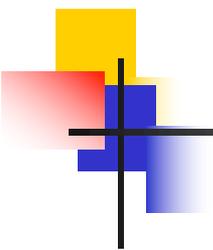


Mención de Ingeniería del Software

Coordinadora: Virginia Mato Abad

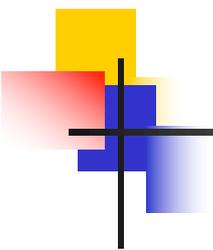
virginia.mato@udc.es

Despacho 3.05 Edificio Área Científica



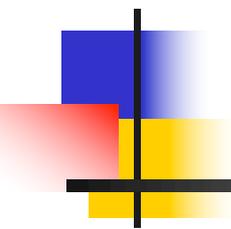
Objetivo

Formar profesionales altamente preparados para la concepción, definición, especificación, diseño, implementación, operación y mantenimiento del software, todo ello, llevado a cabo mediante prácticas adecuadas de gestión y con un aseguramiento y control de calidad adecuado

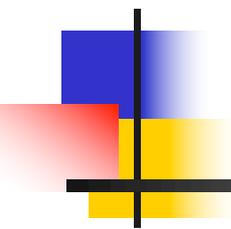


Asignaturas

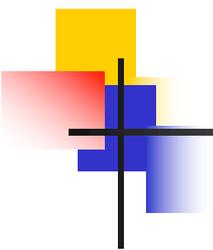
Cuatrimestre	Módulo	Asignatura
Q6	Obligatorio	Arquitectura del Software
		Ingeniería de Requisitos
		Aseguramiento de la Calidad
		Bases de Datos Avanzadas
		Programación Avanzada
Q7	Obligatorio	Metodologías de Desarrollo
		Marcos de Desarrollo
		Validación y Verificación de Software
		Herramientas de Desarrollo
Q8	Obligatorio	Proyectos de Desarrollo Software
Q7	Optativo (elegir 1)	Teoría de la Computación
		Gestión de Procesos de Negocio
		Métodos Estadísticos
Q8	Optativo (elegir 2)	Diseño de Lenguajes de Programación
		Procesamiento de Lenguajes
		Sistemas de Información Empresarial
		Sectores de Negocio



Detalles asignaturas obligatorias

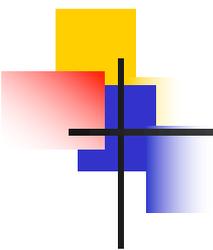


Arquitectura Software (Q6)



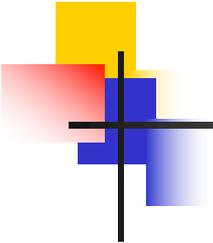
Resumen

- Esta asignatura busca dominar las alternativas actuales de la ingeniería de software para el **diseño de aplicaciones y sistemas a nivel de arquitectura**
- ¿Cómo?
 - Conociendo las **arquitecturas** más **típicas y sus características**
 - Estudiando los **requisitos no funcionales** de los sistemas y su relación con la arquitectura
 - Desarrollando y/o estudiando **sistemas reales**



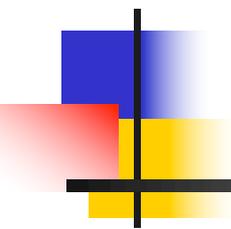
Temario

- **Bloque 1:** Concepto de arquitectura software
- **Bloque 2:** Modelos y arquitecturas de referencia
 - Parámetros de calidad en arquitectura software
 - Arquitecturas software
- **Bloque 3:** Diseño e integración de componentes
Patrones arquitecturales
 - Estrategias y tácticas de diseño de arquitectura
 - Reutilización, reconstrucción, reingeniería
- **Bloque 4:** Trazabilidad y pruebas

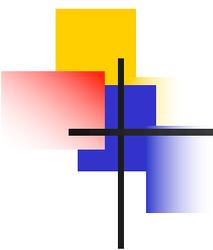


Profesorado

- Laura M. Castro [coordinadora], lcastro@udc.es
Despacho 4.15

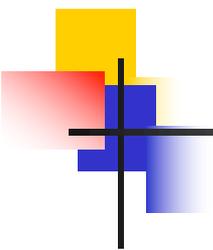


Aseguramiento de la Calidad (Q6)



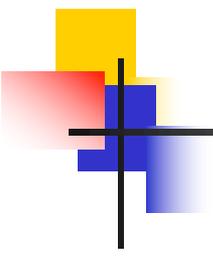
Resumen (1)

- Esta asignatura se centra en el estudio de los mecanismos para asegurar la calidad del desarrollo de software, con el objetivo de producir mejor software. Para ello se comienza con la introducción de los conceptos básicos y la cultura de calidad del software, para continuar con modelos, estándares y métricas de calidad ampliamente aceptados e implantados en las empresas desarrolladoras de software como son, por ejemplo, ISO 9000 y CMMI.



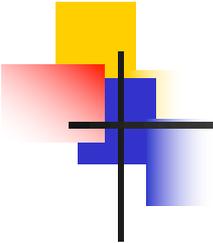
Resumen (y 2)

- La asignatura está enfocada a la aplicación práctica de los conocimientos teóricos, por lo que la nota de prácticas tiene un peso importante en la nota final. Las prácticas capacitarán a los alumnos para definir los aspectos relevantes de un Sistema de Gestión de la Calidad para una empresa desarrolladora de software, así como procedimientos de calidad.



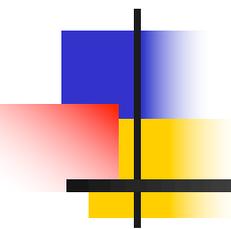
Temario

- Tema 1. ¿Es mejorable el proceso software seguido?
- Tema 2. Calidad y software
- Tema 3. ISO 9000. Aplicación al ámbito del software
- Tema 4. CMMI
- Tema 5. Otros modelos de calidad en el proceso software
- Tema 6. Medición de la calidad del software

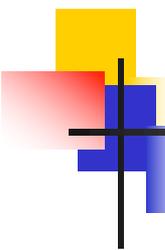


Profesorado

- Javier Andrade Garda. jag@udc.es. Despacho: D 3.08
- Sonia Suárez Garaboa (coordinadora).
ssuarez@udc.es. Despacho: S 3.02



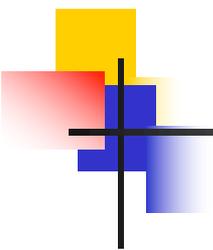
Bases de Datos Avanzadas (Q6)



Resume

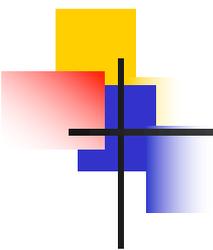
Esta materia céntrase en dous aspectos fundamentais:

- Conceptos das bases de datos relacionais especialmente relevantes de cara ó desenvolvemento de software que actúa contra unha base de datos:
 - Elementos que ofrecen os SXBD relacionais (catálogo, vistas, integridade de datos, seguridade, etc.)
 - Conocer e aplicar técnicas avanzadas de bases de datos, incluíndo o manexo de transaccións
 - Entender as técnicas de optimización usadas por un SXBD relacional
 - Desenvolvemento de aplicacións que accedan a BDs
- Modelado de datos:
 - Modelado conceptual de datos utilizando a linguaxe UML
 - Transformación ó modelo relacional dos modelos conceptuais



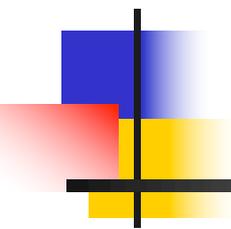
Temario

- Elementos das Bases de Datos relacionais
 - Catálogo, Vistas, Integridade e Seguridade
 - Actividade (disparadores)
- Proceso transaccional
 - Recuperación
 - Concorrenca
- Procesamento de consultas e optimización
- Modelado conceptual (UML) e transformación a relacional
- Desenvolvemento de aplicacións que acceden a bases de datos

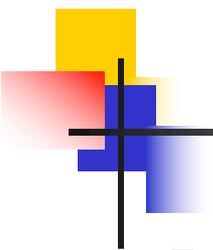


Profesorado

- Miguel Rodríguez Penabad (coordinador)
miguel.penabad@udc.es
Despacho 0.01
- Ángeles Saavedra Places
angeles.saavedra.places@udc.es
Despacho 3.04

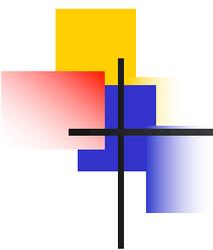


Ingeniería de Requisitos (Q6)



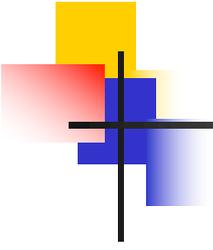
Resumen

- Requisito software
 - Capacidades o condiciones que un sistema debe cumplir
- Ingeniería de requisitos
 - Una aproximación sistemática para elicitar, organizar y documentar los requisitos de un sistema
- Enfoque
 - Requisitos según el Proceso Unificado de desarrollo de software basado en casos de uso
 - Aproximación a los requisitos desde el punto de vista de las metodologías ágiles
 - Práctica:
 - Desde una descripción inicial del problema saber especificar los requisitos del mismo y saber cómo transitar desde estos requisitos hasta el análisis, el diseño y, eventualmente, la codificación



Temario

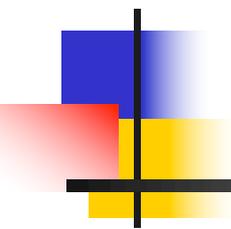
- Bloque I: Introducción
 - 1. Introducción
 - 2. El proceso unificado de desarrollo
- Bloque II: Especificación de Requisitos
 - 3. Los requisitos en la fase de inicio
 - 4. Casos de uso
- Bloque III: De los Requisitos al Análisis y el Diseño
 - 5. Los requisitos en la fase de elaboración
 - 6. Requisitos y diseño
- Bloque IV: Temas adicionales
 - 7. Gestión de requisitos
 - 8. Ingeniería de requisitos en las metodologías ágiles

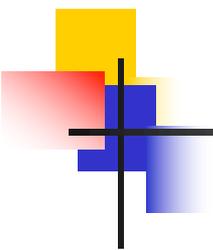


Profesorado

- Miguel Rodríguez Rubio. rodrubio@udc.es.
(coordinador). Despacho: 4.12

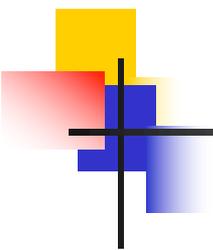
Programación Avanzada (Q6)





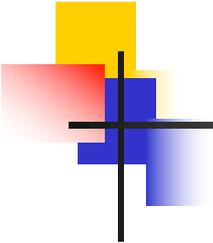
Resumen

- Esta asignatura se centra en el desarrollo de **aplicaciones**, usando una **arquitectura basada en capas** mediante el empleo de **frameworks modernos**
 - Se toma como caso de estudio el desarrollo de una aplicación web SPA
 - Muchas de las ideas aplican, directa o indirectamente, al desarrollo de otro tipo de aplicaciones
- **El enfoque de la asignatura es práctico**, por lo que la nota de la práctica tiene un peso importante en la nota final



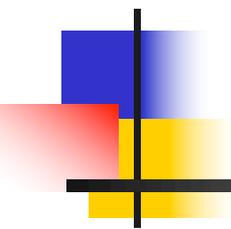
Temario

- Parte I. Introducción
 - Tema 1. Introducción al Desarrollo de Aplicaciones con Frameworks Modernos
- Parte II. Desarrollo de un Backend en Java
 - Tema 2. Capa Acceso a Datos con Spring y JPA
 - Tema 3. Capa Lógica de Negocio con Spring
 - Tema 4. Capa Servicios REST con Spring
- Parte III. Desarrollo de un Frontend Web SPA en JavaScript
 - Tema 5. Desarrollo Basado en Componentes con React
 - Tema 6. Gestión de Estado con Redux
 - Tema 7. Caso de Estudio: Frontend de una Sencilla Tienda de Comercio Electrónico
- Parte IV. Colofón
 - Tema 8. Arquitecturas y Tecnologías Relacionadas

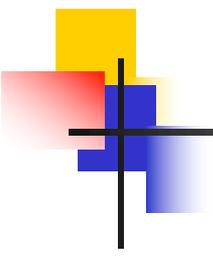


Profesorado

- Manuel Álvarez Díaz. manuel.alvarez@udc.es.
Despacho: 0.18, Facultad de Informática
- Fernando Bellas Permuy (coordinador).
fernando.bellas@udc.es. Despacho: 0.04, Facultad de
Informática
- Juan Raposo Santiago. juan.raposo@udc.es.
Despacho: 3.11, Edificio Área Científica

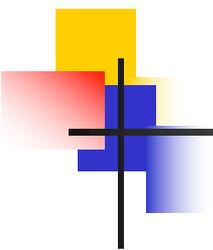


Herramientas de desarrollo (Q7)



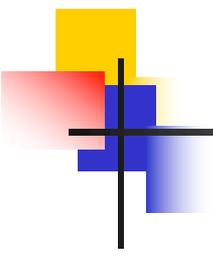
Resumen

- **Herramientas de Desarrollo** proveerá al estudiante del conocimiento necesario sobre **tecnologías y herramientas de apoyo al desarrollo de proyectos software** en un entorno productivo:
 - Herramientas y entornos de desarrollo
 - Herramientas de automatización y ayuda
 - Herramientas de análisis y monitorización
- **Es una asignatura eminentemente práctica,** este hecho se verá reflejado en la forma de evaluación



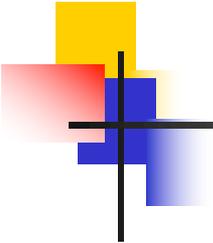
Temario (1)

- Entornos integrados de desarrollo
 - Ej. Eclipse, Netbeans
- Herramientas de automatización de empaquetado
 - Ej. Ant, Maven
- Herramientas de integración continua
 - Ej. Apache Jenkins/Hudson
- Herramientas de control de versiones y trabajo cooperativo
 - Ej. Subversion, Git, CVS



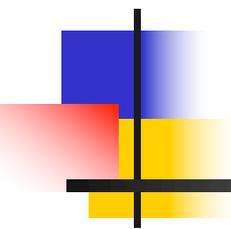
Temario (y 2)

- Herramientas de task/time/bug tracking
 - Ej. Jira, Redmine
- Herramientas de análisis de código y dependencias
 - Ej. Codepro Analytix, Sonar
- Herramientas de análisis de rendimiento y monitorización
 - Ej. JStat, JConsole, JVisualVM

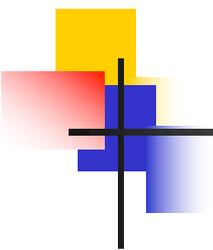


Profesorado

- Javier Parapar López (coordinador)
javierparapar@udc.es Despacho: L 4.2

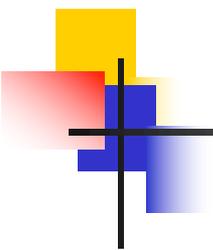


Marcos de desarrollo (Q7)



Resumen (1)

- Diseño e implementación de aplicaciones web empresariales utilizando tecnologías incluidas dentro de la plataforma .NET
- Herramientas fundamentales basadas en el desarrollo en capas/niveles:
- Modelo:
 - Persistencia: Entity Framework
 - Inyección de dependencias: Ninject
 - Librerías de utilidades: Enterprise Library
- Capa Web:
 - ASP.NET + MVC
- Pruebas:
 - VStudio Test Project

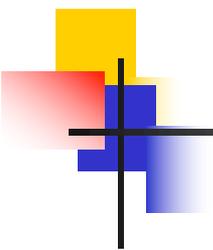


Resumen (y 2)

- Evaluación

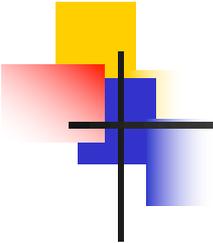
Eminentemente práctica.

Desarrollo aplicación empresarial (reducida) +
asimilación conceptos teóricos (necesarios para su
aplicación en el desarrollo práctico)



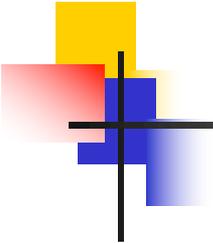
Temario (1)

- Introducción a la plataforma .NET
- El lenguaje de programación C#
- Diseño e Implementación de la Capa Modelo
- Acceso a Bases de Datos
 - ADO.NET
 - Estudio de casos: Diseño e implementación de la capa modelo en aplicaciones web empresariales
 - Entity Framework
 - Inyección de dependencias
 - Ninject
 - Utilidades
 - Enterprise Library: Logging Application Block
 - Test Project



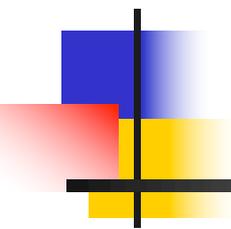
Temario (y 2)

- Diseño e Implementación de la Capa Web
 - Introducción a ASP.NET
 - Componentes principales ASP.NET: Web Forms , server controls, master pages, ...
 - Internacionalización
 - Estudio de casos: Diseño e implementación de la capa vista en aplicaciones web empresariales

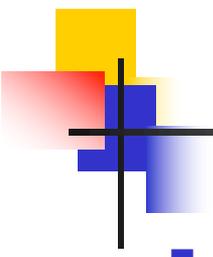


Profesorado

- Marcos Gestal Pose. mgestal@udc.es. (coordinador).
Despacho: Seminario 0.2

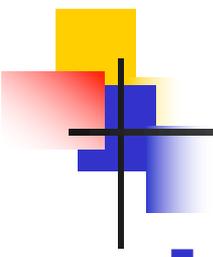


Metodologías de desarrollo (Q7)



Resumen

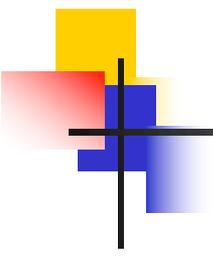
- Exposición detallada de las **metodologías y procesos** que permiten organizar el desarrollo de sistemas y aplicaciones.
- Se centra especialmente en:
 - El **Proceso Unificado de Desarrollo** de software: modelos, fases e iteraciones en la construcción incremental de un sistema dirigido por casos de uso, y centrado en arquitecturas.
 - En la explicación de la oportunidad, alcance y pautas de las **Metodologías Ágiles**.
 - En el marco de la **Programación Extrema**.
- Capacita para **definir los modelos**, desde los aspectos funcionales a los de implementación y despliegue, **validarlos y desarrollarlos** con pautas rigurosas.



Temario (1)

- Parte A. Metodologías y Ciclos de Vida
 - Tema 1. Evolución de las metodologías y sus enfoques.
 - Tema 2. Metodologías clásicas.
 - Tema 3. Encuadre de las metodologías actuales.

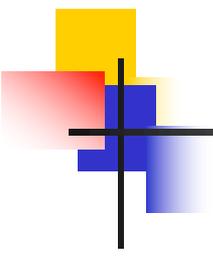
- Parte B. Metodologías para el Desarrollo Incremental.
 - Tema 4. Estructura y alcance del Proceso Unificado de Desarrollo de Software.
 - Tema 5. Proceso
 - dirigido por Casos de Uso.
 - centrado en Arquitecturas
 - iterativo e incremental.



Temario (2)

- Tema 6. Flujos de trabajo.
 - Captura de requisitos
 - Análisis
 - Diseño
 - Implementación
 - Validación y Prueba

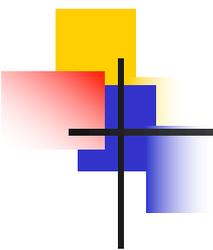
- Parte C. Metodologías ágiles de desarrollo
 - Tema 7. Introducción a metodologías ágiles
 - Motivación, principios básicos y planteamiento general.
 - Tema 8. Scrum
 - Roles, equipos y auto-organización.
 - Planificación de reuniones.
 - Sprints.
 - Gestión e implementación.



Temario (3)

- Tema 9. Programación Extrema (eXtreme Programming, XP)
 - Principios de XP
 - Conceptualización del sistema
 - Planificación: estimación, lanzamientos (releases), iteraciones.
 - Desarrollo: programación por pares, marcos de prueba (testing), diseño y modelado ágil, refactorización, integración.
 - Entrega (delivery).

- Parte D. Complementos.
 - Tema 10. Desarrollo colaborativo.
 - Tema 11. Evolución y mantenimiento del software.
 - Tema 12. El impacto del Legacy.
 - Tema 13. Otros aspectos del desarrollo. Deontología y encuadre legal.

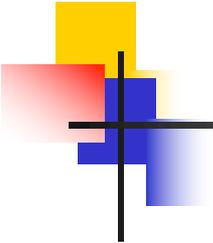


Temario (y 4)

- PRACTICAS.

Las prácticas se basarán en casos reales planteados por empresas colaboradoras.

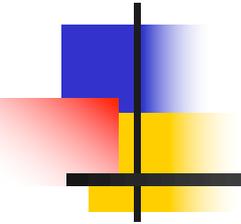
- Uso de herramientas de modelado y generación de código a partir de diagramas UML (p.ej. Rational, MagicDraw, etc).
- Uso de herramientas Java de programación extrema.

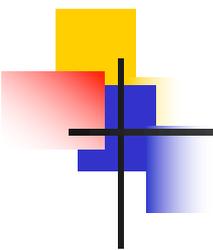


Profesorado

- Miguel Rodríguez Rubio. rodrubio@udc.es.
(coordinador). Despacho: 4.12
- Pedro Cabalar Fernández. cabalar@udc.es. Despacho
4.11

Validación y Verificación del Software (Q7)



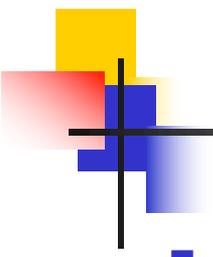


Resumen

- Esta asignatura busca dominar las alternativas actuales de la ingeniería de software para la **validación y verificación del software.**

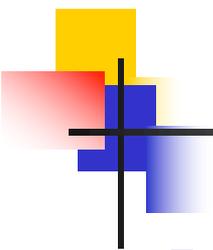
¿Cómo?

- Conociendo técnicas y herramientas de **validación** de software funcional y no funcional a todos los niveles (unidad, integración, sistema)
- Conociendo técnicas y herramientas de **razonamiento automático.**
- Conociendo técnicas y herramientas de **verificación formal.**



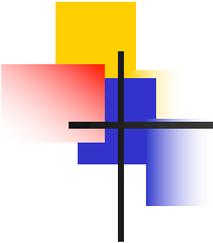
Temario

- Parte I: Validación de software
 - Especificación y ejecución de pruebas
 - Niveles y tipos de pruebas
 - Propiedades y trazabilidad de requisitos
 - Herramientas
 - Gestión de pruebas: planificación, evaluación, métricas y revisiones
- Parte II: Métodos formales y razonamiento automático
 - Introducción: Deducción natural y cálculo de secuentes.
 - Prueba automática utilizando PVS.
 - ¿Qué es y para que se utiliza un theorem prover?
 - Lenguaje de especificación de PVS: Tipos, expresiones, teorías, subtipado.
 - Probador PVS: tácticas, recursión, razonamiento ecuacional.



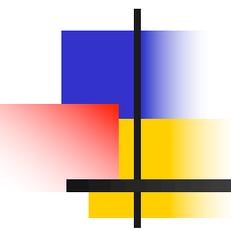
Temario (II)

- Parte III: Comprobación por modelos (Model checking)
 - Introducción a las lógicas modales temporales.
 - Especificación de propiedades: deadlocks, safety, liveness, fairness.
 - Funcionamiento de un comprobador por modelos.
 - Introducción al manejo de una herramienta de model checking: SPIN ó SMV.
- Prácticas:
 - Práctica con herramientas de validación de software.
 - Práctica de verificación formal en PVS.
 - Práctica de model checking (en SPIN o en SMV).

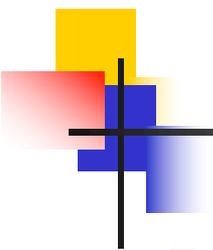


Profesorado

- Pedro Cabalar
cabalar@udc.es
Despacho 4.11
- Laura M. Castro [coordinadora]
lcastro@udc.es
Despacho 4.15
- Gilberto Pérez
gperez@udc.es
Despacho 4.01

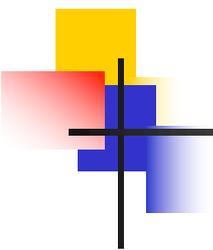


Proyectos de Desarrollo Software (Q8)



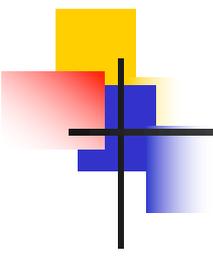
Resumen

- Esta asignatura aborda un problema importante en el desarrollo de proyectos software: el **incumplimiento de plazos y costes**. Dentro del conjunto de técnicas para resolverlo, está la realización de unas estimaciones realistas que hagan que el proyecto concluya con éxito.
- Se abordan distintas **técnicas de estimación** en proyectos de desarrollo software.
- La asignatura tiene un **fuerte componente práctico**, por lo cual la nota de prácticas tiene un peso importante en la calificación final. Se realizarán prácticas adecuadas al temario propuesto.



Temario

- **Tema 1.** Técnicas de estimación en proyectos de desarrollo software:
 - Introducción.
 - Métodos de estimación software.
 - Estimación del tamaño de un proyecto software: Puntos Función, Puntos de Casos de Uso, Puntos Historia.
 - Estimación del esfuerzo, duración y coste de un proyecto software: ISBSG, COCOMO II, etc.
- **Tema 2.** Dimensiones de un proyecto software (peopleware)



Profesorado

- Virginia Mato Abad, virginia.mato@udc.es,
Despacho 3.05 Edificio Área Científica
- Francisco Abel Cedrón Santaefemia,
francisco.cedron@udc.es
Centro Imedir, Edificio Servicios Centrales de Investigación