

### Nombre del Cuso: Laboratorio de análisis y diseño de sistemas II

<b>Categoría:</b> Obligatorio	<b>Semestre:</b> Primer semestre 2025
<b>Docente:</b> Ing. Claudia Rojas	<b>Auxiliar</b> Saul Castellanos
<b>Edificio:</b> MEET	<b>Salón:</b> MEET

#### 1. Descripción del laboratorio

El laboratorio del curso de Análisis y Diseño de Sistemas 2, trata sobre la parte práctica del curso. Se encarga de brindar a los estudiantes herramientas para la aplicación de principios y metodologías para utilizar en el proceso de construcción de software. Se guía a los estudiantes en la aplicación de patrones de diseño de software.

#### 2. Objetivos

##### General

Diseñar la arquitectura de software de un proyecto real, con base en diferentes puntos de vista y perspectivas para alcanzar los atributos de calidad.

##### Específicos

- Identificar y describir conceptos de arquitectura de software para el desarrollo de sistemas informáticos.
- Identificar y describir conceptos de atributos de calidad y tácticas aplicables para alcanzarlos.
- Evaluar y adaptar el diseño de un sistema de software de acuerdo con los diferentes tipos de requerimientos de un proyecto.

#### 3. Metodología

1. Clases magistrales para guiar y asesorar al estudiante.
2. Autoaprendizaje y lectura acerca de las herramientas a utilizar.
3. Prácticas y proyecto.

#### 4. Competencias terminales

Al finalizar el laboratorio el estudiante desarrolla las siguientes competencias:

- Dominar los conceptos básicos de la fase de análisis de la construcción de un proyecto de software.
- Aplicar diferentes tipos de framework en el diseño de arquitecturas de proyectos de software

#### 5. Observaciones

- Cualquier copia de prácticas, tareas o fases de proyecto, será sancionada con la pérdida automática del laboratorio y la sanción correspondiente por parte de la Escuela de Sistemas.
- Las tareas, prácticas y proyectos se entregan en la fecha indicada. Es de hacer notar que la PUNTUALIDAD es importante en la entrega de cualquiera de las actividades, de no cumplirse la nota de la actividad se verá afectada, así como también queda a discreción del auxiliar el recibir alguna fuera del tiempo estipulado.
- No se permite el uso de código bajado de Internet para el desarrollo de prácticas o proyectos.
- Cualquier librería o contenido obtenido de Internet debe de ser aprobado por el auxiliar antes de su uso en cualquier práctica o proyecto.



- El estudiante que no entregue un proyecto o una práctica no tendrá derecho a entregar la siguiente.
- Si alguna persona no puede asistir a la calificación de un proyecto o práctica, debe informar tanto al auxiliar como al catedrático el motivo de su falta, luego se procederá a determinar si se puede o no realizar la calificación del proyecto o práctica y bajo qué términos se efectuará si se realiza. Solamente el catedrático puede autorizar la calificación de algún proyecto o práctica después de la fecha establecida, acompañando la solicitud con una prueba que demuestre el motivo por el cual no pudo asistir a la misma.
- Se requiere de un 70% de asistencia al laboratorio.
- Es obligatorio aprobar el laboratorio con una nota mínima de 61 puntos para tener derecho a examen final y aprobación del curso.
- Copias parciales o totales serán penalizadas con el 100% de la nota y los implicados serán reportados a escuela.

## 6. Contenido temático del laboratorio

### 1. Frameworks

- 1.1. Qué es un framework
- 1.2. Tipos de framework
- 1.3. Beneficios del uso de frameworks
- 1.4. Características comunes de los frameworks
- 1.5. Casos de uso específicos
- 1.6. Diferencias entre framework y Librería

### 2. Elementos de diseño

- 2.1. Patrones de diseño
  - 2.1.1. Definición de patrones de diseño
  - 2.1.2. Importancia y aplicabilidad de patrones de diseño
  - 2.1.3. Notaciones de patrones de diseño
  - 2.1.4. Ejemplos de patrones de diseño
- 2.2. Contexto del diseño
- 2.3. Principios de diseño
- 2.4. Tipos de patrones de diseño
  - 2.4.1. Creacional
    - 2.4.1.1. Singleton
    - 2.4.1.2. Abstract Factory
    - 2.4.1.3. Factory Method
  - 2.4.2. Estructural
    - 2.4.2.1. Adapter
    - 2.4.2.2. Decorator
    - 2.4.2.3. Facade
    - 2.4.2.4. Proxy
  - 2.4.3. Comportamiento
    - 2.4.3.1. Observer
    - 2.4.3.2. Strategy
    - 2.4.3.3 Command
    - 2.4.3.4 Iterator
    - 2.4.3.5 State



- 2.5. Catálogo de estilos
- 2.6. Experiencia de usuario (UX)
- 2.7. Diseño de interfaces de usuario (UI)

### 3. Pruebas

- 3.1. Pruebas funcionales
  - 3.1.1. Pruebas unitarias
  - 3.1.2. Pruebas de aceptación
  - 3.1.3. Pruebas de integración
  - 3.1.4. Pruebas de regresión
- 3.2. Pruebas no funcionales
  - 3.2.1. Pruebas de carga
  - 3.2.2. Pruebas de estrés
  - 3.2.3. Pruebas de escalabilidad
  - 3.2.4. Pruebas de portabilidad

### 4. Seguridad

- 4.1. Introducción a la seguridad de sistemas
- 4.2. Modelado de amenazas
- 4.3. Diseño seguro de software
- 4.4. Pruebas de seguridad
- 4.5. Cumplimiento y Normativas de seguridad

### 5. Despliegue en la nube (Cloud)

- 5.1. Arquitectura por capas
- 5.2. Arquitectura 4 + 1 vistas
- 5.3. MVC
- 5.4. Arquitecturas monolíticas vrs microservicios
- 5.5. PaaS
- 5.6. SaaS
- 5.7. Docker
- 5.8. Infraestructura como código y automatización con Terraform y ansible

### 6. DevOps

- 6.1. Beneficios de DevOps
- 6.2. Ciclo de vida de las aplicaciones
- 6.3. Plan/ Code / Build / Test
- 6.4. Release / Deploy / Operate / Monitor
- 6.5. DevOps Tools
- 6.6. JSON vs XML

## 7. Evaluación de rendimiento académico

Procedimiento de evaluación		Ponderación / 100	Envío enunciado	Entrega
<b>Practica No. I</b>	Entregable único	5	3/02/2025	9/02/2025
<b>Proyecto Fase I</b>	Entregable 1	30	10/02/2025	23/02/2025
<b>Proyecto Fase II</b>	Entregable 2	30	24/02/2025	16/03/2025
<b>Proyecto Fase III</b>	Entregable 3	30	17/03/2025	13/04/2025
<b>Practica No. II</b>	Entregable único	5	13/04/2025	25/04/2025
<b>Nota de promoción</b>		<b>61 pts.</b>		