****

***TÉCNICO SUPERIOR EN DESARROLLO DE SOFTWARE***

**Unidad Curricular:** **Ingeniería de Software i**

**Ubicación en el Diseño Curricular:** 1º año

**Año lectivo:** 2019

**Carga horaria semanal:** 4 hs. cátedra semanales

**Régimen de cursado:** Anual

**Profesora:** Susana Isabel Arce

**FUNDAMENTACIÓN**

Esta unidad curricular tiene la finalidad de introducir al estudiante en el trabajo de Ingeniería, llevando a cabo proyectos con la utilización de métodos y la ayuda de herramientas propias de su ámbito de acción. Reconocer, diferenciar y validar las necesidades que justifican sus proyectos. Documentar escenarios y propuestas que respondan a dichas necesidades contemplando las demandas de artefactos requeridos por las empresas de la región.

**PROPÓSITOS**

Los estudiantes a partir del desarrollo de la unidad adquieren la habilidad para abordar las fases del ciclo de vida de diferentes tipos de software, con la aplicación práctica del conocimiento científico a través de los métodos y técnicas adecuados, para el desarrollo de software.

**OBJETIVOS**

El estudiante tiene que desarrollar la habilidad para generar propuestas de modelos de negocios de proyectos de software.

Identificar y aplicar la metodología adecuada para el desarrollo de diferentes productos de software.

Identificar y establecer los lineamientos formales para el desarrollo de aplicaciones robustas, tomando decisiones técnicas relacionadas con la arquitectura de la aplicación como guía para el diseño de la misma.

**CONTENIDOS**

**Unidad 1**: El software y la ingeniería de software

La naturaleza del software

Definición de software

Dominios de aplicación del software

Software heredado

Ingeniería de software

El proceso del software

La práctica de la ingeniería del software

La esencia de la práctica

Principios generales

Mitos del software

**Unidad 2**: Modelos del proceso

Modelo general de proceso

Definición de actividad estructural

Identificación de un conjunto de tareas

Patrones del proceso

Evaluación y mejora del proceso

Modelos de proceso prescriptivo

Modelo de la cascada

Modelos de proceso incremental

Modelos de proceso evolutivo

Modelos concurrentes

Modelos de los procesos especializados

Desarrollo basado en componentes

El modelo de métodos formales

Desarrollo de software orientado a aspectos

El proceso unificado y sus fases

Modelos del proceso personal y del equipo

Tecnología del proceso

Producto y proceso

**Unidad 3**: Desarrollo ágil

Agilidad y el costo del cambio

Proceso ágil, principios de agilidad y políticas del desarrollo ágil

Factores humanos

Programación extrema de modelos ágiles de proceso

Desarrollo optativo de software

Herramientas para el proceso ágil

**Unidad 4**: Principios que guían la práctica

Principios fundamentales

Principios que guían el proceso, la práctica y toda actividad estructural

Principios de comunicación

Principios de planeación

Principios de modelado

Principios de construcción

Principios de despliegue

**Unidad 5**: Comprensión de los requerimientos

Establecer las bases, identificar participantes y reconocer múltiples puntos de vistas

Trabajar hacia la colaboración

Hacer las primeras preguntas

Indagación de los requerimientos

Despliegue de la función de calidad

Escenarios de uso

Desarrollo de casos de uso

Elaboración y elementos del modelo de requerimientos

Patrones de análisis

Requerimiento de las negociaciones

Validación de los requerimientos

**Unidad 6**: Modelado de los requerimientos

Análisis de los requerimientos

Reglas prácticas del análisis

Enfoques del modelado de requerimientos

Modelado basado en escenarios

Modelos UML

Conceptos de modelado de datos

Modelado basado en clases

Modelado orientado al flujo

Patrones para el modelado de requerimientos

**Unidad 7**: Diseño

El proceso de diseño

Conceptos de diseño

Conceptos de diseño orientados a objeto

Clases de diseño

El modelo del diseño

**METODOLOGÍAS DE TRABAJO**

Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes.

Facilitar el contacto directo con materiales e instrumentos, al llevar a cabo actividades prácticas, para contribuir a la formación de las competencias para el trabajo experimental como: identificación, manejo y control de variables y datos relevantes, planteamiento de hipótesis, trabajo en equipo.

Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.

Observar y analizar fenómenos y problemáticas propias del campo ocupacional. Ejemplo: el proyecto que se realizará durante el curso.

**ACTIVIDADES**

Producir diagramas de clase a partir de problemas correspondientes a diversos dominios. Analizar y discutir sus propiedades y corrección. Representar situaciones determinadas utilizando diagramas UML u otras técnicas. Analizar y discutir sus características y defectos. Modelizar y especificar casos de usos a partir de descripciones de situaciones realistas. Documentar escenarios. Revisar documentos de requerimientos de software utilizando buenas prácticas para determinar su calidad. Realizar revisiones cruzadas de especificaciones. Utilizar las herramientas de soporte en todas las actividades que realicen.

**EVALUACIÓN**

Prácticas de laboratorio

Entrega de reporte de Tópicos investigados

Exposición de temas

Defensa de proyectos.

**BIBLIOGRAFÍA**

Pressman, R. S. Ingeniería del Softwuare – Un enfoque práctico. 7ma. Edición McGraw Hill. Interamericana Editores. 2010

Weitzenfeld, A. Ingeniería de software orientada a objetos. Cengage learning editores. México. 2005.

Borrero, L. Tecnologías de la Información En Internet. Editorial Norma Colombia. 2003

Prof. Prog. Susana Isabel Arce