



UNIVERSIDAD CATÓLICA DEL NORTE

PROGRAMA DE CURSO

I. IDENTIFICACIÓN DEL CURSO	
Carrera: Magíster en Ingeniería Informática	
Unidad responsable: Departamento de Ingeniería de Sistemas y Computación	
Nombre del curso: Ingeniería de Requisitos	
Código: Por definir	
Semestre en la malla:	
Cursos Requisitos: Procesos de desarrollo de software	Requisito para:
II. PROPÓSITO DEL CURSO	
Al finalizar el curso el estudiante estará capacitado para elicitar, analizar y especificar las necesidades de los stakeholders, tomando en cuenta los posibles factores que afectan el desarrollo de software.	
III. UNIDADES TEMÁTICAS	
<ol style="list-style-type: none">1. INTRODUCCIÓN A LA INGENIERÍA DE REQUISITOS<ol style="list-style-type: none">1.1. CONCEPTOS E IMPORTANCIA DE LA IR1.2. CÓMO ESCRIBIR REQUISITOS1.3. EL PROCESO DE IR2. ESPECIFICACIÓN<ol style="list-style-type: none">2.1. TIPOS DE ESPECIFICACIÓN2.2. PROPIEDADES DE UNA ESPECIFICACIÓN2.3. ESTÁNDAR IEEE-8302.4. HERRAMIENTAS DE REQUISITOS3. VALIDACIÓN<ol style="list-style-type: none">3.1. CONCEPTOS BÁSICOS DE VALIDACIÓN3.2. REVISIONES E INSPECCIONES3.3. OTRAS TÉCNICAS DE VALIDACIÓN4. ELICITACIÓN<ol style="list-style-type: none">4.1. PROBLEMAS ASOCIADOS A LA ELICITACIÓN DE REQUISITOS4.2. TÉCNICAS DE ELICITACIÓN4.3. DE LA ELICITACIÓN AL ANÁLISIS5. ANÁLISIS<ol style="list-style-type: none">5.1. CONCEPTOS BÁSICOS DEL ANÁLISIS	

- 5.2. TÉCNICAS DE ANÁLISIS
- 5.3. MODELAMIENTO CONCEPTUAL
- 6. PROTOTIPADO
 - 6.1. PROPÓSITO DE LOS PROTOTIPOS
 - 6.2. PROCESO DE CONSTRUCCIÓN
 - 6.3. EVALUACIÓN DE PROTOTIPOS
- 7. GESTIÓN
 - 7.1. CONTROL DE CAMBIOS
 - 7.2. PLANIFICACIÓN DE ENTREGAS

IV. RESULTADOS DE APRENDIZAJE

- A. Identificar los requisitos funcionales y no funcionales de un sistema de software existente
- B. Redactar una especificación de requisitos en base a un estándar
- C. Inspeccionar requisitos de software para validar y verificar su calidad
- D. Elicitar los requisitos de un sistema de software utilizando las técnicas de educación más adecuadas
- E. Analizar los requisitos funcionales y no funcionales de un sistema de software
- F. Crear un prototipo de un sistema de software
- G. Implementar un sistema de control de cambios para los requisitos funcionales y no funcionales
- H. Crear un modelo de software basado en la especificación de requisitos

V. RECURSOS BIBLIOGRÁFICOS

Textos Guías:

- K. Wiegers. Software Requirements. Microsoft Press, 2nd ed. 2003

Textos o lecturas complementarias:

- G. Kotonya y I. Sommerville. Requirements engineering: Processes and techniques. John Wiley and Sons. 1998
- I. Sommerville & P. Sawyer. Requirements engineering. A good practice guide. John Wiley and Sons. 1997
- Otras lecturas.