

**SEMINARIO DE LENGUAJES  
OPCION PHP, REACT, API REST**

Año 2025

**Carrera/ Plan: (Dejar lo que corresponda)***Licenciatura en Informática Plan 2021/Plan 2015**Licenciatura en Sistemas Plan 2021/Plan 2015**Analista Programador Universitario Plan 2021/Plan 2015**Analista en Tecnología de la Información y la Comunicación  
Plan 2021/Plan 2017***Año:** 2do.**Régimen de Cursada:** Semestral**Carácter (Obligatoria/Optativa):** Obligatoria**Correlativas:** Taller de Programación**Profesor/es:** Raúl Champredonde**Hs. semanales teoría:** 3**Hs. semanales práctica:** 3**FUNDAMENTACIÓN**

*El segundo año de la carrera es un momento adecuado para que el alumno aprenda un nuevo lenguaje de programación y se ponga en contacto con tecnologías como las de internet o aplicaciones para móviles, que lo acerque a las herramientas de uso común en la vida profesional.*

**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.*

**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).*

**COMPETENCIAS**

- CGS2- Comunicarse con efectividad en forma oral y escrita.

*Se evalúa a los alumnos en coloquios en los cuales deben exponer la tarea realizada en forma individual y donde el docente evalúa no sólo los conocimientos sino la claridad de la presentación, su organización y la forma de expresión.*

- CGS4- Aprender en forma continua y autónoma, con capacidad de planificar este aprendizaje.

*Se presenta a los alumnos pequeños problemas a resolver mediante la investigación o búsqueda de información. Se evalúa el nivel de aplicación de los mismos en el proyecto entregado como parte de la aprobación de la cursada.*

- CGS6- Capacidad para interpretar la evolución de la Informática con una visión de las tendencias tecnológicas futuras.

Se presentan diversas tecnologías, y se evalúa el nivel de utilización y su corrección en el proyecto entregado como parte de la aprobación de la cursada.

- CGT1- Identificar, formular y resolver problemas de Informática.

Se pone énfasis en el proceso de identificación de problemas del mundo real, especificación de los mismos como problemas resolubles desde la informática y en el desarrollo de soluciones verificables para los mismos. Se evalúa en base al software desarrollado por el alumno como solución al problema planteado que debe solucionar para aprobar la cursada.

- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.

Se pone énfasis en el uso efectivo de técnicas y herramientas de aplicación en Informática. La evaluación de esta competencia forma parte de las evaluaciones de trabajos entregados y su defensa en coloquio individual.

- LI- CE4 – Realizar y/o evaluar proyectos de desarrollo de software para la resolución 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 – Realizar y/o evaluar proyectos de desarrollo de software para la resolución 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.

### **CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)**

Estudio de un lenguaje de programación en el que se desarrollen aplicaciones concretas. En particular, aplicaciones web, para lo cual resulta necesario un recorrido académico básico por las tecnologías involucradas (cliente/servidor, html, javascript, bases de datos).

### **PROGRAMA ANALÍTICO**

1. Introducción al desarrollo de aplicaciones Web. ¿Cómo funciona la Web? HTTP. Arquitectura Cliente/Servidor. Nombres de dominio. Lenguaje HTML. Requerimiento/respuesta. Tipos de requerimientos. Scripting del lado del cliente (javascript). Scripting del lado del servidor.
2. HTML, CSS
3. Javascript
4. PHP
  - a. Introducción. Sintaxis y semántica del lenguaje. Consideraciones generales del lenguaje. Variables. Tipos de Datos. Operadores y expresiones. Constantes.
  - b. Estructuras de control.
  - c. Subprogramas Funciones. Definición. Invocación. Parámetros formales y actuales. Alcance de una variable: variables locales y globales. Funciones anónimas.
  - d. Matrices Definición. Creación. Matrices asociativas. Matrices multidimensionales. Acceso y manipulación. Strings Definición. Formato. Operaciones básicas.
  - e. Utilización de bases de datos. Introducción al uso de bases de datos y tablas. Introducción a las sentencias elementales SQL. Uso de MySQL.
  - f. Clases y objetos. Definición. Creación. Métodos y propiedades. Herencia.
  - g. Excepciones
  - h. Sesiones. Manejo de sesiones. Cookies
5. API REST
  - a. Concepto de API
  - b. Arquitectura REST
  - c. Desarrollo de API REST
6. React
  - a. Estructura de una aplicación React

- b. JSX
- c. Componentes y componentes función.
- d. Props, estados
- e. Hooks state, effect

## **BIBLIOGRAFÍA**

- [php.net](http://php.net)
- [w3.org](http://w3.org)
- *Desarrollo de Aplicaciones web con php y Mysql.* Manuel Torres Remon. Empresa Editora Macro
- *Desarrollo web con PHP y MySQL. Edición 2018 (Guías Prácticas).* Luis Miguel Cabezas Granado. Anaya Multimedia
- *Learning PHP 7.* Antonio Lopez. Packt Publishing - ebooks Account
- [www.slimframework.com](http://www.slimframework.com)
- <https://laravel-guide.readthedocs.io/en/latest/eloquent/>
- [es.reactjs.org](http://es.reactjs.org)
- [fullstackopen.com](http://fullstackopen.com)
- *Desarrollo Web con React,* Pello Xabier Altadill Izura, Anaya

## **METODOLOGÍA DE ENSEÑANZA**

La actividad curricular se organiza en Teoría, Práctica y explicación de práctica.

La teoría consiste en una presentación y explicación conceptual por parte del docente de los temas centrales donde los alumnos participan haciendo preguntas y planteando dudas.

En la práctica se resuelven ejercicios en forma individual que son planteados en las guías de Trabajos Prácticos. El docente atiende las dudas y consultas de los alumnos para la resolución de los mismos. Las explicaciones de la practica permite la articulación entre la teoría y la practica el docente desarrolla algún ejercicio modelo que resulta importante para la resolución de las ejercitaciones prácticas.

## **EVALUACIÓN**

La aprobación de la cursada consiste en el desarrollo de un proyecto que es entregado y corregido en diferentes etapas. Este trabajo debe cumplir con un conjunto de requisitos enunciados junto con el problema planteado.

La aprobación final de la materia consiste en la defensa individual escrita y oral del proyecto presentado.

## **CRONOGRAMA DE CLASES Y EVALUACIONES**

Clase	Fecha	Contenidos/Actividades
1	10/03/2025	Funcionamiento de la Web. HTTP Nociones de HTML, CSS, JS, XML Nociones de posibles arquitecturas
2	17/03/2025	PHP
3	31/03/2025	PHP. BD. JSON
4	07/04/2025	BD. JSON
5	14/04/2025	API. Arquitectura REST
6	21/04/2025	Desarrollo de una API. Slim, MVC, Eloquent
7	28/04/2025	Desarrollo de una API. Postman. Ejemplo API
8	05/05/2025	HTML, CSS, Javascript
9	12/05/2025	React. Introducción. JSX
10	19/05/2025	React. Componentes función
11	26/05/2025	React. Props. Estados. Hooks state, effect
12	02/06/2025	React. Consumir recursos servidor por una API REST



13	09/06/2025	Ejemplo aplicación React que consume API REST
14	23/06/2025	Integración Frontend - Backend
15	30/06/2025	Integración Frontend – Backend
16	07/07/2025	Repaso

Evaluaciones previstas	Fecha
Entrega/coloquio parcial API REST	06/05/2025
Entrega/coloquio parcial REACT	01/07/2025
Coloquio final	14/07/2025

**Contactos de la cátedra:**

- **Mail (obligatorio):** [rchampre@yahoo.com.ar](mailto:rchampre@yahoo.com.ar)
- **Sitio WEB:**
- **Plataforma virtual: Ideas**
- **Otros:**

Firma del/los profesor/es

**SEMINARIO DE LENGUAJES  
OPCION PHP, REACT, API REST  
REDICTADO**

Año 2025

**Carrera/ Plan: (Dejar lo que corresponda)***Licenciatura en Informática Plan 2021/Plan 2015**Licenciatura en Sistemas Plan 2021/Plan 2015**Analista Programador Universitario Plan 2021/Plan 2015**Analista en Tecnología de la Información y la Comunicación  
Plan 2021/Plan 2017***Año:** 2do**Régimen de Cursada:** Semestral**Carácter (Obligatoria/Optativa):** Obligatoria**Correlativas:** Taller de programación**Profesor/es:** Raúl Champredonde**Hs. semanales teoría:** 3**Hs. semanales práctica:** 3**FUNDAMENTACIÓN**

*El segundo año de la carrera es un momento adecuado para que el alumno aprenda un nuevo lenguaje de programación y se ponga en contacto con tecnologías como las de internet o aplicaciones para móviles, que lo acerque a las herramientas de uso común en la vida profesional.*

**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.*

**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).*

**COMPETENCIAS**

- CGS2- Comunicarse con efectividad en forma oral y escrita.

*Se evalúa a los alumnos en coloquios en los cuales deben exponer la tarea realizada en forma individual y donde el docente evalúa no sólo los conocimientos sino la claridad de la presentación, su organización y la forma de expresión.*

- CGS4- Aprender en forma continua y autónoma, con capacidad de planificar este aprendizaje.

*Se presenta a los alumnos pequeños problemas a resolver mediante la investigación o búsqueda de información. Se evalúa el nivel de aplicación de los mismos en el proyecto entregado como parte de la aprobación de la cursada.*

- CGS6- Capacidad para interpretar la evolución de la Informática con una visión de las tendencias tecnológicas futuras.

Se presentan diversas tecnologías, y se evalúa el nivel de utilización y su corrección en el proyecto entregado como parte de la aprobación de la cursada.

- CGT1- Identificar, formular y resolver problemas de Informática.

Se pone énfasis en el proceso de identificación de problemas del mundo real, especificación de los mismos como problemas resolubles desde la informática y en el desarrollo de soluciones verificables para los mismos. Se evalúa en base al software desarrollado por el alumno como solución al problema planteado que debe solucionar para aprobar la cursada.

- CGT5- Utilizar de manera efectiva las técnicas y herramientas de aplicación de la Informática.

Se pone énfasis en el uso efectivo de técnicas y herramientas de aplicación en Informática. La evaluación de esta competencia forma parte de las evaluaciones de trabajos entregados y su defensa en coloquio individual.

- LI- CE4 – Realizar y/o evaluar proyectos de desarrollo de software para la resolución 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 – Realizar y/o evaluar proyectos de desarrollo de software para la resolución 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.

### **CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)**

Estudio de un lenguaje de programación en el que se desarrollen aplicaciones concretas. En particular, aplicaciones web, para lo cual resulta necesario un recorrido académico básico por las tecnologías involucradas (cliente/servidor, html, javascript, bases de datos).

### **PROGRAMA ANALÍTICO**

1. Introducción al desarrollo de aplicaciones Web. ¿Cómo funciona la Web? HTTP. Arquitectura Cliente/Servidor. Nombres de dominio. Lenguaje HTML. Requerimiento/respuesta. Tipos de requerimientos. Scripting del lado del cliente (javascript). Scripting del lado del servidor.
2. HTML, CSS
3. Javascript
4. PHP
  - a. Introducción. Sintaxis y semántica del lenguaje. Consideraciones generales del lenguaje. Variables. Tipos de Datos. Operadores y expresiones. Constantes.
  - b. Estructuras de control.
  - c. Subprogramas Funciones. Definición. Invocación. Parámetros formales y actuales. Alcance de una variable: variables locales y globales. Funciones anónimas.
  - d. Matrices Definición. Creación. Matrices asociativas. Matrices multidimensionales. Acceso y manipulación. Strings Definición. Formato. Operaciones básicas.
  - e. Utilización de bases de datos. Introducción al uso de bases de datos y tablas. Introducción a las sentencias elementales SQL. Uso de MySQL.
  - f. Clases y objetos. Definición. Creación. Métodos y propiedades. Herencia.
  - g. Excepciones
  - h. Sesiones. Manejo de sesiones. Cookies
5. API REST
  - a. Concepto de API
  - b. Arquitectura REST
  - c. Desarrollo de API REST
6. React
  - a. Estructura de una aplicación React

- b. JSX
- c. Componentes y componentes función.
- d. Props, estados
- e. Hooks state, effect

## **BIBLIOGRAFÍA**

- [php.net](http://php.net)
- [w3.org](http://w3.org)
- *Desarrollo de Aplicaciones web con php y Mysql.* Manuel Torres Remon. Empresa Editora Macro
- *Desarrollo web con PHP y MySQL. Edición 2018 (Guías Prácticas).* Luis Miguel Cabezas Granado. Anaya Multimedia
- *Learning PHP 7.* Antonio Lopez. Packt Publishing - ebooks Account
- [www.slimframework.com](http://www.slimframework.com)
- <https://laravel-guide.readthedocs.io/en/latest/eloquent/>
- [es.reactjs.org](http://es.reactjs.org)
- [fullstackopen.com](http://fullstackopen.com)
- *Desarrollo Web con React,* Pello Xabier Altadill Izura, Anaya

## **METODOLOGÍA DE ENSEÑANZA**

*La actividad curricular se organiza en Teoría, Práctica y explicación de práctica.*

*La teoría consiste en una presentación y explicación conceptual por parte del docente de los temas centrales donde los alumnos participan haciendo preguntas y planteando dudas.*

*En la práctica se resuelven ejercicios en forma individual que son planteados en las guías de Trabajos Prácticos. El docente atiende las dudas y consultas de los alumnos para la resolución de los mismos.*

*Las explicaciones de la practica permite la articulación entre la teoría y la practica el docente desarrolla algún ejercicio modelo que resulta importante para la resolución de las ejercitaciones prácticas.*

## **EVALUACIÓN**

*La aprobación de la cursada consiste en el desarrollo de un proyecto que es entregado y corregido en diferentes etapas. Este trabajo debe cumplir con un conjunto de requisitos enunciados junto con el problema planteado.*

*La aprobación final de la materia consiste en la defensa individual escrita y oral del proyecto presentado.*

## **CRONOGRAMA DE CLASES Y EVALUACIONES**

Clase	Fecha	Contenidos/Actividades
1	25/08/2025	Funcionamiento de la Web. HTTP Nociones de HTML, CSS, JS, XML Nociones de posibles arquitecturas
2	01/09/2025	PHP
3	08/09/2025	PHP. BD. JSON
4	15/09/2025	BD. JSON
5	22/09/2025	API. Arquitectura REST
6	29/09/2025	Desarrollo de una API. Slim, MVC, Eloquent
7	06/10/2025	Desarrollo de una API. Postman. Ejemplo API
8	20/10/2025	HTML, CSS, Javascript
9	27/10/2025	React. Introducción. JSX
10	03/11/2025	React. Componentes función
11	10/11/2025	React. Props. Estados. Hooks state, effect
12	17/11/2025	React. Consumir recursos servidor por una API REST



13	01/12/2025	Ejemplo aplicación React que consume API REST
14	15/12/2025	Integración Frontend - Backend

Evaluaciones previstas	Fecha
Entrega/coloquio parcial API REST	21/10/2025
Entrega/coloquio parcial REACT	09/12/2025
Coloquio final	22/12/2025

**Contactos de la cátedra:**

- **Mail (obligatorio):** [rchampre@yahoo.com.ar](mailto:rchampre@yahoo.com.ar)
- **Sitio WEB:**
- **Plataforma virtual: Ideas**
- **Otros:**

Firma del/los profesor/es