

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

NOMBRE CURSO:	PROGRAMACIÓN
NIVEL:	BÁSICO
DURACIÓN TÉCNICA:	108 HORAS

DESCRIPCIÓN DEL CURSO	
MODALIDAD DE FORMACIÓN	VIRTUAL/PRESENCIAL
METODOLOGÍA	TUTORIZADA: VIRTUAL SINCRÓNICA
PERFIL DEL TUTOR / EJECUTOR:	El tutor debe tener un título en Ciencias de la Computación o campo relacionado, con al menos 3 años de experiencia en desarrollo web y programación en Python. Debe ser experto en HTML, CSS, JavaScript, y Flask, y tener habilidades pedagógicas para enseñar en modalidad virtual y presencial. Se valora experiencia en visualización de datos y capacidad para implementar estrategias metodológicas efectivas.

RESULTADOS DE APRENDIZAJE ESPERADOS
<ul style="list-style-type: none"> • Desarrollar sitios web estáticos con HTML, CSS, JavaScript. • Crear programas básicos en Python para visualización de datos. • Desarrollar sitios web con programas básicos de visualización de datos. • Contar con bases de programación para manejo de datos.

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

COMPETENCIAS

Al finalizar el curso de programación nivel básico, el estudiante será capaz de diseñar, desarrollar e implementar aplicaciones web estáticas y dinámicas, integrando tecnologías de front-end (HTML, CSS, JavaScript) y back-end (Python con Flask) para la visualización de datos, demostrando comprensión y aplicación de los fundamentos de programación y manejo de datos.

Competencias:

- Aplicar los fundamentos de la programación para desarrollar sitios web estáticos utilizando HTML, CSS y JavaScript, comprendiendo la lógica de programación y estructuras de control básicas
- Desarrollar programas en Python para la manipulación y visualización de datos, utilizando estructuras de datos y funciones adecuadas, así como bibliotecas de visualización de datos como Matplotlib y Seaborn.
- Integrar aplicaciones web dinámicas utilizando Flask y visualizar datos en la web, combinando conocimientos de HTML, CSS, JavaScript y Python para desarrollar un proyecto web completo.

ESTRATEGIAS METODOLÓGICAS

Aprendizaje Basado en Proyectos

Los estudiantes trabajan en proyectos prácticos desde el inicio del curso. Cada módulo incluye proyectos incrementales que culminan en un proyecto final integrador. La implementación de la estrategia para cada módulo se realizaría así:

- **Módulo 1:** Crear un sitio web personal utilizando HTML, CSS y JavaScript.
- **Módulo 2:** Desarrollar un programa en Python que realice análisis y visualización de datos de un conjunto de datos específico.
- **Módulo 3:** Desarrollar una aplicación web que integre las habilidades de front-end y back-end para mostrar visualizaciones de datos dinámicas

Inmersión Total y Aprendizaje Intensivo

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

El curso se diseña para ser intensivo y requiere un compromiso total de los estudiantes, similar a los bootcamps. Las sesiones son diarias y se combinan con tareas y prácticas intensivas, de la siguiente manera:

Implementación

- **Horario:** Sesiones de clases teóricas, seguidas con horas de práctica guiada y trabajo en proyectos.
- **Sprint Semanal:** Cada semana se enfoca en un tema o conjunto de habilidades específicas, con un mini-proyecto al final de la semana para consolidar el aprendizaje.

Aprendizaje Colaborativo

Se fomenta la colaboración entre los estudiantes a través de proyectos en equipo y actividades de pair programming. Esta estrategia se implementa a través de:

- **Proyectos en Equipo:** Algunos proyectos se diseñan para ser realizados en equipos, promoviendo la colaboración y la comunicación.
- **Pair Programming:** Actividades regulares donde los estudiantes trabajan en parejas para resolver problemas de programación.
- **Hackathons y Coding Challenges:** Eventos especiales donde los estudiantes colaboran para resolver problemas o desarrollar aplicaciones en un tiempo limitado.

Aprendizaje Basado en Problemas

Los estudiantes aprenden resolviendo problemas complejos y reales que requieren la aplicación de múltiples conceptos y habilidades. Esta se puede implementar a través de:

- **Desafíos Semanales:** Presentar un problema al inicio de la semana que los estudiantes deben resolver utilizando las habilidades adquiridas hasta ese momento.
- **Grupos de Discusión:** Facilitar grupos de discusión donde los estudiantes pueden compartir sus enfoques y soluciones al problema.
- **Presentaciones de Soluciones:** Al final de la semana, los estudiantes presentan sus soluciones y discuten los diferentes enfoques utilizados.

MATERIAL PEDAGÓGICO POR MÓDULO

MATERIAL DIDÁCTICO

Módulo 1: fundamentos de programación:

1. Presentación introductoria sobre programación.
2. Tutoriales básicos en HTML y CSS.
3. Ejercicios prácticos para crear una página web simple
4. Diapositivas sobre la estructura y elementos HTML.
5. Ejemplos de código HTML.
6. Ejercicios para practicar la creación de documentos HTML con enlaces e imágenes.
7. Presentación sobre sintaxis y selectores CSS.
8. Ejemplos de código CSS aplicados a documentos HTML.
9. Ejercicios para practicar el uso de CSS en la estilización de páginas web.

Módulo 2: Programación en Python y Visualización de Datos

1. Guía de instalación de Python y herramientas (Anaconda, Jupyter Notebook).
2. Ejemplos de código básicos. Ejercicios prácticos.
3. Presentación sobre estructuras de datos.
4. Ejemplos de código para cada estructura de datos.
5. Presentación sobre control de flujo.
6. Ejemplos de código de condicionales, bucles y funciones.
7. Presentación sobre gráficos básicos.
8. Ejemplos de código para gráficos de línea, barra y dispersión.

Módulo 3: Integración de Python con Desarrollo Web

1. Presentación en PowerPoint sobre Flask.
2. Tutorial paso a paso de instalación de Flask.
3. Ejemplo de código básico de Flask.

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

4. Presentación sobre rutas y plantillas en Flask.
5. Código de ejemplo con rutas y plantillas.
6. Presentación sobre formularios y validación

RECURSOS EDUCATIVOS GENERALES:

Plataformas Online:

- Documentación oficial de Flask.
- Documentación oficial de SQLAlchemy.
- Documentación oficial de Matplotlib y Plotly.

Libros Recomendados:

- "Flask Web Development" de Miguel Grinberg.
- "Python Data Science Handbook" de Jake VanderPlas.

Comunidades y Foros en línea:

- Stack Overflow.
- Reddit (r/flask, r/learnpython).
- GitHub para compartir y colaborar en código.

ESTRUCTURA DEL CURSO

Módulo 1: Fundamentos de Programación

Nombre de la lección	Subtemas
Introducción a la Programación	<ul style="list-style-type: none"> • ¿Qué es la programación? • Algoritmos y lógica de programación

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

		<ul style="list-style-type: none"> • Variables y tipos de datos • Operadores y estructuras de control (if, else, for, while)
Introducción a HTML y CSS		<ul style="list-style-type: none"> • Estructura básica de un documento HTML • Etiquetas y elementos fundamentales (<html>, <head>, <body>, <p>, <h1>-<h6>, <a>,) • HTML semántico • Introducción a CSS: sintaxis y selectores • Selectores y propiedades CSS (color, font-size, margin, padding etc)
Introducción a JavaScript		<ul style="list-style-type: none"> • Conceptos básicos de JavaScript y su integración en HTML • Variables y tipos de datos en JavaScript • Operadores y estructuras de control (if, else, for, while) • Manipulación básica del DOM
Módulo 2: Programación en Python y Visualización de Datos		
Nombre de la lección	Subtemas	
Fundamentos de Python		<ul style="list-style-type: none"> • Instalación y configuración de Python • Sintaxis básica y estructura de un programa en Python • Variables y tipos de datos en Python • Operadores y estructuras de control (if, else, for, while)

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

Estructuras de Datos y Funciones en Python	<ul style="list-style-type: none"> Listas, tuplas y diccionarios Definición y uso de funciones Parámetros y retorno de valores en funciones Manejo de excepciones
Visualización de Datos con Python	<ul style="list-style-type: none"> Introducción a Matplotlib y Seaborn Creación de gráficos básicos (líneas, barras, dispersión) Personalización de gráficos (etiquetas, títulos, leyendas)
Módulo 3: Integración de Python con Desarrollo Web	
Nombre de la lección	Subtemas
Desarrollo Web con Flask	<ul style="list-style-type: none"> Introducción a Flask Creación de una aplicación web simple con Flask Renderización de plantillas HTML con Flask Envío de datos de Python a HTML
Visualización de Datos en la Web	<ul style="list-style-type: none"> Incorporación de gráficos creados con Python en un sitio web Uso de Chart.js y D3.js para visualización de datos Creación de gráficos interactivos Enlace entre gráficos de Python y JavaScript
Proyecto Integrador Final	<ul style="list-style-type: none"> Diseño y planificación de un proyecto que combine HTML, CSS, JavaScript y Python

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

	<ul style="list-style-type: none"> Implementación de una aplicación web que muestre y visualice datos dinámicamente Pruebas y depuración del proyecto Documentación y presentación del proyecto final
--	--

INTENSIDAD HORARIA

Módulo	Lección	Duración (hrs)
Módulo 1: Fundamentos de Programación	Introducción a la programación	12
	Introducción a HTML y CSS	10
	Introducción a JavaScript	12
Módulo 2: Programación en Python y Visualización de Datos	Fundamentos de Python	12
	Estructuras de Datos y Funciones en Python	10
	Visualización de datos con python	10
Módulo 3: Integración de Python con Desarrollo Web	Desarrollo Web con Flask	12
	Visualización de Datos en la Web	15
	Proyecto Final Integrador	15
TOTAL		108

RÚBRICA DE EVALUACIÓN GENERAL DEL CURSO

COMPETENCIA	ESCALA DE CALIFICACIÓN		
	NO CUMPLE	A MEJORAR	CUMPLE
	01- 30	31-89	90-100

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

Diseñar, desarrollar e implementar aplicaciones web estáticas y dinámicas, integrando tecnologías de front-end (HTML, CSS, JavaScript) y back-end (Python con Flask) para la visualización de datos, demostrando comprensión y aplicación de los fundamentos de programación y manejo de datos.	Debe darse alguna de las siguientes dificultades en el conocimiento:	Deben mejorarse, algunos o varios de los siguientes conocimientos parciales:	Deben cumplirse a satisfacción uno o varios de los siguientes conocimientos:
	Dificultad para comprender y aplicar conceptos básicos como variables, tipos de datos y estructuras de control. -	Comprende parcialmente conceptos básicos, pero presenta errores frecuentes. -	Comprende y aplica conceptos fundamentales sin dificultades. -
	Incapacidad para resolver problemas básicos.	Necesita apoyo constante para resolver problemas básicos.	Resuelve problemas básicos de forma autónoma y correcta.
	Incapacidad para estructurar páginas web con HTML. - Dificultad para aplicar estilos CSS de manera coherente.	Estructura páginas web básicas, pero con errores en el uso de etiquetas y atributos. - Aplica estilos CSS básicos, pero sin coherencia visual.	Diseña y estructura páginas web utilizando HTML de manera correcta. - Aplica estilos CSS de forma coherente y atractiva visualmente.
	Dificultad para entender y aplicar conceptos básicos de JavaScript. - Incapacidad para manipular el DOM adecuadamente.	Aplica conceptos básicos de JavaScript con errores. - Manipula el DOM de forma limitada y con errores frecuentes.	Implementa funcionalidades interactivas utilizando JavaScript correctamente. - Manipula el DOM de manera efectiva y sin errores.
	Dificultad para comprender y	Comprende y aplica conceptos	Comprende y aplica

FORMATO SYLLABUS

PERTENECE AL PROCEDIMIENTO: IDENTIFICACIÓN DEL CURSO

CLASIFICACIÓN: USO INTERNO

	<p>aplicar sintaxis y estructuras básicas de Python.
- Incapacidad para manejar estructuras de datos simples.</p>	<p>básicos de Python con errores frecuentes.
- Maneja estructuras de datos simples, pero necesita mejorar en su manipulación.</p>	<p>correctamente la sintaxis y estructuras básicas de Python.
- Maneja eficientemente estructuras de datos simples.</p>
	<p>Incapacidad para crear y personalizar gráficos con bibliotecas como Matplotlib y Seaborn.
- Dificultad para interpretar y representar datos visualmente.</p>	<p>Crea gráficos básicos, pero con errores en la personalización.
- Necesita mejorar en la interpretación y representación de datos.</p>	<p>Crea y personaliza gráficos correctamente utilizando Matplotlib y Seaborn.
- Interpreta y representa datos visualmente de forma efectiva.</p>
	<p>Dificultad para comprender y aplicar conceptos básicos de Flask.
- Incapacidad para renderizar plantillas HTML y manejar datos en el back-end.</p>	<p>Aplica conceptos básicos de Flask con errores frecuentes.
- Renderiza plantillas HTML de manera limitada y con errores en el manejo de datos.</p>	<p>Desarrolla aplicaciones web básicas utilizando Flask correctamente.
- Renderiza plantillas HTML y maneja datos en el back-end de forma efectiva.</p>
	<p>Incapacidad para planificar y desarrollar un proyecto web completo.
- Dificultad para integrar tecnologías de frontend y backend.</p>	<p>Planifica y desarrolla el proyecto con errores significativos.
- Integra tecnologías de front-end y backend de forma limitada y con errores.</p>	<p>Planifica, desarrolla e implementa un proyecto web completo integrando correctamente tecnologías de frontend y backend.</p>