



**Facultad de Ingeniería
Comisión Académica de Posgrado**

Formulario de Aprobación Curso de Posgrado 2014

Asignatura: Profesión en ingeniería de software

(Si el nombre contiene siglas deberán ser aclaradas)

Profesor de la asignatura ¹: Dr. Ing. Diego Vallespir, Gr 3 DT, InCo, Fing
(título, nombre, grado o cargo, Instituto o Institución)

Profesor Responsable Local ¹:
(título, nombre, grado, Instituto)

Otros docentes de la Facultad: Ing. Lucía Camilloni, Gr 2 10hs, InCo
(título, nombre, grado, Instituto)

Docentes fuera de Facultad:
(título, nombre, cargo, Institución, país)

Instituto o Unidad: Instituto de computación
Departamento o Area: Ingeniería de software

¹ Agregar CV si el curso se dicta por primera vez.

(Si el profesor de la asignatura no es docente de la Facultad se deberá designar un responsable local)

Fecha de inicio y finalización: Segundo semestre
Horario y Salón: A definir

Horas Presenciales: 14
(se deberán discriminar las mismas en el ítem Metodología de enseñanza)

Nº de Créditos: 4
(de acuerdo a la definición de la UdelAR, un crédito equivale a 15 horas de dedicación del estudiante según se detalla en el ítem Metodología de la enseñanza)

Público objetivo y Cupos:
Interesados en temas de profesión en ingeniería de software.

Sin cupo.

(si corresponde, se indicará el número de plazas, mínimo y máximo y los criterios de selección. Asimismo, se adjuntará en nota aparte los fundamentos de los cupos propuestos. Si no existe indicación particular para el cupo máximo, el criterio general será el orden de inscripción en el Depto. de Posgrado, hasta completar el cupo asignado)

Objetivos:

Brindar un panorama general sobre la profesión en ingeniería de software discutiendo diversos aspectos de la misma.

Se busca que a través de la realización de este curso el estudiante pueda:

- Conocer y utilizar el cuerpo de conocimiento de la ingeniería de software (SWEBOK) así como saber de sus limitaciones.
- Conocer sobre los aspectos que definen una profesión y en particular la profesión en la ingeniería de software.

- Conocer y diferenciar distintos elementos que conforman a una profesión en ingeniería, tales como: educación profesional inicial, acreditaciones, desarrollo de habilidades, certificaciones, licenciamiento, desarrollo profesional, sociedades profesionales y código de ética. Como ejemplo se discutirán artículos que presentan cómo está hoy en día la Ingeniería de Software con respecto a cada uno de estos elementos.

Conocimientos previos exigidos: Conocimientos de ingeniería de software

Conocimientos previos recomendados:

Metodología de enseñanza:

(comprende una descripción de las horas dedicadas por el estudiante a la asignatura y su distribución en horas presenciales -de clase práctica, teórico, laboratorio, consulta, etc.- y no presenciales de trabajo personal del estudiante)

Este curso se basa en libros y artículos que los estudiantes deberán leer y comprender durante el curso. Los estudiantes deberán preparar y presentar cada una de las clases a menos de algún tema puntual. Estas clases estarán basadas en el material de lectura. La preparación y presentación de las clases se realizará por parte de grupos de estudiantes conformados para tal fin.

Todas las semanas se asignarán artículos para leer a todos los estudiantes y para presentar por parte de uno o más grupos de estudiantes. Todas las semanas se realizará una prueba parcial individual que buscará evaluar los conocimientos adquiridos de cada estudiante en la semana. Los estudiantes deberán poder leer inglés, ya que la inmensa mayoría de los artículos a estudiar durante el curso estarán en dicho lenguaje.

En la clase semanal se realiza la evaluación semanal, presentaciones por parte de los docentes y las presentaciones por parte de los grupos de estudiantes.

Cada una de las clases tendrá también un espacio de discusión de media hora de los temas presentados. Los estudiantes deberán participar en clase discutiendo cada uno de los temas.

- Horas clase (teórico): 14
- Horas clase (práctico): 0
- Horas clase (laboratorio): 0
- Horas consulta: 0
- Horas evaluación: incluidas en las 14 horas presenciales
 - Subtotal horas presenciales: 14
- Horas estudio: 31
- Horas resolución ejercicios/prácticos: 0
- Horas proyecto final/monografía: 15 (preparación de presentaciones)
 - Total de horas de dedicación del estudiante: 60

Forma de evaluación:

La evaluación consta de tres partes

- Pruebas parciales en cada clase. (Alrededor del 80-85%)

- Presentaciones de los grupos de estudiantes de temas del curso. (Alrededor del 15-20%)



Facultad de Ingeniería Comisión Académica de Posgrado

- Presentaciones de los grupos de estudiantes de temas del curso. (Alrededor del 15-20%)

- Asistencia a clase obligatoria. El estudiante debe asistir al 85% de las clases para poder exonerar el curso. Es importante la asistencia a clase para fomentar la discusión de los temas planteados.

Los porcentajes de las pruebas parciales no pueden ser establecidos de forma fija ya que pueden variar con cada dictado.

El curso se aprueba con el 60% de los puntos. Siendo obligatoria la aprobación de la presentación y el 85% de la asistencia a clase.

Temario:

1. Introducción
 - 1.1. Definición de la ingeniería de software como disciplina.
 - 1.2. Diferencias entre la ciencia de la computación y la ingeniería de software.
 - 1.3. Que son las organizaciones internacionales ACM e IEEE-CS
 - 1.4. Introducción básica a los temas del curso.Referencias: [Parnas 11] [SE2004 cap 2 pp 5-8] [CC2005 cap. 2 pp 13-15] [Shaw 09]
2. Cuerpo de conocimiento en ingeniería de software (SWEBOK).
 - 2.1. El cuerpo de conocimiento y su importancia.
 - 2.2. Áreas que conforman el cuerpo de conocimiento.
 - 2.3. Relación entre el cuerpo de conocimiento, la Educación y la Profesión en ingeniería de software.
 - 2.4. Historia del cuerpo de conocimiento.Referencias: [SWEBOKv3 pp xxxi-xxxiii] [SWEBOK v3 Introducción de cada KA] [Camilloni 2014a - Sección 2]
3. Profesión en ingeniería de software.
 - 3.1. ¿Qué define a una profesión?
 - 3.2. La ingeniería de software como una profesión.Referencias: [McConell 99] [McConell 13]
 - 3.3. Ética en ingeniería de software.Referencias: [Rogerson 02] [Ellis 09 – cap. 17] [SECEPP]
 - 3.4. Licenciamiento en ingeniería de software y otras "credenciales"Referencias: [Bagert 99] [Laplante 13] [Ellis 09 – cap. 18] [Kruchten 08]
 - 3.5. Habilidades del ingeniero de softwareReferencias: [Sedelmaier 12] [Rivera-Ibarra 10] [IEEE-CS 14 – pp 2-20]

Bibliografía:

[Bagert 99] Donald J. Bagert; Licensing Software Engineers: The Computer Society Must Continue to Lead, in COMPSAC, 1999.

[Camilloni 2014a] Lucía Camilloni, Diego Vallespir. Sobre profesión y educación en ingeniería de software, Reporte Técnico InCo/PEDECIBA (versión borrador), 2014.

[CC2005] The Association for Computing Machinery (ACM); The Association for Information Systems (AIS); The Computer Society (IEEE-CS), Computing Curricula 2005 - The Overview Report covering undergraduate degree programs in Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering., 2005.



Facultad de Ingeniería Comisión Académica de Posgrado

-
- [Ellis 09] Ellis, Heidi J.C. , Steven A. Demurjian and J. Fernando Naveda; Software Engineering: Effective Teaching and Learning Approaches and Practices, IGI Global, 2009.
- [IEEE-CS 14] The software engineering competency model (SWECOM), draft version, 2014.
- [Kruchten 08] Philippe Kruchten; Licensing Software Engineers?; IEEE Software 25(6): pp. 35-37;2008
- [Laplante 13] Phillip A. Laplante; An International Perspective on U.S. Licensure of Software Engineers, Technology and Society, vol. 32, no. 1, pp. 28-30, 2013.
- [McConnell 99] Steve McConnell and Leonard Tripp; Professional Software Engineering-Fact or Fiction?, IEEE Software, vol. 16, no. 6, pp. 13-18, Noviembre 1999.
- [McConnell 13] Steven McConnel and Leonard L. Tripp; Software Engineering Professional Practices, en Software Engineering Essentials, Volume II: The Supporting Processes, Richard Hall Thayer and Merlin Dorfman, Eds., 2013, cap. 11
- [Parnas 11] David Parnas; Software Engineering - Missing in Action: A Personal Perspective, IEEE Computer, vol. 44, no. 10, pp. 54-58, 2011.
- [Rivera-Ibarra 10] Rivera-Ibarra, J.G.; Rodriguez-Jacobo, J.; Serrano-Vargas, M.A., Competency Framework for Software Engineers, Software Engineering Education and Training (CSEE&T), 2010 23rd IEEE Conference on , vol., no., pp.33,40, 9-12 March 2010
- [Rogerson 02] Simon Rogerson; The Software Engineering Code of Ethics and Professional Practice:a case for being proactive, in *26 th Annual International Computer Software and Applications Conference*, 2002
- [SE2004] Joint Task Force on Computing Curricula, IEEE Computer Society and Association for Computing Machinery; Software engineering 2004: Curriculum guidelines for undergraduate degree programs in software engineering, Tech. Rep., 2004.
- [SECEPP] Joint Task Force on Software Engineering Ethics and Professional Practices, IEEE Computer Society and Association for Computing Machinery; Software Engineering Code of Ethics and Professional Practice;1999
- [Sedelmaier 12] Sedelmaier, Y.; Landes, D.; A research agenda for identifying and developing required competencies in software engineering, Interactive Collaborative Learning (ICL), 2012 15th International Conference on , vol., no., pp.1,5, 26-28 Sept. 2012
- [Shaw 09] M. Shaw; Continuing prospects for an engineering discipline of software, IEEE Software, vol. 26, no. 6, pp. 64-67, Nov 2009.
- [SWEBOKv3] P. Bourque and R. E. Fairley; Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE Computer Society Press, 2014.
-