



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

SAN LUIS, 13 MAR 2006

VISTO:

El Expte. D-3-1259/05.- mediante el cual el Departamento de Informática de la Facultad de Ciencias Físico-Matemáticas y Naturales eleva el anteproyecto de creación de la Carrera de Postgrado **ESPECIALIDAD EN INGENIERIA DE SOFTWARE**; y

CONSIDERANDO:

Que la propuesta de creación de la Carrera de Postgrado Especialidad en Ingeniería de Software surge de la experiencia realizada por distintos grupos de investigación y docencia del Departamento de Informática de la Facultad de Ciencias Físico-Matemáticas y Naturales de la Universidad, que han alcanzado un grado de desarrollo en la investigación, formación de recursos humanos y equipamiento, suficiente para hacer de la formación de Postgrado una actividad normal y continuada.

Que la experiencia recogida por el grupo responsable en la Maestría en Ingeniería de Software de la Facultad de Ciencias Físico-Matemáticas y Naturales de la Universidad Nacional de San Luis, acreditada y categorizada por CONEAU, y que se desarrolla desde 1997, indica que existe una demanda específica de formación de Postgrado que no se corresponde exactamente con las exigencias de la Maestría que actualmente se ofrece.

Que una Especialización es la más apropiada a la formación profesional y a las aspiraciones de reciclaje / perfeccionamiento de una parte importante de los alumnos de postgrado potenciales.

Que el Departamento de Informática dispone del "saber como" y de los recursos humanos docentes pertinentes como para desarrollar la Especialidad en esta área temática.

Que la presente propuesta se encuentra dentro del marco de las reglamentaciones vigentes de la Universidad Nacional de San Luis y de las emanadas por Ministerio de Educación, Ciencia y Tecnología.

Que la Comisión Asesora de Investigaciones de la Facultad de Ciencias Físico-Matemáticas y Naturales, actuando como Comisión de Postgrado, aprobó por unanimidad el proyecto de creación de la Carrera de Postgrado Especialidad en Ingeniería del Software.

Por ello, en su sesión del día 9 de marzo y en uso de sus atribuciones

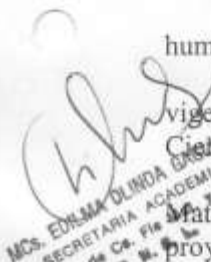
**EL CONSEJO DIRECTIVO DE LA FACULTAD DE
CIENCIAS FISICO-MATEMATICAS Y NATURALES**

ORDENA:

ARTICULO 1º.- Aprobar el Plan de Estudios de la Carrera de Postgrado Especialidad en Ingeniería del Software que, como ANEXO UNICO forma parte de la presente Ordenanza.

Cpde. ORDENANZA N° 002 ± 06


Dra. BORA AMALIA FRANTINI
VICE DECANA
Fac. de Cs. Fís. Mat. y Nat. U.N.S.L.


M.C.S. EDELMIRA D LINDA
SECRETARIA ACADEMICA
Fac. de Cs. Fís. Mat. y Nat. U.N.S.L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

ARTICULO 2°.- Elevar al Consejo Superior de la Universidad Nacional de San Luis para su homologación, según lo establecido en Art. 85, Inc. g, del Estatuto Universitario.

ARTICULO 3°.- Comuníquese, insértese en el Libro de Ordenanzas, publíquese en el Digesto de la Universidad y archívese.

ORDENANZA N° 002.-06

MCS. EDILMA OLINDA GAGLIARDI
SECRETARIA ACADEMICA
Fac. de Cs. Fís. Mat. y Nat.
U. N. S. L.

Lic. DORA AMALIA FRANZINI
VICE DECANA
Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

ANEXO UNICO

ESPECIALIDAD EN INGENIERIA DE SOFTWARE

1. Motivación de la solicitud de creación de la carrera

- 1.1. Disponibilidad del "saber como" y de los recursos humanos docentes existentes en el Departamento de Informática, tales como para encarar una carrera en este ámbito y con este alcance.
- 1.2. Experiencia recogida durante el desarrollo muy exitoso de la Maestría en Ingeniería de Software en la Universidad Nacional de San Luis. Esa experiencia indica que existe una demanda específica de formación de post grado que no se corresponde exactamente con las exigencias de la Maestría que actualmente se ofrece. Una especialización ajustará más a las aspiraciones de reciclaje / perfeccionamiento de una parte importante de los alumnos de post grado potenciales.
- 1.3. Existente posibilidad de articulación y complementación con la Maestría en Ingeniería de Software, acreditada y categorizada por CONEAU, que se está desarrollando actualmente desde 1997.
- 1.4. Disponibilidad de un estándar "de facto" internacionalmente aceptado respecto de los contenidos de carreras de este tipo (Software Engineering Institute - Carnegie Mellon University). Esta propuesta respeta dicho estándar y lo presenta adaptado al caso específico de nuestro entorno.
- 1.5. Disponibilidad de contactos y acuerdos, a nivel internacional, que permitirán un posicionamiento muy competitivo de esta nueva carrera de post grado.


Lic. DORA AMALIA FRAMINI
VICE DECANA
Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.

2. Fundamentos de la solicitud de creación de la carrera

- 2.1. Incremento de la demanda de profesionales formados específicamente para el desarrollo de software con criterios industriales.
- 2.2. Incremento de los requisitos de certificación de calidad para las empresas radicadas en nuestro país.
- 2.3. Incremento de la tendencia a instalar "software factories" internacionales en nuestro país.
- 2.4. Incremento de la tendencia a exportar software desde nuestro país. Dicho software debe responder a estándares de calidad internacionalmente aceptados.
- 2.5. Incremento de la proporción de alumnos de la Maestría en Ingeniería de Software que no provienen del ámbito académico (se desempeñan en la industria) a los que realizar la tesis para obtener el grado de Magister no les interesa así como tampoco les interesan asignaturas como Metodología de la Investigación Científica. La elaboración de una Tesis de Maestría es vista como no conducente por estos alumnos sólo interesados en los aspectos profesionales específicos de la Ingeniería de Software.


MGS. EDILMA OLIVERA
SECRETARIA ACADEMICA
Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

3. Destinatarios de la carrera de post grado "Especialidad en Ingeniería de Software"

- 3.1. Profesionales en Informática (Licenciados en Ciencias de la Computación, Licenciados en Sistemas, Ingenieros en Sistemas y títulos finales de grado similares correspondientes a carreras con planes de estudio que impliquen de cuatro o más años de duración)
- 3.2. Profesionales en Ciencias Económicas (Licenciados en Administración, Actuarios, Licenciados en Economía) que aspiren a obtener la aptitud de integrarse a equipos multidisciplinarios de desarrollo de sistemas de software.
- 3.3. Profesionales en Ingeniería (título de Ingeniero en las diversas especialidades) que deseen poseer la capacidad de integrarse a equipos multidisciplinarios de desarrollo de software.
- 3.4. Profesionales en Ciencias Exactas o Naturales que deseen desarrollar la capacidad de formar parte de equipos de desarrollo de software.
5. Profesionales universitarios, graduados en universidades de gestión pública o privada de nuestro país o del exterior, en carreras de grado de cuatro o más años de duración, que manifiesten una muy firme decisión de integrarse a equipos multidisciplinarios de desarrollo de software.

Nota: Salvo los profesionales con formación de grado específica en Informática, los graduados pertenecientes a otras disciplinas y que no demuestren los conocimientos base necesarios, deberán cursar los siguientes seminarios (20 horas cada uno) como requisito para su confirmación de su aceptación como alumnos de esta carrera de post grado:

- Seminario de paradigmas de programación (algorítmica, funcional, lógica, orientada a objetos)
- Seminario de software de base (sistemas operativos y gestores de bases de datos)
- Seminario de organización del computador y teleinformática

Estructura General de la carrera

El ciclo de estudio está organizado en cuatro cuatrimestres con la siguiente estructura y carga horaria presenciales por cada materia:

<u>Denominación del curso</u>	<u>Carga horaria</u>
Modelos de sistemas de software	80
Métodos de la Ingeniería de Software	80
Administración de proyectos de desarrollo de software	80
Análisis de artefactos de software	80
Arquitectura de sistemas de software	80
Optativa 1	80
Optativa 2	80
Trabajo de Integración Final	50
Total de la Carrera	610

Cpde. ORDENANZA N° 002-06



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Arthur, Lowell Jay, "Rapid Evolutionary Development", Wiley, 1992
- Uzal, R. et al "Rapid Evolutionary Prototyping of Data Base Applications", Software Engineering IASTED Software Engineering Conference, Las Vegas, 1998.
- Connell, J. and L. Shafer, "Object-Oriented Rapid Prototyping". Yourdon Press / Prentice Hall, 1995.
- Rational Software, "SQA Suite." <http://www.rational.com/products/sqa/prodinfo/index.jtmpl>
- Rational Software, "ClearCase." <http://www.rational.com/products/clearcase/prodinfo/index.jtmpl>
- Rational Software, "Performance Studio." <http://www.rational.com/products/pstudio/prodinfo/index.jtmpl>
- Peterson J., Petri Net Theory and the Modelling of Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
- Ghezzi, Mandrioli, Morascas and Pezze, A general way to put time in Petri nets. Proceeding 5th International Workshop on Software Specification and Design, Pittsburgh, May 89, pages 60-67, IEEE Computer Society Press, 1989.
- Brian Hambling, Managing Software Quality, Mc Graw-Hill, 1996.
- M. Zuberek, Timed Petri nets: Definitions, Properties, and applications. Microelectronics and Reliability, 31(4):627-644, 1991.
- Ghezzi, Mandrioli, Morascas and Pezze, A unified high-level Petri Net model for time critical system. IEEE Transactions on Software Ingeniering 17(2), 160-172. 1991.
- Página Web del Instituto de Ingeniería de Software (CMU), <http://www.cmu.edu>,
- Página Web del Dr. Roger Pressman, <http://www.rspa.com>

5.1.2. Métodos de la Ingeniería de Software

5.1.2.1. Contenidos conceptuales mínimos

Evolución de las metodologías de desarrollo de sistemas de software. La etapa de los métodos estructurados, la etapa de la ingeniería de la información, la etapa de la utilización del paradigma de la orientación a objetos para el análisis, diseño e implantación de sistemas de software. Estudio comparativo de las fortalezas y debilidades de métodos correspondientes a cada una de estas etapas mediante un enfoque de casos. Estudio específico de especificaciones de tiempo real. Metodologías Ágiles. XP. Crystal. Scrum. DSDM. UP. UML. Métodos y Modelado específicos para la Web.

5.1.2.2. Carga horaria: 80 horas

5.1.2.3. Tipo de evaluación

5.1.2.3.1. Desarrollo de casos prácticos correspondientes a cada tipo de metodología

5.1.2.3.2. Comprobación formal mediante test de evaluación



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.1.2.4. Bibliografía

- Apuntes de la Cátedra
- Conallen J, 1999, Building Web Applications with UML, Addison-Wesley Pub Co; ISBN: 0201615770
- Fenton, N.E., Pfleeger, S.L., 1997, Software Metrics: a Rigorous and Practical Approach, 2nd Ed., PWS Publishing Company
- Dujmovic, J.J., 1996, "A Method for Evaluation and Selection of Complex Hardware and Software Systems", The 22nd International Conference for the Resource Management and Performance Evaluation of Enterprise Computing Systems. CMG 96 Proceedings, Vol. 1, pp.368-378.
- G. Rossi, D. Schwabe and F. Lyardet, "Integrating Patterns into the Hypermedia Development Process". The New review of Hypermedia and Multimedia, Taylor Graham, 1999
- <http://sern.cpsc.ucalgary.ca/~maurer/ICSE99WS/ICSE99WS.html>
- D. Schwabe, G. Rossi, L. Emerald, F. Lyardet, "Engineering Web applications for reuse", IEEE Multimedia, Spring 2001.
- Springer. Int'l Conference on Web Engineering 2003 (ICWE'03), Oviedo, Spain.
- Rosenfeld, L., Morville, P., 1998, Information Architecture for the World Wide Web, O'Reilly.
- Rossi, G.; Schwabe, D.; Lyardet, F., 1999, Improving Web Information Systems with Navigational Patterns; Proceed. of WWW8 Conference, Toronto, Canada.
- Zuse, H., 1998, A Framework of Software Measurement, Walter de Gruyter, Berlin-NY.
- J.J. Dujmovic, CSC841 Reader: Computer Performance Evaluation. SFSU, 1996.
- E.D. Lazowska et al., Quantitative System Performance. Prentice-Hall, 1984.
- L. Lipsky, Queuing Theory. Maxwell Macmillan, 1992.
- Pressman, Roger S., "Ingeniería del Software", Ed. Mc-Graw Hill - 3ra Edición. 1993.
- Booch, Rumbaugh, Jacobson, "El Proceso de Desarrollo de Software Unificado", Addison-Wesley, 1999.
- Booch, Rumbaugh, Jacobson, "El Lenguaje de Modelado Unificado", Add-Wesley, 1999.
- Booch, Grady, "Object-Oriented analysis and design with applications", The Benjamin/Cummings Publishing Company Inc. 1994.
- Booch, Rumbaugh, Jacobson, et al., "UML Semantics", <http://www.omg.gov>
- Booch, Rumbaugh, Jacobson, "Object Constraint Language Specification", <http://www.omg.gov>
- Booch, Rumbaugh, Jacobson, "UML Notation Guide", <http://www.omg.gov>
- Craig Larman, "UML y Patronos: Introducción al análisis y diseño orientado a objetos". Prentice Hall, 1999.
- B. Thuraisingham, "Web Data Management and Electronic Commerce", CRC Press, 2000
- A. Scharl, "Evolutionary Web Development", Springer 2000
- B. Furht, "Handbook of Internet Computing", CRC 2000

Dra. DORA AMALIA FRANZINI
VICE DECANA
Fac. de Cs. Fís. Mat. y Nat. U.N.S.L.

M.Cs. EDILMA BLINDA GABRIEL
SECRETARIA ACADÉMICA
Fac. de Cs. Fís. Mat. y Nat. U.N.S.L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Alistair Cockburn, "Characterizing People as Non-Linear, First-Order Components in Software Development",
<http://alistair.cockburn.us/crystal/articles/cpanfocisd/characterizingpeopleasnonlinear.html>
- Norman L. Kerth, <http://c2.com/ppr/about/author/norm.html>
- Alianza Ágil, <http://www.agilealliance.org> - Patricio Letelier, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, letelier@dsic.upv.es
- Manifiesto para el Desarrollo de Software Ágil, <http://www.agilemanifesto.org> - Martín Fowler, La Nueva Metodología, <http://www.programacion.net>
- Kent Beck, "Extreme Programming Explained",
<http://www.amazon.com/exec/obidos/ASIN/0201616416/programacione-20>
- Ron Jeffries <http://www.xProgramming.com> - Don Wells
<http://www.extremeProgramming.org>
- Ward Cunningham, página wiki, <http://c2.com/cgi/wiki?ExtremeProgrammingRoadmap>
- Grupo de Discusión XP, <http://www.egroups.com/group/extremeprogramming>
- Mark Paulk, "XP desde la Perspectiva del CMM", <http://www.sei.cmu.edu/cmm/papers/xp-cmm-paper.pdf>
- Alistair Cockburn, <http://members.aol.com/acockburn>
- Crystal, <http://members.aol.com/acockburn/> - Raymond "La Catedral y el Bazar",
<http://es.tldp.org/Otros/catedral-bazar/cathedral-es-paper-00.html>
- Karl Fogel, "Desarrollo de Código Abierto con CVS", Coriolis Group Books; 1st edition, 1999
- Fundación Sidar, <http://www.sidar.org/recur/desdi/cvs>
- Jim Highsmith, <http://www.adaptivesd.com>
- Jim Highsmith, "Adaptive Software Development", Dorset House Publishing Company, Incorporated, 1999
- Stephen R Palmer, John M. Felsing, "A Practical Guide to Feature-Driven Development (The Coad Series)", Prentice Hall PTR, 2002
- Jeff De Luca, FDD, <http://www.featuredrivendevelopment.com>
- Peter Coad, "Java Modeling In Color With UML", Prentice Hall, 1999
- Alistair, Desarrollo de Software Ágil, Addison-Wesley Professional; 1st edition, 2001
- Ken Schwaber y Mike Beedle, "Agile Software Development with Scrum", Prentice Hall; 1st edition, 2001
- Ken Schwaber, <http://www.controlchaos.com> - Jeff Sutherland,
<http://jeffsutherland.com/scrum/index.html>
- Scrum, <http://groups.yahoo.com/group/scrumdevelopment>.
- TogetherSoft, <http://www.togethersoft.com>
- DSDM, <http://www.dsdm.org>
- Jennifer Stapleton, Peter Constable, "DSDM", Addison-Wesley Professional; 1st edition, 1997
- Carlos Reynoso, "Métodos Heterodoxos en Desarrollo de Software",
http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/heterodox.asp



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Jim Highsmith, "Agile Software Development Ecosystems", Boston, Addison Wesley, 2002.
- Scott Ambler, "Agile Modeling: Effective practices for Extreme Programming and the Unified Process", John Wiley & Sons, 2002.
- Scott Ambler, "Agile Modeling and the Unified Process",
<http://www.agilemodeling.com/essays/agileModelingRUP.htm>, 2002.
- Pekka Abrahamsson, Outi Salo, Jussi Ronkainen & Juhani Warsta, "Agile Software Development Methods: Review and Analysis", VTT, 2002
<http://www.biblioteca.secyt.gov.ar>. Journal "Environmental Modelling and Software"
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book Systems Analysis and Design
<http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Software and Systems Modeling
- <http://www.springerlink.com/>. International Journal on Software Tools for Technology Transfer (STTT)
- <http://www.springerlink.com/>. Innovations in Systems and Software Engineering
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.1.3. Administración de proyectos de desarrollo de software

5.1.3.1. Contenidos conceptuales mínimos

Recursos requeridos en un proyecto de software: Su administración específica a lo largo del ciclo de vida del proyecto. Conformación y liderazgo de equipos de desarrollo de software. El concepto de Proceso de Software: Su aplicación. Modelos de ciclo de vida de productos de software. Estudio comparativo de las fortalezas y debilidades de cada modelo de ciclo de vida. Programación y presupuesto de proyectos de software. Administración de riesgos. Gestión de las configuraciones. Aseguramiento de la calidad en proyectos de software. Definición de las áreas clave de proceso asociadas al Capability Maturity Model Integration y sus correspondientes prácticas y componentes tecnológicas. Relación del producto de software con el producto integrado con especial énfasis en el software embebido. Aspectos legales del desarrollo de software: copyright, patentes, etc. La documentación de un proyecto de software: Estándares vigentes. La ingeniería de requerimientos a lo largo de un proyecto.

5.1.3.2. Carga horaria: 80 horas

5.1.3.3. Tipo de evaluación

5.1.3.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.1.3.3.2. Caso práctico integrador

5.1.3.3.3. Test de evaluación formal

5.1.3.4. Bibliografía

- Apuntes de la Cátedra
- Marakas, George, "Decision Support Systems", Prentice Hall, 1999
- Rivas, Alonso, "Auditoría Informática", Díaz de Santos, 1997



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Sage, Andrew, "Systems Management for Information Technology and Software Engineering (Wiley Series in Systems Engineering)", John Wiley, 1995
- Reifer, Donald, "Software Management", IEEE, 1997
- Jenner, Michael, "Software Quality Management", Wiley, 1995
- Ginac, Frank, "Customer Oriented Software Quality Assurance", Prentice Hall, 1998
- Martin, James, "Information Engineering", Prentice Hall, 1990
- Dujmovic J., Informes de Auditoría del Programa 41 del banco Mundial, 1999
- Debnath, N., Uzal, R. et al "Software Quality Assurance in a Reengineering Project Based on Rapid Evolutionary Prototyping Methodology", 2000
- Arthur, Lowell Jay, "Rapid Evolutionary Development", Wiley, 1992
- Dujmovic, J., Uzal, R. et al "Software Quality Assurance issues in a project based on Rapid Evolutionary Prototyping Methodology, SFSU web page, 1999
- Uzal, R. et al "Rapid Evolutionary Prototyping of Data Base Applications", Software Engineering IASTED Software Engineering Conference, Las Vegas, 1998.
- Connell, J. and L. Shafer, "Object-Oriented Rapid Prototyping". Yourdon Press / Prentice Hall, 1995.
- Página Web del Instituto de Ingeniería de Software (CMU) <http://www.cmu.edu>
- James Womack, Daniel Jones y Daniel Roos, "The machine that changed the world: The story of Lean Production", HarperBusiness, 1991.
- <http://www.biblioteca.secyt.gov.ar>. Journal "Advances in Engineering Software"
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Data Communications, the Internet, and Electronic Commerce
- <http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Book. Encyclopedia of Information Systems
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.1.4. Análisis de artefactos de software

5.1.4.1. Contenidos conceptuales mínimos

Análisis de fortalezas y debilidades de productos utilizados en el desarrollo de software.

Herramientas para la definición de especificaciones, herramientas para el diseño de productos de software, herramientas de generación de código, lenguajes y ambientes de programación correspondientes a los diversos paradigmas (lógico, algorítmico, orientado a objetos y funcional). Estudio comparativo de herramientas de testing, de ingeniería reversa y de mantenimiento de productos de software. Uso intenso del concepto de "slicing" de programas en el aseguramiento de calidad, en el testing y en el mantenimiento de productos de software. Estudio detallado del lenguaje funcional Haskell y de sus posibilidades en el ámbito de la especificación, implantación y verificación.

5.1.4.2. Carga horaria: 80 horas



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.1.4.3. Tipo de evaluación

5.1.4.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.1.4.3.2. Evaluación formal mediante test de comprobación.

5.1.4.4. Bibliografía

- Apuntes de la Cátedra

- Paul Jorgensen, "Software Testing- A Craftsman's approach", CRC Press, 1995

- B. Marick, "The Craft of Software Testing - Object based and Object oriented Testing",
Prentice Hall, 1995.

- Dobnath, N., Uzal, R. et al "Software Quality Assurance in a Reengineering Project Based on
Rapid Evolutionary Prototyping Methodology", submitted paper, 2000

- Dujmovic, J., Uzal, R. et al "Software Quality Assurance issues in a project based on Rapid
Evolutionary Prototyping Methodology, SFSU web page, 1999

- ISO/IEC 9126-1991 International Standard, "Information technology – Software product
evaluation – Quality characteristics and guidelines for their use".

- ISO/IEC 14598-5:1998 International Standard, "Information technology -- Software product
evaluation" -- Part 5: Process for evaluators

- R. Jain, The Art of computer systems Performance Analysis. J. Wiley, 1991.

- J.J. Dujmovic, CSc841 Reader: Computer Performance Evaluation. SFSU, 1996.

- A.O. Allen, Probability, Statistics, and Queuing Theory with Computer Science
Applications. Second Edition, Academic Press, 1990.

- A.O. Allen, Introduction to Computer Performance Analysis with Mathematica. Academic
Press, 1994.

- T. Browning, Capacity Planning for Computer Systems. Academic Press, 1995.

- R. Grace, The Benchmark Book. Prentice Hall, 1996. M.K. Molloy, Fundamentals of
Performance Modeling. Macmillan, 1989.

- E.D. Lazowska et al., Quantitative System Performance. Prentice-Hall, 1984.

- L. Lipsky, Queuing Theory. Maxwell Macmillan, 1992.

- S. Trivedi, Probability and Statistics with Reliability, Queuing, and Computer Science
Applications. Prentice-Hall, 1982.

- D. Ferrari et al., Measurement and Tuning of Computer Systems. Prentice-Hall, 1983.

- Cem Kaner, James Bach, Bret Pettichord, "Lessons Learned in Software Testing", Wiley;
1st edition, 2001

- Brian Marick, Testing Foundations, <http://www.testing.com>

- Brian Hambling, Managing Software Quality, Mc Graw-Hill, 1996.

- Bret Pettichord, <http://pettichord.com>

- <http://www.biblioteca.secyt.gov.ar>. Journal "Environmental Modelling and Software"

- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book SysAnalysis and Design

- <http://www.springerlink.com/>. Annals of Software Engineering

- <http://www.springerlink.com/>. Requirements Engineering

- <http://www.springerlink.com/>. Software and Systems Modeling

- <http://www.springerlink.com/>. International Journal on Software Tools for Technology
Transfer (STTT)

- <http://www.iecc.org/>. Journals and Conferences in IEEE Xplore.

Handwritten signature and stamp: AMALIA FRANZINI, VICE DECANA, Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.

Handwritten signature and stamp: M.Cs. EDILMA OLINDA BAGUERI, SECRETARIA ACADEMICA, Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.1.5. Arquitectura de sistemas de software

5.1.5.1. Contenidos conceptuales mínimos

Los proyectos de software de alta complejidad: El nivel de abstracción de la arquitectura. El diseño de la arquitectura de productos de software de alta complejidad. Modelos y métodos formales orientados a la arquitectura. Herramientas de generación de instancias específicas de la arquitectura. Evaluación de la arquitectura de sistemas complejos existentes. Estilos de arquitectura de sistemas de software de mayor difusión. El rol del dominio del espacio de problema en la especialización de una arquitectura general a una familia de aplicaciones en particular

5.1.5.2. Carga horaria: 80 horas

5.1.5.3. Tipo de evaluación

- 5.1.5.3.1. Desarrollo de casos prácticos correspondientes a cada tópico
- 5.1.5.3.2. Test de verificación formal de conocimientos

5.1.5.4. Bibliografía

- Apuntes de la Cátedra
- Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, Second Edition, Addison-Wesley Professional; 2 edition, 2003
- Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nord, Judith Stafford, Documenting Software Architectures: Views and Beyond, Addison-Wesley Professional; 1st edition, 2002
- Mary Shaw, David Garlan, Software Architecture: Perspectives on an Emerging Discipline, Prentice Hall; 1 edition, 1996
- Thomas Erl, Service-Oriented Architecture : A Field Guide to Integrating XML and Web Services, Prentice Hall PTR, 2004
- Luke Hohmann, Beyond Software Architecture: Creating and Sustaining Winning Solutions, Addison-Wesley Professional; 1st edition, 2003
- Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, Peter Sommerlad, Michael Stal, Pattern-Oriented Software Architecture, Volume 1: A System of Patterns, John Wiley & Sons; 1 edition, 1996
- Robert D. Austin, "Measuring and Managing Performance in Organizations", Dorset House Publishing Company, Incorporated, 1996
- Martin Fowler, Patterns of Enterprise Application Architecture, Addison-Wesley Professional; 1st edition, 2002
- Joshua Kerievsky, Refactoring to Patterns (Addison-Wesley Signature Series), Addison-Wesley Professional, 2004
- Philippe Kruchten, "The Rational Unified Process", Add-Wesl Professional; 2 edition, 2000)



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Craig Larman, "Applying UML and Patterns", Prentice Hall PTR; 1st edition, 1997
- <http://www.biblioteca.secyt.gov.ar>. Journal "Environmental Modelling and Software"
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book Theories, Methodologies, and Foundations
- <http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Programming and Computer Software
- <http://www.springerlink.com/>. Software and Systems Modeling
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.springerlink.com/>. International Journal on Software Tools for Technology Transfer (STTT)
- <http://www.springerlink.com/>. Book. Encyclopedia of Information Systems
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

Liq. BORA AMALIA FRANZINI
VICE DECANA
Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.

5.2. Ciclo de cursos optativos

Los siguientes cursos propuestos se dictarán como optativos, los cuales dos de ellos deberán ser seleccionados por los alumnos para cubrir la cantidad total de cursos necesarios según la estructura general de la carrera. Es dable destacar que el siguiente listado no es limitativo a que puedan agregarse en el futuro nuevos cursos optativos conforme al avance de la tecnología y la ciencia.

5.2.1. Sistemas de tiempo real

5.2.1.1. Contenidos conceptuales mínimos

Concepto de Tiempo Real y de Sistema de Tiempo Real. Las características de los Sistemas de Tiempo Real. Requerimientos, Análisis, Diseño e Implementación de Sistemas de Tiempo Real. Especificación y Verificación Formal de Sistemas de Tiempo Real. Modelado y Simulación de Sistemas de Tiempo Real.

5.2.1.2. Carga horaria: 80 horas

5.2.1.3. Tipo de evaluación

- 5.2.1.3.1. Desarrollo de casos prácticos correspondientes a cada tópico
- 5.2.1.3.2. Evaluación formal mediante test de comprobación.

5.2.1.4. Bibliografía

- Apuntes de la Cátedra
- Qing Li, Yao, Caroline "Real-Time Concepts for Embedded Systems", CMPBooks, 2003
- Sifaks, J., Yovine, S., "Compositional Specification of Timed Systems", In 13th Annual Symposium on Theoretical Aspects of Computer Science, pages 347-359, Grenoble, France, LNCS 1046, Springer-Verlag, February 1996

Edilma Olinda Babilonia
SECRETARIA ACADEMICA
Fac. de Cs. Fís. Mat. y Nat. U. N. S. L.



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- E. Clarke, O. Grumberg and D. Long, "Model Checking", Springer-Verlag Nato ASI Series F, Volume 152, 1996.
- Bouyssounouse, Bruno, Embedded Systems Design : The ARTIST Roadmap for Research and Development (Lecture Notes in Computer Science), Springer, 2005)
- M. Bozga, C. Daws, O. Maler, A. Olivero, S. Tripakis y S. Yovine, KRONOS: A model-checking tool for real-time systems, Computer Aided Verification (CAV 98), Lectures Notes in Computer Science 1427, Springer Verlag. 1998.
- R. Alur, C. Courcoubetis, N. Halbwachs, T. Henzinger, P. Ho, X. Nicollin, A. Olivero, J. Sifakis y S. Yovine. The Algorithmic Analysis of Hybrid Systems. Theoretical Computer Science B vol. 137. IEEE Enero 1995.
- C. Daws, A. Olivero, S. Tripakis, and S. Yovine, The tool KRONOS. In Hybrid Systems III, Verification and Control, pages 208-219. Lecture Notes in Computer Science 1066, Springer-Verlag, 1996.
- T.A. Henzinger, X. Nicollin, J. Sifakis, and S. Yovine. Symbolic model checking for real-time systems. Information and Computation, 111(2):193--244, 1994.
- J. Sifakis and S. Yovine, Compositional specification of timed systems. In 13th Annual Symposium on Theoretical Aspects of Computer Science, STACS'96, pages 347--359, Grenoble, France, Fe'vrier 1996. Lecture Notes in Computer Science 1046, Springer-Verlag.
- Peterson J., Petri Net Theory and the Modelling of Systems. Prentice Hall, Englewood Cliffs, New Jersey, 1981.
- Ghezzi, Mandrioli, Morascas and Pezze, A general way to put time in Petri nets. Proceeding 5th International Workshop on Software Specification and Design, Pittsburgh, May 89, pages 60-67, IEEE Computer Society Press, 1989.
- M. Zuberek., Timed Petri nets: Definitions, Properties, and applications. Microelectronics and Reliability, 31(4):627-644, 1991.
- Ghezzi, Mandrioli, Morascas and Pezze. A unified high-level Petri Net model for time critical system. IEEE Transactions on Software Ingeniering 17(2), 160-172. 1991.
- Buy and Sloan. Analysis of Real-Time Programs with Simple Time Petri Nets. ACM 1994.
- High-level Petri Nets – Concepts, Definitions and Graphical Notation. ISO/IEC 15909, 1997, Version 3.4.
- Eric Y. T. Juan, Jeffry J. P. Tsai, Tadao Murata, Yi Zhou, Reduction Methods for Real Time Systems Using Delay Time Petri Nets. IEEE Transaction on Software Engineering. Vol 27, No. 5, May 2001.
- Henzinger, Nicollin, Sifakis and Yovine. Symbolic model checking for real-time systems. Information and Computation, 111(2):193-244, 1994.
- Qing Li, Yao, Caroline "Real-Time Concepts for Embedded Systems", CMPBooks, 2003
- Bouyssounouse, Bruno, Embedded Systems Design : The ARTIST Roadmap for Research and Development (Lecture Notes in Computer Science), Springer, 2005
- Ian Sommerville, Ingenieria de Software (6ª Edición), Pearson Educación, 2002
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

Lic. DORA AMALIA FERRER
VICE DECANA
Fac. de Cs. Fís., Mat. y Nat. U. N. S. L.

MCS. EDILBERTO CLINDA BARRERA
SECRETARIA ACADEMICA
Fac. de Cs. Fís., Mat. y Nat. U. N. S. L.

Cpde. ORDENANZA N° 002-06



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.2.2. Sistemas financieros

5.2.2.1. Contenidos conceptuales mínimos

Evaluación Financiera de Proyectos de Inversión. El proyecto de inversión como flujo de contado ("cash flow"). Criterios de evaluación de un proyecto de inversión. El periodo de recupero. Características. El Valor Actual Neto (VAN) de un Proyecto de Inversión. Cálculo del VAN de distintos proyectos. Utilización del VAN para evaluar proyectos alternativos. La Tasa Interna de Retorno (TIR) de un proyecto. Estudio comparativo de herramientas computacionales orientadas a la evaluación de proyectos de inversión. Riesgo y rendimiento. El rendimiento esperado de una cartera. Modelo de fijación de precios de los activos financieros. Línea del Mercado de Capitales. El intercambio compensatorio riesgo – rendimiento. Cálculo de Beta. Beta de una cartera. La diversificación. Herramientas computacionales necesarias para la gestión del riesgo. Concepto de Cuadro de Mando Integral (BSC). Definición de Indicadores en el Cuadro de Mando Integral (BSC). Indicadores Financieros y no Financieros en el BSC. Ejemplos de BSC en distintos tipos de organizaciones. Selección de herramientas computacionales para implantar el concepto de BSC. Metodología de implantación del concepto de BSC.

5.2.2.2. Carga horaria: 80 horas

5.2.2.3. Tipo de evaluación

5.2.2.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.2.3.2. Evaluación formal mediante test de comprobación.

5.2.2.4. Bibliografía

- Apuntes de la Cátedra
- Brealey, Richard, Myers, Stewart "Dirección Financiera", Mc Graw Hill, 1996
- Kaplan R., Norton D. "Cuadro de Mando Integral" ("Balanced Scorecard)", Gestión 2000, 1997.
- Olve, Nils-Göran, et al "Implantando y Gestionando el Cuadro de Mando Integral", Gestión 2000, HBS Press, mayo del 2000
- Marakas, George, "Decision Support Systems", Prentice Hall, 1999
- Emery, Douglas, Finnerty, "Administración Financiera Corporativa", Prentice Hall, 2000
- Cissell, Robert; Cissel Helen, Flaspohler, David "Matemáticas Financieras", CECSA, 1998
- Emery, Douglas, Finnerty, "Administración Financiera Corporativa", Prentice Hall, 2000
- Mondino, Diana; Pendás, Eugenio "Finanzas para Empresas Competitivas" Granica, 1994
- Welsch, Glenn A. et al. *Presupuestos. Planificación y control de utilidades*. 2ª edición en español. México. Prentice-Hall Hispanoamericana. 1990.
- Van Horne, James C. *Administración Financiera*. 2ª edición en español. México. Prentice-Hall Hispanoamericana. 1993.
- Weston, J. F., Copeland, T. E. *Finanzas en administración*. (2 volúmenes). 3ª edición en español. México. McGraw-Hill. 1988.
- Brealey, Richard, Myers, Stewart "Dirección Financiera", Mc Graw Hill, 1996



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.2.3. Análisis comparativos de paradigmas de programación

5.2.3.1. Contenidos conceptuales mínimos

Características fundamentales de los paradigmas de programación algorítmico, funcional, lógico y orientado a objetos. Análisis comparativo de las fortalezas y debilidades de cada uno de los paradigmas. Desarrollo de aplicaciones con un enfoque multiparadigma. Avances en programación multiparadigma.

5.2.3.2. Carga horaria: 80 horas

5.2.3.3. Tipo de evaluación

5.2.3.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.3.3.2. Evaluación formal mediante test de comprobación.

5.2.3.4. Bibliografía

- Apuntes de la Cátedra
- A Gentle Introduction to Haskell. Version 98. Paul Hudak, John Peterson, Yale University, Joseph Fasel, Los Alamos National Laboratory . Junio, 2000. <http://haskell.org/tutorial/>
- Introducción al lenguaje Haskell. Jose E. Labra G. <http://lsi.uniovi.es/~labra>
- Why Functional Programming Matters. John Hughes, Institutionen f"or Datavetenskap.
- Programación Funcional. Jeroen Fokker. 1996. Universidad de Utrecht. Departamento de Informática
- Haskell The Craft of Functional Programming. Simon Thompson. University of Kent at Canterbury. Addison-Wesley
- Yet Another Haskell Tutorial. Copyright (c) Hal Daume III, 2002-2004. <http://www.isi.edu/~hdaume/htut/>
- Elisa Gallo, Mikel Vergara, European Software Institute, <http://www.esi.es/Berrikuntza>
- Steve McConnell, "Rapid Development", Microsoft Press; 1st edition, 1996
- Alistair Cockburn, "Sobreviviendo Proyectos Orientados a Objetos", <http://www.amazon.com/exec/obidos/ASIN/0201498340/programacione-20>
- Neil Harrison, Brian Foote, Hans Rohnert "Pattern Languages of Program Design 4 (Software Patterns Series) ", Addison Wesley Publishing Company; 1 edition, 1999
- Robert Martin, <http://www.objectmentor.com/resources/articles/RUPvsXP.pdf>
- Robert Charette, The Foundation Series on Risk Management, Volume II: "Foundations of Lean Development", Cutter Consortium, Arlington, 2001
- Mary Poppendieck, "Lean Programming", <http://www.agilealliance.org/articles/LeanProgramming>, 2001.
- Andrew Hunt, David Thomas, "The Pragmatic Programmer: From Journeyman to Master", <http://allconsuming.net>
- James Bach, Satisfice, Inc. <http://www.satisfice.com>
- Cem Kaner J.D., Ph.D., <http://www.kaner.com/>
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Data Communications, the Internet, and Electronic Commerce



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book Theories, Methodologies, and Foundations
- <http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Programming and Computer Software
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.2.4. Tópicos avanzados de especificación y verificación formal de software

5.2.4.1. Contenidos conceptuales mínimos

Métodos Formales. Pro y contra de los Métodos Formales. Estilos de especificaciones
Estilos de desarrollo. Grados de rigor. Especificaciones formales. Especificaciones orientadas a la propiedad. Especificaciones orientadas al modelo. Estilos: aplicativo, imperativo, concurrente. Funciones totales y parciales. Definiciones explícitas e implícitas. Operadores. Módulos. Esquemas. Objetos. Combinación de Técnicas Formales y semi-formales. Captura de requerimientos. Semántica formal en RSL de diagramas de clases en UML. Uso de UML2RSL en un caso práctico.

5.2.4.2. Carga horaria: 80 horas

5.2.4.3. Tipo de evaluación

5.2.4.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.4.3.2. Evaluación formal mediante test de comprobación.

5.2.4.4. Bibliografía

Apuntes de la Cátedra

The RAISE Language Group. "The RAISE Specification Language". Prentice-Hall International, 1992.

The RAISE Method Group. "The RAISE Development Method". Prentice-Hall International, 1995.

- Dasso, A., Funes, A., "Formal Methods in Software Engineering" en Encyclopedia of Information Science and Technology, Idea Group, Enero de 2005.
- Funes, A., George, C., "Formalizing UML class diagrams", "capítulo 8 de UML and the Unified Process, Idea Group, 2003.
- Luqi & Goguen, J., "Formal methods: Promises and problems", IEEE Software, 14(1), págs. 73-85., Ene-Feb 1997.
- Bowen, J., & Hinchey, M., "Seven more myths of formal methods. IEEE Software, 12(3). Julio, 1995.
- Bowen, J., & Hinchey, M., "Ten commandments of formal methods. IEEE Computer. Abril, 1995.
- Bibliografía complementaria descargable desde el sitio del UNU/IIST:
<http://www.iist.unu.edu/raise/>



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- K. Lano. "The B Language and Method : A Guide to Practical Formal Development" Springer. 1996
- Dines Bjorner (Editor), VDM'87: Vdm-A Formal Method at Work: Vdm-Europe Symposium 1987 (Lecture Notes in Computer Science, 252), Springer Verlag, 1987
- A. T. Nakagawa, T. Tamai, A.T. Nakagawa, CAFE: An Industrial-Strength Algebraic Formal Method by Kokichi Futatsugi (Editor), Elsevier Science, 2000
- Deri Sheppard, An Introduction to Formal Specification With Z and Vdm (The McGraw-Hill International Series in Software Engineering), McGraw Hill, 1994.
- K. Lano, The B Language and Method: A Guide to Practical Formal Development (Formal Approaches to Computing and Information Technology), Springer Verlag, 1996
- E. Sekerinski (Editor), K. Sere (Editor), Program Development by Refinement: Case Studies Using the B Method (Formal Approaches to Computing and Information Technology), Springer Verlag, 1999
- Cliff B. Jones, Systematic Software Development Using VDM, Prentice Hall, 2nd Ed. Edition, 1990
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book Theories, Methodologies, and Foundations
- <http://www.springerlink.com/>. Programming and Computer Software
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.springerlink.com/>. Formal Methods in System Design
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.2.5. Ingeniería Web

5.2.5.1. Contenidos conceptuales mínimos

Evolución del diseño en entorno web. Aplicación de los principios y criterios de la Ingeniería de Software en entorno web. Naturaleza específica de las aplicaciones web. Características específicas del ciclo de vida del software en entorno web. Análisis, Requerimientos, Especificaciones y Diseño en entorno web. Implementación de aplicaciones web. Testing de aplicaciones web. Mantenimiento de aplicaciones web.

5.2.5.2. Carga horaria: 80 horas

5.2.5.3. Tipo de evaluación

5.2.5.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.5.3.2. Evaluación formal mediante test de comprobación.

5.2.5.4. Bibliografía

- Apuntes de la Cátedra
- Powell, Thomas, "Web Site Engineering", Prentice Hall, 1999
- Suh, Woojong, "Web Engineering: Principles and Techniques", ISBN 1-59140-432-0, Idea Group, 2005



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- Nielsen, J., 1995-2002, La columna Alertbox (contiene columnas sobre evaluación heurística, entre otras) <http://www.useit.com/alertbox/>
- Nielsen, J., 2000, Designing Web Usability: The Practice of Simplicity, New Riders Publishing; ISBN: 156205810X
- ISO/IEC 14598-5:1998 Information technology -- Software product evaluation -- Part 5: Process for evaluators
- ISO/IEC 9126-1: 2001 International Standard, "Software Engineering – Product Quality – Part 1: Quality model".
- Olsina, L., Godoy, D; Lafuente, G.J; Rossi, G.; 1999, Assessing the Quality of Academic Websites: a Case Study, New Review of Hypermedia and Multimedia (NRHM) Journal, Taylor Graham Publishers, UK/USA Vol 5. ISSN: 1361-4568
- Olsina, L.; Lafuente, G.J; Rossi, G.; 2000, E-commerce Site Evaluation: a Case Study. LNCS of Springer-Verlag; 1st International Conference on Electronic Commerce and Web Technology, London Greenwich, UK.
- Olsina, L.; 2000, Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web. Tesis Doctoral, UNLP, Defendida en Abr. 2000.
- Olsina, L.; Lafuente, G. Pastor, O., 2002, Towards a Reusable Repository for Web Metrics, Proceed. III Workshop on Web Engineering, ICSE, Orlando, US.
- Olsina, L.; Rossi, G, 2002, Measuring Web Application Quality with WebQEM, In IEEE Multimedia Magazine, Vol. 9, N° 4.
- Powell, T.; Jones, D; Cutts, D., 1998, "Web Site Engineering: Beyond Web Page Design", Prentice Hall PTR.
- Rosenfeld, L., Morville, P., 1998, Information Architecture for the World Wide Web, O'Reilly.
- WWW Consortium, 2001, "WAI Accessibility Guidelines: Page Authoring", W3C Working Draft, "WAI Accessibility Guidelines: Page Authoring", <http://www.w3c.org/TR/WD-WAI-PAGEAUTH/>.
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Data Communications, the internet, and Electronic Commerce
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.2.6. Auditoría informática

5.2.6.1. Contenidos conceptuales mínimos

Control y Auditoría: Responsabilidades. La Auditoría como verificación de la eficacia de los mecanismos de Control Interno. Auditoría Interna y Externa. La Auditoría de Estados Contables. La Auditoría Informática: Ámbito de Incumbencia. Capítulos de la Auditoría Informática: Evaluación de los Controles y del Riesgo; Ajuste a los Procedimientos Establecidos, Auditoría de la Seguridad y de la Confidencialidad, Auditoría Informática como soporte a la Auditoría Contable - Financiera; Auditoría de Fraudes (forense). Estándar de la "Information System Audit and Control Association" (ISACA): COBIT. El "Institute of Internal Auditors". El estándar de la "Association of Certified Fraud Examiners" (ACFE). El "American Institute of Certified Public Accountants (AICPA): La certificación CIPT. Normas generales de Auditoría y Marco Legal vigentes en nuestro país. Normas específicas del BCRA.

Cpde. ORDENANZA N° 002-06



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.2.6.2. Carga horaria: 80 horas

5.2.6.3. Tipo de evaluación

5.2.6.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.6.3.2. Evaluación formal mediante test de comprobación.

5.2.6.4. Bibliografía

- Apuntes de la Cátedra
- Hunton, James "et al", "Core Concepts of Information Tecnology Auditing", Wiley, 2004
- Piattini, Mario, "Auditoría Informática: Un enfoque práctico", ISBN 84-7897-444-X, Díaz de Santos, 2001
- Capability Maturity Model Integration <http://www.sei.cmu.edu/cmml/>
- Estándar COBIT de Control y de Auditoría <http://www.isaca.org/cobit>
- Estándares de Calidad específicos (casos de estudio) <http://www.hipaa.org;>
<http://www.hl7.org>
- El estándar ISO 900x.3 <http://www.quality.org/ISO9000>
- Dujmovic J., Informes de Auditoría del Programa 41 del banco Mundial, 1999
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elseiver. Book Theories, Methodologies, and Foundations
<http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.2.7. Temas avanzados de calidad del software

5.2.7.1. Contenidos conceptuales mínimos

Control de Calidad vs. Aseguramiento de Calidad en un Proyecto de Software. Gestión de la Calidad en las diversas fases del Ciclo de Vida de un Proyecto. Herramientas automatizadas de Aseguramiento de la Calidad durante un Proyecto de Software. Aspectos conceptuales e instrumentales del estándar ISO 900.x.3. Estructura, ámbito de aplicación, fortalezas y debilidades del CMMI. Ámbito de aplicabilidad del estándar HL7. Su evolución. Estudio del estándar HIPAA. Gerenciamiento de la Calidad: Calidad como un proceso de mejora continua: Aplicación de este concepto en el ámbito de la Ingeniería de Software y al de la Tecnología Informática. Gerenciamiento del proceso de certificación correspondiente a los estándares de Calidad más difundidos.

5.2.7.2. Carga horaria: 80 horas

5.2.7.3. Tipo de evaluación

5.2.7.3.1. Desarrollo de casos prácticos correspondientes a cada tópico

5.2.7.3.2. Evaluación formal mediante test de comprobación.

Cpde. ORDENANZA Nº 002-06



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

5.2.7.4. Bibliografía

- Apuntes de la Cátedra
- Evans, Isabel, "Achieving Software Quality through Teamwork", Artech House, 2004
- Lewis, Robert, "Independent Verification and Validation: A life cycle Engineering Process for Quality Software", Wiley, 1992
- Capability Maturity Model Integration <http://www.sei.cmu.edu/cmmi/>
- Estándares de Calidad específicos (casos de estudio) <http://www.hipaa.org>;
<http://www.hl7.org>
- El estándar ISO 900x.3 <http://www.quality.org/ISO9000>
- <http://www.biblioteca.secyt.gov.ar>. Book. Editorial Elsevier. Book Theories, Methodologies, and Foundations
- <http://www.springerlink.com/>. Annals of Software Engineering
- <http://www.springerlink.com/>. Lecture Notes in Computer Science
- <http://www.springerlink.com/>. Innovations in Systems and Software Engineering
- <http://www.ieee.org/>. Journals and Conferences in IEEE Xplore.

5.3. Proyecto de Integración Final

Consistirá en un aporte innovador, en el ámbito de la Ingeniería de Software, que implique una dedicación nunca menor a las 50 (cincuenta horas). Este trabajo podrá ser un enfoque conceptual, con una contrastación del tipo "laboratorio" o un trabajo de campo en la industria del software que claramente desplace el límite reconocido como "estado del arte" en el ámbito de problema específico del trabajo.

El trabajo deberá ser dirigido por un profesor de la Especialización o por un profesional con formación de post grado y experiencias adecuados a criterio del Director de la Carrera. La evaluación del Proyecto de Integración Final estará a cargo del director de dicho trabajo junto a dos profesores seleccionados por el Director de la Carrera entre los profesores que hayan participado en el dictado de cursos de la Carrera.

Conceptualmente el trabajo deberá presentar con claridad:

- Descripción del Problema (de lo general a lo particular y destacando su relevancia)
- Descripción detallada de la solución que se propone respecto del problema presentado
- Justificación y sustento de la consistencia y robustez de la solución propuesta

Formalmente el Trabajo Final estará organizado (esquema sugerido por el Dr Eduardo Scarano):

- I. Presentación
 - Portada
 - Prefacio
 - Reconocimiento
 - Abreviaturas



Ministerio de Educación, Ciencia y Tecnología
Universidad Nacional de San Luis
Facultad de Ciencias Físico
Matemáticas y Naturales

- II. Cuerpo
 - Introducción
 - Partes (optativas)
 - Capítulos
 - Sub capítulos
- III. Final
 - Apéndice
 - Notas
 - Referencias
 - Bibliografía
 - Figuras
 - Tablas
 - Glosario

Los Trabajos Finales, en su organización, deberán regirse según los lineamientos del "Manual de Redacción de Escritos de Investigación", Eduardo Scarano - Editorial Macchi - 2004

6. Título a ser expedido

A los alumnos que cursen y aprueben la totalidad de los cursos esenciales, dos de los cursos optativos, y aprueben el Trabajo Final Integrador, se le otorgará el título de post grado de "Especialista en Ingeniería de Software"

ORDENANZA N° 002-06
NP


MCS. EDILMA OLINDA GAGLIARDI
SECRETARIA ACADEMICA
Fau. de Cs. Fis. Mat. y Nat.
U. N. S. L.


Lto. DORA AMALIA FRANZINI
VISE-DECANA
Fau. de Cs. Fis. Mat. y Nat. U. N. S. L.