Facultad de Ingeniería - Universidad Nacional de Cuyo P1- PROGRAMA DE ASIGNATURA			
Asignatura:	**Ingeniería de Software I**		
Profesor Titular:	**Marisa Haderne**		
Carrera:	**Licenciatura en Ciencias de la computación**		
Año: 2019	**Semestre: 5to.**	**Horas Semestre: 80**	**Horas Semana: 5**

OBJETIVOS

- Reconocer los fundamentos de la ingeniería del software como una disciplina científica.
- Aplicar los elementos y conceptos integrados en los procesos de desarrollo de software para la documentación adecuada de un proyecto software.
- Especificar los requisitos software satisfaciendo las necesidades del cliente y optimizando los recursos existentes.
- Aplicar métodos formales para especificación de sistemas de software, especificación formal de sistemas concurrentes y de tiempo real y verificación de especificaciones
- Estimar los costes de un proyecto y determinar los tiempos de desarrollo.
- Aplicar métodos de planificación y estimación de proyectos software. Determinar costos y tiempos de desarrollo.
- Utilizar tecnologías y herramientas actuales y emergentes acordes a las necesidades del entorno.

CONTENIDOS

UNIDAD 1: INTRODUCCIÓN A LA INGENIERÍA DEL SOFTWARE

1.A. Ingeniería del software

Conceptos. Orígenes de la producción del software. Crisis de la industria del software. Software heredado. Desafío de la ingeniería del software. Los principios generales de la IS.

1.B. Sistemas de información. La IS y su relación con otras áreas

Dominios de aplicación del software. Teoría general de sistemas. Sistemas de información: conceptos y características. Sistemas de información, organizaciones y estrategia. Clasificaciones Aspectos éticos y sociales. Nociones de sistemas colaborativos. Problemática actual

1.C. Principios que guían la práctica de la IS

Principios de proceso, de práctica, de comunicación, planeación, modelado, construcción y despliegue.

UNIDAD 2: PROCESO SOFTWARE. MODELOS DE CICLOS DE VIDA

2.A. Procesos

Modelo general de proceso. Etapas fundamentales. Evaluación y mejora de procesos.

2.B. Ciclos de vida. Modelos de proceso prescriptivo

Modelos de ciclos de vida de desarrollo de software. Nociones de ciclo de vida. Ventajas. Ciclos de vida tradicionales y alternativos.

2.C. Otros modelos de proceso

Modelos de proceso especializado. Proceso unificado racional. Desarrollo ágil. Principios y políticas de desarrollo ágil.
Tendencias actuales de modelos de la IS

UNIDAD 3: INGENIERÍA DE LOS REQUERIMIENTOS

3.A. Introducción a los requerimientos del software

Conceptos y características de requerimientos. Requerimientos funcionales y no funcionales. Necesidad de la especificación de requerimientos. Productos de la Ingeniería de Requerimientos.

3.B. El proceso de Ingeniería de requerimientos

El contexto. Dimensiones de la especificación de requerimientos. Estudios de viabilidad. Propiedades deseables de los requisitos. Trazabilidad, manejo de conflictos y validación-verificación. Gestión de requerimientos.

3.C. Métodos formales en la Ingeniería del Software

Introducción a los métodos formales. Sus aplicaciones a la ingeniería del software. Fundamentos matemáticos de los métodos formales. Lenguajes formales para especificación de sistemas de software. Verificación de especificaciones.

UNIDAD 4: LENGUAJES DE ESPECIFICACIÓN. MODELADO DE LOS REQUERIMIENTOS

4.A. Lenguaje unificado de modelado.

Las primeras metodologías y el surgimiento de UML. Objetivos y características Visión general: Vistas, Diagramas, elementos de modelado, mecanismos, extensiones a UML (estereotipos, valores etiquetados, restricciones). Diagramas de Actividad. Modelado a distintos niveles. Modelado de Casos de Uso. Relación con los requisitos. Estructura de un documento

4.B. Modelado de requerimientos

Diagrama de clases. Clases y objetos. Diagrama de secuencia. Diagrama de comunicación. Diagramas de máquinas de estado. Extensiones a UML. Representación de Arquitecturas. Estructura lógica. Componentes. Colaboraciones. Diagramas de estructura de compuestos. Diagrama de componentes.

UNIDAD 5: METRICAS Y ESTIMACION DE PROYECTOS DE SOFTWARE

5.A. Métricas de proceso y de proyecto

Las métricas del proceso y la mejora del proceso de software. Métricas de proyecto. Medición del software. Métricas orientadas a tamaño, a función, a objeto y a casos de uso. Métricas para calidad de software. Medición de la calidad. Integración de métricas dentro del proceso

5.B Métricas de producto

Medidas, métricas e indicadores, principios de medición. Medición de software orientado a meta. Atributos de las métricas de software efectivas. Métricas para el modelo de requerimientos: métrica basada en funciones, métricas para calidad de la especificación. Métricas para el modelo de diseño: métricas del diseño arquitectónico, para diseño orientado a objetos, orientadas a clase. Métricas de diseño de interfaz de usuario, para código fuente y para pruebas

5.C. Estimación para proyectos software

Estimación de proyectos de software. Técnicas de descomposición. Modelos de estimación empíricos. La estructura de los modelos de estimación. Estimación del esfuerzo: Incertidumbres, construcción de los modelos. Gestión de Costos. Retos de la Estimación de Costos Software. Elaboración de Presupuestos. Estimación del Tamaño mediante diferentes técnicas.

***5.D.* Modelos de estimación**

La estructura de los modelos de estimación. El modelo COCOMO II. Estimación para proyectos orientados a objetos, técnicas de estimación especializadas. Estimación para desarrollo ágil. Estimación para webapp. Adquirir o desarrollar software. Outsourcing

UNIDAD 6: PERSPECTIVAS FUTURAS

6.A. Perspectivas futuras

Mejoramiento del proceso de software. Tendencias emergentes en ingeniería del software

METODOLOGÍA DE ENSEÑANZA

La materia se organiza en clases teóricas, clases prácticas y actividades de laboratorio.
En las clases teóricas se brindan los contenidos fundamentales de la asignatura. Se desarrollan actividades de aprendizaje que propicien la aplicación de los conceptos, modelos y metodologías que se van aprendiendo en el desarrollo de la asignatura. Se propicia el uso adecuado de conceptos, y de terminología científico-tecnológica.
Las estrategias pedagógicas permitirán al alumno:

- observar y analizar fenómenos y problemáticas propias del campo ocupacional,
- relacionar los contenidos de esta asignatura con las demás del plan de estudios para desarrollar una visión interdisciplinaria en el estudiante y proponer situaciones que permitan al estudiante la integración de contenidos.

En las clases prácticas se ejercita sobre los temas cubiertos en la teoría, con especial énfasis en actividades de ingeniería del software aplicadas a casos reales. Llevar a cabo actividades prácticas que promuevan el desarrollo de habilidades para la experimentación, el planteamiento de hipótesis, y trabajo en equipo. Se utilizarán recursos pedagógicos basados en tecnologías de la información y la comunicación para permitir una interacción y comunicación docente-alumno continua y eficiente.
En el laboratorio se propicia el uso de las nuevas tecnologías en el desarrollo de los contenidos de la asignatura, fomentando que los alumnos desarrollen habilidades prácticas.
La resolución de problemas debe conducir al desarrollo de las competencias necesarias para la identificación y solución de problemas abiertos de ingeniería, entendiendo como tal aquellas situaciones reales o hipotéticas cuya solución no es única y requiere la aplicación de los conocimientos de las ciencias básicas y de las tecnologías.
Se seleccionarán estrategias de enseñanza que fomenten actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.

Actividad	**Carga horaria por semestre**
Teoría y resolución de ejercicios simples	32
Formación práctica	
Formación Experimental – Laboratorio	23
Formación Experimental - Trabajo de campo	0
Resolución de problemas de ingeniería	25
Proyecto y diseño	0
Total	**80**

BIBLIOGRAFÍA

Bibliografía básica

Autor	Título	Editorial	Año	Ejemplares en biblioteca
Pressman, Roger	Ingeniería de Software. Un enfoque práctico	McGraw-Hill	2010	
Pantaleo, Guilermo - Rinaudo, Ludmila	Ingeniería del software	Alfaomega Grupo editor	2014	
Sommerville, Ian	Ingeniería de Software	Pearson	2011	

		Education		

Bibliografía complementaria

Autor	Título	Editorial	Año	Ejemplares en biblioteca
Piattini, Velthuis	Fábricas de Software	Alfaomega Grupo ed.	2007	
J. Rumbaugh, I. Jacobson, G. Booch	El lenguaje unificado de modelado. Manual de referencia	Pearson Educación	2007	

EVALUACIONES (S/ Ord. 108-10_CS)

El régimen de evaluación se ajusta a lo establecido por la Ordenanza 108/10/CS- Anexo I, y a las normas reglamentarias específicas y resoluciones de casos particulares de la Facultad.

Evaluaciones durante el cursado

Para la evaluación continua en el aula se considerarán la resolución de problemas, trabajos prácticos y evaluaciones parciales, con observación del desempeño individual y grupal y retroalimentación constante. La evaluación incluirá propuestas de autoevaluación que posibiliten la reflexión del alumno como protagonista de su proceso de aprendizaje.

Condición de regularidad tras el cursado

Para adquirir la condición de alumno regular se deberá cumplir con los siguientes requisitos:

- Aprobar los dos parciales, cada uno de ellos con puntaje máximo de 100 puntos. Para la obtención de la regularidad es necesario aprobar las dos evaluaciones con un mínimo de 60 puntos en cada uno de ellos
- Aquellos alumnos que desaprueben alguna de las dos evaluaciones parciales o las dos, podrán rendir su examen recuperatorio correspondiente, con los mismos temas, los cuales se aprueba con 60 puntos sobre un total de l00.
- Aprobar los trabajos prácticos, casos de resolución de problemas entregados en tiempo y forma y actividades evaluadas durante el cursado o sus recuperatorios correspondientes.
- Cumplir con el 70% de asistencia.

Evaluación final

Para la aprobación de la asignatura del estudiante en condición regular, se realizará una evaluación calificativa que apunta a identificar cuánto ha aprendido un alumno en este espacio curricular con el fin de tomar la decisión de acreditación de saberes. Constituye una evaluación de resultados que plantea una actividad integradora, de síntesis, que refleje el trabajo del estudiante, comprensivo y fundamentado, y brinda al docente, una mirada global del proceso de aprendizaje.
Para el examen final, el alumno debe presentarse con la carpeta de trabajos prácticos completa.

Examen final para alumnos libres. El mismo es escrito y se aprueba con un mínimo de 60 puntos sobre un total de 100. Una vez aprobado el examen escrito, el alumno libre deberá rendir un examen oral según los criterios de evaluación mencionados previamente.

Alumnos recursantes.

No hay régimen especial para alumnos recursantes.