



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

(Programa del año 2022)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
METODOS FORMALES	ING. INFORM.	026/1	2- 2022	1° cuatrimestre
		08/15		

II - Equipo Docente

Docente	Función	Cargo	Dedicación
FUNES, ANA MARIA	Prof. Responsable	P.Asoc Exc	40 Hs
SANCHEZ, ALEJANDRO	Prof. Colaborador	P.Adj Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
6 Hs	Hs	Hs	Hs	6 Hs

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

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
21/03/2022	24/06/2022	14	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 la verificación formal de 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 métodos formales como herramienta de producción de software de alta calidad, con especial énfasis en el uso de lenguajes de especificación formal en las etapas iniciales del desarrollo de software.

VI - Contenidos

Unidad I: Aspectos Generales de los Métodos Formales

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

Verificación formal del software. Model Checking. 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, firmas, hechos, predicados, funciones, aserciones y comandos run y check. Uso del analizador.

Unidad V: Especificación Formal del Comportamiento.

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

Unidad VI: Integración de Técnicas Formales y Semi-formales.

Limitaciones de las especificaciones semiformales. Limitaciones de los diagramas de clase UML. El lenguaje OCL. Características. Especificación de Valores iniciales de atributos. Especificación de reglas de derivación (valores de atributos derivados). Especificación del cuerpo de operaciones de consulta. Especificación de invariantes. Especificación de pre y post condiciones de operaciones.

VII - Plan de Trabajos Prácticos

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

VIII - Regimen de Aprobación

Condiciones para REGULARIZAR la asignatura:

- Haber asistido, de manera presencial o virtual según corresponda a la situación epidemiológica por COVID-19 del momento, al menos al 70% de las clases teóricas y prácticas de la asignatura.
- Haber aprobado dos parciales (o sus respectivas recuperaciones) con nota mayor o igual a 6 sobre 10.

Condiciones para PROMOCIONAR la asignatura:

- Haber asistido, de manera presencial o virtual según corresponda a la situación epidemiológica por COVID-19 del momento, al menos al 80% de las clases teóricas y prácticas de la asignatura.
- Haber aprobado dos parciales (o sus respectivas recuperaciones) con nota mayor o igual a 7 sobre 10.
- Haber aprobado un examen integrador con nota mayor o igual a 7 sobre 10.
- En caso de haber obtenido nota de promoción en todas las instancias de evaluación, el alumno aprobará la materia con una nota que surgirá del promedio de las mismas.

Recuperaciones:

-Se otorgarán dos recuperaciones para cada examen parcial. Se considera como válida la última nota obtenida en cada recuperación.

-El examen integrador no se recupera.

Examen Final:

En caso de regularizar la materia, el alumno deberá rendir un examen final, el cual podrá ser oral o escrito, en cualquiera de los turnos de examen establecidos.

Exámenes Libres:

Para aprobar un examen libre, el alumno deberá aprobar en una primera instancia una evaluación sobre temas prácticos de la materia con una nota de al menos 6 sobre 10. En caso de aprobar la evaluación práctica, el alumno además deberá aprobar una evaluación sobre temas teóricos con una nota de 4 sobre 10. La nota final surgirá del promedio de ambas notas.

IX - Bibliografía Básica

- [1] Funes, A., Dasso, A. "Formal Methods Overview" en Encyclopedia of Information Science and Technology, 3ra Edición, IGI Global, 2014.
- [2] George, C. Introduction to RAISE. Report 249 UNU/IIST.
- [3] Haxthausen, A. An Introduction to Formal Methods for the Development of Safety-critical Applications. Technical report, 2010.
- [4] Hung Dang Van, Chris George, Tomasz Janowski, and Richard Moore (Eds). Specification Case Studies in RAISE. FACIT series. Springer-Verlag. 2002.
- [5] Jackson, Daniel; Software Abstractions, The MIT Press, 2012.
- [6] OMG. Object Constraint Language version 2.4 (2014) - <https://www.omg.org/spec/OCL/About-OCL/>
- [7] Peterson, James Lyle (1981). "Petri Net Theory and the Modeling of Systems". Prentice Hall. ISBN 0-13-661983-5.
- [8] The RAISE Language Group. The RAISE Specification Language. Prentice-Hall International, 1992.
- [9] 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] B. Bérard, M. Bidot, A. Finkel, F. Laroussinie, A. Petit, L. Petrucci et al. System and Software Verification, Model-Checking Techniques and Tools. Springer. 1999.
- [3] Bowen, J., & Hinchey, M., "Seven more myths of formal methods". IEEE Software, 12(3). Julio, 1995.
- [4] Bowen, Jonathan. "Formal Specification and Documentation using Z: A Case Study Approach". International Thomson Computer Press (ITCP) Thomson Publishing. ISBN 1850322309. 2003
- [5] Bowen, J., & Hinchey, M., "Ten commandments of formal methods". IEEE Computer. Abril, 1995.
- [6] Funes, A., George, C., "Formalizing UML class diagrams", capítulo 8 de UML and the Unified Process, Idea Group, 2003.
- [7] Huth, Michael & Ryan, Mark . "Logic in Computer Science: Modelling and Reasoning about Systems". Cambridge University Press, 2000.
- [8] Jackson, Daniel. Software Abstractions Volume 2: Patterns of Modelling & Analysis. Draft not for distribution. 2015.
- [9] Luqi & Goguen, J., "Formal methods: Promises and problems", IEEE Software, 14(1), págs. 73-85., Ene-Feb 1997.
- [10] Warmer, Jos & Kleppe, Anneke. The Object Constraint Language, The: Precise Modeling with UML. Addison-Wesley Professional. 1999.

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

El presente programa puede presentar ajustes dada la situación epidemiológica por COVID-19. Toda modificación será acordada y comunicada con el estudiantado e informada a Secretaría Académica.

Correo de contacto: afunes@unsl.edu.ar

XIV - Otros