



## PROGRAMA ANALÍTICO DE ASIGNATURA

**DEPARTAMENTO:** Ingeniería en Sistemas de Información

**CARRERA:** Ingeniería en Sistemas de Información

**NOMBRE DE LA ACTIVIDAD CURRICULAR:** Diseño de Sistemas de Información

**Año Académico:** Plan 23

**Área:** Sistemas de Información

**Bloque:** Tecnologías Aplicadas

**Nivel:** 3º

**Tipo:** Obligatoria

**Modalidad:** Anual

**Cargas horarias totales:**

<i>Horas reloj</i>	<i>Horas cátedra</i>	<i>Horas cátedra semanales</i>
144	192	6

### OBJETIVOS

- Distinguir las metodologías, modelos, técnicas y lenguajes del proceso de diseño.
- Elaborar modelos de diseño de sistemas de información, aplicando diseño de arquitectura.
- Aplicar patrones de diseño de sistemas orientados a un diseño de calidad.
- Emplear conceptos de experiencia de usuario en el diseño del sistema de información.
- Aplicar herramientas de verificación y validación en el diseño, respetando criterios de calidad y seguridad.

### CONTENIDOS

#### Contenidos mínimos

- Diseño de Arquitecturas
- Patrones de Diseño de Sistemas
- Verificación y Validación del Diseño
- Diseño de Experiencia de Usuario
- Diseño de Procesos
- Estrategias de Prototipado y de Diseño de Componentes
- Integración de Sistemas
- Calidad y Seguridad en el Diseño de Sistemas de Información

#### Contenidos analíticos



### **Unidad 1: Diseño y Sistemas**

Introducción al Diseño. Qué es el Diseño. Diseño como Actividad y como Resultado.

Diferenciación entre Diseño de Sistemas de Información y Diseño de Software.

Influencias del Proceso de Diseño: Requerimientos Funcionales, No Funcionales, Reglas de Negocio, Ambiente de Desarrollo y Uso del Sistema, Recursos Humanos y las Herramientas Tecnológicas Disponibles. Atributos de Calidad y Cualidades del Diseño. Concepto de Proceso. Introducción a las metodologías de Desarrollo. Impacto de la metodología en el Diseño.

**Logros pedagógicos:** conocer el concepto de Diseño y comprender las diferencias y similitudes entre el diseño de sistemas y diseño de software. Comprender cómo los requerimientos influyen en el proceso de Diseño. Conocer los atributos de calidad de una solución, comprendiendo su importancia y observando cuál es su impacto en el diseño.

### **Unidad 2: Herramientas de Concepción y Comunicación del Diseño**

Herramientas para apoyar al proceso de análisis, diseño y desarrollo. UML y BPMN como herramienta.

Diagramas y su ubicación en el proceso de diseño:

Estáticos orientados al Diseño (diagrama de clases, diagrama de objetos, diagrama de componentes, diagrama de despliegue)

Dinámicos orientados al Diseño (diagrama de secuencia, diagrama de colaboración)

Diagrama Entidad-Relación. Modelo 4+1.

Herramientas no basadas en Diagramas.

**Logros pedagógicos:** Conocer y aplicar diferentes herramientas disponibles para el proceso de diseño. Comprender la complementariedad de las herramientas para comunicar la solución.

### **Unidad 3: Diseño con Objetos**

Conceptos Básicos del Paradigma de Objetos. Concepto de objeto, instancia, mensaje, clase, herencia, jerarquía, composición, interfaz, desde la visión de Diseño y Vinculado al diseño estructural. Diseño Dirigido por el Dominio. Modelado de Negocio y Procesos. Principios que guían el diseño: principios SOLID y "Design Smells" . Patrones de Diseño. Concepto de Patrón. Ventajas y Desventajas de su uso. Patrones creacionales. Patrones estructurales. Patrones de comportamiento. Refactorización mediante Patrones.

**Logros pedagógicos:** Modelar y construir soluciones aplicando el paradigma de objetos. Aplicar Patrones de Diseño. Justificar las decisiones de Diseño.

### **Unidad 4: Diseño de Interfaz de Usuario**

Concepto UX y CX. Diseño de servicios y productos digitales. Relaciones entre UX/UI. Diseño de interacción, prototipado, diseño visual. Aspectos generales de Accesibilidad

Web. Normas, leyes y herramientas de validación. Patrones de diseño y buenas prácticas.

Heurísticas: Jakob Nielsen, Tognazzini y Don Norman. Patrones de interacción. Conceptos de Diseño para Interfaces Web y Móvil.



**Logros pedagógicos:** Comprender las diferencias entre UX, CX y UI. Conocer los Patrones de Diseño de Interfaz. Aplicar Heurísticas para el Diseño de Interfaces de Usuario.

#### **Unidad 5: Diseño de Datos y Estrategias de Persistencia**

Normalización y Desnormalización como decisión del diseño de datos.

Modelo de Datos. Definición y Concepto de Persistir.

Arquitectura de Persistencia. Archivos: XML. CSV. Serialización. Prevalencia. Base de Datos: Tipos. Relacional, Orientada a Objetos, Multidimensionales, NoSQL: Clave Valor, Orientada a Columnas, Orientada a Documentos. Técnicas de Mapeo Objeto-Relacional/ Mapeo Objeto-Documental.

**Logros pedagógicos:** Comprender las diferentes estrategias de persistencia. Modelar y construir soluciones aplicando el modelo relacional. Aplicar técnicas de mapeo objeto-relacional y objeto-documental. Aplicar técnicas de Normalización y Desnormalización. Justificar las decisiones de Diseño.

#### **Unidad 6: Diseño de Arquitectura**

Definición y Concepto de Arquitectura. Concepto y Vistas de la Arquitectura. Influencias, requerimientos y restricciones del diseño arquitectónico

Patrones y Estilos Arquitectónicos. Componentes de una Arquitectura. Metodologías de Diseño y Evaluación de Arquitecturas. QAW. ATAM.

**Logros pedagógicos:** Conocer y Comprender el concepto de Arquitectura. Analizar y Diseñar soluciones arquitectónicas. Aplicar técnicas y metodologías en el proceso de diseño.

#### **Unidad 7: Integración de Sistemas**

Concepto de Integración. Necesidades. Síncrona y Asíncrona. Modelos de Integración: por base de Datos, por cola de mensajes, RPC, por bus. Representación de los datos: protocolos binarios y de texto. Interoperabilidad.

**Logros pedagógicos:** Conocer y Comprender el concepto de Integración. Analizar y Diseñar e Implementar soluciones de Integración.

#### **Unidad 8: Validación del Diseño**

Objetivo de la construcción de casos de prueba. Tipos de Pruebas. Concepto de Escenario. Diseño de pruebas unitarias y de integración. Diseño de Pruebas sobre Requerimientos No Funcionales: Performance, Carga, Stress. Motivación de Pruebas de Regresión y Automatización de Tests. Validación vs. Verificación.

**Logros pedagógicos:** Conocer y Comprender el concepto de validación. Diseñar e Implementar pruebas que permitan validar los requerimientos. Diseñar e Implementar pruebas que permitan validar el diseño del sistema.

#### **Unidad 9: Diseño y Seguridad**

Concepto de Seguridad de la Información. Tríada de la Seguridad: Confidencialidad, Integridad, Disponibilidad. Conceptos de Diseño y Desarrollo Seguro. Autenticación. Autorización.



*Universidad Tecnológica Nacional*  
*Facultad Regional Buenos Aires*

Mecanismos de Seguridad para Sistemas de Información. Componentes vinculados a la Seguridad de la Información.

**Logros pedagógicos:** Conocer el concepto de Seguridad de la Información. Aplicar estrategias de Diseño y Desarrollo seguro en las soluciones.

### **BIBLIOGRAFÍA OBLIGATORIA**

- Blancarte Iturralde, O. (2020). Introducción a la Arquitectura de Software – Un enfoque práctico. Ed. Oscar Javier Blancarte Iturralde.
- Blancarte Iturralde, O. (2016). Introducción a los Patrones de Diseño – Un enfoque práctico. Ed. Oscar Javier Blancarte Iturralde.
- Booch, Rumbaugh and Jacobson. (2006). Lenguaje de Modelado Unificado. Ed. Addison Wesley-Pearson Educación.
- Cervantes, H. (2016). Designing Software Architectures: A Practical Approach. Ed. Addison-Wesley Professional
- Cooper, A; Reimann, R.; Cronin, D. ; Noessel C. (2014). About Face – The Essentials of Interaction Design. Ed. Wiley.
- Gamma E. (2003) Patrones de Diseño. Ed. Addison Wesley.
- Larman C (2003). Uml y Patrones. Ed. Prentice Hall.
- Martin R. (2017). Clean Architecture: A Craftsman's Guide to Software Structure and Design. Ed. Pearson Education.
- Shvets, A. (2018). Dive Into Design Patterns. Ed. Refactorin .Guru.
- Sommerville, I. (2011). Ingeniería de Software (Novena ed.). Ed. Addison- Wesley.

### **BIBLIOGRAFÍA COMPLEMENTARIA**

- Norman D. (2013). The Design of Everyday Things. Ed. Basic Books.
- Evans E. (2003). Domain-Driven Design: Tackling Complexity in the Heart of Software. Ed. Pearson Education.