



**SEMINARIO DE LENGUAJES
(OPCION JavaScript)**

Año 2021

Carrera/ Plan:

Licenciatura en Informática

Plan 2003-07 / Plan 2012 / Plan 2015

Licenciatura en Sistemas

Plan 2003-07 / Plan 2012 / Plan 2015

Analista Programador Universitario

Plan 2007 // Plan 2015

Analista en TICs

Año: 2°

Régimen de Cursada: Semestral

Carácter: Obligatoria

Correlativas: Algoritmos, Datos y
Programas

Profesor: Paola Amadeo

Hs. semanales: 6 hs.

FUNDAMENTACIÓN

Dentro del marco de la materia genérica Seminario de Lenguajes, esta opción introduce el lenguaje JavaScript. A partir de 1995, la penetración de Internet en todos los ámbitos generó aplicaciones cada vez más complejas y con numerosas capas para cumplir con los requisitos del paradigma centrado en la red. JavaScript es uno de los lenguajes que surgió para dar respuesta a este paradigma siendo hoy uno de los más populares lenguajes de scripts usado para construir aplicaciones front-end, back-end y mobile. Aprender JS en esta instancia de la carrera brinda bases sólidas para materias correlativas así como también para el futuro profesional del estudiante. La metodología aprender haciendo es fundamental en estos cursos, donde el mundo digital requiere competencias de trabajo en equipo, gestión de plataformas para el desarrollo de proyectos, fomenta la motivación y la creatividad.

OBJETIVOS GENERALES

Profundizar los conocimientos obtenidos por el alumno en los primeros cursos vinculados con Algoritmos y Programación, permitiéndole desarrollar un estudio teórico-práctico de algún lenguaje de programación (el lenguaje puede variar con los cambios tecnológicos), poniendo énfasis en el análisis formal de las características del lenguaje y su comparación con los que el alumno conociera a ese momento (típicamente Pascal).

COMPETENCIAS

- CGS2- Comunicarse con efectividad en forma oral y escrita.
- CGS4- Aprender en forma continua y autónoma, con capacidad de planificar este aprendizaje.



- CGS6- Capacidad para interpretar la evolución de la Informática con una visión de las tendencias tecnológicas futuras.
- CGT1- Identificar, formular y resolver problemas de Informática.
- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.
- LI- CE4 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real, especificación formal de los mismos, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.
- LS- CE1 – Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. Especificación formal, diseño, implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software que se ejecuten sobre sistemas de procesamiento de datos, con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.

RESULTADOS DE APRENDIZAJE

- 1.3. Describir los avances informáticos actuales e históricos y demostrar cierta visión sobre tendencias y avances futuros (Básico)
 - 3.1. Definir y diseñar hardware/software informático/de red que cumpla con los requisitos establecidos (Básico)
 - 3.3. Elegir y utilizar modelos de proceso adecuados, entornos de programación y técnicas de gestión de datos con respecto a proyectos que impliquen aplicaciones tradicionales, así como aplicaciones emergentes (Básico)
 - 3.4. Describir y explicar el diseño de sistemas e interfaces para interacción persona-ordenador y ordenador-ordenador (Básico)
 - 3.5. Aplicar las correspondientes competencias prácticas y de programación en la creación de programas informáticos y/u otros dispositivos informáticos (Adecuado)
- 6.1. Organizar su propio trabajo de manera independiente demostrando iniciativa y ejerciendo responsabilidad personal (Básico)
- 6.3. Planificar su propio proceso de aprendizaje autodidacta y mejorar su rendimiento personal como base de una formación y un desarrollo personal continuos (Básico)

CONTENIDOS MINIMOS

Estudio de un lenguaje de programación en el que se desarrollen aplicaciones concretas. En lo posible la oferta de lenguajes será variable y actualizada con el cambio tecnológico



PROGRAMA

Unidad 1 HTML, CSS y JS

Unidad 2. Componentes básicos del lenguaje.

Unidad 3: Estructurando datos

Unidad 4: Eventos y formularios

Unidad 5 Librerías Client-side.

Unidad 6: Server-side JS.

Taller de ética profesional.

BIBLIOGRAFÍA

- JavaScript. <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Standard ECMA-262 <https://www.ecma-international.org/publications/standards/Ecma-262.htm>
- JavaScript: The Definitive Guide, 7th Edition. D. Flanagan (2020) . Ed O`Reilly <https://www.oreilly.com/library/view/javascript-the-definitive/9781491952016/>
- Referencia de JavaScript. <https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia>
- HTML 5.2 W3C Recommendation <https://www.w3.org/TR/html52/>
- HTML5: Entiende el cambio, aproveche su potencial. De Luca, Damián. (2011). Ed. Andina <http://catalogo.info.unlp.edu.ar/meran/opac-detail.pl?id1=5738#.X70rOrN7IPY>
- Pro HTML5 accessibility: building an inclusive web. O Connor, Joshue (2012). Ed. Apress. <http://catalogo.info.unlp.edu.ar/meran/opac-detail.pl?id1=7241#.X70tQLN7IPY>
- Beginning hybrid mobile application development. Panhale, Mahesh (2016) Ed. Apress <http://catalogo.info.unlp.edu.ar/meran/opac-detail.pl?id1=7223#.X70s27N7IPY>
- W3C. Cascading Style Sheets. <https://www.w3.org/Style/CSS/>
- Visual Studio Code (<https://code.visualstudio.com/download>)–Web Storm (<https://www.jetbrains.com/webstorm/>)



UNIVERSIDAD
NACIONAL
DE LA PLATA

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

- <https://nodejs.org/es/>
- Entendiendo los frameworks de JavaScript del lado del cliente
https://developer.mozilla.org/es/docs/Learn/Herramientas_y_pruebas/Lado-del-cliente_JavaScript_frameworks
- 3 librerías JavaScript para la visualización de datos (2019)
<https://www.analyticslane.com/2019/06/07/3-librerias-javascript-para-la-visualizacion-de-datos/>
- Herramientas para desarrollar juegos con HTML5 <http://untitled.es/herramientas-juegos-html5/>



METODOLOGÍA DE ENSEÑANZA

La metodología es del tipo taller, con clases teóricas donde se desarrollan los aspectos conceptuales del lenguaje, que se dictan utilizando videos y presentaciones del tipo slideshow. Se incluyen ejemplos de código para resolver distintas situaciones problemáticas y animaciones que permitan ilustrar los conceptos más complejos durante la clase. Estas clases no son obligatorias y se tomará asistencia.

Las clases prácticas se realizan en la sala de PC de la Facultad, basada en proyectos, los estudiantes plantean sus dudas, trabajan con los ayudantes y se corrigen entre ellos, acompañándose durante todo el proceso. Las prácticas incluyen ejercicios a entregar y tendrán una calificación conceptual. Incluyen una instancia de coloquio donde el docente a cargo del grupo realiza distintas preguntas sobre la temática abordada. La entrega se realiza a través de la plataforma virtual. En los coloquios el estudiante expone la tarea realizada en forma individual y el docente evalúa no sólo los conocimientos sino la claridad de la presentación, su organización y la forma de expresión. Esto se refleja en planillas que conforman documentación de evaluación del coloquio.

Esta metodología se complementa con la plataforma virtual Moodle. Se incluyen autoevaluaciones de carácter optativo para reforzar temas teóricos y los coloquios de los trabajos prácticos utilizando la plataforma. Cada ayudante tiene a cargo un grupo de alumnos, y será su función seguirlo en el desenvolvimiento de la cursada, intentado identificar los puntos más problemáticos para poder resolverlos en forma rápida y no provoque el abandono de la cursada.

A fin de mejorar la comunicación entre los estudiantes y la cátedra, se utilizarán plataformas colaborativas como Slack para informar los distintos eventos y materiales de la cátedra en forma ágil y automática.

Se utilizará también una herramienta para el seguimiento de la cátedra, de la asistencia de los alumnos a las prácticas y del aprovechamiento de cada práctica.

EVALUACIÓN

La **aprobación de la cursada** estará dada por un trabajo final integrador. Tendrá la forma de “desafíos” que deben convertirse en “ideas proyecto” y posteriormente en potenciales desarrollos del alumno. Se trata de que alumno logre abstraer una serie de pasos que respondan a una metodología clásica de investigación y lo ayuden a formarse en esta competencia:

- Búsqueda de bibliografía actualizada sobre el tema.
- Abstracción del desafío/problema como una “idea proyecto a resolver”.
- Expresión sintética de la especificación del proyecto, con recursos humanos requeridos y plan de tareas.



UNIVERSIDAD
NACIONAL
DE LA PLATA

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

- Implementación y defensa oral/escrita de la solución al desafío.

La cátedra acompaña el proceso del alumno, para consolidar sus habilidades para esta competencia.

La evaluación de estas competencias se refleja en una planilla detallada, donde se indica la capacidad del alumno para desarrollar su aprendizaje y la formulación de la solución al desafío en forma autónoma.

En la cátedra se pone énfasis en la capacidad del alumno para conocer técnicas y herramientas de aplicación en Informática (en lo posible siguiendo las tendencias marcadas por el cambio tecnológico) y en la aplicación efectiva de las mismas. La cátedra acompaña el proceso con materiales para que el alumno estudie casos y valore la selección y empleo eficiente de herramientas y técnicas determinadas para cada problema. La evaluación de esta competencia forma parte de las evaluaciones de trabajos prácticos.

La **aprobación de la materia** estará dada por la aprobación de la cursada y un trabajo final. La asistencia a las clases teóricas y las entregas de los ejercicios de las prácticas aportará a la calificación final.



CRONOGRAMA DE CLASES Y EVALUACIONES

Fecha de comienzo: semana 15/03/2021

Fecha de comienzo de prácticas: semana 22/03/2021

Clase	Contenidos/Actividades	Evaluaciones previstas
1	Historia. Conceptos básicos. Estructura de un documento HTML. Tags básicos, semántica y usos. Atributos. Introducción a JS. Historia y estándares. ECMAScript.	
2	Introducción a CSS. Conceptos básicos. Reglas CSS. Elementos, clases e identificadores. JS: Tipos de datos, variables, expresiones, funciones y alcances. HTML input fields.	
3	Estructuras de control: condicionales, iterativas y operadores lógicos.	
4	Manejo de eventos, onclick y submit entre otros.	Entrega ejercicios TP1
5	La API DOM. Formularios. Detección y corrección de errores.	
6	Objetos, propiedades y métodos. Pasaje de parámetros. Funciones flecha.	
7	Formato JSON.	
8	Introducción a Node.js. Server-side	Entrega ejercicios TP2
9	Node.js continuación	
10	Librerías client-side	
11	JQuery.js Chart.js D3.js React	
12	Desarrollo del Trabajo Final	Entrega ejercicios TP3
13	Desarrollo del Trabajo Final	
14	Desarrollo del Trabajo Final	
15	Desarrollo del Trabajo Final	



UNIVERSIDAD
NACIONAL
DE LA PLATA

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

16	Desarrollo del Trabajo Final	Entrega Final 1/7/2021
17	Correcciones y recuperatorios.	
18	Luego de la entrega final se realizará una puesta en común de todos los estudiantes de su trabajo final, haciendo hincapié en los problemas que tuvieron en el desarrollo del mismo y la forma de solucionarlos, así como también haciendo un balance de su desempeño.	

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

seminariojs@info.unlp.edu.ar

<http://catedras.info.unlp.edu.ar>

Firmas del profesor responsable: