



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.
- Diseñen Test charters
- Apliquen testing ágil en sus proyectos reduciendo tiempos y esfuerzos
- 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.
- Automaticen casos de pruebas básicos
- Gestionen defectos de manera temprana.

#### **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 ADMISIÓN

- 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 siete módulos y un Trabajo Final Integrador (TFI) por módulo. La carga horaria total del Programa será de 200 (doscientas) horas:

Modalidad:

SEMIPRESENCIAL: 1 clase de 4 horas por semana, 1 clase de 2 horas virtual ( con actividad planeada para 2 horas con corrección)

Uso de Aula virtual

Clases en laboratorio

### 1.d) MÓDULOS

#### **Módulo 1: Metodologías de desarrollo e Inspección de documentos (20 hs)**

- Calidad- Conceptos Generales
- Modelos de ciclo de vida de desarrollo.
- Metodologías en Cascadas
- Metodologías Agiles
- Herramientas para Revisión de Documentos
- Métodos para conducir inspecciones
- Ejercitación

#### **Módulo 2: Análisis y Diseño de Testing (36 hs)**

- Conceptos de V&V durante el ciclo de vida del software.
- Niveles de Pruebas- Generación de objetivos de pruebas por nivel
- Tipos de pruebas- Generación de objetivos y escenarios de pruebas por tipos.
- Proceso de testing
- Técnicas de testing – Caja blanca, caja negra.
- Diseños de casos de pruebas Formales de sistemas
- Diseño de caso de pruebas Formales de Aceptación

#### **Módulo 3: Gestión de Defectos y Set suite de Testing (32 hs)**

- Conceptos de Set suite, Confección de planillas de ejecución,
- Gestión de Defectos, Reporte, revisión, Monitoreos
- Herramientas de Gestion de Defectos
- Ejercitación

#### **Módulo 4: Téstring Agil (32 hs)**

- Testing Exploratorio



- Confección de test charters
- Métricas y planificación de ejecuciones.
- Técnicas basada en la Experiencia
- Ejercitación

#### **Módulo 5: Fundamentos en Testing Automatizado (20 hs)**

- Conceptos de Modelos de Automatización de pruebas
- Arquitecturas de Proyectos de Automatización
- Herramientas para Grabado y Reproducción
- Herramientas para Arquitecturas Data Driven
- .Herramientas para Arquitecturas basada en modulos
- Herramientas para Arquitecturas híbridas.

#### **Módulo 6: Testing de Accesibilidad (20 hs)**

- Conceptos de Testing de Accesibilidad
- Cuales son los principios del TA
- Estandares internacionales de TA
- Herramientas para TA
- Diseño de Test Charter para casos de TA

#### **Módulo 7: Administración del proceso de prueba (40 hs)**

- Estrategias de Pruebas
- Confección del plan de prueba Agil
- 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 DIRECCIÓN ACADÉMICA**

Coordinador Académico: Ing. Montoya Fanny

Comité académico:

- Ing. Pace Valeria
- Ing Nieto Claudia
- Ing. Montoya Fanny

### **2.b) 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
- - Professional 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