

---

# Presentación asignatura

SO 2023\_2024\_Q2

# Índice

---

- Descripción
- Competencias Técnicas
- Competencias Transversales
- Objetivos
- Metodología docente
- Organización
- Método de evaluación
- Canales de comunicación
  
- <http://docencia.ac.upc.edu/FIB/grau/SO>

# Descripción

---

- El objetivo de esta asignatura es:
  - Presentar los servicios y funcionalidades principales que ofrece el SO a los usuarios
  - Presentar cómo se gestionan internamente estos servicios
  - Dar los criterios necesarios para detectar problemas en un sistema multiusuario

# Competencias Técnicas (de nivel 2)

---

- CT6 - Demostrar conocimiento y comprensión del funcionamiento interno de un ordenador y del funcionamiento de las comunicaciones entre ordenadores.
  - CT6.1 - Demostrar conocimiento y tener capacidad para administrar y mantener sistemas, servicios y aplicaciones informáticas.
  - CT6.3 - Demostrar conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos que nos permita un uso adecuado, administración y diseño, así como la implementación de aplicaciones basadas en sus servicios.
- CT7 – Evaluar y seleccionar plataformas de producción hardware y software para la ejecución de aplicaciones y de servicios informáticos.
  - CT7.3 - Determinar qué factores inciden negativamente en la seguridad y la fiabilidad de un sistema hardware/software, y minimizar sus efectos.

# Competencias Transversales

---

- Actitud adecuada ante el trabajo (nivel 1)
- Asistencia, Puntualidad, Respeto, Motivación, Calidad y Responsabilidad
- ¿Qué tenéis que hacer?
  - En los laboratorios
    - ▶ Ser puntuales
    - ▶ Preparar las sesiones (trabajo previo)
      - Entrega antes de la sesión (Racó)
    - ▶ Hacer las sesiones (completas=2 horas) y **ASIMILARLAS**
      - Entrega al final de la sesión (Racó) + cuestionario
    - ▶ Ofrecer una cierta calidad del trabajo en los labs
  - En clase de teoría
    - ▶ Estar atento y comportarse correctamente
- ¿Cómo se evalúa?
  - Principalmente por el trabajo hecho en los labs

# Objetivos (resumen)

---

- **FUNCIONES: CONOCER Y UTILIZAR** las funciones básicas que ofrece un SO tipo Linux
  - Crear un nuevo “programa”
  - Cambiar su binario
  - Pedir más o menos memoria
  - Enviar/Recibir datos a/desde dispositivos
  - ...
- **FUNCIONALIDAD: CONOCER** los elementos de diseño interno BÁSICOS en un SO

# Metodología Docente (I)

---

- Clases de teoría
  - 2h/setmanales
  - Conceptos + ejemplos
  - Utilizaremos transparencias + pizarra
- 3 sesiones de problemas: ver planificación

# Metodología docente (II)

---

- Clases de laboratorio
  - 2h/setmanales
  - **9 Sesiones de laboratorio en grupos de dos**
    - ▶ Preparación previa → entrega a través del Racó (se cierra al inicio de la clase de la sesión)
    - ▶ Ejercicios en el laboratorio → entrega a través del Racó (se cierra unos minutos antes de que acabe la clase de la sesión)
  - Cuestionario **individual** al final de cada sesión (excepto sesión 1 y sesión 2)
    - ▶ Validación de las competencias y objetivos de las sesiones
  - **2 “simulacros” de examen individuales**
  - **Para optar a la nota de evaluación continua hay que asistir al 80% del las sesiones (7 sesiones)**
    - ▶ Asistir presencialment a toda la classe y hacer y entregar la sesión
  - **Normativa copias:** si se detecta que algún código de los entregados (trabajo previo, sesión o simulacro) es una copia, el estudiante tendrá un 0 de la componente de evaluación continua

# Método de evaluación (I)

---

- **NOTA C. Técnicas = MÁXIMO(0,4 \* EC + 0,6 \* EF, EF)**
  
- **EC = 0,35 \* CT + 0,10 \* ST + 0,30 \* CL + SL**
  - CT: parcial de teoría fuera de horas de clase
    - ▶ Problemas y/o preguntas cortas de los temas 1 y 2
    - ▶ **Sin apuntes**
  - ST: nota de seguimiento de teoría
    - ▶ Pequeño cuestionario sobre los temas 1 y 2 (ver planificación)
    - ▶ **Sin apuntes**
  - CL: parcial de laboratorio fuera de horas de clase
    - ▶ Implementación de código donde se aplican conceptos vistos en las sesiones 1,2,3, i 4
    - ▶ **Sin apuntes** (solo se pueden consultar las páginas del man instaladas en los ordenadores del aula)
  - SL: seguimiento de laboratorio
    - ▶  $SL = 0,10 * \text{Cuestionarios} + 0,05 * \text{Simlab1} + 0,10 * \text{Simlab2}$
    - ▶ Cuestionarios: al final de cada sesión de laboratorio (excepto sesión 1 i sesión 2)
    - ▶ Simlab1 i Simlab2: Simulacros de examen en horas de clase de laboratorio
      - Ejercicio de una hora y cuestionario al final del ejercicio
      - La segunda hora de clase el profesor resuelve el ejercicio
  
- **Normativa copias:** si se detecta que algún código de los entregados (trabajo previo, sesión o simulacro) es una copia, el estudiante tendrá un 0 de la componente de evaluación continua

# Método de evaluación (II)

---

■  $EF = 0,5 * T + 0,5 * L$

- **Examen de todo el curso en la franja de exámenes finales**

- T: final de teoría

- ▶ Preguntas cortas y problemas que evalúan todo el curso

- L: final de laboratorio

- ▶ Implementación de código donde se aplican conceptos vistos en cualquiera de las sesiones de laboratorio

# Canales de comunicación

---

- Ver página web.
  - <http://docencia.ac.upc.edu/FIB/grau/SO>
  - Documentación: planificación del curso, teoría, laboratorios (enunciados, ficheros), ejemplos, exámenes amb soluciones, colección de problemas
- **Racó FIB**
  - Información más dinámica: avisos, convocatoria exámenes, notas, etc
  - Prácticas via web
    - ▶ Trabajo previo laboratorio
    - ▶ Sesiones laboratorio
    - ▶ Exámenes