UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
CONSEJO DIRECTIVO

Campus de la UNA
SAN LORENZO-PARAGUAY

RESOLUCIÓN 25/18/153-00
ACTA 1226/25/08/2025

"POR LA CUAL SE APRUEBA EL PROGRAMA DE ESTUDIO DE LA ASIGNATURA *SEGURIDAD EN APLICACIONES,* DE CARRERAS DE GRADO, SEDE SAN LORENZO, FILIAL VILLARRICA Y FILIAL CORONEL OVIEDO"

VISTO: El Memorando DA/1707/2025 del Director Académico de la FP-UNA, Prof. MSc. Felipe Santiago Uzabal Escurra, 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 Asignatura de las Carreras de Grado.

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 ***"Seguridad en Aplicaciones"*,** la cual es común entre Carreras de Grado.

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.° 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:

25/18/153-01 APROBAR el Programa de Estudio de la Asignatura ***"Seguridad en Aplicaciones"*,** de carreras de Grado, Sede San Lorenzo, Filial Villarrica y Filial Coronel Oviedo, detallado en el ANEXO 115 de la presente Acta.

25/18/153-02 COMUNICAR, copiar y archivar.

Prof. Abg. Joel Arsenio Benítez Santacruz
Secretario

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

TEL/FAX: +595 21 5887000 – C.C. 1130 (Asunción) – 2111 (San Lorenzo)
www.pol.una.py

DEPARTAMENTO DE ENSEÑANZA DE INFORMÁTICA
PROGRAMA DE ESTUDIO

I. IDENTIFICACIÓN

Nivel	Grado				
Asignatura	Seguridad en Aplicaciones				
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.
Licenciatura en Ciencias Informáticas	2026	Sede San Lorenzo/Filial Villarrica/Filial Coronel Oviedo	Electiva	***	Ingeniería de Software I, Base de Datos II, Seguridad Informática, Programación Frontend.

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

Uno de los principales problemas en el desarrollo de software es que la seguridad no es considerada como parte integral de su construcción. Usualmente se piensa en la seguridad del software sólo como un complemento final. Esto lleva a que las aplicaciones queden vulnerables a ataques maliciosos. Tecnologías reactivas de redes como firewalls pueden ayudar a aliviar algunos ataques, pero ellos no abordan el problema real de seguridad: la mala calidad del software. Si queremos resolver este problema de la seguridad informática, necesitamos construir software más seguro.

La asignatura abordará los conceptos asociados con la seguridad informática preferentemente desde el punto de vista de las aplicaciones. Presentará la teoría necesaria acerca de los requerimientos de un sistema seguro a través del establecimiento de políticas de seguridad y de los mecanismos de implementación de estas. Las aplicaciones web son las que han avanzado más rápidamente en la actualidad, por lo que se requiere conocer su funcionamiento y las vulnerabilidades a las que se exponen. En este sentido, se presentarán las vulnerabilidades y ataques a las que estas aplicaciones se exponen y cómo protegerlas. El desarrollo de aplicaciones seguras es otro de los temas principales abordado en este

abordado en este curso, se mostrarán y explicarán los conceptos y técnicas asociados al desarrollo de software seguro desde su idealización, diseño, implementación e implantación final. Finalmente se analizarán en forma detallada las vulnerabilidades que se introducen en la construcción de aplicaciones, como detectarlas y cómo evitar que estas puedan ser un problema de seguridad a nivel de sistemas.

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: Diseño seguro, buenas prácticas de codificación, OWASP, vulnerabilidades comunes y mitigación.

III. COMPETENCIAS DEL PERFIL DE EGRESO ASOCIADAS

1. Aplicar en la práctica profesional los valores humanos, la ética y los mecanismos de seguridad laboral.
2. 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.
3. Aplicar, producir y difundir conocimientos técnicos y científicos en el área de la informática.

IV. ORGANIZACIÓN DE LA ASIGNATURA

Unidades	Contenidos	Resultados de aprendizaje
1. Diseño seguro.	1.1. Principios de diseño de seguridad. 1.2. Modelos de Diseño de Software Seguro (SDL, SAMM, etc). 1.3. Proceso de desarrollo de software seguro: 1.3.1. Definición y descubrimiento de los Objetivos de seguridad. 1.3.2. Marcos de seguridad. 1.3.3. Categoría de vulnerabilidades. 1.3.4. Consideraciones de despliegue. 1.3.5. Directrices de diseño. 1.3.6. Modelado de amenazas . 1.3.7. Diseño arquitectónico seguro. Consideraciones de despliegue e infraestructura. 1.3.8. Revisión de código seguro. 1.3.9. Revisión de despliegue seguro. 1.4. DevSecOps: concepto, componentes y procesos y pipelines. Orquestación de seguridad. Herramientas y automatización.	1. Describe los principios de seguridad orientados a la construcción de aplicaciones. 2. Describe los pasos mínimos para el desarrollo de software seguro. 3. Describe las diferentes etapas del proceso de diseño de software seguro utilizados actualmente. 4. Describe el proceso de desarrollo de software basado en DevSecOps. 5. Utiliza los modelos de desarrollo de software seguro en aplicaciones usuales del mercado. 6. Utiliza el modelado de amenazas para diferentes escenarios y modelos de aplicaciones.
2. Desarrollo de software seguro.	2.1. Diferencia entre falla/error/failure 2.2. Cómo combatir la	1. Describe los escenarios para diferenciar falla, error y failure en los programas.

Unidades	Contenidos	Resultados de aprendizaje
	inseguridad en el software. 2.3. Diferentes tipos de fallas de implementación (de lógica, de diseño, de consideración en la plataforma) 2.4. Técnicas de abordajes para detectar vulnerabilidades en el software. 2.5. Prácticas sugeridas de programación en diferentes plataformas (escritorio, web,móvil). Estándares asociados 2.6. Testing. Concepto. Tipos. Coberturas (sentencia, condicionales, branch y de camino). 2.6.1. Fuzzing 2.6.2. Análisis estático (AS). Conceptos básicos. Límites del AS. FlowAnalysis. Uso de AS en seguridad. 2.6.3. Ejecución simbólica. Concepto. Ejecución simbólica básica orientada a seguridad. 2.6.4. Test de penetración testing orientada a aplicaciones.	2. Describe los diferentes tipos de fallas en la implementación especialmente asociadas a seguridad. 3. Describe las principales mejores prácticas de programación para diferentes entornos de desarrollo particularmente las asociadas a seguridad. 4. Describe las principales estrategias para el desarrollo de verificación de seguridad en el código (análisis estático y dinámico).
3. Seguridad Web.	3.1. Características y funcionamiento de aplicaciones web actuales. 3.2. OWASP: descripción y objetivos. Vulnerabilidades de aplicaciones web Principales ataques web y contramedidas 3.3. Autenticación y autorización. 3.4. Seguridad en Navegadores: modelo de ejecución y principios de	1. Describe las principales características de las aplicaciones basadas en web. 2. Describe las principales vulnerabilidades descritas en OWASP. 3. Describe los mecanismos de autenticación y autorización utilizados en las aplicaciones web. 4. Describe el modelo de amenazas de los servidores web. 5. Describe las características principales de los WAF (Web

Unidades	Contenidos	Resultados de aprendizaje
	seguridad, DOM, JavaScript, SOP, cookies/frames. 3.5. Seguridad de Servidores web. 3.6. Técnicas de detección de vulnerabilidades orientadas a aplicaciones web. 3.7. WAF. Características y funcionamiento para protección de aplicaciones web.	Application Firewall) y su funcionamiento. 6. Identifica las vulnerabilidades presentes en las aplicaciones web utilizando herramientas de uso común en el mercado. 7. Compara los resultados arrojados por herramientas de detección de vulnerabilidades y de ataques (WAF).
4. Vulnerabilidades Comunes y Mitigación.	4.1. Definición de vulnerabilidades, identificación y tipificación. CVE y CWE. 4.2. Los errores más comunes en el software. CWE Top. 4.3. Buffer overflow. Descripción. Funcionamiento.Tipos (stack, formatstring, heap, return-to-libc, ROP) y defensas. 4.4. Integeroverflow. 4.5. Memoria segura: seguridad temporal y seguridad espacial 4.6. Tipeo de datos seguro. Diseño de tipeo de datos por seguridad. 4.7. Estrategias para evitar exploits. 4.8. MITRE ATT&CK.	1. Describe de manera general las vulnerabilidades orientadas a las aplicaciones. 2. Define los errores más comunes en el software y clasificarlos de acuerdo a los estándares actualmente utilizados. 3. Comprende el funcionamiento de las vulnerabilidades de bajo nivel y su tipificación. 4. Utiliza los estándares de facto para la clasificación e identificación de vulnerabilidades (CVE y CWE). 5. Desarrolla ejemplos de identificación de vulnerabilidades de buffer overflow. 6. Describe las estrategias principales para evitar exploits en las aplicaciones. 7. Reconoce la aplicación de la matriz MITRE ATT&CK.

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:** 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 estudiante buscará resolver un problema a través del conocimiento que adquirió en el aula.

- **Aprendizaje basado en proyectos:** el docente propondrá la realización de un proyecto que involucre todos los resultados de aprendizaje de la materia. De esta forma el estudiante participa activamente en su aprendizaje, desarrollando diferentes habilidades para solucionar un problema a través de este proyecto.

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

Pizarras, marcadores, computadoras, proyector, parlantes para multimedia, plataforma virtual, sala de laboratorio equipada para prácticas con sistemas operativos Linux o Windows y acceso a Internet.

VIII. BIBLIOGRAFÍA

- Application Security. ProgramHandBook (2022).D. Fisher. ManningPublications Co.
- Web Application Security. Explotation and Countermeasuresfor Modern web Applications (2020) A. Hoffman. .O'Really Media, Inc.
- The Web applicationHacker'sHandboo (2011). D. Stuttard& M. Pinto. 2nd. Edition .WileyPress.
- Introduction to Computer Security (2018). M. Bishop Addison Wesley.
- Hacking Web Apps. Detecting and Preventing Web Application Security Problems (2012) M. Shema. SyngressPress. Elsevier.
- Thetangled Web. A guide to securing Modern Web Applications (2012). M. Zalewski.
- 24 deadlysins of software security (2010). M. Howard, D. LeBlanc y J. Viega . McGraw Hill.
- Computer Security. Principles and Practices (2018). W. Stalling& L. Brown. 3rd Edition. Pearson Educ. Inc.