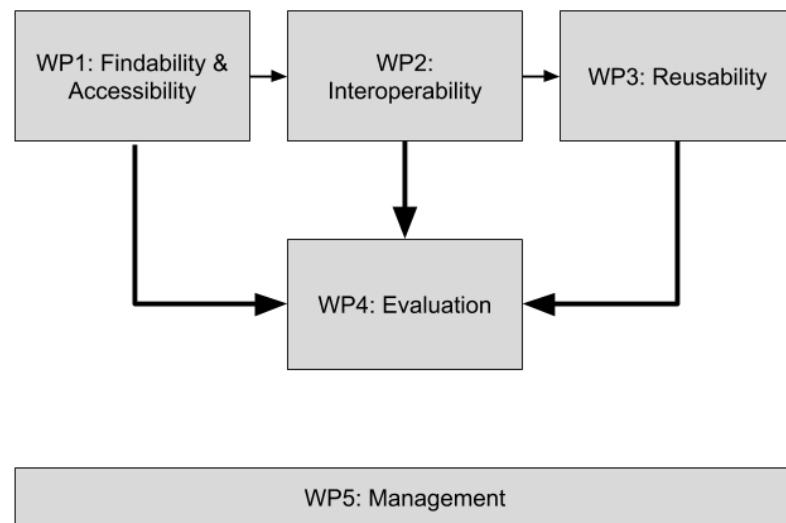


Figure 2. Decomposition of the work plan into Work Packages (WPs).

Work package number	WP1	Start Date or Starting Event	M1
Work package title	Findability and Accessibility		
WP Leader	Eva Estupinyà Piñol (ESP)		
Participants	MOS, RGG, DEV-UDL-1, JELG, DFA, PHD-UO-1, DEV-UO-1		
Objectives:	<p>this work package will be led by Eva Estupinyà Piñol, responsible for the Universitat de Lleida's Library User Services and, particularly, the deployment of the research data repository Dataverse in this institution. There will be additional institutional support through Marta Oliva Solé, who is appointed Coordinator of Digital Transformation by Universitat de Lleida's rector. This will help achieve the main objective of this work package, the integration of ANGLIRU with existing research data repositories.</p> <p>The fundamental requirement to achieve wider research data reuse following the FAIR principles is that it is easier to find and access. This process is guided in repositories by the use of persistent identifiers and metadata. These mechanisms are well-supported by current research data repositories, as shown in Figure 1, so ANGLIRU will integrate them through their API to take care of this part of the research data ingestion and annotation process.</p> <p>The developed integration will also make it easier for current users of this kind of repositories to migrate to ANGLIRU, or that datasets already registered benefit from the additional features that will be provided regarding Findability and Accessibility. After</p>		

ingestion through the integrated repositories, the generated persistent identifier and metadata will be converted into semantic data to bootstrap the generation of the associated knowledge graph (KG). Moreover, in the case of tabular data, it will also be processed to generate a rich semantic representation that will be combined into the dataset knowledge graph. For non-tabular data, the knowledge graph will include semantic annotations of the content.

Findability and accessibility will benefit from machine-readable representations of the dataset data. For instance, datasets can also be located in an automated way based on a combination of metadata and data fields. For instance, users can look for datasets with a certain subject like "biology" and narrow down the search to those including measures related to a specific biological pathway.

T1.1: Integration with existing research data repositories

Leader: EEP; Participants: JELG, MOS, JGI, DEV-UDL-1, PHD-UO-1, DEV-UO-1

Specific objectives: SG10

Develop ANGLIRU dataset ingestion services through the APIs of existing FAIR research data repositories, focusing on Dataverse first and then Zenodo. This includes the ingestion of new dataset, which will be also registered in the underlying repository, and those already present in a repository. As a result of this process, metadata and data representations linked to a persistent identifier will be obtained as generated by the underlying repository.

T1.2: Generation of dataset knowledge graph

Leader: MOS; Participants: DFA, JGI, EEP, DEV-UDL-1, PHD-UO-1, DEV-UO-1

Specific objectives: SG1, SG11

Metadata and data retrieved from the integrated repository are processed to automatically generate a semantic representation that will bootstrap the generation of the dataset knowledge graph. For metadata, the semantic representation will reuse existing schemes like Dublin Core or those registered in centralized registries like the DCC or RDA Metadata Directory. However, a wider range of schemas will be available through the integration of existing taxonomies, thesaurus and ontologies already available in a machine-readable format.

The proposed solutions leverage mechanisms developed by the Linked Data community to facilitate the retrieval of data and metadata using their identifiers in a standardised way. These mechanisms make use of URI and HTTP to provide identifiers that can be resolved to metadata and data, using content negotiation and redirections. This spans too to the schemas used for metadata description, as URIs are also used to identify the metadata fields which point to the schemas where they are defined.

T1.3: Semantic enrichment of the knowledge graph

Leader: RGG; Participants: JGI, DFA, PHD-UDL-1, DEV-UDL-1, PHD-UO-1, DEV-UO-1

Specific objectives: SG3, SG12

The knowledge graph resulting from T1.2 will be enriched by incorporating a semantic representation of the dataset data too. This process cannot be fully automated as it is highly domain-dependent. Following a semi-automatic approach, the user triggering the ingestion will receive assistance from the system based on an analysis of the available dataset metadata and its content.

The automatic part of the process will be based on approaches like Named Entity Linking (Ji et al., 2020), complemented with specific approaches in the case of tabular data³⁷. The results will be then presented to the user who will have the last word in formalising the enrichment of the knowledge graph, for instance choosing among the potential candidates for entity linking. The user experience will be inspired by systems like OpenRefine³⁸, which have successfully shown that lay users can use it to interactively generate knowledge graphs from tabular data.

The contribution made by ANGLIRU in this regard will be to facilitate user experience by preprocessing the tabular data and generating a first proposal for the semantic typing of rows, the mapping of columns to schema properties and cell values to entities.

Deliverables

- D1.1: Integration of ANGLIRU and Dataverse** (M6; Lead author: EEP)
- D1.2: Preliminary Version of the Dataset to KG Converter** (M9, Lead author: MOS)
- D1.3: Preliminary Version of the KG Enricher** (M12, Lead author: RGG)
- D1.4: Integration of ANGLIRU and Zenodo** (M15; Lead author: EEP)
- D1.5: Final Version of the Dataset to KG Converter** (M24, Lead author: MOS)
- D1.6: Final Version of the KG Enricher** (M24, Lead author: RGG)

Work package number	WP2	Start Date or Starting Event	M1
Work package title	Interoperability		
WP Leader	JELG		
Participants	DFA, Dev-UO-1, Dev-UO-2, PhD-UO-1, POP, DGA		

Description: Data users are usually interested in the integration of many different datasets about their topic of interest. Data integration increases its value and the usefulness of the conclusions derived from its analysis. This is particularly true for analysis methods that require high volumes of data, like those based on Artificial Intelligence techniques.

Moreover, the data should interoperate with applications or workflows for analysis, storage, and processing. Consequently, it should be possible to transform it into formats required by the involved tools. In this regard, the use of a Knowledge Graph and a graph-based data model facilitates both integration and interoperation.

RDF graphs provide a very flexible and general data model that can be converted into tabular or tree-based data without any kind of data loss. On the other hand, traditional systems mostly based on tabular or tree-based data pose a great limitation when the underlying data naturally follows more sophisticated models.

To attain that higher flexibility, traditional RDF based technologies use implicit schemas which can evolve and be tailored to the new needs that the system finds along the time. The problem of this approach is that data producers are not able to properly document the contents of their datasets or even ensure that the data they produce is valid which limits

³⁷ <http://www.cs.ox.ac.uk/isg/challenges/sem-tab>

³⁸ <https://openrefine.org>

consumers adoption of their data portals. The appearance of RDF validation technologies like ShEx and SHACL (Labra et al, 18) can mitigate this problem by enabling the use of explicit schemas for knowledge graphs that can be machine-processable.

However, as these technologies have appeared only recently, current implementations are not able to properly handle large knowledge graphs.

T2.1: Knowledge graphs data models and validation

Leader: JELG; Participants: DFA, MOS, Dev-UO-1, Dev-UO-2, PHD-UDL-1

Specific objectives: SO1, SO2, SO3

In this task, we propose the development of data transformation and integration systems that can handle data from different technologies like tabular data and semistructured data (XML, JSON). In WESO, we have already developed ShExML, an integration language inspired by ShEx (García-González et al, 20a).

Another important task is to validate large knowledge graphs. Currently, ShEx/SHACL validators do the validation in-memory which limits their application to large scale knowledge graphs like DBpedia or Wikidata. In this task, we are planning to adapt our SHaClEX implementation to handle this important use case in a way that the temporal results of the validation (the shape assignments) can be externalized to an external component which could gracefully decide to either keep them in memory or store them in other components. As far as we know, our SHaClEX implementation is the only one that handles both ShEx and SHACL, although both languages tackle the problem of validating RDF data, they do it from different perspectives: data description in ShEx vs constraints in SHACL, we will do further research to conciliate between both and to implement translators from one to the other that can be used inside the online RDFShape service (Labra et al 18).

Finally, one requirement that users have requested in practice is the need to automatically infer schemas from RDF data. We have already developed a prototype tool to extract schemas from existing RDF data called sheXer (Fernández et al 18, Boneva et al 19). That tool has already been applied to infer the ShEx schema for the national budget dataset at the Chilean National Library of Congress (Cifuentes et al 20).

T2.2: Schema authoring, visualizations and mappings

Leader: POP, participants: JELG, RGI, DFA, Dev-UO-1, PhD-UO-1, PHD-UDL-1

Specific objectives: SO4

One important aspect of interoperability is to be able to define mappings at the schema or data model level. We have already developed a system to transform XML schemas to ShEx schemas in (García-González et al, 20b). In this project, we are planning to continue this line of work and provide a general service that provides mappings between other data models like Json Schemas or DDLs. On the other hand, the appearance of Shapes ecosystems prompts the development of new tools that can create relationships between schema libraries promoting a new degree of reusability at the schema level.

T2.3: Vocabularies, shapes and rules for knowledge graphs

Leader: DGA, Participants: JELG, JGI, POP, DEV-UO-1, PHD-UO-1, PHD-UDL-1

Specific Objectives: SO5

The appearance of shapes based solutions in the semantic web stack has also triggered new discussions about the interplay between old members of the stack like ontologies and

vocabularies with shapes. Some proposals like SHACL rules³⁹ blur the separation between traditional rules technologies like SWRL or RIF and open new questions about which solution is better. In this task, we are planning to investigate the relationship between these solutions and provide some guidelines about when one approach is better than another.

Another aspect of this line of work is test-driven development of ontologies where the tests are defined using shapes. In WESO, we already created a first prototype solution for the HERCULES project (Labra et al 20) using a continuous integration server that checks compliance of the results inferred by a reasoner using the ontology with the expected results defined in shape maps and validated using shape expressions.

An important aspect of interoperability is modelling and representing provenance information, to that end, annotating RDF triples using RDF* based solutions (Hartig 17) seems also an important aspect to take into account and we are planning to work also on possible extensions of Shapes schemas that handle RDF*.

T2.4: Interoperability between private and research data

Leader: AAJF, Participants: JELG, RGG, DGA, Phd-UO-1, Dev-UO-1, PHD-UDL-1

Specific Objectives: SO6

The privacy concerns of users are motivating the appearance of projects like SOLID⁴⁰ where the users own their data which is stored in POD (Personal Online Datastores). An important aspect of future research platforms is to cooperate with this kind of platforms and interoperate with their data in a way that can be useful to the researcher but also respects user's privacy concerns. In this task, we will work on a service that can access user's data from SOLID platform in an interoperable way.

Deliverables

- D2.1. Data transformation service, M24, JELG
- D2.2. Large scale validation service, M24, JELG
- D2.3. Service for schema extraction from data, M15, DFA
- D2.4. Schema authoring and visualization service, M12, POP
- D2.5. Schema mappings service, M24, AAJF
- D2.6. Continuous integration service based on Shapes, M24, JELG
- D2.7. Prototype KG based on shapes, rules and ontologies, M36, DGA
- D2.8. SOLID interoperability service, M36, AAJF

Work package number	WP3	Start Date or Starting Event	M6
Work package title	Reusability		
WP Leader	Roberto García González		
Participants	JELG, AAJF, DFA, Dev-UO-1, Dev-UO-2, Phd-UO-1, Phd-UO-2		

³⁹ <https://w3c.github.io/shacl/shacl-af/#rules>

⁴⁰ <https://solidproject.org/>

	JGI, RGI, PHD-UDL-1, PHD-UDL-2, DEV-UDL-2
Objectives: Facilitating the reuse of the data is the ultimate goal. This is a complex process both from the technical and the legal standpoint. Thanks to the rich representation of data and metadata as part of the resulting knowledge graph, together with a representation with the same degree of richness of the data reuse conditions and obligations, this process will be largely facilitated by ANGLIRU services.	
First of all, potential reusers will have access to richer descriptions of the dataset thanks to the semantic representations of both the metadata and the data, and their integration with other knowledge graphs, vocabularies and ontologies. Though this might make user interaction more difficult, ANGLIRU will make use of automatically generated user interfaces that allow iteratively exploring and visualising the knowledge graphs structure.	
Reuse licenses will be also represented using semantic technologies, so ANGLIRU will provide services for both data providers and reusers to model the terms under which they are offering or requesting the data. These representations will facilitate offering data under a more diverse set of terms, locating datasets that fit reuser constraints and assisted negotiation of the terms.	
Once the reuse is agreed among the involved parts, Self-Sovereign Identity (SSI) and blockchain mechanisms will be put into practice to provide a trustful environment for them. The objective is to keep track of an immutable and auditable way about the terms and identities of the parties.	
Finally, ANGLIRU will make it easier to reuse the requested data by providing data extraction mechanisms tailored to reuser needs. This will include tabular data enriched with semantic annotations to avoid ambiguities and increase its quality, but also research objects and models to apply Artificial Intelligence techniques.	
T3.1: Dataset structure discovery	
Leader: JGI; Participants: RGI, PHD-UDL-1, Phd-UO-1, Dev-UO-1	
Specific objectives: SO13	
Data is naturally documented as a result of being integrated into a knowledge graph, following the links among different related datasets and the associated vocabularies that unambiguously state the intended interpretation of the dataset metadata and data. To make these detailed descriptions accessible to users, they will be presented through interactive descriptions and visualizations based on the Rhizomer ⁴¹ tool for semantic data exploration.	
T3.2: Reuse licence management	
Leader: RGI; Participants: RGG, PHD-UDL-2, POP	
Specific Objectives: SO14	
To make the terms under which data is made available for reuse clear, even for automated processing or users assistance, licenses will be also modelled using a rich representation language to make them machine-actionable. The contribution will build on top of previous experience with the Copyright Ontology (García and Gil, 2010). The license terms for datasets will be represented using ontologies, capturing the corresponding conditions and obligations and facilitating the automated matching of intended uses and available terms.	

⁴¹ <https://rhizomik.net/rhizomer>

ANGLIRU will feature an assistant during data publishing which helps users pick from predefined licenses modelled using the Copyright Ontology like Creative Commons ones. In this regard, it will be similar to the Public License Selector⁴². However, it will add the possibility for users to go beyond these predefined licenses and allow data providers to configure the license terms they wish, and reusers negotiate the terms building counteroffers.

This will be done in a completely interactive way, freeing users from the burden of modelling the terms, building on top of previous experience in the context of the InVID H2020 project to negotiate licensing terms among journalists and social media owners using semantic technologies and blockchain (García et al., 2019).

The proposed approach opens the door to many data providers with datasets they do not wish to publish under open licenses because they prefer more control over their reuse. This facilitates compliance with existing codes of conduct about data sharing and reuse, like the EU one in agricultural data sharing by contractual agreement⁴³, which require awareness and approval by data providers about reuses of their data, including compensations that do not necessarily need to be monetary. These detailed terms can be captured, including conditions like privileged access by providers to the results of studies that use their data, balancing this way the power among data players, like small farmers and big research institutes with economic incentives.

T3.3: Reuse trust

Leader: RGG; Participants: MOS, PHD-UDL-2, AAJF, Phd-UO-2

Specific Objectives: SO14

To increase trust in the system especially from the data providers perspective, and to provide guarantees about the conditions under which reusers and using the data, ANGLIRU will integrate blockchain technologies to keep immutable and tamper-proof records of the terms of the reuse agreement, and additional information about the life cycle of the data from ingestion to reuse to keep a record of data provenance.

The services will make use of public blockchains (like Ethereum) but also of permissioned ones like those provided by the Alastria⁴⁴ initiative, of which Universitat de Lleida is a member and thus has access to this kind of blockchain nodes. The use of blockchain independent from ANGLIRU will increase the trust parties can put on the platform. For instance, it will be possible to verify agreements' terms independently from the ANGLIRU infrastructure and these will be available if the infrastructure suffers an outage or completely disappears.

Another important requirement for trust is the identity of the involved parties. ANGLIRU will leverage the authentication and authorisation mechanisms of the underlying FAIR platforms. Beyond that, it will generate a trusted identity that aggregates one or more underlying user accounts of the FAIR platforms integrated. This new identity, and the associated authentication and authorization mechanisms, will be based on Decentralised Identifiers (DID) supporting SSI (Self-Sovereign Identity), for instance, blockchain-

⁴² <https://ufal.github.io/public-license-selector>

⁴³ https://copa-cogeca.eu/img/user/files/EU%20CODE/EU_Code_2018_web_version.pdf

⁴⁴ <https://alastria.io>

anchored DIDs.

T3.4: Data Extraction

Leader: RGG; Participants: MOS, PHD-UDL-2, DEV-UDL-2, JELG, DFA, Phd-UO-2

Specific Objectives: SO15

Because the data is based on a graph model, it can be transformed into representations tailored to the tools used for data analysis to facilitate their reuse. Despite this process might produce losses due to the adaptation of less expressive representation languages, this transformation will be documented in the knowledge graph to keep provenance information so its reproducibility and explainability can be guaranteed.

For instance, if the resulting data is tabular, there will be links from columns to the corresponding concepts or dimensions represented by the data, including unambiguous references to the associated entities, like a reference from the cell value "Guadalajara" in the column "City" to the URL "<https://www.wikidata.org/wiki/Q11953>" that unambiguously identifies the Spanish city and avoid the confusion with the Mexican one.

Moreover, the original graph representation will be also used to generate more sophisticated representations that facilitate reuse, like Research Objects Crate⁴⁵, and the application of Artificial Intelligence approaches, especially Knowledge Graph embeddings (KGEs). These are low-dimensional representations of the entities and relations in a knowledge graph. They provide a generalizable context about the overall KG that can be used to infer relations, like insights about molecular property interactions to accelerate drug discovery, or to detect clusters, for instance, to classify the productivity characteristics of different kinds of fruit varieties.

In all cases, ANGLIRU services will guarantee that the extracted data comply with community standards or best practices regarding how data is shared. These standards and best practices will be modelled as data shapes, specifications of these requirements about the attributes being stored or retrieved implemented using ShEx or SHACL.

Deliverables

D3.1: Dataset Discover Service (M18; Lead author: JGI).

D3.2: Dataset Licensing Service (M24, Lead author: RGI)

D3.3: Reuse Trust Service (M30, Lead author: RGG)

D3.4: Data Extraction Service (M33, Lead author: RGG)

Work package number	WP4	Start Date or Starting Event	M12
Work package title	Evaluation through use cases		
WP Leader	DGA		
Participants	JELG, AAJF, Dev-UO-2, Phd-UO-2 RGG, EEP, PHD-UDL-1, DEV-UDL-2		

⁴⁵ <https://www.researchobject.org>

Description: the use cases where the approach will be put into practice have been selected based on the priorities detailed in the "European Strategy for Data" and towards the expected impacts detailed in Section 3. The use cases are:

- **Agriculture and Green Deal data space:** to enhance the sustainability performance and competitiveness of the agricultural sector through the processing and analysis of production and other data, allowing for precise and tailored application of production approaches at farm level. Considering the Green Deal priority actions on climate change, circular economy, zero-pollution, biodiversity, deforestation and compliance assurance.
- **Health data space:** for advances in preventing, detecting and curing diseases as well as for informed, evidence-based decisions to improve the accessibility, effectiveness and sustainability of the healthcare systems. As part of this work package, we will also tackle the complex problem of checking health-related information using knowledge graphs.

The evaluation of all developments made in the context of the previous use cases will be automated through services for FAIRness assessment like the FAIR Evaluation Services⁴⁶ (Wilkinson et al., 2019). This service is strictly focused on detecting and validating behaviours of digital objects that make them machine-readable and reusable, the main focus of the proposal. Moreover, this evaluation services can be integrated through an API so the evaluation process can be incorporated into a Continuous Integration and Deployment development process.

T4.1: Agriculture and Green Deal Use Case

Leader: RGG; Participants: EEP, PHD-UDL-1, DEV-UDL-2

Specific Objectives: SO16

Data is one key element to enhance the sustainability, performance and competitiveness of the agricultural sector. To put it into practice, a common data space for agricultural data geared towards controlled data sharing could facilitate the emergence of a neutral platform for sharing and integrating agricultural data, including both private and public data. This data space can be then exploited in combination with other data on the supply chain and other types of data, such as earth observation or meteorological data, making it possible to apply more precise and tailored production approaches at farm level.

ANGLIRU perfectly fits the previous challenge as it adds to existing research data repositories mechanisms to facilitate data integration to mix heterogeneous sources. Moreover, it facilitates sharing private data, or data that providers do not want to make open but under certain conditions not necessarily economic.

The objective in this use case will be to apply ANGLIRU services to agriculture research data, converted and integrated through Knowledge Graphs with existing KGs for the agri-food domain like FAO's AGROVOC or other vocabularies and ontologies from AgroPortal⁴⁷. Regarding reuse policies, ANGLIRU's services regarding reuse will be applied to implement the EU code of conduct for sharing of agricultural data by contractual

⁴⁶ <https://w3id.org/AmI/FAIR>

⁴⁷ <http://agroportal.lirmm.fr>

agreement developed in 2018 by EU stakeholders, involving – among others – the farming as well as the machinery sector.

To assure that enough agriculture data is collected during the project to guarantee the relevance of the Knowledge Graph resulting from the integration of all datasets, the Federation of Agricultural Cooperatives of Catalonia (FCAC) and the Catalan Cluster of Agricultural Production Means (FEMAC) have been contacted and have expressed their interest in the project. There will also be the participation of the UdL's Agriculture Precision research group, who will contribute research data through the institutional Dataverse.

T4.2: Health and bioinformatics Use Case

Leader: Daniel Gayo Avello; Participants: JELG, POP, AAJF, DFA, Dev-UO-2, PhD-UO-2

Specific Objectives: SO7, SO8, SO9

As we described in the previous work section, in the last years, Jose Emilio Labra Gayo has been participating in several initiatives related to health, life sciences and bioinformatics applying knowledge graphs. One important goal will be a common European health data space, which will be essential for advances in preventing, detecting and curing diseases as well as for informed, evidence-based decisions to improve the accessibility, effectiveness and sustainability of the healthcare systems.

There are currently some on-going initiatives to obtain clinical data following FAIR principles like the FHIR-FAIR track from HL7⁴⁸ and the FAIR4Health project⁴⁹. Two aspects are relevant for this connection, on one hand, the specification of FHIR-RDF format is already using ShEx, so all the technologies developed in the interoperability work package for ShEx like schema authoring, visualization and mapping tools can be leveraged in this task. On the other hand, given there are already a big amount of health data that is based on RDF like UniProt⁵⁰, Bio2RDF⁵¹, Wikidata⁵², etc. and in fact, GeneWiki project has been using Wikidata as a hub to integrate this kind of information (Burgstaller 16). We are actively collaborating with the GeneWiki project to that end and the WikidataIntegrator already uses ShEx validation before ingesting data to Wikidata or other Wikibase instances and we have participated in the development of a protocol to add data about COVID-19 to Wikidata in April 2020 which has recently been accepted in the BMC Biology Journal (Waagmeester et al, 20).

Another line of research which has been promoted by our collaboration with Andra Waagmeester and Wikidata is the need for a system that can generate Wikidata subsets for some specific needs. During the Biohackathon project 35 led by Jose Emilio Labra Gayo we already started a prototype that can generate subsets of Wikidata based on data models defined in ShEx⁵³.

During the biohackathon, several applications and use cases were identified like the possibility to define a subsetting service for Wikidata or other large knowledge graphs which allows the users to obtain subsets or snapshots of those large knowledge graphs at

⁴⁸ <https://confluence.hl7.org/display/FHIR/2021-01+FAIR>

⁴⁹ <https://www.fair4health.eu/>

⁵⁰ <https://www.uniprot.org/>

⁵¹ <https://bio2rdf.org/>

⁵² <https://www.wikipathways.org/>

⁵³ <https://github.com/elixir-europe/BioHackathon-projects-2020/tree/master/projects/35>

some specific points in time that could be used later in their research. In this way, those subsets can be part of their research objects and can help reproducibility of their results.

Another application of the knowledge graphs related to health that will be explored in this use case is their application for fact-checking and the verification of documents containing potential disinformation, like fake news. We are planning to actively collaborate with some organizations fighting disinformation in the health and science domains, like maldita.es⁵⁴ or Salud Sin Bulos⁵⁵ who have also expressed their support to this proposal, to facilitate the detection of fake news about health by automated means using knowledge graphs with health-related facts that can help to detect false or unproven facts in social media.

Deliverables

D4.1: Prototype of the Green Deal Agriculture KG (M24; Lead author: RGG)

D4.2: Green Deal Agriculture KG (M36; Lead author: RGG)

D4.3: FHIR-FAIR integration prototype (M24; Lead author: AAJF)

D4.3: Knowledge graph subsetting service (M36; Lead author: DFA)

D4.4: Knowledge graphs-based fact-checking service (M36, Lead author: DGA)

Work package number	WP5	Start Date or Starting Event	M1
Work package title	Management		
WP Leader	Patricia Ordóñez de Pablos		
Participants	JELG, RGG, RGI, EEP		

Description:

Given that this is a coordinated project we consider necessary to have a separate work package for project management which will involve the coordination, risk analysis and dissemination aspects of the project. The project will start creating a risk assessment and contingency committee formed by JELG, RGG, POP and RGI which will evaluate potential risks of each stage of the project implementation as well as the period before and after the project starts and ends.

T5.1: Project Management

Leader: Patricia Ordóñez de Pablos; Participants: JELG, EEP, RGG

This task will consist of the creation of the risk management and contingency plan. A risk matrix will be employed to position the estimated risks as probability and frequency of occurrence, risks' impact on current and future actions and potential solutions. The risk impact will be estimated on a 5 items scale (marginal, minor, moderate, major, severe) and risk probability evaluated (rare, unlikely, possible, likely, almost certain). The matrix will generate a list of 5 categories of risk priorities to be addressed. The risk management plan will be drafted, for each identified risk, each priority will benefit from at least two possible solutions based on existing resources or the project capability to attract external resources

⁵⁴ <https://maldita.es/>

⁵⁵ <https://saludsinbulos.com/>

to manage severe risks.

We will also prepare the data management plan which will also include a software management plan as in this case software development is an important aspect of this project.

T5.2: Dissemination

Leader: Patricia Ordóñez de Pablos; Participants: JELG, RGI, RGG

During the project, a web page will be created to include information about the different results. The web page will also include links to related projects and other social network aspects like possible blog posts, Twitter entries, etc.

In this task, the project will coordinate the dissemination tasks related to the project like participation in conferences, journal submissions, etc. which will be collected in a research publication report at the end of the project. More information about the dissemination plan can be seen in the section Results diffusion plan.

Deliverables

D5.1: Risk management and contingency plan and reviews (M4, Lead author: POP)

D5.2: Data Management Plan and reviews (M1, M13, M25; Lead author: EEP)

D5.3: Web page of the project (M21, Lead author: POP) this deliverable will also include other social network activities like the creation of Twitter entries, blog posts, links to other projects, etc.

D5.4: Research publications report (M36, Lead author: POP)

Chronogram

In the following figure, we represent the critical dependencies among WPs identified as milestones and connected with the corresponding deliverables.

ID	TASK AND DELIVERABLES	LEADER	Year 1				Year 2				Year 3			
			Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
T1.1	Integration with existing research repositories	EEP												
T1.2	Generation of dataset knowledge graphs	MOS												
T1.3	Semantic enrichment of the knowledge graphs	RGG												
D1.1	Integration of ANGLIRU and Dataverse	ESP												
D1.2	Preliminary Version of the Dataset to KG Converter	MOS												
D1.3	Preliminary Version of the KG Enricher	RGG												
D1.4	Integration of ANGLIRU and Zenodo	ESP												
D1.5	Final Version of the Dataset to KG Converter	MOS												
D1.6	Final Version of the KG Enricher	RGG												
T2.1	Knowledge graphs data models and validation	JELG												
T2.2	Schema authoring, visualization and mapping	POP												
T2.3	Vocabularies, shapes and rules for knowledge graphs	DGA												
T2.4	Interoperability between private and research data	AAJF												
D2.1	Data transformation service	JELG												
D2.2	Large scale validation service	JELG												
D2.3	Service for schema extraction from data	DGA												
D2.4	Schema authoring and visualization service	POP												
D2.5	Schema mappings service	AAJF												
D2.6	Continuous integration service based on Shapes	JELG												
D2.7	Prototype KG based on shapes, rules and ontologies	DGA												
D2.8	SOLID interoperability service	AAJF												
T3.1	Dataset structure discovery service	JGI												
T3.2	Reuse license management	JGI												
T3.3	Reuse trust	RGG												
T3.4	Data extraction	RGG												
D3.1	Dataset Discover Service	JGI												
D3.2	Dataset Licensing Service	RGG												
D3.3	Reuse Trust Service	RGG												
D3.4	Data Extraction Service	RGG												
T4.1	Agriculture and Green Deal	RGG												
T4.2	Health and bioinformatics	DGA												
D4.1	Prototype of the Green Deal Agriculture KG	JGI												
D4.2	Green deal agriculture KG	RGG												
D4.3	Knowledge graph subsetting service for GeneWiki	RGG												

Risk management and contingency plan

We have identified several risks in the proposed chronogram and work packages as well as their possible impact and the probability that they occur. In the next table we show a first analysis although a more complete risk management and contingency plan will indeed be developed as part of the management work package once the project starts.

Risk	Probability	Impact	Mitigation plan
R1. Delays in scientific paper publication	Middle	Very Low	Publishing scientific papers is a process that depends on several factors which are not always under control by a research team. Although in the chronogram there are tasks that imply the publication of a certain amount of scientific papers, it may be possible that the deadlines are delayed. The impact on the overall results of the project is considered very low as publication may succeed later.
R2. Delays in software production deliverables	Low	Low	The research groups have a long experience in all the topics presented in the different work packages. In particular, the main researcher has been years working in knowledge graphs and RDF validation technologies like ShEx and SHACL and more recently in the application of these technologies to research data as part of the HERCULES project. Furthermore, with the enrolment of new developers and PhD students we ensure the availability of enough resources to accomplish the goals. One aspect that we considered in the proposal is that most of the work packages are independent between each other in a way that if there is some delay in one task, other tasks are not directly affected.
R3. Some member of the team leaves the project	Low	Low	This project has a combination of software development based on independent services. In case that some member of the team leaves the project the service-based architecture that we are considering will be able to accommodate to the parts affected and we expect that the corresponding risks that could affect the whole project will be minimized. On the other hand, given that the members of the research group are part of the Master in Web Engineering we consider that finding other people interested to work in these projects will not be difficult as the students from that master are taught Semantic web

			technologies and it seems possible to enrol new team members which already know the basis of these technologies without a very critical delay.
R4. Coordination problems	Low	Low	We consider that the possibility that serious coordination problems inside each research group appear is low as the members of each research team have a long experience working together in several projects. On the other hand, the collaboration between WESO and GRIHO teams is new and the risks of coordination in this case is relevant. To mitigate this risk, the project work plan includes, in WP5, a task devoted to project management with the participation of 4 researchers, two for each institution. This management committee will be responsible for tracking project evolution, detecting potential issues and coordinating responses that solve them.

Justification on the need to hire additional personnel

Given the size of the project and in order to meet the goals and use cases we need to temporally hire new members for both the WESO and GRIHO research teams. Given that the members of the research team have teaching and management assignments, they cannot be involved in software development or hardware configuration and maintenance on a daily basis. Moreover, these researchers cannot focus on research problems that require full-time dedication either, while these problems offer a great opportunity to develop PhD projects. Consequently, two profiles are considered for the personnel to be hired: technical developers and PhD students, to that end, we request the following additional personnel which has already been included in the different work packages:

- The UdL team requests 2 PhD students and 2 developers:
 - PHD-UDL-1: will develop a PhD thesis project about the interactive and semiautomatic generation and integration of knowledge graphs from tabular data, applied to data related to agriculture, food, sustainability and biodiversity, with participation in Tasks 1.3, 2.1, 2.2, 2.3, 3.1 and 4.1.
 - PHD-UDL-2: will focus on a PhD thesis project the use of artificial intelligence and blockchain technologies to provide trustful research data reuse, with participation in Tasks 2.4, 3.2, 3.3 and 3.4.
 - DEV-UDL-1: will contribute to the development of the integration of ANGLIRU with existing research data repositories and the transformation of retrieved data into knowledge graphs. This requires a relevant amount of development effort in the context of WP1 for a period of 30 months, which includes development but its integration with and maintenance during the last year.
 - DEV-UDL-2: will participate in the application of the ANGLIRU services in the agriculture and green deal use case, corresponding to Task 4.1, and also to the generation of data extractions tailored to this use case in Task 3.4.
- The WESO team at the University of Oviedo also requests the incorporation of 2 PhD students and 2 developers:
 - PHD-UO-1: will develop a PhD project about applications of shapes to knowledge graphs validation. In that way, he will participate in all tasks from

the interoperability work package (T2.1-T2.4) as well as in some tasks from the findability and accessibility work package (T1.2, T1.3)

- PHD-UO-2 will develop a PhD project about applications of shapes and knowledge graphs in the health and life sciences domain. To that end he will participate in the use task 4.2 and will also collaborate in T3.3 and T3.4 from the reusability work package.
- DEV-UO-1 will contribute to the development and implementation shapes applications and tools participating in all the tasks in the interoperability work package (T2.1-T2.4) as well as in the tasks T1.1, T1.2 and T1.3 from the findability and accessibility work package.
- DEV-UO-2 will participate in the development of tools for knowledge graph subsetting and the implementation of algorithms for knowledge graph based fact checking in task 4.2.

References

- (Berners-Lee et al., 01) Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):34–44, Mai 2001.
- (Berrueta 06) Searching over Public Administration Legal Documents Using Ontologies, Diego Berrueta Muñoz, Jose Emilio Labra Gayo, Luis Polo Paredes, In International Joint Conference on Knowledge Based Software Engineering IJCKBS,06-2006.
- (Bizer 11) Bizer, Christian; Heath, Tom; Berners, Tim. Linked data: The story so far. En Semantic services, interoperability and web applications: emerging concepts. IGI Global, 2011. p. 205-227.
- (Boneva et al., 19) Iovka Boneva, Jérémie Dusart, Daniel Fernández Álvarez, Jose Emilio Labra Gayo, Semi Automatic Construction of ShEx and SHACL Schemas, Arxiv preprint: <https://arxiv.org/abs/1907.10603>
- (Burgstaller 16) Sebastian Burgstaller-Muehlbacher, Andra Waagmeester, Elvira Mitraka, Julia Turner, Tim E. Putman, Justin Leong, Chinmay Naik, Paul Pavlidis, Lynn M. Schriml, Benjamin M. Good, Andrew I. Su: Wikidata as a semantic framework for the Gene Wiki initiative. *Database J. Biol. Databases Curation* 2016 (2016)
- (Cifuentes et al., 20) Francisco Cifuentes Silva, Jose Emilio Labra Gayo, Daniel Fernández Álvarez, National Budget Law as Linked Open Data: New tools for supporting the sustainability of public finances, *Sustainability*, vol 12, issue 11 - 2020
- (Cifuentes, 19) F. Cifuentes-Silva y J. E. Labra Gayo, Legislative document content extraction based on Semantic Web technologies - A use case about processing the History of the Law, *Extended Semantic Web Conference*, 2019.
- (Cifuentes 11) F. Cifuentes-Silva, C. Sifaqui y J. E. Labra-Gayo, Towards an architecture and adoption process for linked data technologies in open government contexts, de Proceedings of the 7th International Conference on Semantic Systems - I-Semantics '11, 2011.
- (Collins et al., 18) Turning FAIR into Reality, Final report and action plan from the European Commission Expert Group on FAIR Data, Directorate General for Research and Innovation, https://ec.europa.eu/info/sites/info/files/turning_fair_into_reality_1.pdf
- (Fernández et al., 18) Daniel Fernández Álvarez, Herminio García González, Johannes Frey, Sebastian Hellmann, Jose Emilio Labra Gayo, Inference of Latent Shape Expressions Associated to DBpedia Ontology, Poster presented at International Semantic Web Conference, Monterey, California -2018

- (García-González et al. 20a) Herminio García-González, Iovka Boneva, Sławek Staworko, José Emilio Labra-Gayo, Juan Manuel Cueva Lovelle, ShExML: Improving the usability of heterogeneous data mapping languages for first-time users, in PeerJ Computer Science 6:e318, 2020, doi:10.7717/peerj-cs.318 Open Access
- (García-González et al. 20b) Herminio García-González, Jose Emilio Labra-Gayo, XMLSchema2ShEx: Converting XML validation to RDF validation" in Semantic Web Journal, vol. 11, no. 2, pp. 235-253, 2020, doi:10.3233/SW-180329
- (García et al. 19) García, Roberto, et al. Copyright Management of User-Generated Video for Journalistic Reuse. Video Verification in the Fake News Era. Springer, Cham, 2019. p. 223-259.
- (García et al., 10) García, Roberto; GIL, Rosa. Content value chains modelling using a copyright ontology. Information Systems, 2010, vol. 35, no 4, p. 483-495.
- (García et al., 20) L. García C., J. Bolleman, M. Dumontier, S. Jupp, Jose E. Labra Gayo, T. Liener, T. Ohta, N. Queralt-Rosinach, C. Wu, Data validation and schema interoperability, BiohackrXiv preprints, <https://biohackrxiv.org/8qdse/>
- (Hartig, 17) Olaf Hartig: Foundations of RDF* and SPARQL* - An Alternative Approach to Statement-Level Metadata in RDF. In Proceedings of the 11th Alberto Mendelzon International Workshop on Foundations of Data Management (AMW), Montevideo, Uruguay, June 2017
- (Hogan et al., 20) Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutiérrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan F. Sequeda, Steffen Staab, Antoine Zimmermann: Knowledge Graphs. CoRR abs/2003.02320 (2020)
- (Jacobsen et al., 20) A. Jacobsen, R. de Miranda, N. Juty, D. Batista, S. Coles, R. Cornet, M. Courtot, M. Crosas, M. Dumontier, C. Evelo, C. Goble, G. Guizzardi, K. Kryger, A. Hasnain, K. Hettne, J. Heringa, R. Hooft, M. Imming, K. Jeffery, R. Kaliyaperumal, M. Kersloot, C. Kirkpatrick, T. Kuhn, I. Labastida, B. Magagna, P. McQuilton, N. Meyers, A. Montesanti, M. van Reisen, P. Rocca-Serra, R. Pergl, S. Sansone, L. Santos, J. Schneider, G. Strawn, M. Thompson, A. Waagmeester, T. Weigel, M. Wilkinson, E. Willighagen, P. Wittenburg, M. Roos, B. Mons, and E. Schultes, FAIR Principles: Interpretations and Implementation Considerations, Data Intelligence 2020 2:1-2, 10-29
- (Labra et al., 13) J. E. Labra Gayo y J. M. Alvarez Rodríguez, Validating statistical index data represented in RDF using SPARQL queries de RDF Validation Workshop. Practical Assurances for Quality RDF Data, Cambridge, 2013.
- (Labra et al., 14) J. E. Labra Gayo, H. Farham, J. Castro Fernández y J. M. Álvarez Rodríguez, Representing Statistical Indexes as Linked Data Including Metadata about Their Computation Process de Metadata and Semantics Research - 8th Research Conference, MTSR 2014, Karlsruhe, Germany, November 27-29, 2014. Proceedings, 2014.
- (Labra et al., 18) José Emilio Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, Dimitris Kontokostas: Validating RDF Data. Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool Publishers 2018
- (Labra et al., 19) Jose E. Labra-Gayo, H. García-González, D. Fernández-Alvarez, E. Prud'hommeaux, Challenges in RDF Validation. In: Alor-Hernández G., Sánchez-Cervantes J., Rodríguez-González A., Valencia-García R. (eds) Current Trends in Semantic Web Technologies: Theory and Practice. Studies in Computational

Intelligence, vol 815. Springer, Cham. https://doi.org/10.1007/978-3-030-06149-4_6, 2019

(Labra et al., 20) ASIO: A research management system based on semantic technologies, Jose Emilio Labra Gayo José Barranquero Tolosa Guillermo Facundo Colunga Alejandro González Hevia Emilio Rubiera Azcona Daniel Ruiz Santamaría Paulino Álvarez de Ron Ondina, In International Semantic Web Conference, ISWC2020. First virtual workshop on Research data* management for Linked Open Science - DaMaLOS, 2020

(Lieber et al., 20) Lieber, S., Dimou, A., & Verborgh, R. (2020). Statistics about Data Shape Use in RDF Data. In K. Taylor, R. Goncalves, F. Lecue, & J. Yan (Eds.), Proceedings of the 19th International Semantic Web Conference: Posters, Demos, and Industry Tracks (Vol. 2721, pp. 330–335).

(Mons 17) Mons, Barend et al. 'Cloudy, Increasingly FAIR; Revisiting the FAIR Data Guiding Principles for the European Open Science Cloud'. 1 Jan. 2017 : 49 – 56.

(Poveda-Villalón 20) Poveda-Villalón, María, et al. Coming to Terms with FAIR Ontologies. En Knowledge Engineering and Knowledge Management: 22nd International Conference, EKAW 2020, Bolzano, Italy, September 16–20, 2020, Proceedings. Springer Nature, 2020. p. 255.

(Prud'hommeaux et al., 14) Eric Prud'hommeaux, José Emilio Labra Gayo, Harold R. Solbrig: Shape expressions: an RDF validation and transformation language. SEMANTICS 2014: 32-40, Best paper award.

(Shaoxiong 20) JI, Shaoxiong, et al. A survey on knowledge graphs: Representation, acquisition and applications. arXiv preprint arXiv:2002.00388, 2020.

(Thornton, 19) Katherine Thornton, Harold Solbrig, Gregory S. Stupp, José Emilio Labra Gayo, Daniel Mietchen, Eric Prud'hommeaux, Andra Waagmeester: Using Shape Expressions (ShEx) to Share RDF Data Models and to Guide Curation with Rigorous Validation. ESWC 2019: 606-620

(Verborgh 19) Ruben Verborgh, Shaping Linked Data apps, available at: <https://ruben.verborgh.org/blog/2019/06/17/shaping-linked-data-apps/>

(Waagmeester et al., 20a) A. Waagmeester, G. Stupp, S. Burgstaller-Muehlbacher, B. Good, M. Griffith, O. Griffith, K. Hanspers, H. Hermjakob, T. Hudson, K. Hybiske, S. Keating, M. Manske, M. Mayers, D.I Mietchen, E. Mitraka, A. Pico, T. Putman, A. Riutta, Nuria Queralt-Rosinach, L. Schriml, T. Shafee, D. Slenter, R. Stephan, K. Thornton, G. Tsueng, R. Tu, S. Ul-Hasan, E. Willighagen, C. Wu, A. Su, Wikidata as a knowledge graph for the life sciences, eLife, vol. 9(2020)

(Waagmeester et al., 20b) Andra Waagmeester, Egon L. Willighagen, Andrew I Su, Martina Kutmon, Jose Emilio Labra Gayo, Daniel Fernández Álvarez, Quentin Groom, Peter J. Schaap, Lisa M. Verhagen, Jasper J. Koehorst, A protocol for adding knowledge to Wikidata, a case report, BMC Biology, Accepted, 2020

(Wilkinson, 16) Wilkinson, M.D., et al. The FAIR Guiding Principles for scientific data management and stewardship. Scientific data, 2016, vol. 3, no 1, p. 1-9. <https://doi.org/10.1038/sdata.2016.18>

(Wilkinson et al., 17) Wilkinson, M. D., Verborgh, R., Bonino da Silva Santos, L. O., Clark, T., Swertz, M., Kelpin, F. D. L., Gray, A. J. G., Schultes, E., van Mulligen, E. M., Ciccarese, P., Kuzniar, A., Gavai, A., Thompson, M., Kaliyaperumal, R., Bolleman, J., & Dumontier, M. (2017). Interoperability and FAIRness through a novel combination of Web technologies. PeerJ Computer Science, 3, e110.

(Wilkinson et al., 18) Wilkinson, M.D., et al. Evaluating FAIR-compliance through an objective, automated, community-governed framework. *BioRxiv*, 2018, p. 418376. <https://doi.org/10.1101/418376>

(Wilkinson et al., 19) Wilkinson, M.D., et al. Evaluating FAIR maturity through a scalable, automated, community-governed framework. *Sci Data*, 2019, vol. 6, no. 174 (2019). <https://doi.org/10.1038/s41597-019-0184-5>

3. EXPECTED RESULTS IMPACT

Scientific and Technological Impact

ANGLIRU main results will contribute to the state of the art of research data platforms, bringing them closer to the complete fulfilment of the FAIR principles. The contribution will be based on the use of more advanced knowledge representation languages than those typically used in these platforms. The knowledge representations methodologies and technologies will be based on those promoted by the Semantic Web and currently applied to many Knowledge Graph initiatives, like Wikidata or DBpedia.

The traditional barrier for the adoption of semantic technologies has been their complexity, which makes them difficult to adopt especially by communities outside computing disciplines. To overcome this problem, ANGLIRU will be built as an extension of existing FAIR repositories, which have seen big adoption, that complements them while keeping the workflow current users are used to mostly unaltered.

To achieve this, ANGLIRU services will be integrated through the API of existing FAIR repositories, like Dataverse API⁵⁶, so they can be easily applied to datasets already loaded into existing repositories. On the contrary, for new datasets, the basic steps of data ingestion and metadata generation will be delegated to the underlying repository on top of which ANGLIRU is deployed.

On the other hand, the added functionality provided by ANGLIRU, geared towards the generation of a richer representation of the data and metadata available from the FAIR repository, will be facilitated by semi-automatic annotators and validators. All will be orchestrated through wizards that will help users drive this part of the process without hampering their user experience while producing high-quality data and metadata, including licensing terms.

In addition to the impact planned through the integration with existing FAIR repositories, ANGLIRU will also seek validation of its contributions by putting its services into practice in specific areas related to the data spaces identified as strategic in the European Data Strategy, particularly the data spaces for agriculture, Green Deal and health.

This validation process will be eased by the multidisciplinary in the project team, including computer science but also agriculture, health, library and information science experts. Moreover, focusing on these particular data spaces will also serve as a way of leveraging ANGLIRU in the corresponding research communities.

⁵⁶ <https://guides.dataverse.org/en/latest/api/>

Finally, from a scientific standpoint, the features provided by ANGLIRU to facilitate data reusability will also be a relevant impact driver. ANGLIRU will make data available using a wider and richer range of formats. This applies even for tabular formats, which in the case of ANGLIRU will include unambiguous descriptions of columns, values or units of measure thanks to semantic descriptions. Though it is particularly true for more advanced representations, like embeddings generated from the knowledge graphs which can be directly used by Artificial Intelligence tools based on Deep Learning.

Results Diffusion Plan

The Diffusion Plan will be defined and tracked in Task 5.2 and includes the diffusion of the project's results including scientific articles, data, software and other documents. Regarding papers, we expect to submit papers to the following journals in which we have already published:

- Semantic Web Journal
- Peerj Computer Science
- Information Systems
- International Journal on Semantic Web and Information Systems
- Journal of Web Semantics

We are also planning to submit papers to the following conferences in which we have already published papers:

- International Semantic Web Conference (ISWC)
- Extended Semantic Web Conference (ESWC)
- Knowledge Graphs and Semantic Web Conference (KGSWC)
- Semantics Conference (Semantics)
- Wikidata Conference (WikidataCon)
- International Conference on Information Integration and Web-based Applications & Services (iiWAS)

Our goal is to obtain the following publication results during the 3 years duration of the project:

Publication Result	Amount
Number of publications in indexed journals	15
Number of publications in Q1 indexed journals	4
Number of books or book chapters	6
Open access publications	10
International conferences	14

All publications will be made available through institutional repositories and under open access whenever possible, preferably under golden open access, to increase readership and citations. In case that option is not possible or there is not enough funding for article processing fees, journal policies will be analysed to at least make postprints or reprints available. The other project outputs like software and data will be also made openly available as detailed later in the Data Management Plan subsection.

Moreover, as part of Task 5.2, we will also create a web page about the project to collect the different publications and results, as well as promote the use of other social network tools like Twitter, LinkedIn, ResearchGate, Academia, etc. to achieve even wider diffusion of the project outcomes.

Knowledge Transfer Plan

The Dataverse Project, an open-source research data repository, is one of the most used FAIR repositories with more than 65 installations worldwide, more than 100 contributors, 138.000 datasets and more than 29 million file downloads⁵⁷. The Dataverse Project, through its co-principal investigator Mercè Crosas, has shown its interest in the results of the project and its willingness to collaborate so the contributions made by ANGLIRU can be integrated into Dataverse's codebase and reach all its users. In addition to Dataverse, other initiatives that have not confirmed their interest but that will be contacted during the development of the project are also being considered as potential targets for the knowledge transfer plan regarding FAIR repositories, they include: Zenodo, European Open Science Cloud (EOSC), Research Data Alliance (RDA), GoFAIR, CODATA and FAIRsFAIR.

Apart from that, the following people and institutions have already expressed their support for the project by providing letters of support:

- Sören Auer, Professor for Data Science and Digital Libraries, Leibniz University of Hannover, Director TIB Leibniz Information Centre for Science and Technology, German National Library
- Thomas Baker, Dublin Core Metadata Initiative (DCMI) directorate
- Dan Brickley, Schema.org project lead and W3C Community Group chair, Staff Developer Advocate, Google EMEA, London
- Michel Dumontier, Distinguished Professor of Data Science, Institute of Data Science; Brightlands Institute for digital Smart Society
- Hyvönen Eero, Helsinki Centre for Digital Humanities
- Leyla Jael García Castro, Knowledge Management group, Information Centre for Life Sciences, Alemania, RDA Research Metadata Schemas, BioSchemas
- Sebastian Hellmann, Executive Director of the DBpedia Association, Director of the KILT Competence Center at InfAI
- Pablo Hernández, academic research coordinator at Madita.es
- Enric Pedrós, Manager, Catalan Cluster of Agricultural Production Means (FEMAC), Spain
- Carlos Mateos, director COM Salud y coordinador Salud Sin Bulos
- María Hernández Reyes Mora, technical manager at HERCULES project
- Andra Waagmeester, Chair of W3C ShEx Community Group, Micelio BV
- Egon Willighagen, Department of Bioinformatics NUTRIM School of Nutrition and Translational Research in Metabolism Faculty of Health, Medicine and Life Sciences, Maastricht University

We consider that the current project proposal could lead to further knowledge transfer contracts or collaborations with some of these institutions.

Data Management Plan

As part of Task 5.1 in the Management WP, a Data Management Plan (DMP) will be prepared and updated along the project life cycle. This DMP will ensure that the collected data is made available following the FAIR principles and with the objective that the data is as open as possible, respecting the regulations on personal data and ethical aspects.

⁵⁷ <https://dataverse.org/metrics>

The datasets that will be generated by the operation of the project include tabular data, for instance, the results of the evaluation of the FAIRness of the services as integrated with different research data, collections of vocabularies, ontologies and data shapes including their alignments. This kind of data will be saved and managed in institutional network or cloud units and restricted to team members till the associated research is completed. Then, it will be published using the institutional research data management repositories and made available through ANGLIRU too as open data using Creative Commons licenses.

However, most of the data to be generated by the project will not be tabular but graph-based. It will be based on the World Wide Web Consortium (W3C) Resource Description Framework (RDF) standard. Specialised databases are necessary for storing this kind of data and thus, the requested storage and processing servers will be used to deploy this kind of databases. There will be dedicated servers at both institutions' data centres to guarantee the quality and security of the data and the services built on top of these servers.

Access control mechanisms will be configured on the graph databases. Just members of the team participating in the development of the services will have direct access to them. All other kinds of uses will be done using the licensing and reuse mechanisms implemented by ANGLIRU. Regarding user profiles, authentication and authorization will be delegated to the underlying repositories to avoid storing personal data in ANGLIRU. For the identities aggregating different user profiles, Self-Sovereign Identities will be used so all personal data will be kept in user devices. Reuse agreements will be also stored in user devices and encrypted in decentralised storage services, also looking to guarantee users' privacy.

Finally, it is important to note that another important output of the project, beyond data, is software. All source code will be made available following the FAIR principles under open licensing terms compatible with those of the research data repositories ANGLIRU will be built on. In the case of Dataverse, it is the Apache License v2.0 and GNU General Public License v2.0 for Zenodo. The repository used to share the code and provide version control and coordination mechanisms, like in the case of Dataverse or Zenodo, will also be GitHub. Reference instances of the ANGLIRU services, based on the shared source code, will be deployed on both partners servers to guarantee the availability of the software for testing purposes during the project.

Social and Economic Impact

One of the main impacts from the social and economic perspective is that ANGLIRU, as an extension of existing research data repositories, contributes to seed infrastructure for data processing, tools for data sharing and governance mechanisms that facilitate data reuse. This is aligned with the European Data Strategy⁵⁸ towards the generation of European data spaces in different strategic domains with special relevance from an economic and social perspective. These data spaces include one for health, another for the Green Deal and, among others, one for agriculture. ANGLIRU seeks to increase its impacts through two use cases in WP4 that target just these data spaces.

First, to help create a European health data space, which is essential for advances in preventing, detecting and curing diseases as well as for informed, evidence-based decisions to improve the accessibility, effectiveness and sustainability of the healthcare systems. This one has been chosen due to previous experience gained by Universidad de Oviedo in this

⁵⁸https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/european-data-strategy_en

domain, as detailed in the Previous Work subsection in Section 2. Special care will be taken regarding the impact this data space has in keeping society informed and fighting disinformation regarding health, which will include the collaboration with Spanish organisations combating disinformation in the health and science domains who have already expressed their support to this proposal, Salud sin Bulos and Maldita.es.

The second use case focuses on supporting the European agriculture data space that seeks to enhance the sustainability, performance and competitiveness of the agricultural sector. This data space is envisioned in the European Data Strategy as the foundation for the processing and analysis of production and other data, allowing for precise and tailored application of production approaches at the farm level. To support the emergence of this innovative data-driven ecosystem, the strategy also stresses the importance of fair contractual agreements among data providers, including farmers, and reusers. Universitat de Lleida has experience in this particular domain gained through participation in projects dealing with agriculture data or reuse agreements supported by blockchain technologies, as detailed in the Previous Work subsection in Section 2.

Finally, it is also relevant to stress that ANGLIRU is also looking to provide relevant results in connection with the Spanish Artificial Intelligence Strategy (ENIA). This strategy highlights the importance of the creation of data spaces that facilitate the application of Artificial Intelligence in areas that also include health and agriculture, among others.

It also highlights some of the barriers that the construction of these data spaces might encounter. They include the complexities of data sharing, the need to guarantee the quality and interoperability of the data, the importance of balancing the power among data providers and users through frameworks for data governance or privacy issues. ANGLIRU will make relevant contributions that will help overcome these barriers. Existing FAIR repositories have proven their ability to facilitate data sharing, and ANGLIRU will build on top of these features.

Moreover, through Knowledge Graphs, it will help to guarantee data quality and facilitate its integrations. The use of rich representations for reuse licenses and the trust provided by blockchain technologies will help to build a fairer relationship among data providers and reusers, while respecting privacy.

4. TRAINING CAPACITY

WESO research group is part of the Department of Computer Science, University of Oviedo Spain. The department of computer science offers a PhD programme where the PhD students obtain some specific training from their discipline as well as some general training during the 3 years of the PhD. Thanks to the international relationships developed in the last years about RDF, semantic web, knowledge graphs and bioinformatics we collaborate with Egon Willighagen and Michel Dumontier (Maastricht University), Sebastian Hellmann (University of Leipzig), Iovka Boneva (INRIA, Lille), Alasdair J G Gray (Heriot-Watt University), Katherine Thornton (Yale University), Eric Prud'hommeaux (Janeiro Digital), Ruben Verborgh (Ghent University), etc. where the PhD students can do short research stays to opt for the international mention of the PhD doctorate.

Until this date, the PhD students have already done research stays in:

- INRIA, France (Herminio García González) invited by Iovka Boneva. Year 2019
- Leipzig (Daniel Fernández Álvarez). Year 2018, invited by Sebastian Hellmann.

In the course 2020, we had arranged a research stay in IDLab at Ghent University invited by Ruben Verborgh for Paulino Álvarez de Ondina which had to be cancelled because of COVID-19 and has been postponed for next course.

In the last years, the following PhD theses have been defended:

- Author: Jose María Álvarez Rodríguez, Title: *Métodos semánticos de reutilización de datos abiertos enlazados en las licitaciones públicas*, 2012, supervisor: Jose Emilio Labra Gayo. Jose María Álvarez is currently Associate Professor (Profesor Titular) at Universidad Carlos III de Madrid.
- Author: Hernán Sagastegui Chigne, Title: *Desarrollo de Tecnologías Colaborativas para Entornos virtuales de apoyo al aprendizaje*, 2016, supervisor: Jose Emilio Labra Gayo. Hernán Sagastegui has been teacher at the Universidad Nacional de Trujillo, Perú and at Pontificia Universidad Católica de Perú, he currently teaches at the Universidad Privada Antenor Orrego, Perú. He has been director of the Department of Computer Science and later director of the PhD Programme at the Engineering Faculty. He is currently *profesor principal* which is the same as full professor.
- Author: Jose Alberto Benítez Andrade, Title: *Tecnologías Semánticas aplicadas al Análisis de Redes Sociales en el ámbito de la Salud*, 2017, supervisors: Jose Emilio Labra Gayo and Isaías García Rodríguez. He is currently *Ayudante Doctor* in the University of León and has been accredited as *profesor colaborador doctor* by the Spanish Agency ANECA.
- Author: Guillermo Infante Hernández, title: "GeMTe. Generación Semántica de Modelos de Tramitación Electrónica" 2016, supervisor: Aquilino Adolfo Juan Fuente and Benjamín López Pérez. He currently works in the Taiger company as VP of Technology.
- Daniel Fernández Lanvin: He read his PhD in 2007, supervised by Aquilino Adolfo Juan Fuente and Raúl Izquierdo Castanedo. He is currently Associate professor (*profesor Titular de Universidad*) in the University of Oviedo.

Ongoing PhD thesis:

- Herminio García González, *Integración semántica de grandes fuentes de datos heterogéneas*. Supervised by Jose Emilio Labra Gayo and Juan Manuel Cueva Lovelle. It is planned to be defended in January/February 2021.
- Daniel Fernández Álvarez, *Extracción de conocimiento semántico estructurado mediante minería de medios sociales*, supervised by Jose Emilio Labra Gayo and Daniel Gayo Avello
- Paulino Álvarez de Ondina, *Técnicas de descripción y validación de flujos de datos semánticos*, supervised by Jose Emilio Labra Gayo and Aquilino Adolfo Juan Fuente.
- Francisco Cifuentes Silva, *Tecnologías Semánticas en el ámbito Político-Legislativo*, supervised by Jose Emilio Labra Gayo
- Any Navarro Cabrero, *Técnicas para la medición de calidad en Datos Abiertos y Enlazados*, supervised by Jose Emilio Labra Gayo and Patricia Ordóñez de Pablos.
- Carla Sepúlveda, *Modelo de Ingeniería de Requerimientos para el Diseño de Sistemas de datos heterogéneos*, supervised by Jose Emilio Labra Gayo
- Fernando Bautista Sepúlveda, *Uso de técnicas de navegación automática para la validación de reglas de negocio en proyectos de migración*, supervised by Jose Emilio Labra Gayo

- Claudio Andrés Navarro Cruces, *Tecnologías Semánticas Aplicadas al Modelado de Sistemas Ciber-Físicos*, supervised by Jose Emilio Labra Gayo

Training inside the research group is based on semantic web technologies, big data, data analysis and natural language processing which are needed in this field. Inside the research group it is common that a number of synergies appear by the weekly meetings that the group maintains and also by the contracts and collaborations with companies and institutions which allow the PhD students not only to be in contact with research but also with the industrial world. The research group has already applied for an European project proposal which has not yet resolved.

The GRIHO research group (Human-Computer Interaction and Data Integration) is part of the Computer Science and Engineering Department of the Universitat de Lleida. The members of this department participate, through the Doctoral School, in the PhD Programme in Engineering and Information Technology. This PhD program has been awarded the Mention towards Excellence Reference MEE 2011-0481⁵⁹.

In the context of this program, PhD students receive specialized training in the research lines, among others: Distributed Computing, Cryptography and Graphs, Artificial Intelligence, Human-Computer Interaction, Statistics, Decision Models or Game Theory. The Doctoral School also provides support to PhD students regarding access to accommodation, additional training, travel grants and transition to the labour market. All this in compliance with the European Commission's "European Charter for Researchers" and "Code of Conduct for Recruitment of Researchers" that allowed UdL achieved the HRS4R "HR Excellence in Research" recognition.

The research group has established tight collaborations with other research groups in reference institutions worldwide through many collaborations, like pre and postdoc research stays or joint PhD advisorships.

Research stays:

- DERI Galway - **National University of Ireland**, Ireland (2009)
- Human-Computer Interaction - **Stanford University**, USA (2012)
- Agile Knowledge Engineering and Semantic Web - **University of Leipzig**, Germany (2012)
- Haystack Group - **Massachusetts Institute of Technology**, (USA (2014 & 2015)
- Logic & Intelligent Data Group - **University of Oslo**, Norway (2016)
- Department of Commercial Law - **University of Auckland**, New Zealand (2018)
- Value Sensitive Design Research Lab - **University of Washington**, USA (2019)
- Blockchain Lab, **Technical University of Delft**, The Netherlands (2019)
- Web Semantic Oviedo - **Universidad de Oviedo**, Spain (2020)

Defended PhD thesis:

- Héctor Carretié Arangüena, Improving the comparability of financial statements in a changeable context: a Semantic Web-based approach, 2011. Advisors: Beatriz

⁵⁹ Resolution of October 6, 2011, BOE No. 253 of October 20, Annex I

Torvisco Manchón (Universidad Rey Juan Carlos) and Roberto García. Currently Associate Professor at Universidad de Nebrija.

- Llúcia Masip Ardévol, User experience methodology for the design and evaluation of interactive systems, 2013. Advisors: Antoni Granollers and Marta Oliva. Currently User Experience consultant and Frontend developer at GFT Group.
- Josep Maria Brunetti Fernández, Interacting with Semantic Web Data through an Automatic Information Architecture, 2013. Advisor: Roberto García. Currently Solutions Architect and Team Lead at GFT Group.
- Bruno Splendiani, A proposal for the inclusion of accessibility criteria in the authoring workflow of images for scientific articles, 2015. Advisors: Mireia Ribera (Universidad de Barcelona) and Roberto García (UdL). Currently Search Engine Optimization Analyst at iSocialWeb Marketing.
- Teresa Guilera Llados, Empatía en estudiantes de Medicina: estudio psicométrico, biométrico, de evolución y propuestas de intervención psicoeducativa, 2017. Advisors: Jorge Soler and Rosa Gil. Currently a researcher at the Biomedical Research Institute of Lleida.

Ongoing PhD thesis:

- Aitor Corchero Rodríguez, Towards a Semantic Sensor Web Architecture featuring User Interaction and Experts Knowledge CaptureDocument. Advisor: Roberto García. As part of an Industrial Doctorate at the Eurecat Research Centre.
- Jordi Virgili Gomà, Emotion Recognition and Representation using Ontologies by combining EEG, Biometric and Eye Tracking Technologies. Advisor: Rosa Gil and Roberto García. Part-time PhD at the IRTA Research Centre.
- Andrés Felipe Aguirre Aguirre, Propuesta para evaluar la satisfacción de uso en Entornos Virtuales de Aprendizaje. Advisors: Cesar Collazos (Universidad del Cauca, Colombia) and Rosa Gil.
- Roberto Mauricio Cárdenas, Gestión de la propiedad intelectual de objetos de aprendizaje. Advisors: Roberto García and Darío Delgado (Universidad Nacional Abierta y a Distancia, Colombia).
- Nidhi Arora, Explainability of Knowledge Graphs Embeddings (tentative). Sarika Jain (National Institute of Technology, Kurukshetra, India) and Roberto García.

19.2 Resolución

A continuación se incluye la propuesta de resolución definitiva publicada el 3 de Septiembre de 2021. Se incluyen únicamente las 3 primeras páginas con el texto de la resolución, la página 82 del Anexo I que incluye los detalles del proyecto y la página 51 del anexo II que incluye los datos económicos.



MINISTERIO
DE CIENCIA
E INNOVACIÓN



RESOLUCIÓN DE LA PRESIDENCIA DE LA AGENCIA ESTATAL DE INVESTIGACIÓN, POR LA QUE SE CONCEDEN AYUDAS CORRESPONDIENTES A LA CONVOCATORIA 2020 DE «PROYECTOS DE I+D+i» EN EL MARCO DE LOS PROGRAMAS ESTATALES DE GENERACIÓN DE CONOCIMIENTO Y FORTALECIMIENTO CIENTÍFICO Y TECNOLÓGICO DEL SISTEMA DE I+D+i Y DE I+D+i ORIENTADA A LOS RETOS DE LA SOCIEDAD.

Mediante la Orden CNU/320/2019, de 13 de marzo, se aprobaron las bases reguladoras para la concesión de ayudas públicas del Programa Estatal de Generación de Conocimiento y Fortalecimiento Científico y Tecnológico del Sistema de I+D+i y del Programa Estatal de I+D+i Orientada a los Retos de la Sociedad, en el marco del Plan Estatal de Investigación Científica y Técnica y de Innovación 2017-2020.

En aplicación de dichas bases reguladoras, mediante resolución de 11 de noviembre de 2020 de la Presidencia de la Agencia Estatal de Investigación, se aprobó la convocatoria para el año 2020 del procedimiento de concesión de ayudas correspondiente a la convocatoria de «Proyectos de I+D+i» de los Programas Estatales de Generación de Conocimiento y Fortalecimiento Científico y Tecnológico del Sistema de I+D+i y de I+D+i Orientada a los Retos de la Sociedad, en el marco del Plan Estatal de Investigación Científica y Técnica y de Innovación 2017-2020. (Identificador Base de Datos Nacional de Subvenciones: 532399; extracto publicado en el «Boletín Oficial del Estado» de 17 de noviembre de 2020).

En base a los citados textos legales y de conformidad con lo establecido en los artículos 18 y 19 de la citada orden de bases reguladoras, y los artículos 16 y 18 de la resolución de convocatoria, el órgano instructor, a la vista del expediente y del informe de la comisión de evaluación nombrada al efecto, dictó en fecha 18 de junio de 2021 la correspondiente propuesta de resolución provisional, así como una complementaria a la misma el 23 de julio de 2021, y la propuesta de resolución definitiva en fecha 3 de septiembre. Una vez evacuado el trámite de audiencia y previa aceptación de los interesados, esta presidencia

RESUELVE:

1. Conceder, con cargo a las aplicaciones presupuestarias 28.303.000X.711, 28.303.000X.713, 28.303.000X.715, 28.303.000X.717, 28.303.000X.730, 28.303.000X.731, 28.303.463B.740, 28.303.463B.750, 28.303.463B.760 y 28.303.463B.780, según la naturaleza de los beneficiarios y el tipo de ayudas, o aquellas que las sustituyan en ejercicios presupuestarios posteriores al corriente, las ayudas que se especifican en los anexos I y II, correspondientes a los Programas Estatales de Generación de Conocimiento y Fortalecimiento Científico y Tecnológico del Sistema de I+D+i y de I+D+i Orientada a los Retos de la Sociedad, convocatoria de 2020, por importe de 411.303.552,11 €, distribuido en las siguientes anualidades:

Aplicación Económica	2021	2022	2023	2024	TOTAL
28-303-000X-711	88.862,40	144.401,40	322.126,20	0,00	555.390,00
28-303-000X-713	733.405,20	571.870,20	1.083.555,00	1.353.094,60	3.741.925,00
28-303-000X-715	690.653,48	1.069.631,53	2.369.771,69	114.986,30	4.245.043,00
28-303-000X-717	32.089,20	20.291,70	35.392,50	69.526,60	157.300,00
28-303-000X-730	14.764.451,74	20.726.831,91	45.146.832,77	5.352.371,57	85.990.487,99
28-303-000X-731	21.296,00	34.606,00	77.198,00	0,00	133.100,00
28-303-463B-740	1.700.004,99	2.338.646,03	5.069.048,78	547.495,20	9.655.195,00
28-303-463B-750	50.122.305,62	69.113.642,82	150.208.473,59	23.623.727,19	293.068.149,22
28-303-463B-760	72.291,69	55.899,22	105.611,83	134.400,26	368.203,00
28-303-463B-780	2.202.315,23	3.302.102,08	7.280.471,02	603.870,57	13.388.758,90
Total	70.427.675,55	97.377.922,89	211.698.481,38	31.799.472,29	411.303.552,11

2. Desestimar la concesión de ayudas al resto de solicitudes al no haber alcanzado la prioridad suficiente para ser financiadas, no haber superado alguno de los umbrales establecidos para los criterios de evaluación o no haber sido propuesto para financiación el subproyecto coordinador, conforme a la propuesta de resolución definitiva de la convocatoria, publicada el 3 de septiembre en la página web de la Agencia Estatal de Investigación.

CONDICIONES DE LA AYUDA

Todas las entidades beneficiarias deberán cumplir las condiciones que se recogen en el presente documento, así como las establecidas en la resolución de convocatoria, en la orden de bases que la rige y en las instrucciones de ejecución y justificación, que estarán disponibles en la página web de la Agencia, así como en el resto de normativa aplicable.

Fecha de inicio del período de ejecución de los proyectos

La fecha de inicio de ejecución de los proyectos y su duración figura en el anexo I, excepto para los proyectos de tipo JIN cuya fecha de inicio coincidirá con la de incorporación del/de la IP al centro de adscripción.

Costes indirectos

Los costes indirectos, entendidos como gastos generales asignados al proyecto pero que por su naturaleza no pueden imputarse de forma directa, serán el 21 % de los gastos directos concedidos.

Personal investigador

Los/as IP y las personas que participan en los equipos de investigación son los que figuran en las comunicaciones recibidas a través de Facilit@.

Esta resolución se publicará en la página web de la Agencia Estatal de Investigación, surtiendo todos los efectos de notificación practicada.

Esta resolución es definitiva en vía administrativa y contra la misma cabe interponer, potestativamente, recurso de reposición ante el mismo órgano que la dictó en el plazo de un



mes, de acuerdo con lo dispuesto en los artículos 123 y 124 de la Ley 39/2015, de 1 de octubre, del Procedimiento Administrativo Común de las Administraciones Públicas; o directamente, recurso contencioso-administrativo ante los Juzgados Centrales de lo Contencioso-Administrativo, en el plazo de dos meses, conforme a lo establecido en los artículos 9.1.c) y 46.1 de la Ley 29/1998, de 13 de julio, Reguladora de la Jurisdicción Contencioso-Administrativa.

LA PRESIDENTA DE LA AGENCIA ESTATAL DE INVESTIGACIÓN

P.D. (Resolución de 27 de julio de 2018, BOE n.º 184 de 31 de julio)

El director de la Agencia Estatal de Investigación

Fdo.: Enrique Playán Jubillar



FIRMADO

FIRMADO por : ENRIQUE PLAYAN JUBILLAR, A fecha : 28/09/2021 09:36:00
 El documento consta de un total de 204 folios. Folio 85 de 204 - Código Seguro de Verificación: 1417139-31732918. Verificable en <https://serviciosede.mineco.gob.es/csv/> según Orden Ministerial del 24/2/2011

Nº	REFERENCIA	SUB ÁREA	ENTIDAD SOLICITANTE	CENTRO	NIF	COMUNIDAD AUTÓNOMA	AÑOS	FECHA INICIO (*)	Nº CONT. PREDOC
2090	PID2020-118575RB-I00	DER	UNIVERSIDAD DE NAVARRA	FACULTAD DE DERECHO	R3168001J	NAVARRA	4	01/09/2021	0
2091	PID2020-119328GA-I00	EDU	UNIVERSIDAD DE NAVARRA	FACULTAD DE EDUCACIÓN Y PSICOLOGÍA	R3168001J	NAVARRA	3	01/09/2021	1
2092	PID2020-115348RB-I00	ALI	UNIVERSIDAD DE NAVARRA	FACULTAD DE FARMACIA Y NUTRICIÓN	R3168001J	NAVARRA	3	01/09/2021	1
2093	PID2020-119729GB-I00	ESN	UNIVERSIDAD DE NAVARRA	FACULTAD DE FARMACIA Y NUTRICIÓN	R3168001J	NAVARRA	3	01/09/2021	0
2094	PID2020-116009GB-I00	LFL	UNIVERSIDAD DE NAVARRA	FACULTAD DE FILOSOFÍA Y LETRAS	R3168001J	NAVARRA	3	01/01/2021	0
2095	PID2020-113691RB-I00	EYA	UNIVERSIDAD DE NAVARRA	INSTITUTO CULTURA Y SOCIEDAD	R3168001J	NAVARRA	3	01/01/2021	0
2096	PID2020-116128GB-I00	HIS	UNIVERSIDAD DE NAVARRA	INSTITUTO CULTURA Y SOCIEDAD	R3168001J	NAVARRA	3	01/09/2021	0
2097	PID2020-117519GB-I00	HIS	UNIVERSIDAD DE NAVARRA	INSTITUTO CULTURA Y SOCIEDAD	R3168001J	NAVARRA	3	01/09/2021	0
2098	PID2020-120589RA-I00	EYA	UNIVERSIDAD DE NAVARRA	INSTITUTO CULTURA Y SOCIEDAD	R3168001J	NAVARRA	3	01/09/2021	0
2099	PID2020-120590GB-I00	ART	UNIVERSIDAD DE NAVARRA	INSTITUTO CULTURA Y SOCIEDAD	R3168001J	NAVARRA	4	01/09/2021	0
2100	PID2020-112792GA-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2101	PID2020-113259RA-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2102	PID2020-115069GB-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2103	PID2020-116135GB-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2104	PID2020-118803GB-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2105	PID2020-118807RB-I00	EYF	UNIVERSIDAD DE NAVARRA	INSTITUTO DE ESTUDIOS SUPERIORES DE LA EMPRESA. BARCELONA	R3168001J	CATALUÑA	3	01/09/2021	0
2106	PID2020-112957RA-I00	EYF	UNIVERSIDAD DE NAVARRA	UNIVERSIDAD DE NAVARRA	R3168001J	NAVARRA	3	01/09/2021	0
2107	PID2020-113896GB-I00	AYF	UNIVERSIDAD DE OVIEDO	DPTO. BIOLOGÍA DE ORGANISMOS Y SISTEMAS	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2108	PID2020-113062RB-I00	BTC	UNIVERSIDAD DE OVIEDO	DPTO. BIOLOGÍA FUNCIONAL	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2109	PID2020-117185RB-I00	CAN	UNIVERSIDAD DE OVIEDO	DPTO. BIOQUÍMICA Y BIOLOGÍA MOLECULAR	Q3318001I	PDO.ASTURIAS	3	01/09/2021	1
2110	PID2020-118394RB-I00	FOS	UNIVERSIDAD DE OVIEDO	DPTO. BIOQUÍMICA Y BIOLOGÍA MOLECULAR	Q3318001I	PDO.ASTURIAS	3	01/09/2021	1
2111	PID2020-114478RB-C22	EDU	UNIVERSIDAD DE OVIEDO	DPTO. CIENCIAS DE LA EDUCACIÓN	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2112	PID2020-114707GB-I00	DER	UNIVERSIDAD DE OVIEDO	DPTO. CIENCIAS JURÍDICAS BÁSICAS	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2113	PID2020-113895GB-C33	ICA	UNIVERSIDAD DE OVIEDO	DPTO. CONSTRUCCIÓN E INGENIERÍA DE FABRICACIÓN	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2114	PID2020-116096RB-I00	DER	UNIVERSIDAD DE OVIEDO	DPTO. DERECHO PÚBLICO	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2115	PID2020-113486RB-I00	LFL	UNIVERSIDAD DE OVIEDO	DPTO. FILOLOGÍA ANGLOGERMÁNICA Y FRANCESA	Q3318001I	PDO.ASTURIAS	3	01/09/2021	1
2116	PID2020-113951GB-I00	FYA	UNIVERSIDAD DE OVIEDO	DPTO. FÍSICA	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2117	PID2020-115269GB-I00	POL	UNIVERSIDAD DE OVIEDO	DPTO. GEOGRAFÍA	Q3318001I	PDO.ASTURIAS	4	01/09/2021	0
2118	PID2020-118223RB-C21	CTA	UNIVERSIDAD DE OVIEDO	DPTO. GEOLOGÍA	Q3318001I	PDO.ASTURIAS	4	01/09/2021	0
2119	PID2020-112726RB-I00	INF	UNIVERSIDAD DE OVIEDO	DPTO. INFORMATICA	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2120	PID2020-117912RB-C21	INF	UNIVERSIDAD DE OVIEDO	DPTO. INFORMATICA	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2121	PID2020-119082RB-C22	TCO	UNIVERSIDAD DE OVIEDO	DPTO. INFORMATICA	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0
2122	PID2020-117282RB-I00	IEA	UNIVERSIDAD DE OVIEDO	DPTO. INGENIERÍA ELÉCTRICA, ELECTRÓNICA, DE COMPUTADORES Y SISTEMAS	Q3318001I	PDO.ASTURIAS	3	01/09/2021	0

ANEXO I. DATOS GENERALES

(*) Tipo JIN: fecha incorporación IP al centro de adscripción

«PROYECTOS DE I+D+i» - CONVOCATORIA 2020

82/124

Nº	REFERENCIA	Financiación (€)						
		Total concedido ¹	Por concepto de gasto		Por anualidades			
			Costes directos	Costes indirectos	2021	2022	2023	2024
2099	PID2020-120590GB-I00	42.350,00	35.000,00	7.350,00	8.639,40	5.463,15	9.528,75	18.718,70
2100	PID2020-112792GA-I00	27.757,40	22.940,00	4.817,40	4.441,18	7.216,92	16.099,30	0,00
2101	PID2020-113259RA-I00	13.987,60	11.560,00	2.427,60	2.238,02	3.636,78	8.112,80	0,00
2102	PID2020-115069GB-I00	50.118,20	41.420,00	8.698,20	8.018,91	13.030,73	29.068,56	0,00
2103	PID2020-116135GB-I00	74.197,20	61.320,00	12.877,20	11.871,55	19.291,27	43.034,38	0,00
2104	PID2020-118803GB-I00	10.551,20	8.720,00	1.831,20	1.688,19	2.743,31	6.119,70	0,00
2105	PID2020-118807RB-I00	36.360,50	30.050,00	6.310,50	5.817,68	9.453,73	21.089,09	0,00
2106	PID2020-112957RA-I00	38.078,70	31.470,00	6.608,70	6.092,59	9.900,46	22.085,65	0,00
2107	PID2020-113896GB-I00	133.100,00	110.000,00	23.100,00	21.296,00	34.606,00	77.198,00	0,00
2108	PID2020-113062RB-I00	211.750,00	175.000,00	36.750,00	33.880,00	55.055,00	122.815,00	0,00
2109	PID2020-117185RB-I00	266.200,00	220.000,00	46.200,00	42.592,00	69.212,00	154.396,00	0,00
2110	PID2020-118394RB-I00	411.400,00	340.000,00	71.400,00	65.824,00	106.964,00	238.612,00	0,00
2111	PID2020-114478RB-C22	30.371,00	25.100,00	5.271,00	4.859,36	7.896,46	17.615,18	0,00
2112	PID2020-114707GB-I00	24.200,00	20.000,00	4.200,00	3.872,00	6.292,00	14.036,00	0,00
2113	PID2020-113895GB-C33	9.317,00	7.700,00	1.617,00	1.490,72	2.422,42	5.403,86	0,00
2114	PID2020-116096RB-I00	44.165,00	36.500,00	7.665,00	7.066,40	11.482,90	25.615,70	0,00
2115	PID2020-113486RB-I00	58.080,00	48.000,00	10.080,00	9.292,80	15.100,80	33.686,40	0,00
2116	PID2020-113951GB-I00	133.100,00	110.000,00	23.100,00	21.296,00	34.606,00	77.198,00	0,00
2117	PID2020-115269GB-I00	268.620,00	222.000,00	46.620,00	54.798,48	34.651,98	60.439,50	118.730,04
2118	PID2020-118228RB-C21	181.500,00	150.000,00	31.500,00	37.026,00	23.413,50	40.837,50	80.223,00
2119	PID2020-112726RB-I00	45.738,00	37.800,00	7.938,00	7.318,08	11.891,88	26.528,04	0,00
2120	PID2020-117912RB-C21	65.703,00	54.300,00	11.403,00	10.512,48	17.082,78	38.107,74	0,00
2121	PID2020-119082RB-C22	43.802,00	36.200,00	7.602,00	7.008,32	11.388,52	25.405,16	0,00
2122	PID2020-117282RB-I00	113.619,00	93.900,00	19.719,00	18.179,04	29.540,94	65.899,02	0,00
2123	PID2020-112587RB-I00	242.000,00	200.000,00	42.000,00	38.720,00	62.920,00	140.360,00	0,00
2124	PID2020-113052GB-I00	45.738,00	37.800,00	7.938,00	7.318,08	11.891,88	26.528,04	0,00
2125	PID2020-117265GB-I00	145.200,00	120.000,00	25.200,00	23.232,00	37.752,00	84.216,00	0,00
2126	PID2020-114736GB-I00	59.290,00	49.000,00	10.290,00	9.486,40	15.415,40	34.388,20	0,00
2127	PID2020-117259RB-I00	106.480,00	88.000,00	18.480,00	17.036,80	27.684,80	61.758,40	0,00
2128	PID2020-115204RB-I00	181.500,00	150.000,00	31.500,00	29.040,00	47.190,00	105.270,00	0,00
2129	PID2020-118376RA-I00	84.700,00	70.000,00	14.700,00	13.552,00	22.022,00	49.126,00	0,00
2130	PID2020-119087RB-I00	181.500,00	150.000,00	31.500,00	29.040,00	47.190,00	105.270,00	0,00
2131	PID2020-113473GB-I00	145.200,00	120.000,00	25.200,00	23.232,00	37.752,00	84.216,00	0,00
2132	PID2020-117021GB-I00	48.400,00	40.000,00	8.400,00	7.744,00	12.584,00	28.072,00	0,00
2133	PID2020-112839RB-I00	85.910,00	71.000,00	14.910,00	13.745,60	22.336,60	49.827,80	0,00
2134	PID2020-114172RB-C21	281.325,00	232.500,00	48.825,00	45.012,00	73.144,50	163.168,50	0,00
2135	PID2020-115313RB-I00	146.410,00	121.000,00	25.410,00	23.425,60	38.066,60	84.917,80	0,00
2136	PID2020-118499GB-C32	14.762,00	12.200,00	2.562,00	3.011,45	1.904,30	3.321,45	6.524,80
2137	PID2020-118499GB-C33	11.737,00	9.700,00	2.037,00	2.394,35	1.514,07	2.640,83	5.187,75
2138	PID2020-112585RA-I00	39.809,00	32.900,00	6.909,00	6.369,44	10.350,34	23.089,22	0,00
2139	PID2020-115183RB-C21	76.109,00	62.900,00	13.209,00	12.177,44	19.788,34	44.143,22	0,00
2140	PID2020-112506GB-C43	60.500,00	50.000,00	10.500,00	12.342,00	7.804,50	13.612,50	26.741,00

ANEXO II. DATOS ECONÓMICOS

I Presupuesto total concedido = presupuesto total financiable

«PROYECTOS DE I+D+i» -2020

19.3 Informe de Valoración científico-técnica

En las siguientes páginas se incluye el informe de valoración científico técnica del proyecto.



INFORME DE VALORACIÓN CIENTÍFICO - TÉCNICA

Convocatoria 2020 Proyectos de I+D+i - RTI Tipo Coord	
Referencia:	PID2020-117912RB-C21
Área:	Tecnologías de la información y de las comunicaciones
Subárea:	Ciencias de la computación y tecnología informática
Investigador/a principal	JOSE EMILIO LABRA GAYO
Título:	APLICANDO GRAFOS DE CONOCIMIENTO PARA LA INTEROPERABILIDAD DE DATOS DE INVESTIGACION

(De acuerdo con los principios de discrecionalidad técnica e independencia, este informe de valoración científico técnica se elabora teniendo en cuenta el consenso adoptado en la comisión técnica, una vez analizadas las valoraciones cualitativas de los informes de experto y las opiniones de los expertos de la comisión, y a la vista del conjunto de proyectos presentados a cada área temática. Por ello, este informe no tiene por qué asumir, en todos sus términos, las valoraciones aportadas por los expertos en sus informes, que pueden ser discrepantes y que se reconocen inevitables y legítimas siempre que no incurran en errores técnicos o fácticos graves y manifiestos).

CRITERIOS DE EVALUACIÓN

1. Calidad y viabilidad de la propuesta

Valoración de 0 a 40: 36.0

Umbral: 30

1.1. Calidad

Valoración 0 a 30: 27.5

La propuesta se sustenta en la siguiente idea: la de enriquecer los datos de investigación ¿que pueden formar grandes bases de datos; mediante grafos de conocimiento, que vayan más allá de los metadatos que actualmente se consideran. Ello permitirá su compartición de forma simple, así como su interoperabilidad y reusabilidad. La unificación de datos de investigación es un tema de gran actualidad e interés, potenciado por los esquemas emergentes de ciencia abierta y la deseable compartición de recursos científicos. Se plantea como un proyecto coordinado de dos subproyectos, cuya coordinación está justificada. Con unos objetivos claros, se distinguen los perseguidos por cada uno de los subproyectos. Los miembros del equipo de investigación han liderado con anterioridad otros proyectos de los Planes Nacionales en temas relacionados, por lo que esta propuesta significa una continuación en una línea de trabajo mantenida en el tiempo, en donde se ha conseguido visibilidad internacional. Se valora la calidad en la escritura de la memoria.

1.2 Viabilidad

Valoración 0 a 10: 8.5

Se plantea un único plan de trabajo en el que participan investigadores de los dos subproyectos, que se estructura en paquetes de trabajo que, a su vez, se componen de varias tareas. Se proponen dos casos de uso considerando los dominios de salud y agricultura, que se evalúan en el último paquete de trabajo. Esta estructura y la distribución de tareas entre investigadores parece adecuada. Se analizan algunas posibles contingencias. Se considera que el proyecto es viable, dados los resultados previos de los grupos solicitantes. Con respecto al presupuesto de este subproyecto C21, las partidas de personal y de adquisición de inventariable parecen sobredimensionadas. Teniendo en cuenta estas consideraciones, el carácter competitivo de la convocatoria y las disponibilidades presupuestarias se propone una reducción del presupuesto.

2. Equipo de investigación

Valoración 0 a 30: 26.48

Umbral: 20

El IP muestra un sólido CV, con publicaciones de calidad, experiencia en la dirección de proyectos y una buena internacionalización. Cuenta con una larga trayectoria en investigación sobre web semántica y grafos de conocimiento, con relevancia internacional. No hay duda de su capacidad para liderar el proyecto coordinado (y este subproyecto). El equipo de investigación, con CVs más modestos, también presenta una actividad interesante en los temas relacionados con el proyecto, con colaboraciones externas y experiencia en el desarrollo de repositorios de datos de investigación. Se considera que, junto con el equipo del subproyecto 2, están suficientemente capacitados para llevar a cabo el plan de trabajo presentado.

3. Impacto

Valoración de 0 a 30: 25.7

3.1. Impacto científico-técnico de los resultados esperados**Valoración 0 a 20: 17.0**

El plan de publicaciones es ambicioso: se estiman 15 artículos en revistas JCR, más varios capítulos de libro, y comunicaciones en congresos internacionales. El avance en el conocimiento esperable se focaliza en la investigación sobre grandes bases de datos para su interoperabilidad y reusabilidad. Los métodos desarrollados se aplicarán en dos dominios reales: salud y agricultura. Es interesante constatar que la propuesta parece haber conseguido apoyos de miembros prominentes de la comunidad internacional de web semántica, y que una plataforma de datos de investigación (Dataverse) ha mostrado interés por los resultados del proyecto.

3.2. Impacto social y económico de los resultados**Valoración 0 a 10: 8.7**

Se estima que los resultados de esta propuesta tendrán un impacto social moderado a corto plazo, que puede aumentar si se logran completamente los objetivos previstos. El plan de difusión es razonable (página web, redes sociales, conferencias, artículos de divulgación) aunque no excepcional. En base a las aplicaciones desarrolladas sobre datos de salud y agricultura, el proyecto puede tener algo más de impacto social.

VALORACIÓN FINAL**Puntuación de 0 a 100: 88.18****OTROS ASPECTOS A CONSIDERAR**

- a) Aspectos relacionados con zonas polares o campañas oceanográficas.(cumplimentar solo en los proyectos que proceda)
- b) Condiciones específicas para la ejecución de determinados proyectos.(cumplimentar solo en los proyectos con aspectos relacionados con las condiciones o implicaciones recogidas en el Anexo IV de la convocatoria)

INFORME-EXPERTO 1

CRITERIOS DE EVALUACIÓN

1. Calidad y viabilidad de la propuesta

1.1. Calidad

Valoración: A (Excepcional)

El proyecto propone avanzar en el conocimiento mediante la construcción de grandes espacios de datos compartidos a partir de distintas bases de datos. Para ello, se pretenden utilizar técnicas que permitan combinar distintos formatos de datos, así como generar conocimiento de los mismos. Se trabajará para conseguir la interoperabilidad y reusabilidad de datos. Tratar de unificar los datos de distintas fuentes es un campo de investigación muy interesante e importante. Los objetivos están bien definidos, y se separan los objetivos que persigue cada uno de los subproyectos. Estos objetivos se complementan muy bien para conseguir el objetivo global del proyecto. Los miembros del equipo de investigación han liderado con anterioridad otros proyectos de los Planes Nacionales en temas relacionados con los de esta propuesta, por lo que supone una continuación a una línea de trabajo consolidada, en la que el IP ha demostrado tener visibilidad internacional. Los subproyectos no necesitan el uno del otro, pero sí se complementan bien y juntos crean sinergia para lograr la interoperabilidad y reusabilidad de datos.

1.2. Viabilidad

Valoración: B (Excelente)

La metodología estructura el trabajo de este proyecto en una serie de paquetes de trabajo, cada uno compuesto por varias tareas. De los paquetes de trabajo, unos son más orientados al subproyecto 1 y otros al subproyecto 2, aunque en todos participan investigadores de ambos subproyectos. En un último paquete de trabajo se propone la evaluación de los métodos desarrollados en dos casos reales, relacionados con la salud y con la agricultura. La distribución de las tareas entre los dos subproyectos parece adecuada. La estructuración que se ha hecho en paquetes de trabajo y tareas parece apropiada, y además permite realizar fácilmente un seguimiento preciso de los progresos del proyecto y comprobar si se avanza según lo planificado. La metodología cuenta con la identificación de riesgos que se puedan dar y sus planes de contingencia, pero no se identifican posibles problemas en la consecución de las distintas tecnologías que se pretenden desarrollar, y de la integración entre ellas. Por ejemplo, hay riesgos que pueden afectar seriamente la viabilidad del proyecto, como por ejemplo en las integraciones de ANGLIRU con Zenodo y Dataverse, o en la generación del KG Enricher, sólo en el paquete de trabajo 1. Los riesgos identificados son sólo a nivel de retrasos en la publicación de artículos y la producción de software, o de recursos humanos (que un miembro del equipo abandone el proyecto o problemas de coordinación). En este último sentido, se podrían considerar posibles problemas por retrasos en la contratación de personal. El equipo no cuenta con recursos suficientes como para afrontar todas las tareas que tiene asignadas, aunque resulta excesivo que soliciten dos ingenieros (ambos a tiempo completo durante 30 meses) y dos estudiantes de doctorado. Con respecto al equipamiento, se solicitan un switch, tres servidores y un frontend para infraestructura HPC que suponen un elevado coste y no se antojan imprescindibles. Por otro lado, creo que sí sería necesario el servidor de almacenamiento.

2. Equipo de investigación

Valoración: B (Excelente)

El IP de este proyecto coordinado cuenta con una excelente trayectoria en el campo de la web semántica y de los grafos de conocimiento, demostrando relevancia en la comunidad internacional. Tiene una gran actividad a nivel de artículos y proyectos de investigación y de transferencia. Tiene una trayectoria destacada en el campo de la web semántica, con muchos años de experiencia y publicaciones de impacto. También cuenta con experiencia reciente en el uso de bases de datos heterogéneas. El equipo de investigación tiene experiencia contrastada en el desarrollo de repositorios de datos para fines de investigación. El equipo cuenta con multitud de colaboraciones con grupos extranjeros, que en algunos casos se ha materializado en proyectos internacionales.

3. Impacto

3.1. Impacto científico-técnico de los resultados esperados

Valoración: B (Excelente)

La principal contribución del proyecto será avanzar en el estado del arte de la investigación sobre grandes plataformas de bases de datos, permitiendo la interoperabilidad con distintas bases de datos, así como su reusabilidad. Las técnicas desarrolladas se utilizarán en dos aplicaciones reales, relacionadas con datos sobre salud y agricultura. El hecho de poder construir las plataformas de datos a partir de bases de datos de distintos formatos y características, supone ya una importante contribución, ya que en multitud de ámbitos científicos se trabaja con grandes cantidades de datos y, en muchos casos, se ven limitados por la existencia de bases de datos con distintas características. El IP cuenta con el interés de uno de los investigadores principales del proyecto Dataverse, uno de los principales repositorios de datos de investigación en la literatura, que ha mostrado interés en los resultados de este proyecto y la posibilidad de incluir sus resultados. Además, el IP cuenta con muestras de interés de multitud de investigadores nacionales y extranjeros en el proyecto. El plan de publicaciones es bastante ambicioso, con 15 publicaciones en revistas JCR, varios libros o capítulos de libro, así como conferencias internacionales.

3.2. Impacto social y económico de los resultados

Valoración: B (Excelente)

El proyecto considera un paquete de trabajo dedicado a la gestión del proyecto y a la difusión de los resultados. Se contempla la creación de una página web en la que se mostrarán los resultados obtenidos en el proyecto, incluyendo enlaces a proyectos relacionados y a redes sociales. Además, se prevé la participación en conferencias y el envío de artículos de investigación a revistas. Como se muestra en las aplicaciones seleccionadas, el proyecto puede tener un impacto considerable a nivel social, ya que los espacios unificados de datos en salud y agricultura pueden ayudar a mejorar el bienestar.

VALORACIÓN FINAL

Valoración: B=Muy bueno

OTROS ASPECTOS A CONSIDERAR

- a) Capacidad formativa.(cumplimentar solo cuando se haya solicitado la inclusión del proyecto en la convocatoria de contratos predoctorales para la formación de doctores)

Valoración: B (Excelente)

En el programa de doctorado en ingeniería informática ofrecido por la Universidad de Oviedo los estudiantes reciben formación específica en su disciplina, además de formación genérica. El grupo de investigación cuenta con un gran abanico de grupos extranjeros con los que colabora habitualmente, donde el estudiante puede realizar estancias de investigación cortas. En los últimos 10 años, el equipo de investigación ha dirigido 5 tesis doctorales, cuyos autores trabajan en la universidad como profesores (la mayoría de ellos) o en el sector privado. Además, tienen otras 9 tesis doctorales en curso en la actualidad. En el grupo de investigación hacen a menudo reuniones para intentar crear sinergias entre las líneas de investigación de sus componentes. También los estudiantes de doctorado tienen contacto con el mundo industrial, lo que complementa su formación.

- b) Aspectos relacionados con zonas polares o campañas oceanográficas. (cumplimentar solo en los proyectos que proceda)

No aplica

- c) Condiciones específicas para la ejecución de determinados proyectos.(cumplimentar solo en los proyectos con aspectos relacionados con las condiciones o implicaciones recogidas en el Anexo IV de la convocatoria)

No aplica

INFORME-EXPERTO 2

CRITERIOS DE EVALUACIÓN

1.Calidad y viabilidad de la propuesta

1.1. Calidad

Valoración: B (Excelente)

La adecuación al reto 7 de economía, sociedad y cultura digitales es adecuada y aporta soluciones novedosas. La propuesta está bien justificada. La hipótesis de partida y los objetivos son relevantes y están justificados en su contexto de aplicación. Los resultados de los proyectos anteriores son buenos aunque el impacto es moderado. La coordinación de los dos proyectos está justificada adecuadamente y aporta valor añadido. La tecnología a desarrollar se aplica en múltiples dominios multidisciplinares.

1.2. Viabilidad

Valoración: B (Excelente)

La distribución de actividades en los paquetes de trabajo está bien planificada y es coherente con los objetivos. La metodología incorpora algunos elementos novedosos y contempla validaciones en múltiples áreas de uso. El plan de contingencia es realista. La distribución de tareas entre los dos grupos coordinados es adecuada. El conjunto de resultados previos avala en parte la propuesta. El trabajo puede depender en exceso de los contratos de personal (con coste alto) y de los contratos FPI.

2. Equipo de investigación

Valoración: B (Excelente)

La experiencia en la temática propuesta de los IPS y de los dos equipos es alta y sus publicaciones relevantes. En conjunto, los resultados previos avalan los objetivos planteados. Las colaboraciones internacionales no se detallan en el proyecto más allá de su posible uso para las estancias formativas y el apoyo al proyecto.

3. Impacto

3.1. Impacto científico-técnico de los resultados esperados

Valoración: B (Excelente)

El impacto científico-técnico puede ser muy alto por su posible uso en temáticas muy diversas en el contexto del reto abordado, de un carácter más tipo aplicado que en la frontera de conocimiento. Los mecanismos de difusión científico-técnica son adecuados. El plan de transferencia y gestión de datos es adecuado.

3.2. Impacto social y económico de los resultados

Valoración: B (Excelente)

El plan de difusión a la sociedad es adecuado y destacable su carácter abierto. El plan de contactos con empresas también es adecuado y favorecerá el impacto del proyecto.

VALORACIÓN FINAL

Valoración: B=Muy bueno

OTROS ASPECTOS A CONSIDERAR

a) Capacidad formativa.(cumplimentar solo cuando se haya solicitado la inclusión del proyecto en la convocatoria de contratos predoctorales para la formación de doctores)

Valoración: B (Excelente)

El programa de formación es adecuado, la capacidad formativa es alta y el contexto científico-técnico de los dos grupos es propicio para una buena formación. El número de tesis simultáneas supervisadas puede ser excesivo (9 actualmente más las dos propuestas).

b) Aspectos relacionados con zonas polares o campañas oceanográficas. (cumplimentar solo en los proyectos que proceda)

N/A

c) Condiciones específicas para la ejecución de determinados proyectos.(cumplimentar solo en los proyectos

con aspectos relacionados con las condiciones o implicaciones recogidas en el Anexo IV de la convocatoria

N/A

INFORME-EXPERTO 3

CRITERIOS DE EVALUACIÓN

1. Calidad y viabilidad de la propuesta

1.1. Calidad

Valoración: A (Excepcional)

Después de analizar la documentación aportada, considero que tanto el contenido como la presentación del proyecto son de una calidad excepcional, basada en una sólida experiencia de los participantes tanto en desarrollo como en transferencia de tecnología, y se ajusta perfectamente al reto 7 (Economía digital...), y (en segundo orden) al reto 6 (...science for society...). En este sentido, creo que se cumple la necesaria adecuación a la convocatoria. El proyecto está muy bien explicado, tanto en contenidos, origen histórico e interés, como en lo que respecta al plan de trabajo, temporización, parcelación del trabajo, explicación de objetivos y entregables previstos. Las tablas elaboradas, la bibliografía aportada, la especificidad de la presentación de ideas y la explicación de la actuación de los diferentes miembros del equipo de investigación y del grupo de trabajo permiten una buena evaluación. Con respecto al punto de partida del proyecto, está sólidamente justificado por el trabajo realizado por los dos grupos de investigación implicados en los años anteriores, lo que también da confianza sobre la posibilidad real del proyecto de completarse con éxito. Debe tenerse en cuenta que los objetivos están dirigidos a la obtención de herramientas informáticas concretas que facilitarían mucho el acceso a los datos de investigación, tema de actualidad y de transcendencia social, puesto todavía hay muchos aspectos a mejorar relacionados con la gestión, reusabilidad, interoperabilidad de los datos de investigación, que deberían estar a disposición para todos los investigadores, en el contexto de la ciencia abierta y la deseable potenciación de la compartición de recursos científicos. Los objetivos marcados en este proyecto, en este sentido, son muy prometedores. Con respecto al equipo de investigación, ha estado implicado en el desarrollo de muchos proyectos, con éxito, hasta donde es posible valorar con la información aportada. Sólo se podría cuestionar, como comentaré en otros apartados, la relativamente baja tasa de publicaciones en revistas científicas (en número, no en calidad) de algunos de los miembros del equipo de investigación (aunque la producción científica de los IPs es relevante), comparada con otros grupos, pero esto puede ser debido a que el grupo tiene centrada su actividad en la producción de software y en proyectos eminentemente aplicados. La necesidad de coordinación entre los dos grupos también está muy justificada, puesto que son expertos en capacidades complementarias que son necesarias para llevar a cabo el proyecto de forma exitosa. El proyecto está adaptado a esa coordinación, por lo que opino que sería difícil desarrollarlo sin la participación de alguno de los dos grupos. Como he dicho anteriormente, la estrategia de coordinación, muy bien explicada, es perfectamente factible.

1.2. Viabilidad

Valoración: B, Muy bueno

El esfuerzo puesto en la realización de la memoria, en la explicación de sus fases, de los participantes que analizarán cada objetivo, la segmentación del trabajo y la experiencia de la que disponen los dos grupos, permiten calificar las actividades propuestas como óptimas para la obtención de los objetivos. La metodología propuesta es muy técnica, y escapa en parte a mis conocimientos, pero viene avalada por la solvencia de los grupos implicados y por una muy precisa especificación tanto de fuentes, relaciones y contenido. Son también adecuadas las tareas y las fechas en las que se realizarán, junto con la personalización de los trabajos y su partición, perfectamente explicadas en el cronograma. Aunque la contingencia es apuntada en la memoria, no creo que el proyecto pueda pasar por fases críticas para su desarrollo. El equipo humano, por tanto, comprende especialistas sobresalientes en todos los temas necesarios, y, por la memoria, entiendo que se dispone de los equipos necesarios que no se solicitan. Sin embargo, la contratación prevista me parece excesiva para los objetivos marcados, dado que el equipo ya cuenta con investigadores formados y los nuevos contratos no se harían a personal tan altamente cualificado como ellos, por lo que entiendo que la mayor parte del trabajo se deberá desarrollar por parte de los investigadores que ya figuran en el proyecto. Como en el caso del subproyecto 2, la contratación de dos investigadores/técnicos, para cada uno de los dos proyectos, me parece innecesaria. El resto de las partidas creo que están justificadas, dadas las necesidades de difusión y colaboración para llevar a cabo el proyecto. Finalmente, como he explicado en el apartado anterior, creo que el reparto entre tareas entre los dos grupos participantes es óptimo.

2. Equipo de investigación

Valoración: B, Muy bueno

El perfil profesional de los investigadores parece adecuado para el proyecto, ya que han mostrado su capacidad para el desarrollo de proyectos similares, alguno de tanta relevancia como el que se solicita en relación a la responsabilidad sobre su desarrollo (si han participado en proyectos europeos). Aunque las iniciativas tanto de desarrollo de material dentro de proyectos científicos como de transferencia son suficientes para demostrar su capacidad, sin embargo los artículos de investigación y comunicaciones a congresos publicados de algunos de los miembros del equipo de investigación no son demasiado numerosos. La capacidad de liderazgo del IP1 está también demostrada por su trabajo desde hace más de diez años, y la competencia de los miembros para el proyecto pienso que está demostrada, ya que su participación en otros proyectos indican que su formación y experiencia profesional es la adecuada. Se colabora habitualmente con grupos internacionales especializados en el tema. El papel que cada miembro tiene en el proyecto está muy bien explicado en la memoria.

3. Impacto

3.1. Impacto científico-técnico de los resultados esperados

Valoración: B, Muy bueno

La ejecución exitosa del proyecto supondría, en mi opinión, un avance muy relevante en la gestión, clasificación, acceso y reutilización de los datos de investigación, permitiendo integrar tanto formatos de documento como diferentes plataformas según su modo de acceso, aportando más información sobre las características de esos datos. Se facilitaría así la compartición de datos entre diferentes grupos de investigación, lo que podría suponer un cambio importante en la forma de investigar. Los planes al respecto de la difusión en medios científicos de los resultados y su transferencia son muy detallados y, desde mi punto de vista, completamente adecuados.

3.2. Impacto social y económico de los resultados

Valoración: C, Bueno

El impacto social, que puede ser relevante, está bien explicado y es adecuado. Como he indicado, se trata de un proyecto que puede conducir a una mejora importante del sistema de gestión de datos de investigación, y se pueden encontrar aplicaciones, tal y como se indica, dentro del plan nacional de desarrollo de inteligencia artificial y en otros temas, igualmente relevantes. No se menciona la participación activa de los usuarios finales para la validación de los resultados, al menos al nivel de grupos concretos de investigadores en otras áreas.

VALORACIÓN FINAL

Valoración: B, Muy bueno

OTROS ASPECTOS A CONSIDERAR

a) Capacidad formativa.(cumplimentar solo cuando se haya solicitado la inclusión del proyecto en la convocatoria de contratos predoctorales para la formación de doctores)

Valoración: B, Muy bueno

Se explica detalladamente el plan de formación de los posibles estudiantes que pudieran participar, y la información aportada permite entender que el grupo tiene una capacidad formativa elevada, y que además trabajan activamente en la formación de nuevos doctores, al igual que en el caso del otro subproyecto. Los contactos nacionales e internacionales y la implementación de los equipos de trabajo en proyectos de transferencia parecen indicar que la calidad de esa formación, además de la capacidad de formar profesionales capaces de desarrollar su trabajo científico en otros ámbitos distintos de las universidades, son elevadas.

b) Aspectos relacionados con zonas polares o campañas oceanográficas. (cumplimentar solo en los proyectos que proceda)

No procede

c) Condiciones específicas para la ejecución de determinados proyectos.(cumplimentar solo en los proyectos con aspectos relacionados con las condiciones o implicaciones recogidas en el Anexo IV de la convocatoria)

No procede

INFORME-EXPERTO 4

CRITERIOS DE EVALUACIÓN

1. Calidad y viabilidad de la propuesta

1.1. Calidad

Valoración: C, Bueno

La propuesta se declara centrada en el 7º reto, sobre economía, sociedad y cultura digitales, con implicaciones en el 6º reto, ciencias sociales y humanidades y ciencia con y para la sociedad. El objetivo principal del proyecto está relacionado con la creación de espacios de datos de investigación y por tanto con una posible contribución a los objetivos de digitalización y la aplicación de una cultura basada en datos para resolver problemas sociales y económicos. La selección del reto 6º se justifica a través de la aplicación de los resultados del proyecto a los dominios de agricultura y sanidad. Si bien estos objetivos caen dentro de lo que se podría esperar en relación al reto 7º, el alcance de los casos de uso definidos en la propuesta genera dudas sobre el impacto real que pueda tener el proyecto en comunidades existentes de usuarios de datos de investigación. Los objetivos de la propuesta están centrados en un área técnica muy acotada, relacionada con el desarrollo de infraestructuras para la gestión de datos mediante grafos de conocimiento. En ese sentido, dichos objetivos son descritos de forma suficiente, aunque con cierta ambigüedad, y presentan un grado limitado de originalidad. La lectura atenta de la propuesta no revela innovaciones significativas más allá del estado del arte actual en tecnologías semánticas, que son el foco de la propuesta. La propuesta parece centrarse en la generación de infraestructura semántica a un nivel más técnico que de investigación en un dominio, el de los datos de investigación bajo el paradigma FAIR, en el que multitud de iniciativas Europeas, canalizadas en gran medida a través de la European Open Science Cloud (EOSC), están actualmente en marcha. No se aprecian muestras significativas de participación relevante de los proponentes en dichas actividades, más allá de menciones como el proyecto HERCULES a nivel nacional y cierta conexión a través de Mercè Crosas, que no es parte de esta propuesta, con Dataverse. Infraestructuras clave como Zenodo son mencionadas rápidamente como algo externo al proyecto con lo que se intentará establecer contacto durante la ejecución del mismo, aunque no queda claro cómo ni parece existir un plan establecido en ese sentido. La propuesta tiene en cuenta de manera lateral tecnologías basadas en blockchain y menciona la conexión con herramientas utilizadas por investigadores en Inteligencia Artificial, aunque en general no desgrana cuáles ni cómo se relacionan con la propuesta. Sí menciona knowledge graph embeddings como una manera alternativa de representar knowledge graphs en un espacio vectorial. Sería razonable en el marco de una propuesta como esta esperar cierta implicación de áreas de la IA como el procesamiento de lenguaje natural o visión por computador. Sin embargo, los datos en los que se centra la propuesta parecen ser siempre (semi)estructurados. No hay mención a datos de investigación no estructurados como los que se puedan encontrar en el texto o figuras de publicaciones científicas. De hecho, la premisa de la propuesta parece ser que el formato último en el que deberían encontrarse los datos de investigación sea el de un grafo de conocimiento, siendo otros formatos como CSVs o bases de datos relacionales considerados como algo a transformar, y todo el trabajo propuesto se orienta en esa dirección. A pesar de las indudables ventajas que para un investigador pueda suponer el formato de grafo, la realidad es que la resolución de los problemas a los que se enfrentan los investigadores en la gestión de datos de investigación y la consecución de los objetivos FAIR no siempre es una solución técnica, ni tiene por qué pasar necesariamente por adoptar dicho formato.

1.2. Viabilidad

Valoración: C, Bueno

Las actividades del proyecto están divididas de manera razonable en dos subproyectos, uno centrado en temas de interoperabilidad y otro en reutilización. La estructura de paquetes de trabajo técnicos se enfoca en cada una de las dimensiones FAIR: WP1 en Findability y Accessibility, WP2 en Interoperability y WP3 en Reusability. Llama la atención que en algunos WP como WP1 se habla de los objetivos del WP mientras que en otros como WP2 se pasa directamente a describir el trabajo a realizar en él. WP1 parece centrarse en el uso de identificadores únicos y metadatos pero no aclara si el proyecto generará DOIs o no y no deja claro el alcance de su API. Tampoco queda claro si el proyecto se limitará a la ingesta de datos de repositorios existentes de datos de investigación o si también contribuirá a ellos. En general, no queda claro cuál será el modelo de gobernanza de datos propuesto por el proyecto y cómo se coordinará a través de los distintos repositorios. Una figura habría sido muy útil en este sentido. La propuesta habla de servicios para la integración de datos y de extracción e interoperabilidad de esquemas. Una pregunta que cabría hacerse es si se espera como parte de los resultados del proyecto el desarrollo de un vocabulario, recomendación o estándar que los integre. La coordinación a nivel internacional de un estándar de esas características sería un objetivo de gran impacto al que el proyecto podría contribuir. También se habla de la generación de un grafo de conocimiento de datasets que facilite la reutilización de datos por parte de los usuarios. Me habría gustado ver un análisis más detallado de las limitaciones reales de los catálogos existentes que se pretenden resolver con este enfoque en el marco del proyecto. El proyecto planea ofrecer asistentes a los usuarios para el enriquecimiento semántico del grafo de conocimiento sin detallar cómo se va a concienciar a la comunidad de usuarios de la necesidad de hacer ese esfuerzo adicional en su trabajo diario. En general, la propuesta da la sensación de generar infraestructura sin tener tan en cuenta como sería deseable a su futuro usuario, el investigador. Esto genera serias dudas sobre sus posibilidades reales de impacto y adopción en dichas comunidades. En WP3 llama la atención que se dé por hecho que la visualización en formato de grafo es la metáfora visual más adecuada para los usuarios investigadores de los dominios propuestos. Se aprecia el énfasis en temas de confianza en el uso de los datos pero cuáles son los requisitos en ese sentido o limitaciones actuales que necesitan ser satisfechos en las comunidades de investigadores concretas a las que el proyecto se dirige? Las actividades propuestas y los recursos humanos, materiales y de infraestructura con las que cuenta el equipo presentan debilidades como la ausencia de investigadores en las áreas de uso propuestas (agricultura y medicina) que puedan ayudar a establecer requisitos reales para esas comunidades sobre las tecnologías propuestas y que puedan dar lugar a una validación creíble. De hecho, no se observa por ejemplo un entregable específico de requisitos de usuario o estudios en los que los usuarios tengan un papel relevante en WP4, que parece centrarse más en el desarrollo de prototipos. No se mencionan datasets concretos en estas comunidades que vayan a ser objeto de estudio en el proyecto. No queda claro tampoco si la aparentemente modesta contribución científica de la propuesta permitirá el desarrollo con éxito de las tesis doctorales ligadas a ella. El plan de contingencia de riesgos es tremadamente genérico y carece de elementos específicos al proyecto. Se incluye un data management plan como parte de los objetivos de WP5.

2. Equipo de investigación

Valoración: C, Bueno

El equipo de investigación tiene buenas contribuciones científico-técnicas y es conocido dentro de su área de investigación. Muy buena internacionalización tanto por su participación en proyectos internacionales como por su colaboración con grupos internacionales de elevado prestigio. La especialización y conocimientos de los IPs es buena para llevar a cabo los objetivos del proyecto según se han descrito en la propuesta. Asimismo, los IPs tienen una larga y reconocida trayectoria de participación en proyectos anteriores financiados por el Plan Estatal de I+D+I y han generado un número considerable de publicaciones en la temática del proyecto. También es destacable su participación en actividades de transferencia tecnológica y con empresas, desde la investigación a la sociedad. Se echa en falta investigadores en las áreas de uso y comunidades propuestas (agricultura y medicina) que puedan ayudar a establecer requisitos reales para esas comunidades sobre las tecnologías propuestas y que puedan dar lugar a una validación creíble.

3. Impacto

3.1. Impacto científico-técnico de los resultados esperados

Valoración: C, Bueno

Los resultados de esta propuesta darán lugar a un moderado avance en el conocimiento científico-técnico dentro del área. El plan de divulgación científica presenta debilidades. Por ejemplo, las actividades de diseminación son muy convencionales y genéricas y el plan de difusión se centra en conferencias y revistas de la comunidad semántica, sin incluir venues relacionados con las comunidades de usuarios en agricultura y medicina. La contribución de los resultados de la propuesta al reto seleccionado se considera moderado, así como el plan de transferencia y valorización de los resultados dentro del reto seleccionado. Por otro lado, la propuesta parece haber recabado apoyos de miembros prominentes de la comunidad internacional de web semántica. El plan de transferencia es limitado y parece centrado en Dataverse, aparentemente sin tener seriamente como objetivo desde el origen iniciativas claves a nivel Europeo como EOSC. No se observan indicadores de rendimiento (key performance indicators) que permitan monitorizar de manera cuantitativa el grado de cumplimiento de los objetivos del proyecto durante su ejecución.

3.2. Impacto social y económico de los resultados

Valoración: D, Aceptable

Los resultados de esta propuesta tendrán un impacto social moderado, con un plan de divulgación de resultados con debilidades, como se ha indicado anteriormente. La dimensión de género está relativamente poco desarrollada. La participación de usuarios finales de los resultados del proyecto es en apariencia escasa. A medio plazo no se observa un gran potencial de desarrollo económico asociado a la propuesta aunque la temática en la que se centra (la convergencia hacia los objetivos FAIR en datos de investigación) es importante para los retos seleccionados. No se observan key performance indicators que permitan monitorizar de manera cuantitativa el grado de cumplimiento de los objetivos del proyecto durante su ejecución.

VALORACIÓN FINAL

Valoración: D, Aceptable

OTROS ASPECTOS A CONSIDERAR

a) Capacidad formativa.(cumplimentar solo cuando se haya solicitado la inclusión del proyecto en la convocatoria de contratos predoctorales para la formación de doctores)

Valoración: C, Bueno

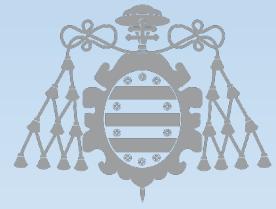
La capacidad formativa de los grupos de investigación participantes en la propuesta está fuera de duda y queda avalada por la cantidad de tesis y otros trabajos supervisados por los IPs que han generado. Sin embargo, la propuesta genera ciertas dudas en cuenta a su capacidad de generar investigación que pueda derivar en una tesis doctoral. Los aspectos técnicos tienen gran relevancia en la redacción de la propuesta pero no quedan claras las posibles preguntas de investigación que un doctorando desarrollando su tesis en el contexto de este proyecto podría plantear con el fin de generar contribuciones científicas de calidad e impacto.

b) Aspectos relacionados con zonas polares o campañas oceanográficas. (cumplimentar solo en los proyectos que proceda)

No procede

c) Condiciones específicas para la ejecución de determinados proyectos.(cumplimentar solo en los proyectos con aspectos relacionados con las condiciones o implicaciones recogidas en el Anexo IV de la convocatoria)

No procede



Universidad de Oviedo

20. Referencias bibliográficas

- [1] P. Abrahamsson, M.A. Babar y P. Kruchten. “Agility and Architecture: Can They Coexist?” En: *IEEE Software* 27.2 (mar. de 2010), páginas 16-22. DOI: 10.1109/ms.2010.36 (véase página 129).
- [2] Heather Adkins, Betsy Beyer y Paul Blankinship. *Building Secure and Reliable Systems: Best Practices for Designing, Implementing, and Maintaining Systems*. OREILLY MEDIA, 7 de abr. de 2020. 558 páginas. ISBN: 1492083127 (véase página 129).
- [3] Razieh Alidoosti. “Ethics-driven Software Architecture Decision Making”. En: *2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C)*. IEEE, mar. de 2021. DOI: 10.1109/icsa-c52384.2021.00022 (véase página 130).
- [4] J. Baragry y K. Reed. “Why is it so hard to define software architecture?” En: *Proceedings 1998 Asia Pacific Software Engineering Conference (Cat. No.98EX240)*. IEEE Comput. Soc. DOI: 10.1109/apsec.1998.733577 (véase página 125).
- [5] Len Bass. *DevOps : a software architect's perspective*. Old Tappan, NJ: Addison-Wesley, 2015. ISBN: 9780134049847 (véase página 129).
- [6] Len Bass, Rick Kazman y Paul Clements. *Software Architecture in Practice: 3rd edition*. SEI Series in Software Engineering. Addison Wesley, 1 de oct. de 2012. ISBN: 0321815734 (véanse páginas 126, 128, 149, 170, 171, 174, 193).
- [7] Betsy Beyer y col. *Site Reliability Engineering*. O'Reilly UK Ltd., 30 de abr. de 2016. 524 páginas. ISBN: 149192912X (véanse páginas 129, 176).
- [8] Barry Boehm. *Software engineering economics*. Englewood Cliffs, New Jersey: Prentice-Hall, 1981. ISBN: 0138221227 (véase página 128).
- [9] Pierre Bourque. *SWEBOK : guide to the software engineering body of knowledge*. Los Alamitos, CA: IEEE Computer Society, 2014. ISBN: 0769551661 (véase página 133).

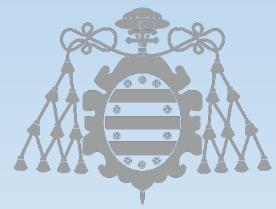
- [10] Amy Brown. *The architecture of open source applications : elegance, evolution, and a few fearless hacks*. Mountain View, Calif: CreativeCommons, 2011. ISBN: 9781257638017 (véase página 182).
- [11] Peter C. Brown, Henry L. Roediger y Mark A. McDaniel. *Make it Stick*. Harvard University Press, 14 de mayo de 2014. 313 páginas. ISBN: 0674729013 (véase página 185).
- [12] Simon Brown. *Software architecture for developers*. Volumen 1 y 2. LeanPub, 2012. URL: <https://softwarearchitecturefordevelopers.com/> (véanse páginas 129, 169, 201).
- [13] Frank Buschmann. *Pattern-oriented software architecture : a system of patterns*. Chichester New York: Wiley, 1996. ISBN: 9780471958697 (véase página 128).
- [14] Frank Buschmann. *Pattern-Oriented Software Architecture V4 - A Pattern Language for Distributed Computing*. Wiley, 16 de mar. de 2007. 638 páginas. ISBN: 0470059028 (véase página 128).
- [15] Coral Calero, M. Angeles Moraga y Mario Piattini, edición. *Software Sustainability*. S.l: Springer Nature, 2021. ISBN: 9783030699697 (véase página 130).
- [16] Nora Jones Casey Rosenthal. *Chaos Engineering*. O'Reilly UK Ltd., 1 de mayo de 2020. ISBN: 1492043869 (véase página 129).
- [17] S. R. Chidamber y C. F. Kemerer. "A metrics suite for object oriented design". En: 20.6 (jun. de 1994), páginas 476-493. DOI: 10.1109/32.295895 (véase página 172).
- [18] Paul Clements, Felix Bachmann y Len Bass. *Documenting Software Architectures: Views and Beyond*. Addison Wesley, 1 de oct. de 2010. 537 páginas. ISBN: 0321552687 (véanse páginas 149, 168, 169, 171, 194).
- [19] Paul Clements y Mary Shaw. "The Golden Age of Software ArchitectureRevisited". En: *IEEE Software* 26.4 (jul. de 2009), páginas 70-72. DOI: 10.1109/ms.2009.83 (véase página 128).
- [20] Comisión de Calidad de la Escuela de Ingeniería Informática. *Memoria del Grado de Ingeniería Informática del Software*. Informe técnico. Universidad de Oviedo – ANECA, 2014. URL: https://calidad.uniovi.es/c/document_library/get_file?p_l_id=2535677&folderId=4419021&name=DLFE-133202.pdf (véase página 166).
- [21] James O. Coplien y Neil B. Harrison. *Organizational Patterns of Agile Software Development*. Prentice Hall, 1 de jul. de 2004. 432 páginas. ISBN: 0131467409 (véase página 128).
- [22] Jeffrey Dean y Sanjay Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters". En: *Proceedings of the 6th Conference on Symposium on Operating Systems Design and Implementation - Volume 6*. OSDI'04. San Francisco, CA: USENIX Association, 2004, página 10 (véase página 175).
- [23] John Dewey. *Democracy and Education: An Introduction to the Philosophy of Education*. The Pennsylvania State University, 1 de feb. de 1997. 378 páginas. ISBN: 0684836319 (véase página 156).
- [24] G. K. A. Dias. "Evolvement of Computer Aided Software Engineering (CASE) Tools: AUser Experience". En: *International Journal of Computer Science and Software Engineering* 6.3 (2017), páginas 55-60. ISSN: 2409-4285 (véase página 128).
- [25] Murat Erder, Eoin Woods y Pierre Pureur. *Continuous Architecture in Practice: Software Architecture in the Age of Agility and Devops*. Addison Wesley Pub., 21 de mayo de 2021. 336 páginas. ISBN: 0136523560 (véanse páginas 129, 197).
- [26] Alicia Escribano. *El aprendizaje basado en problemas: una propuesta metodológica en educación superior*. Third. Madrid: Narcea, 2018. ISBN: 9788427715752 (véase página 156).

-
- [27] Eric J. Evans. *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison Wesley, 1 de ago. de 2003. 529 páginas. ISBN: 0321125215 (véanse páginas 172, 202).
 - [28] George Fairbanks. *Just Enough Software Architecture: A Risk-Driven Approach*. Marshall y Brainerd, 1 de sep. de 2010. 360 páginas. ISBN: 0984618104 (véanse páginas 129, 169-171, 174, 195).
 - [29] Roy Thomas Fielding. “Architectural Styles and the Design of Network-based Software Architectures”. Tesis doctoral. University of California, Irvine, 2000. URL: <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm> (véase página 174).
 - [30] Brian Foote y Joseph W. Yoder. “Big Ball of Mud”. En: *Pattern Languages of Program Design*. Editado por N. Harrison, B. Foote y H. Rohnert. Volumen 4. 2000, páginas 654-692. URL: <http://www.laputan.org/mud/mud.html> (véase página 172).
 - [31] Neal Ford, Rebecca Parsons y Patrick Kua. *Building Evolutionary Architectures*. O'Reilly UK Ltd., 8 de oct. de 2017. ISBN: 1491986360 (véanse páginas 129, 170, 195).
 - [32] Martin Fowler. *Refactoring : improving the design of existing code*. Reading, MA: Addison-Wesley, 1999. ISBN: 9780201485677 (véase página 128).
 - [33] Martin Fowler. *Patterns of Enterprise Application Architecture*. Addison Wesley, 1 de nov. de 2002. 533 páginas. ISBN: 0321127420 (véase página 128).
 - [34] Erich Gamma y col. *Design Patterns: Elements of Reusable Object-Oriented Software*. Prentice Hall, 1 de dic. de 1995. ISBN: 0201633612 (véase página 128).
 - [35] María Antonia García de León y Marisa García de Cortázar. “Universidades y Universitarios”. En: *Revista de Educación (M.E.C.)* (1992). URL: <https://www.educacionyfp.gob.es/dam/jcr:ee6887fc-8bb1-4d9e-b6ce-c23d94c3b93a/re199206-pdf.pdf> (véase página 105).
 - [36] Jeff Garland. *Large-scale software architecture : a practical guide using UML*. Chichester New York: J. Wiley, 2003. ISBN: 9780470848494 (véase página 128).
 - [37] José Gasset. *Misión de la universidad*. Madrid: Alianza Editorial, 1930. ISBN: 9788437633572 (véase página 101).
 - [38] Anne Gentle. *Docs like code : write, review, test, merge, build, deploy, repeat*. Austin, TX: Just Write Click, 2017. ISBN: 9781387081325 (véase página 184).
 - [39] Mahbouba Gharbi y col. *Software Architecture Fundamentals*. Dpunkt.Verlag GmbH, 21 de feb. de 2019. ISBN: 3864906253 (véase página 137).
 - [40] Graham Gibbs. *Using assessment to support student learning*. Leeds Metropolitan University, 2010 (véase página 161).
 - [41] Robert Godwin-Jones. “Emerging Technologies, Blogs and Wikis: Environments for On-line Collaboration”. En: *Language, Learning and Technology* 7.2 (mayo de 2003), páginas 12-16 (véase página 159).
 - [42] Robert S. Hanmer. *Pattern-Oriented Software Architecture For Dummies*. WILEY, 4 de ene. de 2013. 384 páginas. ISBN: 1119963990 (véanse páginas 128, 174).
 - [43] HEA. *A marked improvement: transforming assessment in higher education*. Informe técnico. 2012. URL: https://www.heacademy.ac.uk/system/files/A_Marked_Improvement.pdf (véase página 160).
 - [44] Christine Hofmeister y Robert Nord. *Applied Software Architecture*. Addison Wesley, 1 de nov. de 1999. 432 páginas. ISBN: 0201325713 (véase página 128).

- [45] Gregor Hohpe. *The Software Architect Elevator: Transforming Enterprises with Technology and Business Architecture*. O'Reilly Media, Inc, USA, 24 de abr. de 2020. 350 páginas. ISBN: 1492077542 (véanse páginas 125, 129, 168, 177, 196).
- [46] Gregor Hohpe y Bobby Woolf. *Enterprise Integration Patterns*. Addison Wesley, 1 de ene. de 2004. 480 páginas. ISBN: 0321200683 (véase página 128).
- [47] Rick Kazman Humberto Cervantes. *Designing Software Architectures: A Practical Approach*. Addison Wesley Publishing Company, 13 de mayo de 2016. 320 páginas. ISBN: 0134390784 (véanse páginas 170, 200).
- [48] ISO/IEC 25010. *ISO/IEC 25010:2011, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. Informe técnico. 2011 (véase página 169).
- [49] David Farley Jez Humble. *Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation*. Addison Wesley, 1 de jul. de 2010. 512 páginas. ISBN: 0321601912 (véanse páginas 129, 171, 199).
- [50] Catherine Tang John Biggs. *Teaching for Quality Learning at University*. McGraw-Hill Education Ltd, 1 de sep. de 2011. ISBN: 0335242758 (véanse páginas 152, 185).
- [51] Rick Kazman y col. “Experience with performing architecture tradeoff analysis”. En: *Proceedings of the 21st international conference on Software engineering - ICSE '99*. ACM Press, 1999. DOI: 10.1145/302405.302452 (véase página 128).
- [52] Michael Keeling. *Design It!* O'Reilly UK Ltd., 1 de nov. de 2017. ISBN: 1680502093 (véase página 198).
- [53] Cynthia Andres Kent Beck. *Extreme Programming Explained: embrace change*. Addison Wesley, 1999. ISBN: 0321278658 (véase página 128).
- [54] Gene Kim y col. *The DevOPS Handbook*. IT Revolution Press, 6 de oct. de 2016. 480 páginas. ISBN: 1942788002 (véase página 129).
- [55] Michael Kircher y Prashant Jain. *Pattern-Oriented Software Architecture: Patterns for Resource Management*. Wiley, 1 de jun. de 2004. 284 páginas. ISBN: 0470845252 (véase página 128).
- [56] Donald E. Knuth. “Computer programming as an art”. En: *Communications of the ACM* 17.12 (dic. de 1974), páginas 667-673. DOI: 10.1145/361604.361612 (véase página 128).
- [57] Jay Kreps. *I Heart Logs: Event Data, Stream Processing, and Data Integration*. O'Reilly Media, 28 de oct. de 2014. 60 páginas. ISBN: 1491909382 (véase página 175).
- [58] Phillippe Kruchten. “Architecture blueprints—the "4+1"view model of software architecture”. En: *Tutorial proceedings on TRI-Ada '91 Ada's role in global markets: solutions for a changing complex world - TRI-Ada '95*. ACM Press, 1995. DOI: 10.1145/216591.216611 (véanse páginas 128, 168).
- [59] Robert C. Martin. *Clean Code*. Prentice Hall, 1 de mar. de 2009. 464 páginas. ISBN: 0132350882 (véase página 128).
- [60] Robert C. Martin. *Clean Architecture*. Prentice Hall, 10 de sep. de 2017. 432 páginas. ISBN: 0134494164 (véanse páginas 172, 173, 194).
- [61] Nathan Marz. *Big data : principles and best practices of scalable real-time data systems*. Shelter Island, NY: Manning, 2015. ISBN: 9781617290343 (véase página 175).
- [62] José Medina. *La docencia universitaria mediante el enfoque del aula invertida*. Barcelona: Octaedro, 2016. ISBN: 9788499218427 (véase página 155).

-
- [63] N. Medvidovic y R.N. Taylor. “A classification and comparison framework for software architecture description languages”. En: *IEEE Transactions on Software Engineering* 26.1 (2000), páginas 70-93. DOI: 10.1109/32.825767 (véase página 128).
 - [64] Charles W. Moore. “Architecture: Art and Science”. En: *Journal of Architectural Education* (1947-1974) 19.4 (jun. de 1965), página 53. DOI: 10.2307/1424258 (véase página 127).
 - [65] Mark Richards Neal Ford. *Fundamentals of Software Architecture*. O'Reilly, 6 de mar. de 2020. ISBN: 1492043451 (véanse páginas 129, 168-170, 173, 174, 192).
 - [66] Sam Newman. *Building microservices : designing fine-grained systems*. Sebastopol, CA: O'Reilly Media, 2015. ISBN: 9781491950357 (véanse páginas 129, 175).
 - [67] Sam Newman. *Monolith to Microservices: Sustaining Productivity While Detangling the System*. O'Reilly UK Ltd., 1 de dic. de 2019. ISBN: 1492047848 (véanse páginas 129, 175, 197).
 - [68] Michael Nygard. *Release It!* The pragmatic programmers. O'Reilly UK Ltd., 1 de feb. de 2018. 350 páginas. ISBN: 1680502395 (véanse páginas 129, 176, 196).
 - [69] R. Pawson. “Naked Objects”. En: 19.4 (jul. de 2002), páginas 81-83. DOI: 10.1109/ms.2002.1020291 (véase página 173).
 - [70] Dewayne E. Perry y Alexander L. Wolf. “Foundations for the study of software architecture”. En: *ACM SIGSOFT Software Engineering Notes* 17.4 (oct. de 1992), páginas 40-52. DOI: 10.1145/141874.141884 (véase página 128).
 - [71] Murat Erder Pierre Pureur. *Continuous Architecture: Sustainable Architecture in an Agile and Cloud-Centric World*. Morgan Kaufmann Pub. Inc., 5 de nov. de 2015. 324 páginas. ISBN: 0128032847 (véase página 129).
 - [72] Philip Race. *The lecturer's toolkit : a practical guide to assessment, learning and teaching*. Abingdon, Oxon New York, NY: Routledge, 2020. ISBN: 9780367182298 (véanse páginas 153, 161).
 - [73] Chris Richardson. *Microservice Patterns*. Manning, 1 de jun. de 2019. ISBN: 1617294543 (véase página 129).
 - [74] Ridder-Symoens, Hilde de, edición. *A History of the University in Europe*. Volumen 3. Universities in the Nineteenth and Early Twentieth Centuries (1800–1945). Cambridge University Press, 1991. DOI: <https://doi.org/10.1017/CBO9780511599507> (véase página 101).
 - [75] Nick Rozanski y Eoin Woods. *Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley Professional, 2011. ISBN: 0132906120 (véase página 199).
 - [76] Walter Ruegg, edición. *A History of the University in Europe*. Volumen 1. Universities in the Middle Ages. Cambridge University Press, 2004. DOI: <https://doi.org/10.1017/CBO9780511599507> (véase página 101).
 - [77] Sanjay Sarma. *Grasp : the science transforming how we learn*. New York: Doubleday, 2020. ISBN: 9780385541824 (véase página 185).
 - [78] Douglas Schmidt. *Pattern-oriented software architecture*. Chichester, West Sussex, England New York, NY: Wiley, 2000. ISBN: 9780471606956 (véase página 128).
 - [79] Ken Schwaber. *Agile software development with Scrum*. Upper Saddle River, NJ: Prentice Hall, 2002. ISBN: 9780130676344 (véase página 128).

-
- [80] M. Shaw y D. Garlan. *Software Architecture: Perspectives of an emerging discipline*. Prentice Hall, 1996 (véase página 128).
 - [81] Matthew Skelton. *Team topologies : organizing business and technology teams for fast flow*. Portland, OR: IT Revolution, 2019. ISBN: 1942788827 (véanse páginas 169, 200).
 - [82] Fritz Solms. “What is software architecture?” En: *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference on - SAICSIT '12*. ACM Press, 2012. DOI: 10.1145/2389836.2389879 (véase página 125).
 - [83] Gernot Starke y col. *Arc42 by example*. Leanpub, 2019 (véase página 169).
 - [84] Adam Tornhill. *Software Design X-Rays: Fix Technical Debt with Behavioral Code Analysis*. O'Reilly UK Ltd., 1 de abr. de 2018. ISBN: 1680502727 (véase página 177).
 - [85] *What is your definition of Software Architecture?* Informe técnico 3854. Carnegie Mellon University – Software Engineering Institute, 2017. URL: https://resources.sei.cmu.edu/asset_files/FactSheet/2010_010_001_513810.pdf (véase página 125).
 - [86] Titus Winters, Tom Mansreck e Hyrum Wright. *Software Engineering at Google: Lessons Learned from Programming Over Time*. O'Reilly UK Ltd., 1 de mar. de 2020. 583 páginas. ISBN: 1492082791 (véanse páginas 129, 169, 198).
 - [87] Zhonghua Yang y Keith Duddy. “CORBA: a platform for distributed object computing”. En: *ACM SIGOPS Operating Systems Review* 30.2 (abr. de 1996), páginas 4-31. DOI: 10.1145/232302.232303 (véase página 174).



Universidad de Oviedo

Índice alfabético

A

Acoplamiento	172
actividades síncronas no presenciales	187
Acyclic Dependencies Principle	172
ADD	170
ADD 3.0	170
Agentes móviles	174
alineación constructiva	153
Allocation	175
Amazon	129, 175
Análisis de comportamiento de código	177
API	172
Aprendizaje autorregulado	156
Aprendizaje basado en problemas	162
Aprendizaje colaborativo	156
Arboles de calidad	169
architecting	126
architectural drift	170
Architecture Trade-off Analysis Method	170
Arquitectura basada en servicios	174
Arquitectura del Software	127, 160
Arquitectura en capas	172
arquitectura en capas	174
Arquitectura Kappa	175
Arquitectura Lambda	175
Arquitectura limpia	172
Arquitecturas basadas en eventos	173
arquitecturas de referencia	171
arquitecturas de software en dominios específicos	171
Arquitecturas orientadas a servicios	174
artefacto	169

B

Artes	101
Aspectos	172
ATAM	170
atributos de calidad	169
Attribute Driven Design	170
auditoría de configuraciones	171
Automatización de la construcción	171
automatizaciónn de la construcción	171
AWS Lambda	175
B2B	176
Baas	175
Backend as a Service	175
Backpressure	176
Balanceo de carga	176
base de datos compartida	174
baseline	171
bases de datos	145
Batch	173
batch	175
batch layer	175
becas Erasmus	160
becas FARO	160
becas SICUE Séneca	160
Behavioural Code Analysis	177
big ball of mud	172
big data	175
BitTorrent	174
Blockchain	174
Bolonia	101
boundary context	172
BPM	176

BPMN	176
Bulkheads	176
Bus de mensajería	174
Business Intelligence	176
Business Process Management	176
Business Process Modeling and Notation	176
Business to Business	176

C

canal de despliegue	176
Canales de distribución	175
Canary releases	176
Carnegie-Mellon University	133
Catálogo de patrones	171
CBAM	170
Centro de Cálculo	115
Chat	159
Ciencias Sociales	101
Circuit breaker	176
Clases invertidas	155
Clean architecture	172
Cliente-servidor	173, 174
Cliente-servidor con caché	173
Cliente-servidor sin estado	173
cloud computing	175
cobertura de código	171
Cohesividad	172
Command-query responsibility segregation	173
Commercial Off-the-self	171
commits	186
Common Closure Principle	172
Common Reuse Principle	172
Competencias	165
Competencias específicas	165
Competencias generales	165
competencias generales	165
Complejidad ciclomática de McCabe	169
Componentes y conectores	173
Conocimiento	172
configuración	176
continous deployment	176
Continuous delivery	129
continuous delivery	171, 176
convocatoria extraordinaria	187
CORBA	174
Correo electrónico	159
Cost Benefit Analysis Method	170
COTS	171
CQRS	173, 175
CRM	176
Crosscutting concern	172
CRUD	173
Cummulative component dependency	171
Customer Relationship Management	176
código arquitectónicamente evidente	170

Código bajo demanda	174
código bajo demanda	174
Código móvil	174

D

Data-context-interacción	173
DCI	173
debates	158
deployment pipeline	176
Desarrollo basado en pruebas	171
Desconocidos en arquitectura del software	170
despliegue continuo	176
Despliegue progresivo	176
deterioro estructural	175
Deuda técnica	170
Devops	129
devops	171
Diferencias de comprensión	170
difusión de responsabilidad	169
Discord	158
Diseño de software	172
disponibilidad	169
Disposición	175
Docker	176
Doctorado en Informática	122
Domain-driven design	172
Don't repeat yourself	172
DRY	172
DSLs	174

E

EAI	176
ECM	176
edge computing	130, 175
Ensayos	162
Enterprise Application Integration	176
Enterprise Content Management	176
Enterprise Resource Planning	176
entorno de desarrollo	175
entorno de producción	175
entorno de pruebas	175
entrega continua	171, 176
entregables	186
Erosión de la arquitectura	170
ERP	176
escalabilidad	175
escenarios de calidad	169
Escuela Universitaria de Ingeniería Técnica Infor- mática de Oviedo	115
Estilo arquitectónico	171
Estilo hexagonal	172
Estudios Técnicos	101
estímulo	169
EUITIO	115

- Evaluación de prácticas 185
- Evaluación de teoría 185
- Evaluación remota 174
- Event Sourcing 175
- Event sourcing 174
- Evolución del contexto 170
- expertos del dominio 172
- Extreme programming 128
- Exámenes con preguntas elección múltiple 161

F

- FaaS 175
- Factor del autobús 169
- Factorización 177
- Fail fast 176
- Feature toggles 176
- Fitness function 170
- Flipped classroom 155
- Flujo de datos 173
- fog computing 130, 175
- Foros de discusión 159
- FOSS 171
- Free Open Source Software 171
- Function as a Service 175

G

- General Responsibility Assignment Software Patterns 172
- Gestión de configuraciones 171
- Gestión de conocimiento 176
- Gestión de dependencias 171
- git 171
- GNUTella 174
- Google 129, 175
- Google Cloud Functions 175
- Google Meet 158
- Governor 176
- Grafo de dependencias 171
- GRASP 172

H

- Hadoop 175
- Handshaking 176
- herramientas de trabajo en equipo 171
- Hub & Spoke 174

I

- ignorancia colectiva 169
- Incidencia arquitectónica 170
- Informe post-mortem 176
- Inrupt 188
- integración continua 171

- Interfaces fluidos 172
- Intérpretes 174
- invocación de procedimiento remoto 174
- ISO 25010 169
- issues 186
- Item de configuración 171

J

- Jay Kreps 175

K

- Kahoot 185
- Keep it simple, stupid 172
- KISS 172

L

- Lack of Cohesion of Methods 172
- layers 172
- legacy software 190
- Lenguaje de dominio específico 174
- Lenguaje de patrones 171
- Lenguaje ubicuo 172
- Ley de Brook 169
- Ley de cambio continuo 177
- Ley de complejidad creciente 177
- Ley de Conway 169
- Ley de Demeter 172
- Ley de Postel 172
- Ley de Reforma Universitaria 104
- Leyes 101
- Leyes de Lehman 177
- leyes de Murphy 157
- Licenciatura de Físicas 115
- Licenciatura de Matemáticas 115
- Listas de correo 159
- log de eventos 174
- logging 176
- logs distribuidos 175
- Lérida 104
- línea base 171
- Línea de producto software 177

M

- mainframes 130
- Manifiesto ágil 128
- Maniobra inversa de Conway 169
- mapa de stakeholders 169
- MapReduce 175
- Marcos de aplicaciones 171
- marcos de aplicaciones 173
- Massive Open Online Courses 152
- Master-slave 173

master-slave 175
 Maven 171
 Mecalux EasyWMS 176
 Medicina 101
 medida de respuesta 169
 mensajería 174
 Mercados de aplicaciones 175
 Microkernel 174
 Microservicios 174
 microservicios 173
 Microsoft 129, 175
 Microsoft Teams 158
 Misión de la Universidad 101
 Modelo vista-controlador 173
 Modelos centrados en datos 173
 Modelos de actores 173
 monitorización 176
 monolitos distribuidos 175
 MOOC 152
 Moodle 158, 185
 Ms Azure Functions 175
 MTBF 169
 MTTR 169
 Muestras 162
 Mult-armed bandits 176
 MVC 173
 Máster Universitario en Ingeniería Web 118
 Métrica CCD 171
 Métrica LCOM 172

N

Naked objects 173
 Napster 174
 npm 171

O

on premises 175
 Ordenador 154
 organizador previo 154
 Ortega y Gasset 101
 Oxford 101

P

PAC 173
 Palencia 104
 París 101
 Patrón arquitectónico 171
 Patrón bróker 174
 Peer-to-peer 174
 Perdida por proceso 169
 Personas interesadas 169
 Pipes and filters 173
 Pipes and filters con interfaz uniforme 173

pipes and filters con interfaz uniforme 174
 Plan de Estudios 115
 Plataformas 171
 PLM 176
 PLugins 174
 Portafolios 162
 preguntas-problema 154
 Presentaciones 162
 Presentación-abstracción-control 173
 Principio de robustez 172
 Principios SOLID 172
 Problemas en arquitectura del software 170
 Product Lifecycle Management 176
 profiler 171
 program slicer 171
 Programación orientada a aspectos 172
 propiedad colectiva del código 171
 Proyector 154
 Proyectos heredados 177
 Proyectos legacy 177
 Pruebas A/B 176
 pruebas de carga 176
 Pruebas de ejecución 162
 Pruebas escritas 161
 Pruebas orales 162
 Pruebas prácticas 162
 Publish-subscribe 173
 pull requests 186
 Página Web 159
 Pérdida por proceso 169
 Pósteres 162

R

realimentación 186
 Refactoring 128
 Reflection 174
 Registro de decisiones arquitectónicas 170
 release de software 171
 repositorios de información 145
 Representational State Transfer 174
 respuesta 169
 REST 174
 Resultados de aprendizaje 165
 Reuse/Release Equivalence Principle 172
 riesgos en arquitectura del software 170
 risk storming 170
 Rol del arquitecto de software 169
 Roy T. Fielding 174
 Rúbrica 185

S

Salamanca 104
 Salerno 101
 SAP 176

SCM	176
seguimiento de incidencias	171
SEI	133, 134
seminarios	158
Serverless	175
Servidor replicado	173
serving layer	175
Sistemas colaborativos de edición	159
Sistemas de control de versiones	171
sistemas de control de versiones	171
Slack	158
SOAP	174
software como servicio	177
software heredado	190
Solid	188
spped layer	175
Stable Abstractions Principle	172
Stable Dependencies Principle	172
Stakeholders	169, 170
Steady state	176
Supply Chain Management	176
SWBoK	134
síntesis final	154

T

Tabla de valoración de riesgos	170
talleres de atributos de calidad	169
TCP/IP	172
TDD	171
Teams	185
Technical debt	170
technology stacks	171
Teología	101
Teoría del caos	176
Test harnesses	176
Test-driven development	128, 171
The Open Group Architecture Framework	176
tiempo de ejecución	173
Tiempo medio de recuperación	169
Tiempo medio entre fallos	169
tiers	172
Tim Berners-Lee	188
Timeout	176
TOGAF	176
Trabajo Fin de Grado/Máster	160
Trabajos	162
Trabajos Fin de Grado	159
transferencia de ficheros	174
Trasparencias	154
Táctica arquitectónica	170

U

UDDI	174
Unknowns	170

V

Valladolid	104
versioneado semántico	171
versión de software	171
Vídeoconferencia	159

W

Warehouse Management Systems	176
wiki	159
wikipedia	159
WMS	176
WSDL	174

Y

YAGNI	172
-------------	-----

Z

Zachman framework	176
Zapiens	185
Zoom	158