

FACULTAD
DEPARTAMENTO TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES (TIC)

Materia:	Fundamentos de Programación para
NRC:	10614
Código:	09686
Grupo:	001
Programa/Semestre:	DMI 03
Periodo Académico:	202220
Intensidad Semanal:	3
Créditos:	4

PROFESORES

Santiago Martin Moreno Benavides

DESCRIPCIÓN

En este curso se busca que el estudiante tenga los conocimientos básicos en el área de programación, que le permitan elaborar aplicaciones interactivas que brinden soluciones a procesos, sistemas, productos y servicios a través de la adecuada selección de estructuras de código

OBJETIVOS

GENERALES

Conocer los conceptos basicos en el area de programación, con lo cual pueda desarrollar aplicaciones interactivas usando la adecuada seleccion de estructuras de programación.

TERMINALES

- Reconocer los fundamentos del pensamiento algorítmico y las características generales de un lenguaje de programación para la creación de procesos, sistemas, productos y servicios que dan solución a problemas de diseño de interacción.
- Construir algoritmos a través de la utilización de estructuras de selección y repetición para la creación de aplicaciones con interfaces digitales interactivas.
- Identificar el paradigma de programación orientado a objetos como mediadores de interacción en las interfaces digitales que apoyan procesos, sistemas, productos y servicios.

ESPECÍFICOS

Unidad 1. Introducción a HTML y CSS

- Reconocer las características y partes que debe tener una página web básica
- Usar los distintos elementos HTML para hacer productos web útiles
- Entender la relación entre un archivo HTML y uno CSS
- Reconocer los elementos CSS para poder ubicar elementos HTML en un proyecto de pantalla
- Usar CSS para poder cambiar la forma de elementos HTML
- Usar hojas de estilo para maquetar una página web

Unidad 2: Introducción al pensamiento algorítmico y Javascript

Reconocer las características de los lenguajes de programación. Especialmente Javascript
Utilizar diagramas en la creación de soluciones de programación
Identificar las variables existentes en un problema.
Seleccionar los tipos de datos necesarios para almacenar los datos relacionados a un problema.
Usar las operaciones, estados y valores necesarios para la solución de un problema.
Utilizar las diferentes estructuras de selección y decisión (if/ if-else/switch) para solucionar un problema.
Identificar los componentes básicos de las estructuras repetitivas, inicialización, condición, incremento.
Usar las estructuras de repetición while, do-while y for en la solución de problemas que impliquen la utilización de ciclos.
Reconocer los diferentes elementos que intervienen en las interfaces digitales.

Unidad 3: Eventos y Manejo del DOM

Entender qué es un evento del usuario y cómo usarlo para proporcionar interactividad
Identificar los elementos del DOM
Modificar el DOM en tiempo de ejecución a través del lenguaje de programación

Unidad 4: Programación Orientada a Objetos (POO) y modelo de datos

Reconocer los elementos fundamentales del paradigma de programación orientado a objetos.
Usar las estructuras asociadas a la Programación Orientada a Objetos
Modelar los datos que muestra una página web
Entender la serialización de datos usando JSON
Descargar JSON para mostrarlos en la página

Unidad 5: Herramientas de control de versión

Usar la herramienta GIT para seguir los cambios de un proyecto de programación
Almacenar repositorios de código en GitHub
Actualizar y Descargar repositorios de código
Trabajar en equipo de forma remota a través de GitHub

DE FORMACIÓN DE VALORES Y COMPETENCIAS

Trabajaremos con presentaciones en las cuales el estudiante podrá exponer como desarrollo los ejercicios y proyecto propuestos en la clase, esto con el fin de brindar al estudiante herramientas que le ayuden a comunicarse y expresar su forma de abordar un problema, skill importante en el mundo laboral.

METODOLOGÍA

Este curso tiene dos componentes, con igual importancia:

Componente Teórico: Consiste en la discusión de los diferentes temas y el hacer ejercicios que permitan ponerlos en práctica, esto en el salón de clase. Este componente se desarrolla en tres (3) horas por semana.

Componente Práctico: Consiste en la resolución de ejercicios y su implementación en en los 2 diferentes lenguajes de programación, esto en la sala de cómputo. Para esta componente se han destinado dos (2) horas por semana.

Antes de la clase

- Estudiar el material asignado en la programación del curso.
- Contestar las preguntas que contiene el material, así como las preguntas adicionales que el profesor entregue.

- Resolver los ejercicios propuestos en el material, así como los ejercicios adicionales que se le entreguen.
- Asistir a las sesiones de monitoria y tutoría, si requiere algún apoyo adicional.

De ninguna manera se considera preparar la clase leer a la carrera el material, justo antes de entrar al salón.

Durante la clase

- Plantear las dudas que le quedaron durante el proceso de estudio del tema a tratar.
- Participar en las actividades de revisión y consolidación de conceptos que proponga el profesor.
- Trabajar en la solución de los problemas de aplicación que se propongan. Después de la clase
- Establecer las relaciones entre los temas tratados en la clase y el conocimiento previamente adquirido.
- Resolver los ejercicios de aplicación del tema, que tienen un nivel de complejidad mayor al de los ejercicios que resolvió previamente.
- Si se trata de la clase que cierra la unidad debe, además, trabajar en el mini-proyecto de integración de conocimiento: identificar las relaciones entre los conceptos recién aprendidos y los conceptos previamente trabajados.

Este curso utilizará múltiples herramientas para desarrollar los objetivos propuestos, como entorno de desarrollo se usarán Visual Studio Code para Javascript, HTML y CSS, no obstante, los objetivos de aprendizaje son independientes del lenguaje y el entorno de programación seleccionado. Todos los grupos de Fundamentos de Programación para el Diseño tienen un conjunto de trabajos prácticos que buscan desarrollar en el alumno la capacidad de traducir a un lenguaje de programación la solución dada a diferentes problemas. Estos trabajos serán asignados a lo largo del semestre, y su solución debe incluir tanto análisis como código y deben ser entregadas a través de la plataforma en las fechas y horas especificadas. Adicionalmente, los profesores podrán asignar otras tareas complementarias que le deberán ser entregadas personalmente, en los horarios y fechas especificadas.

Al final del semestre se planteará un problema más complejo, el cual será resuelto en grupos (Proyecto Final)

Metodología específica:

Se trabajarán diferentes metodologías específicas durante el semestre que son base para el desarrollo de software e implementación de prototipos. Es muy importante apoyarse con asistencias a tutorías de donde podrán recibir oportunas retroalimentaciones del proceso y un adecuado acompañamiento.

ESQUEMA DE EVALUACIÓN:

Los cálculos para la nota definitiva se realizarán así:

Proyecto final. Entrega 1: 25%

Proyecto final. Entrega 2: 25%

Proyecto final. Entrega 3: 25%

Seguimientos: 25%

ESQUEMA DEL CURSO

Descripción	Comentarios	Peso
Proyecto final. Entrega 1	N/A	25%
Proyecto final. Entrega 2	N/A	25%
Proyecto final. Entrega 3	N/A	25%

Descripción	Comentarios	Peso
Seguimientos	N/A	25%

BIBLIOGRAFÍA

TEXTOS GUÍAS

OTROS

- Simpson, Kyle. "You don't know javascript".
- Haverbeke, Marijn. "Eloquent javascript a modern introduction to programming".
- Jiménez, Luz Elena. "Conceptos Básicos de Programación con Java". Tercera Edición - U. Icesi, 2005
- Casey Reas and Ben Fry. Processing. A programming handbook for visual designers and artists. MIT Press.
- Ejemplos de diseño visual: Rashid , Karim . "Digipop". Tashen 2004.
- Greenberg Ira. Processing: creative coding and computational art. Friends of ED Press. 2007
- Scolari, Carlos Alberto. Hacer clic: hacia una sociosemiótica de las interacciones digitales.
- Scolari, Carlos Alberto. Hipermediaciones: elementos para una teoría de la comunicación digital interactiva.

http://www.icesi.edu.co/departamentos/tecnologias_informacion_comunicaciones/cursos/09686/home/

- <https://developer.mozilla.org/es/docs/Web/JavaScript>

RECURSOS

Nombre	Descripción	Necesario para la clase
Editor de Texto	Pueden utilizar cualquiera de los disponibles para la plataforma que uds usen, tres de los mas populares son: VsCode, Atom y Sublime Text.	Si