



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

**DESARROLLO SEGURO DE
APLICACIONES**

Carrera/ Plan:

Licenciatura en Sistemas

Plan 2003-07 / Plan 2012 /Plan 2015

Ingeniería en Computación

Año: 2015

Régimen de Cursada: Semestral

Carácter: Optativa

Correlativas:

Proyecto de Software

Año 2015

Profesor: Einar Lanfranco

Hs semanales: 6 hs

FUNDAMENTACIÓN

Esta materia permitirá articular contenidos con las asignaturas de la carrera relacionadas al desarrollo de software complementándola desde una visión de los requerimientos no funcionales, que son transversales a todas las aplicaciones que se desarrollan en la actualidad

.El objetivo de la materia es brindar al alumno una visión global de los problemas de seguridad que afectan al software, permitiendo no sólo identificar vulnerabilidades, amenazas e incidentes sino brindar los conceptos y herramientas que permitan desarrollar un software donde las cuestiones inherentes a la seguridad estén siempre presentes. Hoy en día prácticamente ningún sistema funciona de manera aislada, por el contrario, las aplicaciones tienen fuerte interrelación con otros softwares (aplicaciones, librerías externas, etc), funcionan en red y en general son multiusuario, constituyéndose todas estas cuestiones en un riesgo de seguridad . El desarrollo de aplicaciones debe abordarse desde diferentes perspectivas y la seguridad es una de ellas. Por estos motivos la temática abordada en “Desarrollo seguro de aplicaciones” son relevantes para la formación del perfil profesional de un informático interesado en el desarrollo de software.

OBJETIVOS GENERALES

- *Abordar en profundidad los conceptos relacionados a los problemas de seguridad en aplicaciones de software.*
- *Conocer los mecanismos de protección existentes que permiten minimizar la ocurrencia de los mencionados problemas*
- *Desarrollo de un trabajo integrador que articulando con el trabajo integrador de la materia Proyecto de Software que signifique para el alumno una aplicación concreta de los conocimientos de seguridad adquiridos en la materia utilizando como objetivo el trabajo de la materia anterior.*

CONTENIDOS MINIMOS



- Acercamiento al mundo del software libre, teniendo como objetivo que el alumno genere algún aporte concreto a nivel de desarrollo a proyectos que respeten estos tipos de licencia.

El alumno recibirá clases teóricas incluyendo aspectos avanzados de temas específicos de Ingeniería de Software y cuestiones de seguridad que afectan al software en general. Estos conceptos teóricos serán acompañados por una intensa tarea de desarrollo (individual o en equipos) que consistirá en detectar y solucionar problemas de seguridad.

PROGRAMA ANALÍTICO

Organizar y describir por unidades los diferentes temas y subtemas que se van a desarrollar en dicho curso.

Unidad I: Software Libre: conceptos, licencias y herramientas.

Unidad II: Contenidos básicos de Seguridad y Privacidad.

Unidad III: Top ten de los problemas de seguridad en aplicaciones web.

Unidad IV: Herramientas para desarrollo seguro en web

Unidad V: Vulnerabilidades, Exploits y Zero Days

Unidad VI: Problemas en la cloud: el top nine 2013. Entenderlos y prevenirlos.

METODOLOGÍA DE ENSEÑANZA

La asignatura consolida la formación experimental y profesionalizante del alumno ubicándolo en un entorno de trabajo similar al real y cotidiano.

La teoría y práctica se encuentran estrechamente vinculadas. Estas instancias son semanales. La teoría trabaja lineamientos conceptuales aplicados que se van a utilizar en el desarrollo de los trabajos prácticos.

En general en los prácticos los alumnos resuelven los TPs utilizando herramientas de soporte y desarrollo actuales típicamente se selecciona aquellas desarrolladas por la comunidad de software libre. Estas instancias son supervisadas por los docentes.

El desarrollo de la materia tiene un hilo conductor y es mediante la aplicación de técnicas de ingeniería de software de desarrollo ágil, donde existe alto nivel de interacción con los alumnos.

Se trabaja con los siguiente recursos:

- 1- Guías, diapositivas, videos, libros, tutoriales y especificaciones de estándares a utilizar.
- 2- Cañón, PC, demostraciones de usos de herramientas con ejemplos en vivo.
- 3- Herramientas: Subversion, servidor web, IDEs de desarrollo, frameworks de explotación, debuggers, sistema de requerimientos.
- 4- Plataforma de e-learning.

EVALUACIÓN

Algunos prácticos incluirán ejercicios de entrega obligatoria o evaluación online por medio de la plataforma virtual. Las entregas formarán parte del trabajo final para la aprobación de la cursada.

Al finalizar la cursada se toma una evaluación escrita integradora con sus correspondientes recuperatorios.

*La **aprobación de la materia** estará dada por la aprobación de los trabajos prácticos, las evaluaciones inline y el resultado de un trabajo integrador formado por todas las entregas parciales necesarias para la aprobación de la cursada mas una extensión del mismo. La asistencia a las clases teóricas aportará a la calificación final.*



La nota promedio de todos los items descriptos será la nota final de la materia

BIBLIOGRAFÍA OBLIGATORIA

1. Owasp Top 10 -2013 The Ten Most Web Application Security Risks. Editado por OWASP (Open Wep Application Security Project)
2. Owasp Top 10 -2007 THE TEN MOST CRITICAL WEB APPLICATION SECURITY VULNERABILITIES . Editado por OWASP (Open Wep Application Security Project)
3. Artículos técnicos de especialistas sobre Buffer Overflow: Smashing The Stack For Fun And Profit - <http://www.phrack.com/issues.html?issue=49&id=14&mode=txt>
4. Herramientas de seguridad Open Source. Top 125, <http://sectools.org/tag/web-scanners/>
5. Git Branching → <http://git-scm.com/book/en/Git-Branching>
6. The Notorious Nine Cloud Computing Top Threats in 2013 → https://downloads.cloudsecurityalliance.org/initiatives/top_threats/The_Notorious_Nine_Cloud_Computing_Top_Threats_in_2013.pdf

BIBLIOGRAFÍA COMPLEMENTARIA

1. Recursos varios de SANS , <http://www.sans.org/security-resources/>
2. Desarrollo web ágil con Symfony2, autor: Javier Eguiluz
3. Symfony - The Book, por SensioLabs, <http://symfony.com/doc/current/book/index.html>
4. H. Kniberg, Scrum and XP from the Trenches, InfoQ
5. K. Schwaber/M. Beedle , Agile Software Development with Scrum
6. Scrum in five minutes, Softhouse



CRONOGRAMA DE CLASES Y EVALUACIONES

Clase	Contenidos/Actividades	Evaluaciones previstas
1	Presentación de la materia. Conceptos básicos de Seguridad y Privacidad. Software Libre, licencias, usos, productos.	
2	Uso de sistemas de control de versiones. Git Software Cliente-Servidor. Instalación del entorno que utilizaremos	
3	Presentación de OWASP Diferencias: Modelo basado en riesgo vs Modelo basado en cantidad de vulnerabilidades. OWASP TOP TEN 2013	Evaluación online sobre conceptos básicos
4	OWASP TOP TEN 2013 - A1- Injection	Recuperación del trabajo de Proyecto de Software montandoló en un servidor de Subversion
5	OWASP TOP TEN 2013- A2- Broken Authentication and Session Management	
6	•OWASP TOP TEN 2013- A3- Cross-Site Scripting (XSS)	
7	OWASP TOP TEN 2013- •A4-Insecure Direct Object References •A5-Security Misconfiguration •A7-Missing Function Level Access Control	Evaluación OWASP
8	•OWASP TOP TEN 2013- •A6-Sensitive Data Exposure	
9	•OWASP TOP TEN 2013- •A8-Cross-Site Request Forgery (CSRF)	



**UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA**

10	<ul style="list-style-type: none">•OWASP TOP TEN 2013-•A9-Using Components with Known Vulnerabilities•A10-Unvalidated redirects and Forwards	
11	CVEs, Zero Days, Exploits, Conceptos generales sobre vulnerabilidades Como generar un Patch	Evaluación OWASP
12	The Notorious Nine Cloud Computing Top Threats in 2013	
13	Planteo extensión del trabajo final	Evaluación: Problemas en la Cloud Vulnerabilidades
15		
16		
17		
18		

Contacto de la cátedra (mail, página, plataforma virtual de gestión de cursos):

Mail de contacto: enar@info.unlp.edu.ar

Plataforma virtual: <https://catedras.info.unlp.edu.ar>

Firmas del/los profesores responsables: