

PROGRAMA DE LABORATORIO

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS



INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2 - SECCIÓN N

CÓDIGO:	0771	PUNTEO NETO LABORATORIO:	6 CRÉDITOS CLAR
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	770- Introducción a la Programación y Computación I 107 - Matemática Intermedia 1 795 - Lógica Matemática 960 - Matemática de Cómputo 1	POST REQUISITO:	777 - Org. Lenguajes y Compiladores 1 964- Org. Computacional 772 - Estructura de Datos 601 - Inv. de operaciones 1
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	PRIMER SEMESTRE 2026

Descripción del Laboratorio

El laboratorio del curso 'Introducción a la programación y computación 2' está diseñado para complementar la teoría impartida en clase, mediante la aplicación práctica de los conceptos fundamentales. Su propósito es fomentar el aprendizaje activo, el pensamiento computacional y el desarrollo de habilidades para la resolución de problemas usando métodos algorítmicos y herramientas actuales.

Resumen de Ponderaciones y Tiempo de Auto-aprendizaje

TIPO	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Actividades en Clase	0	0
Proyectos	90	300
Prácticas	0	223
Tareas	0	0
Examen Final	10	0
TOTAL	100	523

Equipo Académico

Coordinador del Área

Nombre Marlon Francisco Orellana Lopez	Correo Electrónico marlonorellana2005@gmail.com
--	---

Docente

Nombre Edwin Estuardo Zapeta Gómez	Correo Electrónico estuardo.zapeta@gmail.com
--	--

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día				X	X	
Horario				17:20 - 19:00	17:20 - 19:00	
Lugar				Meet	Meet	

Tutor(es)

Nombre	Tobías Rafael Zamora Santos	
Correo Electrónico Institucional	2921019450101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20 - 19:0				
	Lugar		Meet				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Índice

Descripción del Laboratorio.....	1
Resumen de Ponderaciones y Tiempo de Auto-aprendizaje.....	1
Equipo Académico.....	2
Coordinador del Área.....	2
Docente.....	2
Tutor(es).....	2
Competencias Vinculadas al Perfil del Egresado.....	4
Competencias Específicas.....	4
Competencias Generales.....	4
Competencias del Laboratorio.....	5
Competencia(s) Específica(s).....	5
Competencia(s) General(es).....	5
Diseño Didáctico.....	6
Sesión de Diagnóstico.....	6
Sesión No. 2, Unidad No. 1 - Introducción a C#.....	7
Sesión No. 3, Unidad No. 2 - Introducción a C#, Manejo de versiones.....	8
Sesión No. 4, Unidad No. 3 - Procesamiento de datos XML.....	9
Sesión No. 5, Unidad No. 4 - Estructura De Datos.....	10
Sesión No. 6, Unidad No. 4 - Estructura De Datos.....	11
Sesión No. 7, Unidad No. 4 - Estructura De Datos.....	12
Sesión No. 8, Unidad No. 5 - HTML5 y CSS.....	13
Sesión No. 9, Unidad No. 6 - Entorno para desarrollo Web en C#.....	14
Sesión No. 10, Unidad No. 7 - JavaScript Object Notation (JSON), Unidad No. 8 - Acceso a datos web.....	15
Sesión No. 11, Unidad No. 8 - Acceso a datos web.....	16
Rúbrica de Evaluación.....	17
Normativa Académica y Ética del Curso.....	17
Bibliografía.....	18

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	Demuestra pensamiento crítico, actitud investigativa y rigor analítico en el planteamiento y la resolución de problemas complejos.
2	Aplica los conocimientos de su disciplina en la elaboración, fundamentación y defensa de argumentos para prevenir y resolver problemas complejos en su campo profesional, identificando y aplicando innovaciones.
3	Demuestra destreza y habilidad en la selección, uso y adaptación de herramientas metodológicas, tecnológicas, equipos especializados y en la lectura e interpretación de datos, pertinentes al contexto de su ejercicio profesional.
4	Toma decisiones profesionales con base en fundamentos teóricos, datos e información pertinente, válida y confiable.
5	Identifica oportunidades y riesgos para la innovación y adaptación de conocimientos y tecnologías para resolver problemas.

Competencias Generales

No.	Competencia
1	Aplica principios básicos de ingeniería, ciencias de computación y sistemas de información y comunicación, en la formulación y resolución adecuada de problemas complejos.
2	Maneja e Interpreta adecuadamente datos masivos, sean estos estructurados o no estructurados, facilitando su visualización e interpretación de forma eficaz en apoyo a la toma de decisiones.
3	Aplica conocimientos tecnológicos con ética profesional, respetando y cuidando los recursos naturales, humanos y financieros.
3	Aplica estándares de calidad, eficiencia y seguridad en la implementación adecuada de soluciones de software, hardware y TIC en general.
4	Actualiza permanentemente sus conocimientos relacionados con TIC en general, apoyándose en las estrategias de aprendizaje apropiadas.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en C# e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas.	Crear
2	Desarrolla aplicaciones web con Flask o Django en C# para soportar arquitecturas backend y frontend bien definidas.	Desarrollar
3	Implementa comunicación cliente-servidor mediante protocolos HTTP/HTTPS para desarrollar APIs seguras y eficientes.	Implementar
4	Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Crear
5	Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	Aplicar
6	Aplica tuplas, listas, diccionarios nativos de C# en programas funcionales para almacenar y manipular conjunto de datos.	Aplicar

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Aplica técnicas de control de versiones, como Git Flow, branching, merging y versionamiento semántico en GIT para asegurar la calidad del software mediante un historial de cambios estructurado.	Aplicar
2	Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	Aplicar

Diseño Didáctico

Sesión de Diagnóstico

Evaluación de conocimientos previos

Se aplicará una actividad diagnóstica con el objetivo de identificar el nivel de conocimientos y habilidades que los estudiantes poseen al inicio del curso. No influye en la nota final, pero es obligatoria para todos los estudiantes.

Tipo de Actividad	Descripción
Cuestionario interactivo	Se utilizarán plataformas como kahoot o quizziz para evaluar el conocimiento previo de los estudiantes.

Presentación del tutor

El tutor se presenta formalmente al grupo, compartiendo su formación académica, experiencia profesional y educativa, así como sus expectativas sobre el curso. También se abordan aspectos como normas de convivencia, canales de comunicación, disponibilidad para consultas y métodos de acompañamiento.

Presentación de los estudiantes

Se escogen un grupo de estudiantes al azar. En su presentación, se les pedirá que compartan información básica como su nombre, intereses personales o profesionales, experiencias previas relacionadas con el curso y sus expectativas. Esta actividad busca promover la interacción, el reconocimiento entre pares y la construcción de un entorno participativo y respetuoso.

Presentación del programa del curso

Se presenta el contenido del programa del curso, se aclaran dudas y se fomenta el compromiso del estudiante con su aprendizaje.

Evaluación de conocimientos del laboratorio actual

Se realiza una evaluación o práctica que permite conocer el grado de familiaridad de los estudiantes con las herramientas, entornos o competencias técnicas necesarias para el laboratorio actual.

Tipo de Actividad	Descripción
Ejercicios prácticos	Ejercicios que engloban los temas vistos en el laboratorio.

Sesión No. 2, Unidad No. 1 - Introducción a C#

Área Actitudinal (Saber ser)

Nombre Del Valor: Responsabilidad
<p>"No se puede escapar de la responsabilidad del mañana evadiéndola hoy."</p> <p>La responsabilidad fomenta que el estudiante cumpla con las tareas asignadas, entregue sus actividades en el tiempo establecido y aproveche los recursos disponibles de manera efectiva.</p>

Área de Conocimiento (Saber)

Competencia(s)	
<p>Aplica los fundamentos de Python mediante la instalación del lenguaje, configuración del entorno de desarrollo y el uso de estructuras básicas (variables, condicionales, funciones e iteraciones) para la resolución de problemas iniciales de programación.</p>	
Tema	Subtema
Introducción y Presentación	Instalación de C#
Introducción y Presentación	Instalación del entorno de desarrollo
Introducción y Presentación	Variables, Expresiones y sentencias
Introducción y Presentación	Ejecuciones condicionales
Introducción y Presentación	Funciones
Introducción y Presentación	Iteración
Introducción y Presentación	Strings

Área de Habilidades (Saber Hacer)

Competencia	
Crea programas básicos en Python aplicando estructuras de control y funciones.	
Tipo de Actividad	Ponderación
Ejercicio	0

Sesión No. 3, Unidad No. 2 - Introducción a C#, Manejo de versiones

Área Actitudinal (Saber ser)

Nombre Del Valor: Responsabilidad
El estudiante cumple puntualmente con las actividades asignadas y utiliza de manera adecuada las herramientas tecnológicas (Python y Git) para garantizar el desarrollo correcto de la práctica.

Área de Conocimiento (Saber)

Competencia(s)	
Aplica los principios de la programación orientada a objetos y las prácticas de control de versiones mediante Git para estructurar, versionar y mantener proyectos de software de manera organizada y colaborativa.	
Tema	Subtema
Programación Orientada a Objetos en Python	Introducción al paradigma de POO
Programación Orientada a Objetos en Python	Clases y Objetos(Métodos y Atributos)
Programación Orientada a Objetos en Python	Pilares de la POO (Herencia, Abstracción, Polimorfismo, Encapsulamiento)
Git Workflow	Introducción a GIT y Control de versiones
Git Workflow	Fundamentos de control de versiones con Git y su Configuración Inicial para el versionamiento de proyectos
Git Workflow	Git Flow: Implementación de Troncales y Ramas
Git Workflow	Branching, Merging y Utilización de Etiquetas (Tags) en Git

Área de Habilidades (Saber Hacer)

Competencia	
Crea un proyecto en Python aplicando POO y lo versiona correctamente utilizando Git.	
Tipo de Actividad	Ponderación
Ejercicio	0

Sesión No. 4, Unidad No. 3 - Procesamiento de datos XML

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad y Puntualidad
Promover la asistencia, puntualidad y compromiso en la ejecución de actividades del laboratorio.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender y aplicar técnicas de procesamiento de datos XML utilizando diferentes librerías y métodos.	
Tema	Subtema
Lectura de actividades y XML	Introducción a bases de XML
Lectura de actividades y XML	Procesamiento de datos con librería XmlDocument
Lectura de actividades y XML	Procesamiento de datos con librería XDocument
Lectura de actividades y XML	Introducción XPathNavigator
Lectura de actividades y XML	Procesamiento de datos con librería XPathNavigator
Lectura de actividades y XML	Lectura XML
Lectura de actividades y XML	Escritura XML

Área de Habilidades (Saber Hacer)

Competencia	
Crea y manipula documentos XML.	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 5, Unidad No. 4 - Estructura De Datos

Área Actitudinal (Saber ser)

Nombre del valor: Precisión y pensamiento lógico
Fomentar la atención al detalle y la lógica en la resolución de problemas de programación.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender y aplicar estructuras de datos nativas de Python para organizar y procesar información de manera eficiente.	
Tema	Subtema
Tuplas, Listas y Diccionarios Nativos de Python	Introducción estructuras de datos
Tuplas, Listas y Diccionarios Nativos de Python	Listas nativas de C#
Tuplas, Listas y Diccionarios Nativos de Python	Ficheros
Tuplas, Listas y Diccionarios Nativos de Python	Diccionarios
Tuplas, Listas y Diccionarios Nativos de Python	Tuplas
Tuplas, Listas y Diccionarios Nativos de Python	Expresiones regulares
Tuplas, Listas y Diccionarios Nativos de Python	Introducción a TDA's

Área de Habilidades (Saber Hacer)

Competencia	
Crear y manipular estructuras de datos.	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 6, Unidad No. 4 - Estructura De Datos

Área Actitudinal (Saber ser)

Nombre del valor: Integridad
<p>La integridad se aplica cuando los ingenieros actúan con honestidad, asegurando que sus sistemas sean seguros y confiables. Evitan comprometer la calidad por presiones externas o atajos.</p> <p>"La integridad es hacer lo correcto, incluso cuando nadie está mirando." – C.S. Lewis</p>

Área de Conocimiento (Saber)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Tema	Subtema
TDA	Concepto Lista enlazada simple
TDA	Operaciones en lista enlazada simple
TDA	Concepto Lista enlazada doble
TDA	Operaciones en lista enlazada doble
TDA	Concepto de lista circular
TDA	Operaciones en listas circulares
TDA	Diferencia entre lista circular y no circular

Área de Habilidades (Saber Hacer)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 7, Unidad No. 4 - Estructura De Datos

Área Actitudinal (Saber ser)

Nombre del valor: Respeto
El respeto se aplica cuando valoran las opiniones y contribuciones de colegas y usuarios, creando un entorno inclusivo."El respeto por nosotros mismos guía nuestra moral; el respeto por los demás guía nuestras maneras." – Laurence Sterne

Área de Conocimiento (Saber)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Tema	Subtema
TDA y graphviz	Concepto de lista doblemente circular
TDA y graphviz	Operaciones básicas en lista doblemente circular
TDA y graphviz	Ventajas y comparación frente a otras listas
TDA y graphviz	Qué es Graphviz
TDA y graphviz	Cómo se usa Graphviz para TDA's
TDA y graphviz	Graficar TDA's con graphviz

Área de Habilidades (Saber Hacer)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 8, Unidad No. 5 - HTML5 y CSS

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Con perseverancia enfrentan desafíos técnicos con determinación hasta encontrar soluciones viables. No se rinden ante errores o problemas complejos en el desarrollo. "El éxito no es la ausencia de obstáculos, sino el coraje para empujar a través de ellos." – James Allen

Área de Conocimiento (Saber)

Competencia	
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	
Tema	Subtema
Introducción a HTML, CSS, JS	Introducción a HTML5
Introducción a HTML, CSS, JS	Estructura básica de HTML5
Introducción a HTML, CSS, JS	Componentes básicos de HTML5
Introducción a HTML, CSS, JS	Introducción a CSS
Introducción a HTML, CSS, JS	Estructura básica de CSS
Introducción a HTML, CSS, JS	Componentes básicos de CSS
Introducción a HTML, CSS, JS	Ejemplos prácticos

Área de Habilidades (Saber Hacer)

Competencia	
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 9, Unidad No. 6 - Entorno para desarrollo Web en C#

Área Actitudinal (Saber ser)

Nombre del valor: Honestidad
La honestidad es el compromiso de actuar con verdad, transparencia y rectitud, evitando engaños o manipulaciones en el trabajo realizado. "La honestidad es el primer capítulo en el libro de la sabiduría." – Thomas Jefferson

Área de Conocimiento (Saber)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tema	Subtema
Introducción a servidores web	Entorno para desarrollo Web en C#
Introducción a servidores web	.NET
Introducción a servidores web	Introducción a .NET y Configuración del Entorno
Introducción a servidores web	Rutas y Vistas
Introducción a servidores web	Razon Views
Introducción a servidores web	Manejo de Formularios y Datos de Usuario
Introducción a servidores web	Ejemplos prácticos

Área de Habilidades (Saber Hacer)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 10, Unidad No. 7 - JavaScript Object Notation (JSON), Unidad No. 8 - Acceso a datos web

Área Actitudinal (Saber ser)

Nombre del valor: Colaboración
La colaboración es el acto de trabajar conjuntamente con otros para alcanzar un objetivo común, compartiendo ideas y recursos de manera efectiva. "Si quieres ir rápido, ve solo. Si quieres llegar lejos, ve acompañado." – Proverbio africano

Área de Conocimiento (Saber)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tema	Subtema
Estructura y Manejo de Datos JSON	Sintaxis de JSON, JSON vs XML
Estructura y Manejo de Datos JSON	Tipos de datos JSON
Estructura y Manejo de Datos JSON	JSON Parse, Stringify, Objects, Arrays
Acceso y Manejo de Datos Web mediante HTTP en C#	Internet como origen de datos
Acceso y Manejo de Datos Web mediante HTTP en C#	Protocolo HTTP y Códigos de error
Acceso y Manejo de Datos Web mediante HTTP en C#	Peticiones HTTP en C# y librería en C# para realizar peticiones HTTP
Acceso y Manejo de Datos Web mediante HTTP en C#	Lectura de archivos binarios mediante peticiones HTTP y Análisis de XML en Peticiones

Área de Habilidades (Saber Hacer)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tipo de Actividad	Ponderación
Actividad	0

Sesión No. 11, Unidad No. 8 - Acceso a datos web

Área Actitudinal (Saber ser)

Nombre del valor: Autonomía
La autonomía es la capacidad de trabajar de manera independiente, tomando decisiones responsables y gestionando tareas sin supervisión constante. "La libertad significa responsabilidad; por eso la mayoría de los hombres le temen." – George Bernard Shaw

Área de Conocimiento (Saber)

Competencia	
Implementa comunicación cliente-servidor mediante protocolos HTTP/HTTPS para desarrollar APIs seguras y eficientes.	
Tema	Subtema
Interacción y Seguridad en APIs y Datos Web	Desplazamiento a través de los nodos
Interacción y Seguridad en APIs y Datos Web	Análisis JSON en Peticiones
Interacción y Seguridad en APIs y Datos Web	Interfaces de programación de aplicaciones
Interacción y Seguridad en APIs y Datos Web	API
Interacción y Seguridad en APIs y Datos Web	Seguridad y uso de API's
Interacción y Seguridad en APIs y Datos Web	Autenticación y autorización en API's
Interacción y Seguridad en APIs y Datos Web	Manejo de respuestas asíncronas y promesas

Área de Habilidades (Saber Hacer)

Competencia	
Implementa comunicación cliente-servidor mediante protocolos HTTP/HTTPS para desarrollar APIs seguras y eficientes	
Tipo de Actividad	Ponderación
Actividad	0

Rúbrica de Evaluación

Cada una de las actividades del laboratorio (proyectos, prácticas, tareas y otras) cuenta con una rúbrica de evaluación específica, la cual está detallada en el documento que se entrega al estudiante al momento de asignar la actividad. Estas rúbricas describen los criterios de evaluación, niveles de desempeño esperados y la ponderación correspondiente de cada aspecto evaluado.

Es **responsabilidad del estudiante** leer detenidamente la rúbrica asignada antes de iniciar el desarrollo de la actividad. Comprender los criterios de evaluación no solo permite orientar adecuadamente el trabajo, sino también mejorar el desempeño académico y fomentar la autorregulación del aprendizaje.

En caso de no recibir la rúbrica al momento de la asignación, el estudiante **debe solicitarla directamente al tutor académico**, ya que constituye una herramienta esencial para el cumplimiento de los objetivos de aprendizaje y la evaluación transparente.

Normativa Académica y Ética del Curso

En concordancia con el perfil del estudiante de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, se espera un alto nivel de compromiso con la excelencia académica y la ética profesional. Por ello, que se establece los siguientes lineamientos de carácter obligatorio que regulan el comportamiento académico del estudiante:

Plagio y copias

- Todo proyecto será sometido a verificación para confirmar su autoría y originalidad, con la finalidad de evitar cualquier plagio, copia o que la actividad no haya sido realizada por el estudiante.
- Cualquier evidencia de lo antes descrito en las distintas actividades será sancionada con una calificación de 0 (cero) y el caso será reportado al Docente quien a su vez informará a la Escuela de Ciencias y Sistemas para su seguimiento institucional.

Prórrogas y reposiciones

- No se otorgarán prórrogas para entregas de actividades.
- No se permitirá la reposición de proyectos bajo ninguna circunstancia.

Requisitos para evaluación final del curso

- Es obligatorio aprobar el laboratorio para tener derecho a la evaluación final del curso.
- La calificación de prácticas, proyectos y otras actividades que se indique será asignada de forma presencial, en la fecha y hora establecidas por el tutor académico.

Asistencia

- Para obtener la nota del laboratorio, se requiere un mínimo del 80% de asistencia a las sesiones de laboratorio.
- En caso de inasistencia, sólo se aceptarán justificaciones válidas respaldadas por constancia oficial.

Entregas

- No se aceptarán entregas tardías de tareas, prácticas, exámenes cortos, exámenes finales o proyectos sin justificación.

Medio oficial de entrega

La plataforma UEDI de la Facultad será el único medio oficial para la entrega de actividades del curso.

Bibliografía

- AprendeAProgramar.com. (s. f.). *Introducción a XML*.
https://www.aprendeaprogramar.com/index.php?option=com_content&view=article&id=483
- Atlassian. (s. f.). *Comandos básicos de Git*.
<https://www.atlassian.com/es/git/glossary#commands>
- Código Facilito. (2015, febrero 22). *MVC (Model, View, Controller) explicado*.
<https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- DesarrolloWeb.com. (s. f.). *Manual de XML*.
<https://desarrolloweb.com/manuales/manual-xml.html>
- Django Project. (s. f.). *Django documentation*.
<https://docs.djangoproject.com/en/5.2/>
- GeeksforGeeks. (2025, abril 8). *HTTP full form Hypertext Transfer Protocol*.
<https://www.geeksforgeeks.org/http-full-form/>
- GeeksforGeeks. (s. f.). *Introduction to circular doubly linked list*.
<https://www.geeksforgeeks.org/dsa/introduction-to-circular-doubly-linked-list/>
- GitLab. (s. f.). *Control de versiones*.
<https://about.gitlab.com/es/topics/version-control/>
- Graphviz. (s. f.). *Graphviz - Graph visualization software*.
<https://graphviz.org/>
- Llamas, L. (s. f.). *Curso C# y .NET Aprende C#, el lenguaje más potente y versátil*.
<https://www.luisllamas.es/curso-csharp/>
- Lucena, P. (2023, mayo 6). *¿Qué es el framework? Maestrías y MBA*.
<https://www.cesuma.mx/blog/que-es-el-framework.html>
- MDN Web Docs. (2025, marzo 21). *Códigos de estado de respuesta HTTP*. Mozilla Foundation.
<https://developer.mozilla.org/es/docs/Web/HTTP/Reference/Status>
- MDN Web Docs. (s. f.). *HTML: Lenguaje de etiquetas de hipertexto*. Mozilla Foundation.
<https://developer.mozilla.org/es/docs/Web/HTML>
- MDN Web Docs. (s. f.). *MDN Web Docs*. Mozilla Foundation.
<https://developer.mozilla.org/es/>
- MDN Web Docs. (s. f.). *XML*. Mozilla Foundation.
<https://developer.mozilla.org/es/docs/Web/XML>
- Muñoz, J. D. (2017, noviembre 17). *Qué es Flask y ventajas que ofrece*. OpenWebinars.
<https://openwebinars.net/blog/que-es-flask/>
- Olanrewaju, T. (2024, diciembre 10). *How Django's MVT architecture works: A deep dive into models, views, and templates*. freeCodeCamp.
<https://www.freecodecamp.org/news/how-django-mvt-architecture-works/>
- OpenAPI Initiative. (s. f.). *OpenAPI Initiative*.
<https://www.openapis.org/>
- Platzi. (2018, febrero 21). *Qué es Frontend y Backend: principal diferencia y ejemplos*.
<https://platzi.com/blog/que-es-frontend-y-backend/>
<https://docs.python.org/3/faq/general.html#what-is-python>
- TechKrowd. (s. f.). *Lenguaje HTML - HTML en español*.
<https://lenguajehtml.com/>
- TechKrowd. (s. f.). *Python tipos de datos*.
https://techkrowd.com/programacion/python/python-tipos-de-datos/#google_vignette
- W3Schools. (s. f.). *Linked lists types*.
https://www.w3schools.com/dsa/dsa_data_linkedlists_types.php