



## PROGRAMA ANALÍTICO DE ASIGNATURA

**DEPARTAMENTO:** Ingeniería en Sistemas de Información

**CARRERA:** Ingeniería en Sistemas de Información

**NOMBRE DE LA ACTIVIDAD CURRICULAR:** Ingeniería y Calidad de Software

**Año Académico:** Plan 2023

**Área:** Desarrollo de Software

**Bloque:** Tecnologías Aplicadas

**Nivel:** 4º

**Tipo:** Obligatoria

**Modalidad:** Cuatrimestral

**Cargas horarias totales:**

<i>Horas reloj</i>	<i>Horas cátedra</i>	<i>Horas cátedra semanales</i>
72	96	6

### OBJETIVOS

- Comprender los componentes de un proyecto de Ingeniería de Software. - Aplicar estándares internacionales de certificación en Ingeniería de Software. - Emplear las métricas de software que se aplican al desarrollo de software. - Aplicar técnicas y herramientas de auditoría de software
- Reconocer los componentes de un proyecto de Ingeniería de Software para gestionar con éxito todo el ciclo de vida del desarrollo/mantenimiento de software
- Aplicar estándares internacionales de certificación en Ingeniería de Software para asegurar la entrega de procesos/productos de software de alta calidad
- Emplear las métricas de software que se aplican al desarrollo de software para evaluar de manera objetiva y, desde distintas ópticas, proyectos, procesos y productos de software
- Aplicar técnicas y herramientas de auditoría de software que permitan la identificación de errores y posibiliten la elaboración de oportunidades de mejora

### CONTENIDOS

#### Contenidos mínimos

- Software e Ingeniería de Software
- Disciplinas de la Ingeniería de Software
- Gestión de Configuración de Software



- Modelos de Calidad de Software.
- Aseguramiento de Calidad del Producto de Software
- Validación y Verificación
- Enfoques en el desarrollo de software
- Despliegue de Software
- Métricas y Estimaciones de Software
- Auditoría de Software
- Plan de desarrollo y mantenimiento de software

### **Contenidos analíticos**

#### **Unidad 1: Introducción a la Ingeniería del SW**

Conceptos generales. Características del SW. Problemas habituales en el desarrollo del SW (nuevos proyectos, mantenimiento, administración y control). Errores más comunes (conceptos erróneos, reacciones ante los problemas). Razones por las que se cometen errores y persisten los problemas. Visibilidad. Causa y efecto. El enfoque de la Ingeniería de SW en la resolución de problemas.

**Logros Pedagógicos:** Comprender la visión ingenieril del desarrollo de SW. Aprender a reconocer los errores, sus causas típicas y sus acciones correctivas.

#### **Unidad 2: Modelos de Calidad**

Concepto de calidad. Gestión de la Calidad. Visiones de la Calidad. Contribuyentes a la calidad. Costos de la "No Calidad". Modelos de Calidad orientados a Productos de SW. Presentación del Modelo de Calidad ISO 25000. Modelos de Calidad orientados a Procesos de SW. Modelos de Madurez para el Desarrollo y Mantenimiento de Software.

**Logros Pedagógicos:** Desarrollar habilidades para evaluar objetivamente la calidad tanto en procesos de desarrollo y mantenimiento de software como de los productos generados. Obtener una visión general de modelos de calidad y su utilidad para evaluar organizaciones y productos. Comprender cómo conducir una evaluación.

#### **Unidad 3: Enfoques y Plan de Desarrollo y Mantenimiento de SW**

Objetivos y alcances. Dimensiones de un proyecto de D&M de SW: Roles y actores principales. Estimaciones: tamaño, esfuerzo y costo. Presentación de métodos. Aspectos a tener en cuenta en la calendarización y en la asignación de recursos. El modelo Cynefin. Aplicaciones prácticas de Scrum y Kanban. TimeBox Development.

**Logros Pedagógicos:** Adquirir habilidades para establecer un claro entendimiento entre los clientes y el equipo del proyecto sobre el plan que dirigirá el proyecto y sus componentes asociados. Comprender los contextos operativos del proyecto para tomar las decisiones adecuadas de planificación.



#### **Unidad 4: Auditoría y Gestión de la Configuración del SW**

Identificación y mantenimiento de los componentes de un proyecto y de un producto. Origen de los cambios. Factores de complejidad del proceso. Necesidad del SCM. La línea base. Identificación de componentes. Control de cambios. Control de versiones. Administración del código fuente. Integración Continua y Delivery Continuo. Despliegue Continuo. Auditoría de la Configuración. Auditorías funcionales, físicas y de procesos. Análisis de Impacto. Herramientas de SCM. Armado de un Plan de SCM. Implementación de SCM en una cultura DevSecOps.

**Logros Pedagógicos:** Desarrollar habilidades para identificar, organizar y controlar las modificaciones al software que está siendo construido por un equipo de desarrollo, maximizando la productividad y minimizando los errores.

#### **Unidad 5: Aseguramiento de la Calidad**

Introducción al Quality Assurance. Costo y detección de errores. Revisiones, inspecciones, walkthroughs. Peer reviews. Principios básicos. Elementos para revisar. Formal vs Informal. El factor psicológico. Estándares. Métricas para la calidad. Armado de un Plan de SQA

**Logros Pedagógicos:** Desarrollar habilidades para evaluar objetivamente los procesos ejecutados contra la descripción de los mismos y los procedimientos asociados.

#### **Unidad 6: Testing de SW**

Introducción al Testing de Software. Prueba de software. Principios básicos. Equivocación, defecto y falla. Tipos de prueba. Ciclo de vida de la prueba. Economía de la Prueba. Pruebas unitarias (caja negra, caja blanca, caja gris). Armado de condiciones y casos de prueba. Partición y Clases de Equivalencia. Grados de Cobertura de la Prueba y Complejidad Ciclomática. Prueba de integración. Otras pruebas: volumen, stress, performance, seguridad, concurrencia, etc. Prueba de aceptación de usuario. Pruebas de Exploración. Herramientas & Automatización del testing. Armado del Plan de Pruebas.

**Logros Pedagógicos:** Desarrollar habilidades para evaluar objetivamente los productos construidos contra los estándares y las especificaciones.

#### **Unidad 7: Métricas**

Métricas: concepto. Tipos de Métricas (orientadas al proceso, al producto, a la gestión de proyectos, etc.). Implementación de un Programa de Métricas. El paradigma Objetivo-Pregunta-Métrica.

**Logros Pedagógicos:** Desarrollar la capacidad de mediciones para ser usada como apoyo a la administración de un proyecto de Ingeniería de SW y al proceso de toma de decisiones.

#### **BIBLIOGRAFÍA OBLIGATORIA**

- Aniche, M (2022). Effective Software Testing. Ed. guide Manning.
- Ciceri C. (2022). Software Architecture Metrics. O'Reilly.
- Farley, D. (2021). Modern Software Engineering: Doing What Works to Build



Better Software Faster 1st Edition. Addison-Wesley.

- IEEE Computer Society . (2014). *Software Engineering Body of Knowledge*
- Jorgensen, P. deVries, B. (2022). *Software Testing - A Craftsman's Approach, Fifth Edition*. Auerbach Publications.
- Kennet, R. Baker, E. (2019). *Process Improvement and CMMI® for Systems and Software* 1st Edition. Auerbach Publications.
- McConnell, S. (2019). *More Effective Agile: A Roadmap for Software Leaders*. Construx Press.
- Pressmann, R. (2021). *Ingeniería de Software – Un Enfoque Práctico. – Novena Edición* McGraw-Hill
- Rodriguez Daniel, Sicilia Miguel Angel, Sanchez Salvador. (2012). *Ingeniería del Software - Un Enfoque desde la Guía SWEBOK*. Ibergarceta Publicaciones.
- Rodriguez Daniel, Sicilia Miguel Angel, Sanchez Salvador. (2012). *Ingeniería del Software - Un Enfoque desde la Guía SWEBOK*. Ibergarceta Publicaciones.(SWEBOK). IEEE Computer Society.
- Steve McConnell. (2019). *More Effective Agile: A Roadmap for Software Leaders*. Construx Press.
- Sommerville, I. (2019). *Engineering Software Products: An Introduction to Modern Software Engineering*. Pearson.

#### **BIBLIOGRAFÍA COMPLEMENTARIA**

- Frederick Brooks Jr. (1995). *Mythical Man-Month, The: Essays on Software Engineering, Anniversary Edition*. Addison-Wesley Professional.