

Programa de Asignatura

00 Código:	F	C	Asign
	T	417	20

01. Facultad: Tecnología Informática / Carrera: Analista Programador

02. Asignatura: T104 – 20 – TRABAJO FINAL.

03. Año lectivo: 2022

04. Año de Cursada: 3º

05. Cuatrimestre: 1º

06. Carga horaria semanal: 6

07. Créditos:

08. Equipo Docente

Profesor Titular: Ing. Cardacci Darío Guillermo

Adjuntos:

Auxiliar:

09. Asignaturas correlativas previas y posteriores:

Correlativas previas:	14. Lenguajes de Última Generación
	15. Metodologías de Desarrollo de Sistemas II
	16. Base de Datos
Correlativas posteriores:	No posee

10. Fundamentación:

a- Aporte específico de la asignatura a la formación académico profesional

Es fundamental para la formación del futuro analista programador integrar los conocimientos adquiridos, así como implementar las técnicas y las buenas prácticas que impone la programación de sistemas informáticos.

La refactorización de piezas de código y la aplicación de técnicas avanzadas de programación aplicadas en un caso práctico testigo, hacen que los desarrollos posean una estructura constructiva más apta a los requerimientos profesionales.

En particular, las técnicas de programación aplicadas en la asignatura potencian al modelo orientado a objetos trabajado en asignaturas previas. Poder integrar conocimientos de análisis y diseño de sistemas, bases de datos y programación, desde un andamiaje teórico y una aplicación práctica, le permiten al alumno redimensionar y potenciar los conocimientos adquiridos en la carrera. Este es el propósito de esta asignatura.

La tarea de los analistas de sistemas excede el mero desarrollo de software. Dentro del área profesional que les incumbe, deben dominar las técnicas y metodologías que le permitirán realizar una correcta administración de los recursos informáticos.

Los profesionales preparados para desarrollar esta actividad deben manifestarse con soltura en las actividades que involucran los aspectos que se indican a continuación:

- Habilidad y predisposición para lograr crear ámbitos virtuales de solución a problemas reales, utilizando y administrando eficientemente los recursos informáticos.
- Incorporar las destrezas para lograr transformar la observación de la realidad compleja a una representación simplificada y sistémica.
- Desarrollar las habilidades que permitan implementar soluciones probadas y estándares que otorgan seguridad y eficiencia en las actividades llevadas a cabo.
- Capacidad y motivación para innovar en los conceptos referidos a la programación.
- Conocimiento para la toma de decisiones que permitan comprender, adaptar y aplicar conceptos altamente calificados para el desarrollo de software.
- Desarrollar la capacidad para trabajar en forma individual y/o grupal.
- Poder tomar decisiones bajo estrictas valoraciones técnicas, éticas y morales.

b- El /los marco/s conceptual/es que sustenta/n el enfoque de la asignatura

La asignatura aborda la forma de desarrollar software utilizando el modelo orientado a objetos, las especificaciones técnicas generadas con el lenguaje de modelado unificado y la implementación de bases de datos normalizadas. También explora las tecnologías asociadas que permiten lograr software de alta calidad respetando la documentación técnica y las normas internacionales que establecen las buenas prácticas para la obtención del software.

c- La articulación de la asignatura con asignaturas previas y posteriores

Articulación Vertical:

La articulación con otras asignaturas se produce a partir de la orientación de aquellas al desarrollo del sistema que esta asignatura propone y a la integración teórica y práctica que Trabajo Final promueve.

La articulación con las asignaturas Metodología de Desarrollo de Sistemas I y II se produce a partir del aporte que éstas realizan al análisis y la modelización de sistemas, así como el uso de una simbología estándar para la representación de los mismos. De las asignaturas Lenguajes de Última Generación y Programación Orientada a Objetos se obtienen los conocimientos de programación y acceso a datos para desarrollar un sistema de información.

Para la implementación también se construye un modelo de datos y la base de datos asociada, temas tratados en la asignatura Base de Datos.

Articulación Horizontal:

Articula con Programación II la que refuerza las técnicas avanzadas de programación.

11. Competencias / sub-competencias y resultados de aprendizaje a las que tributa la asignatura:

Competencias del perfil	Subcompetencia	Nivel de dominio de la subcompetencia	Resultados de aprendizaje
2- Colaborar en la análisis, planificación, programación, implementación y mantenimiento de desarrollo de software de forma efectiva.	2-1- Capacidad para seleccionar el paradigma de programación más adecuado para enfocar los procesos de construcción de un software	2	T4-17-20-2-1-2-RA1: [Comprende] + [las características de los programas orientados a objetos] + [para formular software de calidad] + [utilizando métodos y estrategias estandarizadas]
5 - Integrar tareas técnicas de servicios informáticos en instituciones públicas, en empresas de producción y servicios públicos y privados, para realizar aportes que se incorporen de manera sustantiva desde la programación.	5.1 Capacidad para actuar técnicamente con eficiencia en diferentes tipos de organizaciones	3	T4-17-20-5-1-3-RA2: [Organiza] + [Las actividades que se utilizan en el desarrollo de software] + [Para formular software eficientemente] + [Utilizando los procesos que propone la disciplina]
6 - Actuar con autonomía, responsabilidad y compromiso social ante los desafíos del ejercicio profesional.	6.1 Capacidad para buscar y clasificar fuentes de información académica válida	3	T4-17-20-6-1-3-RA3: [Establece] + [Clasificaciones jerárquicas de información científico-técnica] + [Para aplicar en el desarrollo de software] + [Aplicando criterios de acceso múltiples]

	6.2 Capacidad para aplicar valores morales y éticos en el ejercicio profesional	3	T4-17-20-6-2-3-RA4: [Propone] + [Prácticas éticas durante el proceso de desarrollo] + [Para garantizar el producto y la relación con los integrantes del equipo] + [Considerando el escenario donde se radica la práctica]
<i>7 - Competencia para desempeñarse de manera efectiva en equipos de trabajo, conforme al perfil profesional desarrollado.</i>	7.1 Capacidad para reconocer y respetar los puntos de vista y opiniones de otros miembros del equipo y llegar a acuerdos.	3	T4-17-20-7-1-3-RA5: [Valora] + [La opinión técnica de los pares] + [Para garantizar una visión equilibrada en la solución de software] + [Considerando los estándares propuestos por el UP y UML]
	7.2 Capacidad para expresarse con claridad y de socializar las ideas dentro de un equipo de trabajo.	3	T4-17-20-7-2-3-RA6: [Construye] + [Un discurso claro y preciso] + [Para transmitir al equipo las consignas de trabajo] + [Considerando los características y formación de los integrantes]
	7.3 Capacidad para promover una actitud participativa y colaborativa entre los integrantes del equipo.	3	T4-17-20-7-3-3-RA7: [Crea] + [Un ambiente de trabajo] + [Para estimular la participación de los integrantes del equipo] + [Teniendo en cuenta la multiplicidad de opiniones que se generan]
<i>8 - Competencia para comunicarse de forma efectiva y eficiente a fin de optimizar el intercambio de ideas y conceptos en su entorno laboral.</i>	8.1 Capacidad para seleccionar las estrategias de comunicación en función de los objetivos y de los interlocutores y de acordar significados en el contexto de intercambio.	3	T4-17-20-8-1-3-RA8: [Decide] + [Entre distintas formas comunicacionales] + [Para lograr el objetivo propuesto en el proyecto de manera eficiente y eficaz] + [Considerando el contexto de difusión de la información]
	8.2 Capacidad para producir e interpretar textos técnicos y presentaciones públicas.	3	T4-17-20-8-2-3-RA9: [Formula] + [Objetivamente los modelos de software y datos] + [Para transmitir la información involucrada en el desarrollo del producto software] + [Considerando la interpretación]

			que realizarán los que accedan a ella]
--	--	--	--

12. Unidades de desarrollo de los contenidos:

Unidad 1: DESARROLLO DEL ANÁLISIS Y DISEÑO DE UN SISTEMA. (CASO PRÁCTICO)

Refinamiento de la selección del dominio del problema. Refinamiento de la selección del sistema a programar.

Alcance del sistema. Requerimientos. Requerimientos funcionales y no funcionales.

Documento Visión. Consideraciones tecnológicas para el desarrollo. UML lenguaje de modelado unificado. Modelo de análisis orientado a objetos. Planteo iterativo e incremental del sistema.

Casos de uso. Especificaciones de casos de uso. Diagrama de Clases. Diagrama de Secuencia.

Trabajo final práctico: Desarrollo del documento visión y las especificaciones de caso de uso. Diagrama de secuencia y diagrama de clases.

Tiempo: 18 hs

Unidad 2: DESARROLLO DE LA ESTRUCTURA DE DATOS. (CASO PRÁCTICO)

Concepto de repositorio de datos. Bases de datos. Bases de datos relacionales. Dependencia funcional. Claves. Claves candidatas. Claves primarias. Entidades. Relaciones. Semántica de una relación. Modalidad y cardinalidad de una relación. Claves foráneas. Normalización. Primera, segunda y tercera forma normal. DER (Diagrama entidad relación).

Trabajo final práctico: Desarrollo del diagrama DER.

Tiempo: 18 hs

Unidad 3: PROGRAMACIÓN DEL SISTEMA. (CASO PRÁCTICO).

Consideraciones para la selección de la plataforma de desarrollo, la selección del lenguaje de programación y la base de datos.

Desarrollo de los módulos de servicio del sistema. Gestión de acceso. Encriptación. Backup.

Desarrollo del módulo de acceso a datos. Consideraciones de ORM (mapeador objeto relacional). Manejo de transacción y aseguramiento de la integridad de los datos.

Desarrollo del módulo de presentación. Consideraciones sobre la vista y usabilidad del sistema.

Desarrollo del módulo de negocio. Consideraciones para generar la información que requiere el sistema planteado.

Pruebas. Pruebas unitarias. Pruebas de rendimiento. Pruebas de sistema.

Trabajo final práctico: Programación del sistema planteado. Entrega de la carpeta con la documentación desarrollada en la unidad I y II más el sistema programado.

Tiempo: 60 hs

13. Metodología:

a. Encuadre metodológico de la asignatura

La asignatura plantea un abordaje metodológico sustentado en diferentes estrategias. Temporalmente las actividades se encuentran distribuidas en 16 semanas. En el primer encuentro sincrónico se presenta la asignatura, los resultados de aprendizaje esperados y la distribución de las distintas actividades que se desarrollarán. En particular se explica la distribución temporal del cronograma afectado al trabajo final que realiza el alumno. Para los siguientes encuentros sincrónicos se le solicita al alumno que previamente aborde el material que será tratado desde las guías de orientación desarrolladas para la asignatura.

En los encuentros se orienta al alumno en los aspectos conceptuales que ha abordado y los lineamientos de avance sobre el trabajo final. Se propone que los alumnos trabajen grupalmente en la resolución de problemas generales que deban aplicar los temas desarrollados en los orientadores y los encuentros. Deberán presentar la solución a través de los foros y actividades planteadas en la plataforma on line. En los encuentros, como actividad de cierre se reflexiona grupalmente sobre las conclusiones a las que se ha arribado y como esto aporta a los resultados de aprendizaje finales que se espera lograr. Finalmente se plantea la actividad a desarrollar de manera asincrónica y los objetivos a alcanzar para el siguiente encuentro sincrónico. Se sugiere realizar:

- a. La lectura del orientador.
- b. El avance establecido para el trabajo final.
- c. La resolución de lo correspondiente a la guía de revisión conceptual y la guía de trabajo práctico.
- d. La participación en el foro correspondiente a la unidad de la asignatura que se está abordando.
- e. Cuando la situación lo amerite la resolución de un desafío, una indagación bibliográfica o la resolución de un trabajo integrador que se desarrollará en algunas oportunidades de manera grupal y en otras de forma individual.

Durante toda la cursada se promueve la articulación e integración de los contenidos a través de actividades que remitan a la puesta en práctica de los conceptos teóricos.

Se tendrá especial atención a la participación de los alumnos en los espacios de intercambio y colaboración, para poder rescatar la elaboración conceptual que vayan construyendo acerca de los temas tratados. Los textos de actualidad y las notas periodísticas sobre tecnología de punta, serán propuestos para ser leídos y tratados en la cursada. Se fomentará la participación de los alumnos en la formulación de soluciones a los problemas de la práctica profesional. Se desarrollarán espacios de intercambio de conocimientos entre los alumnos, para que puedan comparar los desarrollos individuales de su trabajo práctico final. Se alentará el desarrollo de investigaciones sobre temas abordados en la asignatura.

b. Contextos de desarrollo de las actividades formativas

Las actividades formativas se llevarán a cabo en dos ámbitos diferente.

Los ámbitos utilizados serán: El aula virtual en el LMS institucional y los encuentros sincrónicos.

De manera complementaria y de forma asincrónica el alumno desarrollará el trabajo final de carrera.

Se propone como complemento a las estrategias de aprendizaje, el empleo de tecnologías digitales que favorezcan el desarrollo de actividades (individuales o grupales) participativas y colaborativas que promuevan la autonomía del estudiante en el proceso de enseñanza sobre la base de siguientes actividades:

- Profundización de contenidos: Tutoriales, videos, lecturas complementarias.
- Demostraciones: Resultado de procesos deductivos
- Resolución de problemas: Indagación de respuestas sobre problemas simulados a fin de crear una solución.
- Foros: Discusión colaborativa sobre temas de interés de la asignatura.

Se promoverá la utilización del aula invertida con el objetivo de potenciar los trabajos de integración que permitan desarrollar los resultados de aprendizaje establecidos en el programa de la asignatura para ir acercándolos paulatinamente a las problemáticas profesionales características de la carrera.

Actividad formativa	Ámbito	Resultado de aprendizaje	Contenidos comprometidos	Producción de los estudiantes	Observaciones
Planificación de actividades	LMS	Organiza las actividades que se utilizan en el desarrollo de software para formular software eficientemente	Unidad 1	Trabajo Final	

		utilizando los procesos que propone la disciplina			
Jerarquización de contenidos	LMS	Establece clasificaciones jerárquicas de información científico-técnica para aplicar en el desarrollo de software aplicando criterios de acceso múltiples.	Unidad 1	Trabajo Final	
Liderazgo	LMS	Propone prácticas éticas durante el proceso de desarrollo para garantizar el producto y la relación con los integrantes del equipo considerando el escenario donde se radica la práctica.	Unidad 1 Unidad 2	Trabajo Final	
Trabajo en grupo	LMS	Valora la opinión técnica de los pares para garantizar una visión equilibrada en la solución de software considerando los estándares propuestos por el UP y UML	Unidad 1 Unidad 2	Trabajo Final	
Comunicación efectiva	LMS	Construye un discurso claro y preciso para transmitir al equipo las consignas de trabajo considerando los características y formación de los integrantes	Unidad 1 Unidad 2	Trabajo Final	
Trabajo en grupo	LMS	Crea un ambiente de trabajo para estimular la participación de los integrantes del equipo teniendo en cuenta la multiplicidad de	Unidad 3	Trabajo Final	

		opiniones que se generan			
Comunicación efectiva	LMS	Decide entre distintas formas comunicacionales para lograr el objetivo propuesto en el proyecto de manera eficiente y eficaz considerando el contexto de difusión de la información	Unidad 3	Trabajo Final	
Resolución de problemas	LMS	Formula objetivamente los modelos de software y datos para transmitir la información involucrada en el desarrollo del producto software considerando la interpretación que realizarán los que accedan a ella	Unidad 3	Trabajo Final	

c. Recursos didácticos

Se utilizarán los siguientes recursos didácticos:

- Guía de orientación de trabajo final
- Orientadores por unidad
- Ejercicios Integradores
- Material multimedial
- Código de programación
- Foros
- Docs de Microsoft
- Learn de Microsoft

14. Procedimiento de evaluación y criterios de promoción:

a. Modalidad y Criterios de Evaluación

Parciales.

Los exámenes parciales son tres. Cada uno coincide con una fase del desarrollo del sistema que realiza el alumno. El primero se deberá entregar el documento visión. El segundo consta de la entrega de las especificaciones de caso de uso, el diagrama de secuencia, el diagrama de clases y el diagrama DER. También se incluirán las correcciones menores que se solicitaron en la devolución del primer parcial. En el tercer parcial se deberá entregar el programa desarrollado y los instrumentos solicitados en el primer y segundo parcial en su versión definitiva.

Los parciales son obligatorios e individuales. Las fechas para su realización serán las publicadas en el campus virtual. Si el alumno no realiza un examen parcial o los errores son sustanciales se calificará con una nota insuficiente.

Recuperatorios.

Si un examen parcial posee errores se le solicitará al alumno que corrija los mismos dentro de los plazos previstos en el campus virtual. Si el alumno no aprueba alguno de los parciales accederá a la instancia de recuperatorio de materia en los plazos indicados en el campus. Si el alumno no realiza las correcciones en los plazos previstos se considerará insuficiente el recuperatorio y se calificará como insuficiente el parcial que estaba recuperando.

Aspectos comunes a parciales y recuperatorios.

Los exámenes parciales y recuperatorios incluirán diversas técnicas para poder observar desde distintos ángulos el objeto de evaluación. Los mismos ponderan la capacidad resolutoria del alumno para solucionar los problemas planteados, utilizando las herramientas que incorporó a lo largo de la carrera.

Trabajos Prácticos.

a. Trabajo práctico final. Consta de la programación de un sistema y se divide en tres etapas.

La primera etapa aborda la elaboración de un documento visión donde se comprenda claramente como el sistema colabora en facilitar, mejorar y potenciar los procesos administrativos y/o productivos de una organización. El alcance del sistema se expresará a través de las gestiones que se plantean y estas a su vez se dividirán en los módulos técnicos que conformarán el desarrollo. También se elaborará versiones preliminares de las especificaciones de casos de uso del sistema.

En la segunda etapa se trabaja sobre el refinamiento de las especificaciones de casos de uso, el diagrama de clases y los diagramas de secuencia, como resultado de las diversas iteraciones. El objetivo es poder observar cómo serán las estructuras constructivas y los procesos fundamentales que generarán la información solicitada en los requerimientos. En esta etapa también se obtienen los prototipos y versiones primarias de los módulos

programados.

La tercera etapa es la etapa final, en ella se terminará de refinar las especificaciones de la segunda etapa, obteniéndose la versión final de las especificaciones y se finalizará con la programación del sistema. Se entrega la carpeta que condensa la documentación del sistema y el sistema programado. El trabajo es individual y se deberá entregar una copia impresa y otra en soporte magnético al docente.

b. Trabajo de Indagación. Los trabajos de indagación estarán referidos a temas de la asignatura. Los mismos tendrán como objetivo profundizar los conocimientos adquiridos en la carrera en general y esta asignatura en particular.

Nota conceptual.

La nota conceptual estará sustentada en la participación del alumno en los distintos espacios de aprendizaje que disponga el campus virtual, el interés demostrado en la asignatura y el empeño puesto de manifiesto en las actividades de estudio.

Objetivos de la evaluación.

Conocer el grado de internalización de los contenidos.

Verificar los errores y cambios conceptuales en las estructuras cognitivas de los alumnos.

Garantizar la transferencia del conocimiento.

Identificar dificultades y contradicciones en el proceso de aprendizaje.

Reconocer la integración de aprendizajes significativos.

Contenidos de la evaluación.

Se tendrán en cuenta los contenidos conceptuales, procedimentales y actitudinales.

b. Evidencias

Las evidencias que darán cuenta del logro de los objetivos planteados en el curso son:

- a. Desarrollo del trabajo final de carrera.
- b. Trabajo de indagación.
- c. Rúbricas.
- d. Foros
- e. Resolución de los ejercicios integradores.
- f. Parciales

c. Requisitos de aprobación de la asignatura

Para aprobar la asignatura y acceder al examen final o coloquio el alumno deberá tener sus dos exámenes parciales y el trabajo final de carrera aprobado.

El alumno deberá tener como mínimo cuatro notas para poder promediar sin contar los recuperatorios de parciales.

Para aprobar la asignatura los estudiantes deberán cumplir los siguientes requerimientos:

1. Asistencia: Los estudiantes deben contar con una asistencia mínima del 70%.
2. Aprobar mediante alguna de estas modalidades:
 - a. Evaluación final “integradora coloquial”: accederán a este régimen de evaluación aquellos alumnos cuyo promedio de cursada se encuentre comprendido entre 6 y 10 puntos. Los alumnos podrán presentarse a la mesa examinadora en grupos de no más de tres personas. La evaluación se realizará en forma individual, debiendo demostrar el dominio de la asignatura como unidad y la capacidad de asociarla con otras materias del plan de estudio ya cursadas.
 - b. Examen final: acceden a este régimen de evaluación aquellos alumnos que han obtenido durante su cursada un promedio comprendido entre 4 y 5,99 puntos. El alumno se presentará en forma individual ante un tribunal examinador, el que interrogará sobre el programa de la asignatura, debiendo demostrar su capacidad de integrarla y relacionarla con otras asignaturas del plan de estudio.

Examen recuperatorio: En las carreras de grado, aquellos alumnos cuyo promedio sea inferior o igual a 3,99 puntos (obtenidos en las dos o tres evaluaciones parciales) y tengan una asistencia igual o mayor al 50%, deberán rendir un Examen Recuperatorio de Materia en la misma fecha que se indica para el Examen Final.

Habiendo aprobado el Examen Recuperatorio de la Materia, con nota mínima de 4 (cuatro) puntos, estarán en condiciones de acceder al Examen Final en el siguiente llamado, con la previa inscripción realizada 48 horas hábiles antes del mismo.

Para rendir Examen Recuperatorio de la Materia no es necesario presentar Permiso de Examen, pero deberá registrarse con no menos de 48 horas hábiles de anticipación. Los alumnos de la modalidad a distancia deberán registrarse con no menos 72 horas hábiles de anticipación.

El derecho a rendir Examen Recuperatorio de la Materia tendrá validez por dos años o dos presentaciones a Exámenes Recuperatorios de la Materia, lo que se produzca primero.

Objetivos de la evaluación.

Validar que el alumno alcanzó los resultados de aprendizaje con el nivel de dominio planteado para la asignatura y el grado de internalización de los contenidos.

Verificar los errores y cambios conceptuales en las estructuras cognitivas de los alumnos.

Garantizar la transferencia del conocimiento.

Reconocer la integración de aprendizajes significativos.

Contenidos de la evaluación.

Se tendrán en cuenta los contenidos conceptuales, procedimentales y actitudinales que denoten la apropiación de los resultados de aprendizaje esperados.

15. Bibliografía

Obligatoria:

- Cardacci Dario y Booch, Grady. 2013. **Orientación a Objetos. Teoría y Práctica.** Pearson Argentina. Buenos Aires, Argentina.
- Gamma, Erich; Helm, Richard; Johnson, Ralph y otros. 2003. **Patrones de diseño: elementos de software orientado a objetos reusable.** Pearson Educación. Madrid.
- Balena, Francesco. 2003. **Programación avanzada con Microsoft Visual Basic.Net.** McGraw-Hill. México.

Ampliatoria:

- Braude, Eric J. 2003. **Ingeniería de software: una perspectiva orientada a objetos.** Alfaomega. México.
- Date, C. J. 2001. **Introducción a los sistemas de bases de datos.** 7a.ed. Addison Wesley Longman México.
- Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D. 2007. **Introducción a la teoría de autómatas, lenguajes y computación.** 3a. ed. Pearson Educación. Madrid.
- Laudon, Kenneth C.; Laudon, Jane Price. 1996. **Administración de los sistemas de información: organización y tecnología.** 3a. Ed. Prentice Hall Hispanoamericana. México.
- Martin, James; Odell, James J. 1997. **Métodos orientados a objetos: conceptos fundamentales.** Prentice Hall Hispanoamericana. México.
- Martin, James; Odell, James J. 2002. **Análisis y diseño orientado a objetos.** Prentice Hall Hispanoamericana. México.
- O'Brien, James A.; Marakas, George M.; Herrero Díaz, María Jesús (Traductor); y otros. 2006. **Sistemas de información gerencial.** 7a. ed. McGraw-Hill Interamericana. México.
- Palma Méndez, José Tomás; Garrido Carrera, María del Carmen; Sánchez Figueroa, Fernando; y otros. 2003. **Programación concurrente.** Thomson Editores. Madrid.
- Pfleeger, Shari Lawrence; Ruíz de Mendarozqueta, Alvaro; Quiroga, Elvira (Traductor). 2002. **Ingeniería de software: teoría y práctica.** Pearson Education. Buenos Aires.
- Piattini, Mario; Villalba, José; Ruiz, Francisco; Bastanchury, Teresa; Polo, Macario; Martínez, Miguel Ángel; Nistal, César. 2001. 2003. **Mantenimiento del software: modelos, técnicas y métodos para la gestión del cambio.** Alfaomega.
- Piattini Velthuis, Mario G.; García Rubio, Félix O. 2003. **Calidad en el desarrollo y mantenimiento del software.** Alfaomega. México.

- Pressman, Roger S. 2005. **Ingeniería del software: un enfoque práctico**. 6a.ed. México. McGraw-Hill.
- Schach, Stephen R.; Fernández, Esther (Traductor); Guerrero, Ekaterina (Traductor) y otros. 2006. **Ingeniería de software clásica y orientada a objetos**. 6a. Ed. McGraw-Hill. México.
- Sommerville, Ian; Alfonso Galipienso, María Isabel (Traductor); Botía Martínez, Antonio (Traductor) y otros. 2005. **Ingeniería del software**. 7a. ed. Pearson Educación. Madrid.
- Weitzenfeld, Alfredo. 2005. **Ingeniería de software orientada a objetos con UML, Java e Internet**. Thomson Editores. Bogotá.