



Campus de la UNA
SAN LORENZO-PARAGUAY

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
CONSEJO DIRECTIVO

RESOLUCIÓN 25/18/162-00
ACTA 1226/25/08/2025

“POR LA CUAL SE APRUEBA EL PROGRAMA DE ESTUDIO DE LA ASIGNATURA TESTING DE SOFTWARE, DE LA CARRERA INGENIERÍA EN INFORMÁTICA – PLAN 2026, SEDE SAN LORENZO”

VISTO: El Memorando DA/1707/2025 del Director Académico de la FP-UNA, Prof. MSc. Felipe Santiago Uzabal Ecurra, con el cual remite el Memorando CCPTCC/030/2025 de la Comisión Coordinadora del Proyecto de Transformación Curricular de Carreras de Grado de la FP-UNA, en el que presenta la propuesta de Programa de Estudio de las Asignaturas de la Carrera Ingeniería en Informática.

CONSIDERANDO: La Ley 4995/2013 de Educación Superior, el Estatuto de la Universidad Nacional de Asunción y las deliberaciones sobre el tema.

Que la Comisión Coordinadora del Proyecto de Transformación Curricular de Carreras de Grado, solicita la aprobación del Programa de Estudio de la asignatura **“Testing de Software”**, de la carrera Ingeniería en Informática – Plan 2026.

Que los programas fueron elaborados conforme a las disposiciones establecidas por el Consejo Nacional de Educación Superior (CONES) en materia de **créditos académicos**, según lo dispuesto en la Resolución CONES N.º 221/2024, que regula el *Sistema de Créditos Académicos – Paraguay* y los criterios para su publicación en las carreras de grado.

**EL CONSEJO DIRECTIVO DE LA FACULTAD POLITÉCNICA
RESUELVE:**

25/18/162-01 APROBAR el Programa de Estudio de la Asignatura **“Testing de Software”**, de la carrera Ingeniería en Informática – Plan 2026, Sede San Lorenzo, detallado en el ANEXO 124 de la presente Acta.

25/18/162-02 COMUNICAR, copiar y archivar

Prof. Abg. Joel Arsenio Benítez Santacruz
Secretario



Prof. Ing. Silvia Teresa Leiva León, MSc.
Presidenta



Campus de la UNA
SAN LORENZO-PARAGUAY

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
CONSEJO DIRECTIVO

Resolución 25/18/162-00 Acta 1226/25/08/2025
ANEXO 124

DEPARTAMENTO DE ENSEÑANZA DE INFORMÁTICA
PROGRAMA DE ESTUDIO

I. IDENTIFICACIÓN

Nivel		Grado									
Asignatura		Testing de Software									
Carrera		Plan		Sede/Filial		Carácter		Semestre		Prerrequisitos	
Ingeniería en Informática		2026		Sede San Lorenzo		Electiva 1		Séptimo		Seguridad Informática, Programación de Aplicaciones 2, Bases de Datos 2, Ingeniería de Software 1.	
Semanal					Periodo						
HT	HP	HTD	HTI	HS	PA	THTD	THTI	THA	CA-PY		
2	2	4	4	8	18	72	72	144	5		

*HT: Horas Teóricas semanales.

*HP: Horas Prácticas semanales.

*HTD: Horas semanales de Trabajo académico con acompañamiento Docente.

*HTI: Horas semanales de Trabajo académico Independiente del estudiante.

*HS: Horas Semanales (HTD+HTI).

*PA: Periodo Académico en semanas.

*THTD: Total de Horas de Trabajo académico con acompañamiento Docente (HTD*PA).

*THTI: Total de Horas de Trabajo académico Independiente del estudiante (HTI*PA).

*THA: Total de Horas de trabajo Académico (THTD+THTI).

*CA-PY: Créditos académicos de la asignatura.

II. FUNDAMENTACIÓN

En el mundo actual, donde los ciclos de desarrollo de software son cada vez más cortos y las demandas de calidad más altas, las pruebas de software juegan un rol crucial para garantizar la estabilidad y funcionalidad de los sistemas. La automatización de pruebas se ha convertido en una herramienta esencial, permitiendo realizar verificaciones rápidas y repetitivas, reduciendo el tiempo y los costos asociados al testing manual. Esta asignatura está diseñada para dotar a los estudiantes de las habilidades necesarias para implementar estrategias de automatización eficientes, utilizando herramientas modernas y técnicas avanzadas que optimizan la cobertura y mejoran la calidad del software en entornos de desarrollo ágil y continuo.

En relación a la naturaleza de la asignatura, se aborda de manera teórico-práctica, se combinarán conceptos teóricos con ejercicios prácticos. La organización de la asignatura se basa en los ejes temáticos, se incluyen conceptos fundamentales como: Principios de testing, técnicas de diseño de tests, herramientas de automatización, testing basado en propiedades.

III. COMPETENCIAS DEL PERFIL DE EGRESO ASOCIADAS

1. Comunicarse en las lenguas oficiales del país y en una lengua extranjera.
2. Liderar y trabajar en equipo con eficacia y responsabilidad tomando decisiones basadas en evidencias-

- 3. Aplicar en la práctica profesional los valores humanos, la ética y los mecanismos de seguridad laboral.
- 4. Actuar proactivamente frente a los problemas sociales y ambientales.
- 5. Adaptarse respetuosamente a contextos nuevos o adversos, así como a diversidades personales, disciplinares y culturales.
- 6. Actualizarse permanentemente mediante la obtención y gestión autónoma de información de calidad, utilizando tecnología de la información y comunicación.
- 7. Evaluar el comportamiento de diversos fenómenos disciplinares e interdisciplinares relacionados con la informática con una visión de sistema, mediante modelos teóricos validados y actualizados, capaces de abarcarlos integralmente en un contexto de incertidumbre.
- 8. Seleccionar, utilizar y construir instrumentos innovadores asociados al ejercicio de la ingeniería informática.
- 9. Aplicar, producir y difundir conocimientos técnicos y científicos en el área de la ingeniería informática.
- 10. Planificar, proyectar, diseñar y ejecutar proyectos sostenibles e integrales para la resolución de problemas, la mejora y la innovación en el ámbito de la ingeniería informática.
- 11. Interpretar, modelar y comunicar información referida a la ingeniería en informática en forma gráfica.

IV. ORGANIZACIÓN DE LA ASIGNATURA

Unidades	Contenidos	Resultados de aprendizaje
1. Fundamentos del Testing y la Automatización.	1.1. Introducción a los principios básicos del testing de software. 1.2. Conceptos de automatización de pruebas: cuándo y por qué automatizar. 1.3. Tipos de pruebas: unitarias, integración, funcionales y regresión. 1.4. Diferencia entre pruebas manuales y automatizadas. 1.5. Beneficios y limitaciones de la automatización.	1. Identifica los principios clave del testing de software y los beneficios de la automatización. 2. Compara las pruebas manuales con las automatizadas para determinar su aplicación en diferentes contextos.
2. Herramientas de Automatización de Pruebas.	2.1. Introducción a las herramientas de automatización más utilizadas. 2.2. Instalación y configuración de entornos de automatización. 2.3. Primer script automatizado: estructura, ejecución y reporte.	1. Selecciona e instala herramientas de automatización adecuadas para diferentes tipos de pruebas. 2. Desarrolla y ejecuta scripts automatizados básicos utilizando herramientas.

Unidades	Contenidos	Resultados de aprendizaje
	2.4. Gestión de proyectos de pruebas automatizadas.	
3. Diseño y Estrategias de Pruebas Automatizadas	3.1. Buenas prácticas en el diseño de pruebas automatizadas. 3.2. Estrategias para maximizar la cobertura de pruebas. 3.3. Diseño de pruebas unitarias y de integración automatizadas. 3.4. Pruebas funcionales y de regresión automatizadas. 3.5. Criterios para priorizar qué pruebas automatizar.	1. Diseña casos de prueba automatizados que optimicen la cobertura y calidad del software. 2. Implementa estrategias para identificar las pruebas más adecuadas para la automatización.
4. Pruebas Basadas en Propiedades.	4.1. Introducción al concepto de Property-BasedTesting (PBT). 4.2. Diferencias entre pruebas basadas en ejemplos y en propiedades. 4.3. Uso de herramientas. 4.4. Ejemplos de pruebas automatizadas basadas en propiedades.	1. Explica la diferencia entre las pruebas tradicionales basadas en ejemplos y las pruebas basadas en propiedades. 2. Implementa pruebas automatizadas utilizando frameworks de PBT para validar el comportamiento del software.
5. Automatización de Pruebas de Rendimiento.	5.1. Introducción a las pruebas de rendimiento (performance testing). 5.2. Herramientas de automatización para pruebas de carga y estrés. 5.3. Diseño y ejecución de pruebas automatizadas para evaluar el rendimiento del sistema. 5.4. Interpretación de los resultados y ajustes del sistema basados en las pruebas.	1. Diseña e implementa pruebas automatizadas de rendimiento utilizando herramientas especializadas. 2. Analiza los resultados de pruebas de carga y estrés para optimizar el rendimiento del sistema..
6. Evaluación y Reporte de Resultados Automatizados	6.1. Métodos de análisis de los resultados de pruebas automatizadas. 6.2. Generación de reportes automáticos de resultados de pruebas. 6.3. Análisis de métricas clave: tasa de fallos, cobertura, tiempo de ejecución. 6.4. Mejora continua basada en los resultados de pruebas.	1. Interpreta los resultados de pruebas automatizadas para detectar áreas críticas en el software. 2. Genera reportes automatizados de calidad del software y planifica acciones correctivas.

V. ESTRATEGIAS DIDÁCTICAS

En el desarrollo del programa se aplicarán estrategias didácticas conducentes a la apropiación teórica y la ejecución práctica de procesos y procedimientos, a saber:

- **Aprendizaje basado en problemas:** estrategia de enseñanza donde se busca resolver un problema a través del conocimiento que adquirió en el aula, el estudiante toma liderazgo de su aprendizaje e identifica la importancia de su aprendizaje y el conocimiento.
- **Aprendizaje basado en proyectos:** metodología donde el estudiante participa activamente en su aprendizaje, desarrollando diferentes habilidades para solucionar un problema a través de un proyecto, y que pueda implementarse para la mejora del contexto.
- **Aprendizaje cooperativo colaborativo:** conjunto de métodos de instrucción para la aplicación en grupos pequeños, de entrenamiento y desarrollo de habilidades mixtas (aprendizaje, desarrollo personal y social) donde cada componente del grupo es responsable tanto de su aprendizaje como del de los restantes.
- **Debate:** exposición por parte del docente de los conceptos básicos por unidad, con materiales de lectura y ejemplos orientados a la enseñanza de las competencias específicas de la asignatura. El docente asume el rol de expositor y buscará generar el debate a través de preguntas sobre lo expuesto y desde la participación de los estudiantes.

La elección particular de la estrategia didáctica aplicada será explícita en el Planeamiento de la Asignatura, de acuerdo con el perfil de los estudiantes, los recursos disponibles y el contexto educativo.

VI. ESTRATEGIAS EVALUATIVAS

Procesos de producción grupales e individuales, pruebas individuales orales y/o escritas durante el desarrollo de las unidades con diálogos e interpretaciones que los estudiantes realicen sobre los contenidos, debates, retroalimentación en casos necesarios y actividades que amplíen el conocimiento.

Con fines de calificación y promoción se aplicará el Reglamento Académico vigente en la institución que prevé valoraciones de proceso y final.

VII. MEDIOS AUXILIARES

Aula o plataforma virtual, pizarras acrílicas, proyector, marcadores, borrador de pizarra acrílica, equipo de audio, ordenadores, wifi, celulares, plataformas de videoconferencia y foros.

VIII. BIBLIOGRAFÍA

- Foundations of Software Testing: ISTQB Certification - Dorothy Graham, Rex Black, Erik van Veenendaal, 4th Edition, 2015.
- Agile Testing: A Practical Guide for Testers and Agile Teams - Lisa Crispin, Janet Gregory, 1st Edition, 2009.
- Effective Software Testing: A Developer's Guide - Maurício Aniche, 2nd Edition, 2022.
- Continuous Testing for DevOps Professionals: A Practical Guide - Eran Kinsbruner, 1st Edition, 2020.
- Practical Test Automation: A Guide to Software Testing in the Real World - Manfred Baumgartner, Thomas Biszak, 1st Edition, 2021.

