



Universidad Tecnológica Nacional – FRC  
DIPLOMATURA SUPERIOR en  
“TESTING DE SOFTWARE”

## 1. PROPUESTA CURRICULAR

### 1.a) OBJETIVOS

#### Que los asistentes:

- ▶ Entiendan la razón y el propósito del aseguramiento de la calidad del software.
- ▶ Comprendan los atributos del aseguramiento de la calidad, la verificación, validación y las pruebas de software.
- ▶ Diseñen y apliquen planes adecuados de prueba.
- ▶ Conozcan e implementen técnicas de estimación de testing.
- ▶ Describan los conceptos de v&V durante el ciclo de vida del software
- ▶ Describan métodos para conducir inspecciones y revisiones.
- ▶ Implementen técnicas de automatización de casos de prueba para los diversos sistemas.

#### Objetivos específicos:

- ▶ Analizar los conceptos fundamentales de pruebas de software en base al aseguramiento de la calidad del software
- ▶ Diseñar casos de pruebas, planes de pruebas y especificaciones de pruebas utilizando técnicas apropiadas.
- ▶ Planificar, especificar, ejecutar y evaluar pruebas de software.
- ▶ Utilizar herramientas para mejorar la efectividad y costo de las pruebas de software.
- ▶ Analizar los elementos críticos para la gestión del proceso de pruebas de software.
- ▶ Conducir revisiones e inspecciones eficaces y eficientes.
- ▶ Aplicar técnicas estadísticas para evaluar y predecir la densidad de defectos.
- ▶ Reconocer cuando es factible automatizar en relación costo beneficio.
- ▶ Conocer las diversas herramientas que están en el mercado para automatizar casos de prueba.

### 1.b) CONDICIONES DE ADMISION

- ▶ Profesionales universitarios de carreras tecnológicas
- ▶ Graduados de tecnicaturas superiores universitarias
- ▶ Graduados de Institutos Superiores en carreras vinculadas con las Tecnologías Informáticas.
- ▶ Tester tanto del ámbito privado como estatal

### 1.c) CARGA HORARIA TOTAL

La Diplomatura se estructura en cinco módulos y un Trabajo Final Integrador (TFI), con una carga horaria de 40 (cuarenta) horas. La carga horaria total del Programa será de 200 (doscientas) horas:

$$200 \text{ Horas} = (5 \text{ módulos} \times 40 \text{ Horas}) + \text{TFI}$$



Todos los módulos tendrán una carga presencial (en el aula y en el laboratorio) y la Obligación de realizar tareas prácticas en forma semipresencial

## 1.d) MODULOS

### **Módulo 1: Aseguramiento de la calidad del software (40 hs)**

**Docente Responsable: Ing. Sheila Cravero- Ing. Fanny Montoya**

- Calidad- Conceptos Generales
- Costos de la Mala Calidad
- Modelos de Calidad
- Aseguramiento de la calidad del software
- Revisiones de software.
- Herramientas para Revisión de Documentos
- Ejercitación
  - ▶ Calidad, Visiones, perspectivas. Costo de Calidad, Gestión de Calidad. Pilares de la calidad., Calidad de software, Mediciones., Aseguramiento de la Calidad. Principios, Características. (ISTQB). Fundamentos del testing, porque es necesario, que es el testing, principios del testing, proceso fundamental de la prueba, la psicología del testing. Modelos de Calidad- ISO- CMMI- SIX SIGMA. Certificaciones de TEsting. Carrera. Procesos, Procesos formales, Metodología Ágil. Gestión de la configuración y uso de herramientas, etc.....

### **Módulo 2: Verificación, Validación e inspección ( 40 hs )**

**Docente Responsable: Ing. Fanny Montoya**

- Conceptos de V&V durante el ciclo de vida del software.
- Conceptos de SOA testing.
- Defectos
- Métodos para conducir inspecciones
- Revisiones personales
- Pruebas de escritorios
- Inspecciones Fagan
- Herramientas para Revisión e inspección.
- Ejercitación
  - ▶ Conceptos de V&V durante el ciclo de vida del software. Niveles de prueba, Tipos de prueba, Ejercicios. Concepto y Ejemplos Casos de Prueba. Formatos de Caso de prueba, Convención de nombres, Criterios de Aceptación, Datos de Prueba, Matriz de trazabilidad de pruebas, Concepto de un plan de prueba, Resultados de la prueba, Script de pruebas, Diseño de un Caso de prueba. Ejercitación. Revisión de Casos de pruebas, Buenas Prácticas. Gestión del Defecto, Reporte de defecto. Técnicas de Revisión, Importancia del rol del tester, Funciones, responsabilidades, Conceptos de SOA testing. Métodos para conducir inspecciones, Herramientas para Revisión e inspección. Creación, mantenimiento y ejecución de Casos de prueba, reporte de avance, creación del entorno de prueba, Metodologías de específicas de testing., Identificar las condiciones de prueba, datos de prueba y objetivos de la prueba. Estrategias de Regresión y Mantenimiento, etc.....



### **Módulo 3: Técnicas de pruebas sistema ( 40 hs)**

**Docente Responsable: Ing. Sheila Cravero**

- Desarrollo de casos de pruebas basados en requerimientos
- Pruebas basadas en el análisis de cobertura y en el de estructura
- Pruebas funcionales
- Pruebas de sistema, alfa ,beta y de Aceptación
- Pruebas de Regresión.
- Herramientas para la ejecución Automatizadas de pruebas
- Herramientas para diseño y desarrollo de pruebas
- Ejercitación
  - ▶ Técnicas de Pruebas, diferenciar entre niveles, tipos y técnicas. Ejercitación
  - ▶ Mejoras de Casos de pruebas.Conceptos de automatización, Importancia del Rol tester-desarrollador, Creación de casos de pruebas automáticas, manejo de 2 herramientas diferentes.Etc....

### **Módulo 4: Técnicas de pruebas de software Unitarias e Integración ( 40 hs)**

**Docente Responsable: Ing Cecilia Luna**

- Desarrollo de casos de pruebas Unitarias.
- Pruebas Unitarias
- Manejo de excepciones
- Herramientas de Automatización.
- Pruebas de Integración
- Ejercitación
  - ▶ Concepto: Prueba Unitaria,Propiedades de las pruebas unitarias,Excusas comunes para no realizar las pruebas unitarias,Beneficios de las pruebas unitarias,Enfoques de desarrollo de pruebas unitarias Tipos de pruebas unitarias (Black Box, White Box Test, Test Harness, Mock objects)Introducción a "Visual Studio Team System Testing Framework",Estructura de las pruebas unitarias ,Pruebas unitarias controladas por datos,Manejo de objetos simulados en las pruebas unitarias con el objeto de evitar dependencias. (Mock objects) ,Pruebas unitarias de la base de datos.,Pruebas unitarias de métodos privados,Cobertura de código,Análisis estático de código,Administrar pruebas con visual studio, Presentación de las diferentes herramientas de automatización disponibles en el mercado para realizar pruebas unitarias, dependiendo del lenguaje de programación. ,Concepto: Prueba de Integracion, Beneficios de las pruebas de integración,Trabajar con proyectos de pruebas,etc...

### **Módulo 5: Administración del proceso de prueba (40 hs)**

**Docente Responsable: An./Cdor. Magris Sergio . Ing Valeria Pace**

- Estrategias de Pruebas
- Confección del plan de prueba
- Estimación
- Control del cambio
- Elementos críticos para la gestión del proceso de prueba del software
- Evaluación de las pruebas



- Enfoque organizacional
  - Herramientas para planificación de pruebas
  - Herramientas de evaluación.
- Ejercitación
- ▶ Estrategias de Pruebas, Confección del plan de prueba. Estimación, Presupuesto.
  - ▶ Control del cambio, Elementos críticos para la gestión del proceso de prueba del software, Análisis de Riesgo y Costo. Evaluación de las pruebas, Enfoque organizacional, Herramientas para planificación de pruebas, Herramientas de evaluación. Importancia del líder de prueba. Responsabilidades. Autogestión de actividades, Mejoras Continuas. Reporte de Avance, etc....

## 2. PROPUESTA ORGANIZATIVA

### 2.a) MODALIDAD DE DIRECCION ACADEMICA

Coordinador Académico: Ing. Montoya Fanny

Comité académico:

Ing. Montoya Fanny  
Ing. Cravero Sheila  
Ing. Luna Cecilia  
Ing. Pace Valeria  
Cr/An. Magris Sergio

**2.b) FECHA DE INICIO:** Lunes de abril de 2011.

**2.c) HORARIO DE CURSADO:** Lunes 18:30 a 22:30 hs / Sábados: 09:00 a 13:00 hs.

### 2.d) BIBLIOGRAFÍA

- ▶ ISTQB Syllabus-2007/2010 (Thomas Müller, Dorothy Graham, Debra Friedenber and Erik van Veendendal)
- ▶ BS 7925-2:1998. Software Component Testing.
- ▶ IEEE 610.12:1990. Standard Glossary of Software Engineering Terminology.
- ▶ IEEE 829:1998. Standard for Software Test Documentation.
- ▶ IEEE 1008:1993. Standard for Software Unit Testing.
- ▶ IEEE 1012:2004 Standard for Verification and Validation Plans
- ▶ IEEE 1028:1997. Standard for Software Reviews and Audits.
- ▶ IEEE 1044:1993. Standard Classification for Software Anomalies.
- ▶ IEEE 1219:1998. Software Maintenance.
- ▶ ISO/IEC 2382-1:1993. Data processing - Vocabulary - Part 1: Fundamental terms.
- ▶ ISO 9000:2005. Quality Management Systems – Fundamentals and Vocabulary.
- ▶ ISO/IEC 9126-1:2001. Software Engineering – Software Product Quality – Part 1: Quality characteristics and sub-characteristics.
- ▶ ISO/IEC 12207:1995. Information Technology – Software Life Cycle Processes.
- ▶ ISO/IEC 14598-1:1999. Information Technology – Software Product Evaluation - Part 1: General Overview.



- ▶ The art of software testing 2nd Edition 2004 Glenford Myers, John Wiley
- ▶ Quality is Free The art of making quality certain by Philip B. Crosby
- ▶ Handbook of Software Quality Assurance 3rd Edition by Gordon Schulmeyer & James McManus
- ▶ Quality Management for projects and programs by Lewis Ireland
- ▶ Software testing techniques- Boris Beizer.
- ▶ Software system testing and Quality Assurance –Van Nostrand Reinhold electrical/computer science and engineering series
- ▶ Black-Box Testing: Techniques for Functional Testing of Software and Systems- Boris Beizer.
- ▶ Manage Software Testing-Peter Farrell Vinay. Software Testing: Testing Across the Entire Software Development Life Cycle- Everett, Gerald D., D. / McLeod, Raymond
- ▶ Profesional Software Testing with Visual Studio 2005 Team System. Autores: Tom Arnold, Dominic Hopton, Andy Leonard y Mike Frost.
- ▶ Test-Driven Development in Microsoft .NET. Autores: James W. Newkirk and Alexei A. Vorontsov

### **Bibliografía Complementaria:**

- ▶ Libros:
  - Unit Test Frameworks. Autor: Paul Hamill
  - Pragmatic Unit Testing in C# with NUnit. Autores: Andrew Hunt y David Thomas
  - Effective Gui Test Automation. Autores: Kanglin Li & Mengqi Wu
- ▶ MSDN:
  - Trabajar con pruebas unitarias -> <http://msdn.microsoft.com/ess/library/ms182515.aspx>
  - Tutoriales -> <http://msdn.microsoft.com/es-es/library/ms182417.aspx>

### **Papers/Artículos Básicos Recomendados**

- ▶ Essential Components of an SOA Quality Foundation- Jim Murphy\_
- ▶ Agile Load Checking for SOA Quality Breaking Performance Barriers when Load Testing Service-Oriented Architectures-Mindreef
- ▶ The foundation of SOA Quality- Mindreef