

Programa de Asignatura

Programación Avanzada

A. Antecedentes Generales

Unidad Académica	Facultad de Diseño							
Carrera	Diseño Interacción Digital							
Código	DIID214							
Ubicación en la malla	Semestre 3 / Año 2							
Créditos	8							
Tipo de asignatura	Obligatorio	x	Electivo		Optativo			
Duración	Bimestral		Semestral	x	Anual		Otro	
Módulos semanales	Clases Teóricas	1	Clases Prácticas	1	Ayudantía		1	
Horas académicas	Clases	x	Ayudantía		Otras horas por periodo completo			
Pre-requisito	Pensamiento Computacional y Programación							

B. Aporte al Perfil de Egreso

El curso de "Programación Avanzada" fortalece el desarrollo del pensamiento computacional y algorítmico, avanzando en las habilidades analíticas del estudiante. Introduce la Programación Orientada a Objetos (OOP) y profundiza el dominio en distintos lenguajes de programación. Además, se enfoca en la gestión y análisis de bases de datos, la interconexión de dispositivos y plataformas, y el desarrollo de sistemas tecnológicos funcionales e interactivos. A

través de proyectos aplicados, los estudiantes aprenden a integrar aspectos funcionales, interactivos y expresivos en sus propuestas.

Esta asignatura, que pertenece a la Línea de Representación y Tecnología, se dicta en el ciclo de Bachillerato, aportando en el desarrollo de las competencias: Dominio de Herramientas Tecnológicas y Procesos de Producción, Transformación Digital, y Pensamiento Sistémico.

C. Competencias y Resultados de Aprendizaje Generales que desarrolla la asignatura

Competencias Genéricas	Resultados de Aprendizaje Generales
<ul style="list-style-type: none"> Transformación Digital 	<p>Desarrolla pensamiento computacional y algorítmico avanzado, aplicando estrategias de resolución de problemas en contextos de diseño e interacción.</p> <p>Implementa la Programación Orientada a Objetos (OOP) en distintos lenguajes de programación para estructurar sistemas modulares y escalables.</p> <p>Construye y despliega interfaces interactivas avanzadas, utilizando frameworks modernos y buenas prácticas de desarrollo web.</p> <p>Maneja bases de datos y APIs, aplicando estrategias de gestión y visualización de información para enriquecer la funcionalidad de proyectos interactivos.</p> <p>Interconecta dispositivos y plataformas, explorando la comunicación entre diferentes tecnologías para el desarrollo de experiencias interactivas.</p> <p>Integra herramientas de programación y tecnologías emergentes, considerando principios de diseño y experiencia de usuario en el desarrollo de proyectos.</p>
<p>Competencias Específicas</p> <ul style="list-style-type: none"> Pensamiento Sistémico Dominio de Herramientas Tecnológicas y Procesos de Producción 	

D. Unidades de Contenidos y Resultados de Aprendizaje

Unidades de contenido	Competencia	Resultados de Aprendizaje
Unidad 1: Fundamentos de Programación Avanzada e Interactividad <ul style="list-style-type: none"> Revisión de estructuras básicas: algoritmos, estructuras de control y lógica. Introducción a proyectos interactivos simples. 	Transformación Digital Pensamiento Sistémico	Desarrolla pensamiento computacional y algorítmico avanzado aplicando estrategias de resolución de problemas en diseño e interacción.
Unidad 2: Programación Orientada a Objetos (OOP) <ul style="list-style-type: none"> Conceptos básicos: clases, objetos, herencia, encapsulación. Implementación de OOP en lenguaje escogido (Javascript, Java, Python u otro). Ejercicios prácticos: simulaciones, sistemas modulares. 	Dominio de Herramientas Tecnológicas y Procesos de Producción Pensamiento Sistémico	Implementa la Programación Orientada a Objetos (OOP) en distintos lenguajes de programación para estructurar sistemas modulares y escalables.
Unidad 3: Desarrollo Web e Interactividad Compleja <ul style="list-style-type: none"> Frameworks (Django, Ruby on Rails, Codeinginter, Angular, o similares) Creación de interfaces interactivas. Comprensión de la arquitectura de software Integración del DOM y eventos. Gestión de ambientes virtuales 	Dominio de Herramientas Tecnológicas y Procesos de Producción Pensamiento Sistémico	Construye y despliega interfaces interactivas avanzadas utilizando frameworks modernos y buenas prácticas de desarrollo web.

<p>Unidad 4: Bases de Datos y APIs</p> <ul style="list-style-type: none"> ▪ Modelado y gestión de datos NoSQL. ▪ Consumo e integración de APIs externas. ▪ Proyectos prácticos: aplicaciones con datos en tiempo real. 	<p>Dominio de Herramientas Tecnológicas y Procesos de Producción</p> <p>Pensamiento Sistémico</p>	<p>Maneja bases de datos y APIs, aplicando estrategias de gestión y visualización de información para enriquecer la funcionalidad de proyectos interactivos.</p> <p>Interconecta dispositivos y plataformas, explorando la comunicación entre diferentes tecnologías para el desarrollo de experiencias interactivas.</p>
<p>Unidad 5: Integración de contenidos en proyecto aplicado</p> <ul style="list-style-type: none"> ▪ Diseño y desarrollo de un sistema que combine los conocimientos adquiridos. ▪ Enfoque en funcionalidad, interactividad y escalabilidad. 	<p>Transformación Digital</p> <p>Pensamiento Sistémico</p> <p>Dominio de Herramientas Tecnológicas y Procesos de Producción</p>	<p>Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso, enfocándose en funcionalidad, interactividad y escalabilidad.</p> <p>Interconecta dispositivos y plataformas, explorando la comunicación entre diferentes tecnologías para el desarrollo de experiencias interactivas.</p>

E. Estrategias de Enseñanza-Aprendizaje

Este curso es de carácter teórico-práctico, donde las competencias se desarrollan a través de ejercicios y proyectos de programación avanzada, con un enfoque en la gestión de bases de datos, desarrollo web, y la interconexión de sistemas. Los estudiantes implementan conceptos de Programación Orientada a Objetos y trabajan con tecnologías modernas para diseñar sistemas funcionales e interactivos. Además, adquieren conocimientos aplicables a diversos lenguajes de programación, fortaleciendo sus habilidades técnicas en contextos prácticos y creativos.

Este curso utiliza diversas metodologías de enseñanza-aprendizaje, tales como:

- Estudio y análisis de casos.
- Clase expositiva, con uso de herramientas digitales.
- Aprendizaje basado en proyectos.
- Ejercicios prácticos para desarrollo autónomo, individuales y grupales.
- Esquemas y organizadores gráficos.
- Aula invertida: estudio en casa para resolver desafíos técnicos durante la clase.
- Presentación oral y/o de proyectos.

F. Estrategias de Evaluación

El curso incorpora un enfoque de evaluación integral y continua, compuesto por tres proyectos intermedios y un Proyecto Final. Cada proyecto intermedio se realiza al finalizar una unidad temática clave, con el objetivo de verificar el aprendizaje continuo y la aplicación práctica de los conceptos desarrollados.

Los proyectos intermedios siguen un enfoque constructivista, donde los estudiantes deben integrar y aplicar progresivamente todos los conocimientos obtenidos hasta el momento. Esto promueve una consolidación continua del aprendizaje y fomenta el desarrollo de habilidades técnicas y de diseño interactivo.

El Proyecto Final sintetiza los conocimientos técnicos y teóricos adquiridos a lo largo del curso. En este proyecto, los estudiantes enfrentan un desafío que integra diversas dimensiones, tales como la programación orientada a objetos, el desarrollo web interactivo, la gestión de bases de datos y la interconexión de sistemas. Este enfoque asegura que los estudiantes no solo comprendan la teoría, sino que también sean capaces de aplicar sus conocimientos en situaciones prácticas y complejas.

- Proyectos Intermedios: 70% de la nota final.
- Proyecto 1: Aplicación de OOP.
- Proyecto 2: Desarrollo de una interfaz interactiva.
- Proyecto 3: Aplicación con manejo de datos y APIs.

- Proyecto Final: 30% de la nota final.
- Creación de un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.

La asistencia mínima requerida para aprobar los cursos dictados por la Facultad de Diseño es de un **75% del período comprendido desde el inicio del semestre hasta la última clase del mismo**. No considera el período de exámenes.

G. Recursos de Aprendizaje

Recursos obligatorios

Publicaciones

- McCarthy, L., Reas, C., & Fry, B. (2015). Getting started with P5.js: Making interactive graphics in JavaScript and processing. Maker Media, Inc.
- Soon, W., & Cox, G. (2020). Aesthetic programming: A handbook of software studies. Open Humanities Press.
- Lutz, M. (2013). Learning Python. O'Reilly Media.
- Crockford, D. (2008). JavaScript: The Good Parts. O'Reilly Media.

Recursos en la web

- home | p5.js. (n.d.). Retrieved May 22, 2024, from <https://p5js.org/>
- Documentación oficial de Python (<https://docs.python.org>) y JavaScript.
- Sitios web como MDN Web Docs y W3Schools para referencias rápidas.
- APIs abiertas para proyectos (e.g., OpenWeatherMap, Spotify API).

Recursos complementarios

Publicaciones

- Shiffman, D. (2015). Learning Processing. Morgan Kaufmann.
- Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly Media.

Recursos en la web

- Shiffman, D. (n.d.). The Coding Train - YouTube. Retrieved May 22, 2024, from <https://www.youtube.com/@TheCodingTrain>
- CreativeApplications.Net – Projects, tools and platforms at the intersection of art, media and technology. (n.d.). Retrieved May 22, 2024, from <https://www.creativeapplications.net>

Programación Avanzada

Calendarización Asignatura

Nº	Unidades y Contenidos	Resultados de Aprendizaje	Actividades en clase	Evaluación y Retroalimentación
1	Unidad 1: Fundamentos de Programación Avanzada e Interactividad Introducción al curso y repaso de fundamentos básicos de programación.	Comprender el alcance del curso y consolidar conocimientos previos.	Clase lectiva Discusión abierta	Evaluación inicial sobre conocimientos previos.
2	Unidad 1: Fundamentos de Programación Avanzada e Interactividad Introducción al contenido y repaso de fundamentos básicos de programación.	Desarrolla pensamiento computacional y algorítmico avanzado.	Ejercicios prácticos en clase.	Evaluación formativa mediante ejercicios.
3	Unidad 1: Fundamentos de Programación Avanzada e Interactividad Introducción al curso y repaso de fundamentos básicos de programación.	Desarrolla pensamiento computacional y algorítmico avanzado aplicando estrategias de resolución de problemas en diseño e interacción.	Actividad práctica: desarrollo de un proyecto simple.	Retroalimentación individual sobre proyectos.
4	Unidad 2: Programación Orientada a Objetos (OOP)	Implementa la Programación Orientada a Objetos (OOP) en distintos lenguajes de programación.	Clase expositiva y ejercicios guiados.	Retroalimentación sobre ejercicios.

	Conceptos básicos: clases y objetos.			
5	Unidad 2: Programación Orientada a Objetos (OOP) Conceptos básicos: clases y objetos.	Implementa la Programación Orientada a Objetos (OOP) en distintos lenguajes de programación para estructurar sistemas modulares.	Ejercicios prácticos individuales.	Retroalimentación sobre técnicas de OOP.
6	Unidad 2: Programación Orientada a Objetos (OOP) Conceptos básicos: clases y objetos.	Implementa la Programación Orientada a Objetos (OOP) en distintos lenguajes de programación para estructurar sistemas modulares y escalables.	Avance en proyecto de la unidad.	
7	Unidad 3: Desarrollo Web e Interactividad Compleja Trabajo con Frameworks	Construye y despliega interfaces interactivas avanzadas utilizando frameworks modernos.	Clase expositiva y configuración inicial de proyectos.	Entrega de Proyecto 01. Exposición oral y/o de proyectos individuales. Evaluación mediante pauta de evaluación.
8	Unidad 3: Desarrollo Web e Interactividad Compleja Trabajo con Frameworks	Construye y despliega interfaces interactivas avanzadas utilizando frameworks modernos.	Taller práctico: creación de componentes interactivos.	Retroalimentación sobre configuración, diseño y funcionalidad de componentes.
9	Unidad 3: Desarrollo Web e Interactividad Compleja Trabajo con Frameworks	Construye y despliega interfaces interactivas avanzadas utilizando frameworks modernos.	Actividad práctica: desarrollo de un formulario dinámico.	Evaluación formativa de la funcionalidad implementada. Entrega de Proyecto 02
10	Unidad 4: Bases de Datos y APIs Introducción a bases de datos NoSQL.	Maneja bases de datos aplicando estrategias de gestión y visualización de información.	Clase teórica y configuración inicial de bases de datos.	Retroalimentación sobre configuraciones realizadas.

11	Unidad 4: Bases de Datos y APIs Introducción a APIs.	Maneja bases de datos y APIs para enriquecer la funcionalidad de proyectos interactivos.	Actividad práctica: uso de una API pública en un proyecto.	Retroalimentación sobre integración de datos.
12	Unidad 4: Bases de Datos y APIs Visualización de Data utilizando APIs.	Maneja bases de datos y APIs para enriquecer la funcionalidad de proyectos interactivos.	Taller: creación de una visualización interactiva con datos en tiempo real. Trabajo en Proyecto 03	Evaluación del proyecto de visualización.
13	Unidad 5: Integración de contenidos en proyecto aplicado Definición del proyecto final y planificación. Trabajo en Examen.	Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.	Discusión grupal sobre Examen, ideas de proyectos y objetivos.	Entrega de Proyecto 03 Retroalimentación sobre
14	Unidad 5: Integración de contenidos en proyecto aplicado Definición del proyecto final y planificación. Trabajo en Examen.	Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.	Trabajo práctico en Examen. Asesoría individual del profesor.	Retroalimentación sobre las propuestas iniciales y progreso de Examen.
15	Unidad 5: Integración de contenidos en proyecto aplicado Definición del proyecto final y planificación. Trabajo en Examen.	Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.	Taller práctico en Examen y pruebas de funcionalidad.	Evaluación formativa del desarrollo avanzado.

16	Unidad 5: Integración de contenidos en proyecto aplicado Definición del proyecto final y planificación. Trabajo en Examen.	Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.	Trabajo práctico en Examen y revisión final con el profesor.	Retroalimentación detallada y últimas correcciones.
17	Unidad 5: Integración de contenidos en proyecto aplicado Presentación Examen.	Diseña y desarrolla un sistema funcional e interactivo que combine los conocimientos adquiridos durante el curso.	Presentación oral y demostración del proyecto.	Entrega de Examen. Evaluación final del proyecto y retroalimentación integral en base a rúbrica.