



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: Mgter. Natacha Lascano- Ing. Sheila Cravero**

- 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

#### **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

#### **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



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

**Docente Responsable: Ing. Ivana Rojas**

- Desarrollo de casos de pruebas Unitarias.
- Pruebas Unitarias
- Manejo de excepciones
- Herramientas de Automatización.
- Pruebas de Integración
- Ejercitación

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

**Docente Responsable: An./Cdor. Magris Sergio**

- 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

## **2. PROPUESTA ORGANIZATIVA**

### **2.a) MODALIDAD DE DIRECCION ACADEMICA**

Coordinador Académico: Ing. Montoya Fanny

Comité académico:

- Lascano Natacha
- Montoya Fanny
- Sheila Cravero
- Ivana Rojas
- Magris Sergio

**2.b) FECHA DE INICIO:** Lunes 12 de abril de 2010.

**2.c) HORARIO DE CURSADO:** Lunes 18:30 a 22:30 hs / Sábados: 09:00 a 13:00 hs. (cada 15 días aproximadamente)

### **2.d) BIBLIOGRAFÍA**

- 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/es-es/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