



Ministerio de Cultura y Educación  
 Universidad Nacional de San Luis  
 Facultad de Ciencias Físico Matemáticas y Naturales  
 Departamento: Informática  
 Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2018)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
METODOS FORMALES	ING. INFORM.	026/1	2- 2018	1° cuatrimestre
		08/15		

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
FUNES, ANA MARIA	Prof. Responsable	P.Asoc Exc	40 Hs
SANCHEZ, ALEJANDRO	Responsable de Práctico	JTP Semi	20 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	2 Hs	2 Hs	1 Hs	5 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
12/03/2018	23/06/2018	15	75

### IV - Fundamentación

Las computadoras son empleadas para múltiples tareas, donde la presencia de fallas puede traer aparejado severas consecuencias no sólo a nivel de pérdidas económicas sino de vidas humanas.

En las últimas décadas, las computadoras han pasado a jugar un rol esencial en el control aeroespacial, en la aeronavegación, en el control de trenes, autos, reactores nucleares y equipamiento hospitalario, por dar simplemente algunas pocas de las aplicaciones más críticas. Por esta razón es que resulta de vital importancia que los sistemas de computación que están por detrás sean altamente confiables.

Para construir software de alta calidad, libre de errores y confiable, una de las vías más importantes, sin descartar otras alternativas, es definiendo el dominio y los requerimientos a través del uso de especificaciones formales. Una de las principales ventajas de las especificaciones formales es que no son ambiguas y permiten "razonar" sobre las propiedades deseables del sistema especificado, conduciendo al desarrollo de software de calidad.

### V - Objetivos / Resultados de Aprendizaje

El objetivo de la materia es que los alumnos tengan la posibilidad de incursionar en la aplicación de los métodos formales como herramienta de producción de software de alta calidad.

## VI - Contenidos

### Unidad I: Introducción

Introducción al curso y a los Métodos Formales. El papel de los Métodos Formales en Ingeniería de Software. Pros y contras de los Métodos Formales. Un panorama de los Métodos Formales. Diversas clasificaciones de los Métodos Formales. Estilos de especificaciones. Estilos de desarrollo. Grados de rigor.

### Unidad II: Especificaciones formales

Enfoques de especificación de un sistema. Especificaciones basadas en Lógica de Primer Orden y Teoría de Conjuntos. Especificaciones aplicativas, imperativas, concurrentes. Especificaciones orientadas al modelo. Especificaciones algebraicas. Especificaciones orientadas a la propiedad. Especificaciones algebraicas en RSL.

### Unidad III: Verificación formal de sistemas de software y de sus propiedades.

Verificación formal del software. Sistema formal. Sintaxis, semántica y sistema de prueba. Relación de refinamiento. Stepwise refinement. Refinamiento de funciones. Refinamiento de datos. Relación de Refinamiento en el Método Formal RAISE.

### Unidad IV: Lightweight Formal Methods.

Métodos Formales Livianos. Alloy. Especificaciones formales y verificación en Alloy. La lógica de Alloy: átomos, relaciones, constantes, funciones predefinidas, operadores, restricciones, expresiones, declaraciones, multiplicidades. El lenguaje Alloy: módulos, signaturas, hechos, predicados, funciones, aserciones y comandos run y check. Uso del analizador.

### Unidad V: Especificación formal del comportamiento.

Concepto de Redes de Petri. Modelado del comportamiento con Redes de Petri. Análisis de una Red de Petri. Árbol de alcanzabilidad. Ecuaciones Matriciales. Algunas extensiones.

### Unidad VI: Combinación de Técnicas Formales y Semi formales.

Limitaciones de los diagramas de clase UML. El lenguaje OCL. Características. Especificaciones de invariantes y operaciones en OCL.

## VII - Plan de Trabajos Prácticos

1. En aula: Introducción a los Métodos Formales.
2. En aula: Especificaciones algebraicas en RSL.  
En laboratorio: de los temas del práctico de aula.
3. En aula: Modelado del comportamiento con Redes de Petri  
En laboratorio: de los temas del práctico de aula.
4. Análisis de Redes de Petri  
En laboratorio: de los temas del práctico de aula.
5. En aula: Especificación y Verificación en Alloy.  
En laboratorio: de los temas del práctico de aula.
6. En aula: OCL

## VIII - Regimen de Aprobación

Existen dos modalidades de aprobación de la materia:

a) Regularización más Exámen Final:

a.1) Para regularizar la materia se deberá:

- Tener como mínimo un 70% de asistencia a clases prácticas.

- Aprobar dos evaluaciones parciales o alguna de sus dos respectivas recuperaciones con una nota mínima de seis (6) sobre

diez (10).

a.2) Aprobar un examen final en los turnos establecidos. La modalidad del mismo puede ser oral o escrita.

b) Promoción sin Examen Final: Para promocionar la materia se deberá:

- Tener como mínimo un 80% de asistencia a clases prácticas y teóricas.

- Aprobar dos evaluaciones parciales o alguna de sus dos respectivas recuperaciones con una nota mínima de siete (7) sobre diez (10).

Alumnos Libres: Se admitirán exámenes de alumnos libres, en cuyo caso el alumno deberá aprobar en primera instancia un examen sobre los temas prácticos de la materia para luego pasar a un examen teórico. La nota final del examen será un promedio de ambas evaluaciones.

## **IX - Bibliografía Básica**

[1] Bowen, J., & Hinchey, M., "Seven more myths of formal methods". IEEE Software, 12(3). Julio, 1995.

[2] Bowen, J., & Hinchey, M., "Ten commandments of formal methods". IEEE Computer. Abril, 1995.

[3] Funes, A., Dasso, A. "Formal Methods Overview" en Encyclopedia of Information Science and Technology, 3ra Edición, IGI Global, 2014.

[4] George, C.; "Introduction to RAISE". Report 249 UNU/IIST.

[5] Hung Dang Van, Chris George, Tomasz Janowski, and Richard Moore (Eds). "Specification Case Studies in RAISE". FACIT series. Springer-Verlag. 2002.

[6] Jackson, Daniel; "Software Abstractions", The MIT Press, 2012.

[7] Object Management Group (OMG); Object Constraint Language OMG Available Specification Version 2.4, Feb 2014

[8] Peterson, James Lyle (1981). "Petri Net Theory and the Modeling of Systems". Prentice Hall. ISBN 0-13-661983-5.

[9] Peterson, James L. (1977). "Petri Nets". ACM Computing Surveys. 9 (3): 223–252. doi:10.1145/356698.356702

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

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

## **X - Bibliografía Complementaria**

[1] Alagar, V. & Periyasamy, K. "Specification of Software Systems". Springer-Verlag New York, Inc. 1998.

[2] Funes, A., George, C., "Formalizing UML class diagrams", capítulo 8 de UML and the Unified Process, Idea Group, 2003.

[3] Jean-Raymond Abrial. "The B-Book: Assigning Programs to Meanings". Cambridge University Press, 1996. ISBN 0-521-49619-5.

[4] Jonathan Bowen. "Formal Specification and Documentation using Z: A Case Study Approach". International Thomson Computer Press (ITCP) Thomson Publishing. ISBN 1850322309. 2003

[5] Luqi & Goguen, J., "Formal methods: Promises and problems", IEEE Software, 14(1), págs. 73-85., Ene-Feb 1997.

[6] Michael Huth and Mark Ryan. "Logic in Computer Science: Modelling and Reasoning about Systems". Cambridge University Press, 2000.

[7] Steve Schneider. "The B-Method: An Introduction". Palgrave, Cornerstones of Computing series, 2001. ISBN 0-333-79284-X.

## **XI - Resumen de Objetivos**

Incursionar en los métodos formales como herramienta de producción de software de alta calidad.

## **XII - Resumen del Programa**

Introducción a los Métodos Formales. Especificaciones formales. Verificación formal de sistemas de software y de sus propiedades. El Método RAISE. Otros estilos de especificaciones formales. Redes de Petri. Métodos Formales Livianos. Alloy. OCL.

## **XIII - Imprevistos**

**XIV - Otros**

--