



| CLASIFICACIÓN DE LA INFORMACIÓN |   |                     |  |                   |  |
|---------------------------------|---|---------------------|--|-------------------|--|
| Pública                         | X | Pública Clasificada |  | Pública Reservada |  |

## INFORME MENSUAL EJECUCIÓN CONTRACTUAL

Bogotá D.C julio de 2026

Señora

**SONIA ROCIO REY HORTA**

Supervisor(a) contrato nro. **CO1.PCCNTR.9167582**

Coordinadora De Formación Profesional

Dependencia Centro de Diseño y Metrología

Bogotá

**Asunto:** Informe mensual de ejecución contractual julio de 2026

**Referencia:** No CO1.PCCNTR.9167582 del año 2026

Jheyson Fabian Villavisan Buitrago, identificado con la cédula de ciudadanía nro. 1.012.355.635, en mi calidad de contratista del SENA, en cumplimiento del Contrato de Prestación de Servicios de la referencia, a continuación, presento el Informe de actividades realizadas en el mes objeto de cobro.

**Valor y forma de Pago:** TREINTA Y SIETE MILLONES OCHOCIENTOS NOVENTA Y NUEVE MIL NOVECIENTOS SETENTA Y SEIS PESOS (\$37.899.976,00) MCTE. Para cada contrato. Esta suma será pagada por el SENA al contratista de la siguiente manera: a) Ocho (8) pagos iguales correspondientes a los meses de febrero y septiembre de 2026 por valor de CUATRO MILLONES SETECIENTOS TREINTA Y SIETE MIL CUATROCIENTOS NOVENTA Y SIETE PESOS (\$ 4.737.497,00) M/CTE., cada uno.

Plazo: Será hasta el 30 de septiembre de 2026.

**Plazo:** Será hasta el 30 de septiembre de 2026.

**Objeto:** Prestar servicios profesionales de carácter temporal en formación profesional integral de acuerdo con la red de conocimiento y/o área temática en los niveles de formación titulada, modalidad presencial, para Articulación con Educación Media, programada por el Centro de Diseño y Metrología para la vigencia 2026.



## Ejecución mensual de actividades

| Nro. | Obligaciones                                    | Acciones realizadas                                                                                                                                                                                                                                                                                                                                                                                                        | Evidencias                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------|-------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1    | <b>Ejecución de la Formación</b>                | <p>-Se diseña y ejecuta las guías de “arreglos y matrices” y “Estructuras de control y cíclicas” para la ficha # 3450403 grado decimo Colegio Magdalena Ortega Nariño.</p> <p>-Se diseñan y ejecutan las guías de Django: Formularios, Autenticaciones, Relaciones Many to Many y Busquedas y Paginación. Para las fichas # 3191267 y #3192453 grado once Colegio República del Ecuador y colegio República de Panamá.</p> | <p>Enlace de la guía “Arreglos y Matrices”: <a href="#">guia_arreglos_y_matricez.pdf</a></p> <p>Enlace de la guía “Estructuras de control y cíclicas”: <a href="#">guia_estructura_de_control_y_ciclicas.pdf</a></p> <p>Enlace de las guías “Django-Formularios”:<br/><a href="#">GFPI-F-135FormatoGuiadeAprendizaje - Guia 05 - Python Django - Formularios.pdf</a></p> <p>Enlace de las guías “Django-Autenticaciones”:<br/><a href="#">GFPI-F-135FormatoGuiadeAprendizaje - Guia 06 - Python Django - Autenticacion.pdf</a></p> <p>Enlace de las guías “Django-Relaciones ManytoMany”:<br/><a href="#">GFPI-F-135FormatoGuiadeAprendizaje - Guia 08 - Python Django - Relaciones Many to Many (2).pdf</a></p> <p>Enlace de las guías “Busquedas y Paginación”:<br/><a href="#">GFPI-F-135FormatoGuiadeAprendizaje - Guia 07 - Python Django - Busquedas y Paginacion (1).pdf</a></p> |
| 2    | <b>Evaluación y Seguimiento del Aprendizaje</b> | -Se realiza seguimiento de asistencias de las fechas 8, 13, 15, 22 y 29 de julio de 2026 de                                                                                                                                                                                                                                                                                                                                | Toma de lista de asistencia de los aprendices colegio República de Ecuador.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |



|   |                                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                                         | <p>la ficha # 3191267 grado once Colegio Republica del Ecuador.</p> <p>-Se realiza seguimiento de asistencias de las fechas 9, 14, 16, 23 Y 30 de julio de 2026 de la ficha # 3192453 grado once Colegio República de Panamá.</p> <p>- Se realiza seguimiento de asistencias de las fechas 10, 17, 24, 27 Y 31 de julio de 2026 de la ficha # 3450403 grado decimo Colegio Magdalena Ortega Nariño.</p>                                                                                          | <p>Enlace: <a href="#">01-FORMATO DE ASISTENCIA-JULIO ECUADOR.xlsx</a></p> <p>Toma de lista de asistencia de los aprendices colegio República de Panamá. Enlace: <a href="#">01-FORMATO DE ASISTENCIA-JULIO PANAMÁ.xlsx</a></p> <p>Toma de lista de asistencia de los aprendices colegio Magdalena Ortega de Nariño. Enlace: <a href="#">01-FORMATO DE ASISTENCIA-JULIO MAONA.xlsx</a></p>                                                                                              |
| 3 | <b>Inducción y Acompañamiento</b>       | <p>-Se realiza la gestión, revisión, ajustes y firma del formato bitácora de los aprendices del grado once del colegio República del Ecuador con número de ficha #3191267.</p> <p>Se realiza la gestión, revisión, ajustes y firma del formato bitácora de los aprendices de la ficha # 3192453 grado once Colegio República de Panamá.</p> <p>Se diligencia el formato 165 selección etapa productiva de los aprendices de la ficha # 3450403 grado decimo Colegio Magdalena Ortega Nariño.</p> | <p>Enlace Evidencia de creación de carpetas de proyecto productivo en repositorio: <a href="#">01. FICHA 3191267 11°</a></p> <p>Enlace Evidencia de creación de carpetas de proyecto productivo en repositorio: <a href="#">01. FICHA 3192453 11°</a></p> <p>Enlace Evidencia de creación de carpetas de proyecto productivo en repositorio: <a href="#">GFPI-F-165FormatoSeleccionModificacionAlternativaEtapasProductiva (2).pdf</a></p> <p><a href="#">02. FICHA 3450403 10°</a></p> |
| 4 | <b>Gestión de Evidencias y Reportes</b> | <p>-Se documentan en el repositorio digital las evidencias de las actividades de aprendizaje para las fichas:</p> <p>* Colegio República de Ecuador ficha 3191267</p>                                                                                                                                                                                                                                                                                                                            | <p>Enlace a los repositorios digitales con las evidencias de las actividades de aprendizaje</p> <p><a href="#">Repositorio Evidencias Ecuador</a></p>                                                                                                                                                                                                                                                                                                                                   |



|   |                                                |                                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                            |
|---|------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                                                | * Colegio República de Panamá ficha #3192453                                                                                                                                                                                                                                                                                                                                                             | <a href="#">Repositorio Evidencias Panama</a>                                                                                                                                                                                                                                                              |
|   |                                                | * Colegio Magdalena Ortega Nariño                                                                                                                                                                                                                                                                                                                                                                        | <a href="#">RepositorioEvidencias MagdalenaOrtegaNariño</a>                                                                                                                                                                                                                                                |
| 5 | <b>Uso y Conservación de Recursos</b>          | No se ejecuta esta obligación durante este periodo                                                                                                                                                                                                                                                                                                                                                       | N/A                                                                                                                                                                                                                                                                                                        |
| 6 | <b>Cumplimiento de Normativa Institucional</b> | <p>- Se realizan las siguientes reuniones presenciales con el equipo de articulación según requerimiento de la coordinación de la articulación con la media:</p> <p>01 julio – Acta No 20 – “Reunión presencial – Capacitación pedagógica”</p> <p>02 julio – Acta No 21 – “Socialiación Guías y Anexos informe semestral”</p> <p>16 julio – Acta No 22 – “Revisión estado carpeta informe semestral”</p> | <p>Actas generadas de la reunión con firmas de participación</p> <p><a href="#">Acta 20 Fortalecimiento Pedagogico..pdf</a></p> <p><a href="#">Acta 21 Gestión de Reportes, Socialización Guías, Legalización Horas.pdf</a></p> <p><a href="#">GOR-F-084 Acta-22- Coordinacion de Procesos-V02.pdf</a></p> |

A continuación, relaciono los desplazamientos que realicé previo a la presentación de este informe. Una vez finalizado cada desplazamiento presenté al ordenador del gasto el informe en el Formato para legalización del desplazamiento, en el que se describieron las actividades desarrolladas y los resultados. Cada informe de legalización cuenta con el visto bueno del supervisor.

Se lista a continuación el soporte de la legalización de los desplazamientos realizados, los cuales forman parte integral del presente informe de ejecución contractual.

| ÍTEM | NRO. DE LA ORDEN DE VIAJE | LUGAR DE DESPLAZAMIENTO | FECHA DE DESPLAZAMIENTO INICIAL | FECHA DE DESPLAZAMIENTO FINAL |
|------|---------------------------|-------------------------|---------------------------------|-------------------------------|
| 1    |                           |                         |                                 |                               |
| 2    |                           |                         |                                 |                               |

Para el trámite de la cuenta me permito adjuntar: (i) Documentos electrónicos enunciados como evidencias del cumplimiento de las obligaciones contractuales, (ii) los desplazamientos realizados y (iii) el pago de la planilla de seguridad social y parafiscal nro. 9507003500 de la planilla referente al mes de junio.



Cordialmente,

A handwritten signature in black ink, appearing to read 'Jheyson'.

Jheyson Fabian Villavisan Buitrago  
Contratista  
C.C. No. 1.012.355.635

A handwritten signature in black ink, appearing to read 'Sonia Roció'.

Sonia Roció Rey Horta  
Supervisor del contrato CO1.PCCNTR.9167582 de 2026

A handwritten signature in black ink, appearing to read 'Jhon Alberto Altamar R.'.  
REVISADO POR: JHON ALBERTO ALTAMAR R.

## PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

### FORMATO GUÍA DE APRENDIZAJE

#### IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto: DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL. Fase del Proyecto: Diseño
- Actividad de Proyecto: Desarrollar el software de acuerdo con el diseño y a la plataforma de desarrollo.
- Competencia: 220501096 Desarrollar la solución de software de acuerdo con el diseño y metodologías de desarrollo.
- Resultados de Aprendizaje Alcanzar: 669169 - Rap 1. Resolver procesos lógicos a través de la implementación de algoritmos y el lenguaje de programación seleccionado.
- Duración de la guía (24-horas)

#### 2. PRESENTACIÓN

Hasta ahora, hemos trabajado con variables que almacenan un único dato a la vez. Sin embargo, la mayoría de los problemas del mundo real requieren manejar colecciones de datos: la lista de notas de un curso, los nombres de los jugadores de un equipo, los asientos de una sala de cine. Los **arreglos** son estructuras de datos que nos permiten almacenar un conjunto de datos del mismo tipo en una sola variable, de forma ordenada. Se dividen principalmente en **vectores** (unidimensionales, como una lista) y **matrices** (bidimensionales, como una tabla), y son fundamentales para escribir programas eficientes y potentes.

#### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

##### 3.1 Actividad de reflexión inicial.

En parejas definir realizar la siguiente actividad, siguiendo las siguientes reglas:

Cada jugador dibujará dos cuadrículas de 10x10 en su papel:

- "Mi Flota": En esta cuadrícula, el jugador posicionará sus barcos en secreto.
- "Radar Enemigo": Esta cuadrícula estará en blanco y se usará para marcar los disparos realizados al oponente.
- Las filas se numerarán del 0 al 9 y se identificarán con la letra "i" y las columnas también del 0 al 9 y se identificarán con la letra "j".
- Cada jugador deberá organizar su flota en la cuadrícula "Mi Flota". Usando los siguientes "barcos" (pueden representarlos con un código, por ejemplo, el número '1' para un barco):
- 1 Portaaviones: Ocupa 4 casillas ([ ] [ ] [ ] [ ])
- 2 Acorazados: Ocupan 3 casillas cada uno ([ ] [ ] [ ])
- 3 destructores: Ocupan 2 casillas cada uno ([ ] [ ])
- Los barcos solo pueden colocarse en horizontal o vertical, nunca en diagonal.

### Desarrollo de la actividad

El juego se desarrolla por turnos. El objetivo es "disparar" a las coordenadas del oponente para hundir su flota.

1. **¡Fuego!:** En su turno, un jugador "dispara" a una coordenada del oponente diciendo en voz alta una fila y una columna. Por ejemplo: "**¡Disparo a i3, j5!**".
2. **Procesando el Disparo:** El jugador que recibe el ataque debe mirar su cuadrícula "Mi Flota" en la coordenada [3][5] y responder:
  - "**¡Agua!**": Si en esa casilla no hay parte de un barco. Marcar con valor '0'.
  - "**¡Tocado!**": Si en esa casilla hay parte de un barco. Marcar con un valor '1'.
  - "**¡Hundido!**": Si el disparo acierta en la última parte intacta de un barco.
3. **Registrando los Resultados:**
  - El jugador que disparó anota el resultado en su "Radar Enemigo". Puede usar una '**0**' para Agua y una '**1**' para Tocado.
  - El jugador que recibió el disparo también marca la casilla en su propia flota para saber qué partes de sus barcos han sido alcanzadas.

### Fin del Juego

El primer jugador en **hundir todos los barcos** del oponente gana la batalla.

Tiempo estimado: 1 hora

## 3.2 Identificar los conocimientos necesarios para el uso de arreglos y matrices.

**Arreglo (Array):** Estructura de datos de tamaño fijo que almacena una colección de elementos del mismo tipo de dato. Los elementos se guardan en posiciones de memoria contiguas.

**Arreglo Unidimensional (Vector):** Es un arreglo que se puede visualizar como una lista o una fila de casillas.

Declaración: En PSeInt, se usa la palabra clave "*Dimension*".

Dimension nombre\_vector[tamaño];

**Arreglo Bidimensional (Matriz):** Es un arreglo que se puede visualizar como una tabla, cuadrícula o tablero, con filas y columnas.

Declaración:

Dimension nombre\_matriz[numero\_filas, numero\_columnas];

### 3.2.1 Actividades de apropiación del conocimiento (PSeInt) arreglos

Observe el siguiente video:

[https://www.youtube.com/watch?v=ywVUXaQ1N8s&list=PLaxZkGILWHGU8Es\\_SDHr8sLkxDUVPTz\\_N&ab\\_channel=DiscoDurodeRoer](https://www.youtube.com/watch?v=ywVUXaQ1N8s&list=PLaxZkGILWHGU8Es_SDHr8sLkxDUVPTz_N&ab_channel=DiscoDurodeRoer)

A continuación, implemente el siguiente algoritmo en PSeInt:

#### 3.2.1.1 Ejercicio Promedio de Notas (Vectores)

Crear un programa que almacene 5 notas en un vector, las sume y calcule el promedio.

2. Diseño (Pseudocódigo):

Proceso PromedioNotas

Dimension notas[5];

Definir suma, promedio Como Real;

Definir i Como Entero;

Para i <- 1 Hasta 5 Hacer

    Escribir "Ingrese la nota ", i, ":";

```

        Leer notas[i];
    FinPara
    suma <- 0;
    Para i <- 1 Hasta 5 Hacer
        suma <- suma + notas[i];
    FinPara
    promedio <- suma / 5;
    Escribir "El promedio de las notas es: ", promedio;
FinProceso

```

### 3.2.1.2 Ejercicio número mayor y menor:

Diseñe el diagrama de flujo y el pseudocódigo de un algoritmo de un programa que pida al usuario ingresar **10 números enteros**. El programa debe encontrar y mostrar cuál es el **número más grande** y cuál es el **número más pequeño** de los que se ingresaron.

### 3.2.1.3 Ejercicio usuarios

Crea un programa que primero almacene **8 nombres** en un arreglo. Luego, debe pedir al usuario que escriba un nombre para buscarlo en el arreglo. El programa debe indicar si el nombre **fue encontrado** o **no fue encontrado**.

**Problema:** Crea un sistema que administre un inventario de productos. Para ello, deberás utilizar tres arreglos paralelos con una capacidad para 10 productos:

1. nombresProductos (Tipo Caracter)
2. preciosProductos (Tipo Real)
3. stockProductos (Tipo Entero)

El índice de cada arreglo debe corresponder al mismo producto. Por ejemplo, los datos en la posición [2] (nombresProductos[2], preciosProductos[2] y stockProductos[2]) pertenecen todos al tercer producto del catálogo.

El programa debe presentar un menú con las siguientes opciones:

1. Agregar Producto: Permite al usuario ingresar el nombre, precio y stock de un nuevo producto en la primera posición disponible.
2. Mostrar Catálogo: Muestra una lista de todos los productos con su nombre, precio y stock.
3. Buscar Producto por Nombre: Pide un nombre al usuario y, si lo encuentra, muestra todos sus datos (nombre, precio y stock). Si no, informa que no existe.
4. Realizar Venta: Pide el nombre de un producto. Si lo encuentra y tiene stock ( $> 0$ ), reduce el stock en 1 y notifica la venta exitosa. Si no hay stock, debe informarlo.
5. Mostrar Productos Agotados: Recorre el inventario y muestra una lista de todos los productos cuyo stock es igual a 0.
6. Salir.

### 3.3.2 Actividades de apropiación del conocimiento (PSeInt) matrices

Observe el siguiente video: [https://www.youtube.com/watch?v=Wdu6CuMuq5c&ab\\_channel=MiltonJara](https://www.youtube.com/watch?v=Wdu6CuMuq5c&ab_channel=MiltonJara)

A continuación, implemente el siguiente algoritmo en PSeInt:

**3.2.2.1 Ejercicio tablas de multiplicar** Crea un programa que genere una matriz de 10x10. Cada celda de la matriz debe contener el resultado de multiplicar su número de fila por su número de columna (considerando que la primera fila/columna es la 1, no la 0). Al final, el programa debe imprimir la matriz completa en la pantalla, mostrando la tabla de multiplicar del 1 al 10.



3.2.2.2 Crea un programa que trabaje con una matriz de 6x6 llena de números enteros (pueden ser ingresados por el usuario o generados aleatoriamente). El programa debe hacer lo siguiente:

1. Mostrar la matriz completa.
2. Calcular y mostrar la suma de los valores de cada cuadrante. La matriz de 6x6 se puede dividir en cuatro cuadrantes de 3x3:
  - Cuadrante Superior Izquierdo: Filas 0-2, Columnas 0-2.
  - Cuadrante Superior Derecho: Filas 0-2, Columnas 3-5.
  - Cuadrante Inferior Izquierdo: Filas 3-5, Columnas 0-2.
  - Cuadrante Inferior Derecho: Filas 3-5, Columnas 3-5.

**Tiempo estimado:** 12 horas.

#### **4. ACTIVIDADES DE EVALUACIÓN (consolidación de conocimientos)**

De forma individual diseñe el pseudocódigo, el diagrama de flujo y posteriormente implementa en PSeInt el algoritmo que solucione el siguiente problema:

Una pequeña tienda necesita un programa para registrar las ventas diarias de 3 de sus productos durante una semana (de lunes a viernes, 5 días). El programa debe permitir ingresar las ventas de cada producto para cada día, y al final debe mostrar un reporte completo.

Tareas a Realizar:

1. Análisis y Diseño

Vectores: Necesitará un vector de 3 posiciones para guardar los nombres de los productos (ej: "Producto A", "Producto B", "Producto C").

Matrices: Necesitará una matriz de 3x5 para guardar las ventas. La fila 1 corresponderá al Producto A, la fila 2 al Producto B, etc. La columna 1 será el lunes, la 2 el martes, etc.

El programa debe usar ciclos anidados para pedir las ventas. El mensaje debe ser claro, ej: "Ingrese ventas del [Nombre del Producto] para el día [Número del día]:".

Deberá calcular: El total de ventas para cada producto (sumar todas las columnas de una misma fila). El total de ventas para cada día (sumar todas las filas de una misma columna).

El programa debe:

1. Pedir los 3 nombres de los productos y guardarlos en un vector.
2. Pedir las ventas para cada producto y cada día, guardándolas en la matriz.
3. Mostrar un reporte final que incluya: La tabla de ventas completa. El total de ventas por cada producto. El total de ventas por cada día.

**Tiempo estimado:** 8 horas.

#### **5. GLOSARIO DE TERMINOS**

- Arreglo (Array): Una estructura de datos que permite almacenar una colección de múltiples elementos del mismo tipo bajo un único nombre de variable.
- Vector (Arreglo Unidimensional): Un arreglo con una sola dimensión, que se puede imaginar como una lista, una fila o una columna de casillas.
- Matriz (Arreglo Bidimensional): Un arreglo con dos dimensiones (filas y columnas), que se puede imaginar como una tabla, una cuadrícula o un tablero.

- Índice (Index / Subscript): Un número entero que indica la posición específica de un elemento dentro de un arreglo. En un vector se necesita un índice; en una matriz se necesitan dos (uno para la fila y otro para la columna).
- Elemento (Element): Cada uno de los valores individuales almacenados dentro de un arreglo.
- Dimensión (Dimension): En PSeInt, es la palabra clave que se utiliza para declarar un arreglo y especificar su tamaño (o tamaños, en el caso de una matriz).
- Recorrer (Traverse): La acción de visitar o procesar cada uno de los elementos de un arreglo de forma sistemática, generalmente utilizando una estructura cíclica como un ciclo `Para`.

## PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

### FORMATO GUÍA DE APRENDIZAJE

#### IDENTIFICACIÓN DE LA GUÍA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto: DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL. Fase del Proyecto: Diseño
- Actividad de Proyecto: Desarrollar el software de acuerdo con el diseño y a la plataforma de desarrollo.
- Competencia: 220501096 Desarrollar la solución de software de acuerdo con el diseño y metodologías de desarrollo.
- Resultados de Aprendizaje Alcanzar: 669169 - Rap 1. Resolver procesos lógicos a través de la implementación de algoritmos y el lenguaje de programación seleccionado.
- Duración de la guía (24-horas)

#### 2. PRESENTACIÓN

Una vez dominadas las estructuras secuenciales, el siguiente paso en la construcción de algoritmos es dotarlos de inteligencia para tomar decisiones y la eficiencia para repetir tareas. Las **estructuras de control** modifican el flujo de ejecución de un programa. Las **estructuras condicionales** ('Si-Sino', 'Según') permiten que un algoritmo elija entre diferentes caminos de acción basados en el cumplimiento de una condición lógica. Por otro lado, las **estructuras cíclicas o repetitivas** ('While', 'For') permiten ejecutar un bloque de instrucciones múltiples veces, ya sea un número determinado de veces o hasta que una condición deje de cumplirse. Se espera que el aprendiz adquiera la habilidad de traducir problemas simples a secuencias lógicas cíclicas de instrucciones.

#### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

##### 3.1 Actividad de reflexión inicial.

Según la siguiente receta o lista de pasos secuenciales (algoritmo) para "Preparar Huevos Revueltos para una persona".

1. Romper 2 huevos en un tazón.
2. Agregar una pizca de sal.
3. Batir los huevos.
4. Calentar un sartén con aceite.
5. Verter los huevos en el sartén.
6. Cocinar por 3 minutos.
7. Servir en un plato.

Discusión 1. Discuta con todo el grupo los siguientes escenarios:

"Van a probar los huevos y se dan cuenta de que les falta sal. "

¿Qué parte del algoritmo tendrían que 'decidir' cambiar?

¿Qué harían si, por el contrario, ya estuvieran salados?"

**SI** (falta sal) **ENTONCES** (agregar un poco más).

Discusión 2. "El paso 3 dice 'batir los huevos'."

¿Es solo un movimiento y ya? ¿O lo hacen una y otra vez?

**¿MIENTRAS los huevos no estén bien mezclados?**

Discusión 3. "La receta es para una persona. Si tuvieran que hacer huevos para toda su familia (digamos, 4 personas):

¿repetirían todo el proceso 4 veces? ¿O qué harían de forma más inteligente?"

**PARA** cada una de las 4 personas, repetir ciertos pasos (como romper 2 huevos).

Realice la siguiente lectura <https://keepcoding.io/blog/que-son-estructuras-de-control-en-programacion/>.

Tiempo estimado: 25 minutos

### **3.2 Identificar los conocimientos necesarios para elaborar algoritmos con estructuras de control.**

Las estructuras condicionales son la base de la toma de decisiones en un programa. Permiten ejecutar un bloque de código solo si se cumple una determinada condición.

Visualice el siguiente video:  
[https://www.youtube.com/watch?v=RaWfeVgkWbE&ab\\_channel=Cegamer](https://www.youtube.com/watch?v=RaWfeVgkWbE&ab_channel=Cegamer)

Preguntas para reflexionar:

1. ¿Cuál es la diferencia fundamental entre un condicional simple (SI...ENTONCES) y un condicional doble (SI...ENTONCES...SINO)?
2. Describe una situación de la vida real donde aplicarías un SI...ENTONCES...SINO.
3. En el diagrama de flujo, ¿qué representa la figura del rombo y por qué tiene dos salidas ("Verdadero" y "Falso")?

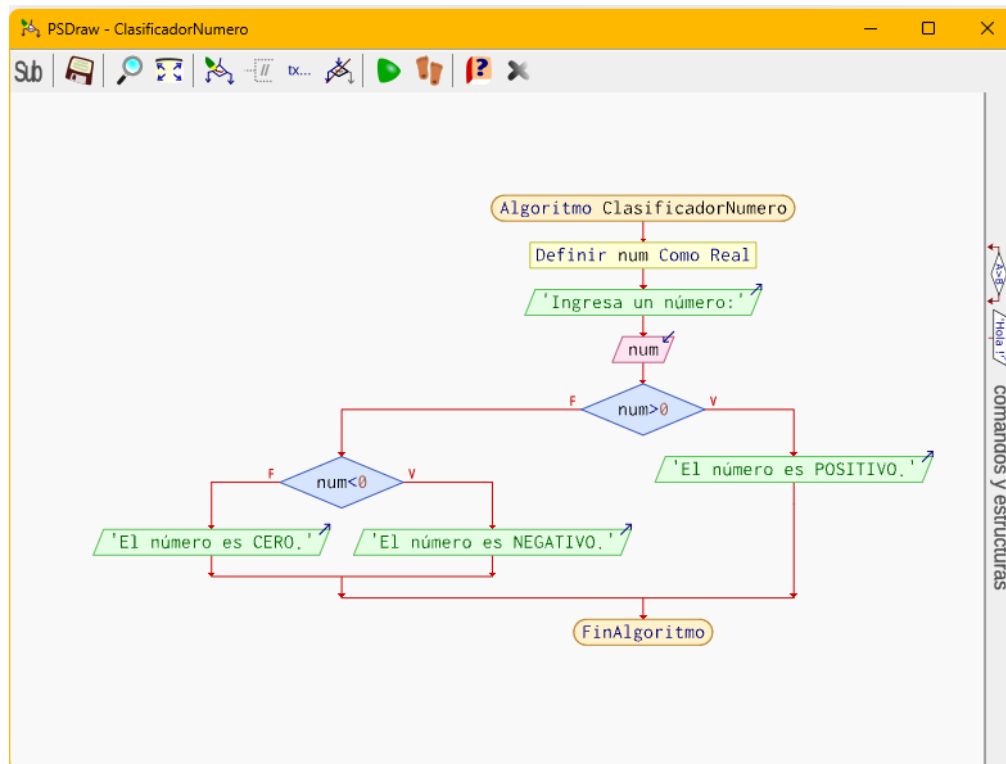
#### **3.2.1 Actividades de apropiación del conocimiento (PSeInt)**

Visualice el siguiente video:  
[https://www.youtube.com/watch?v=DIOBCZPFcaY&ab\\_channel=MarcosRios](https://www.youtube.com/watch?v=DIOBCZPFcaY&ab_channel=MarcosRios)

- Siga el ejemplo e implemente en PSeInt el código para determinar si un aprendiz es o no mayor de edad.

### 3.2.1.1 Ejercicio Clasificador de Números

Objetivo: Implemente el siguiente diagrama de flujo que pide un número al usuario y determine si es positivo, negativo o cero.



### 2.2.1.2 Ejercicio Taquilla de Parque de Diversiones

Objetivo: Crea un algoritmo que calcule el precio de la entrada a un parque de diversiones. El programa debe solicitar la edad y la estatura (en metros) del visitante

#### Reglas de Precios:

- Menores de 4 años entran gratis.
- Entre 4 y 12 años, el precio base es de \$15.000. Pero, si miden menos de 1.30m, tienen un 10% de descuento.
- Mayores de 12 años pagan una tarifa única de \$25.000.

### 2.2.1.3 Ejercicio Recomendador de Menú

**Objetivo:** Diseña un algoritmo que funcione como un recomendador de menú para un restaurante. El programa debe hacer dos preguntas: "¿Qué tipo de comida prefieres? (Carne, Vegetariana, Pescado)" y "¿Cuál es tu presupuesto? (Bajo, Medio, Alto)".

#### Reglas de Recomendación:

- **Carne:**
  - Bajo: "Te recomendamos la Hamburguesa Clásica."
  - Medio: "Te recomendamos el Lomo a la Pimienta."
  - Alto: "Te recomendamos el Corte T-Bone importado."
- **Vegetariana:**
  - Bajo: "Te recomendamos la Ensalada del Chef."

- Medio: "Te recomendamos la Lasaña de Berenjenas."
- Alto: "Te recomendamos el Risotto de Champiñones y Trufa."
- **Pescado:**
- Bajo: "Te recomendamos los Tacos de Pescado."
- Medio: "Te recomendamos el Salmón a la Plancha."
- Alto: "Te recomendamos la Paella Marinera."

Si el usuario ingresa una opción no válida en cualquiera de las preguntas, el programa debe mostrar "Opción no válida en el menú."

**3.2.1.4 Problema:** Un profesor necesita un programa para analizar las calificaciones de un grupo de 5 aprendices. El programa debe hacer lo siguiente:

\*Ingresar la calificación de cada aprendiz, por cada calificación ingresada, determinar si el estudiante "Aprobó" (calificación  $\geq 3.0$ ) o "Reprobó" (calificación  $< 3.0$ ) y mostrar el mensaje inmediatamente.

**Tiempo estimado:** 4 horas.

### 3.3 Identificar los conocimientos necesarios para elaborar algoritmos con estructuras cíclicas condicionadas.

El ciclo "while" o ciclo "mientras" es un "**ciclo controlado por condición inicial**", lo que significa que el proceso se repite solo si una condición es verdadera, evaluándose esta antes de cada ejecución del cuerpo del ciclo.

Realice la siguiente lectura <https://dcodingames.com/el-ciclo-while/> y responda:

- ¿Cómo funciona el ciclo while?
- ¿Cuáles son las partes fundamentales del ciclo while?
- ¿Por qué es crucial que la variable de control cambie su valor dentro del cuerpo de un ciclo while?
- ¿Cuándo es más apropiado utilizar un ciclo while en comparación con otros tipos de ciclos como do-while?

#### 3.3.1 Actividades de apropiación del conocimiento (PSeInt)

Visualice el siguiente video: [https://www.youtube.com/watch?v=CNdd2np4zHE&ab\\_channel=MarcosRios](https://www.youtube.com/watch?v=CNdd2np4zHE&ab_channel=MarcosRios)

- Siga el ejemplo para crear un programa en PSeInt que simule un "login". Debe pedir una contraseña hasta que el usuario ingrese la correcta.

##### 3.3.1.1 Ejercicio 2 Simulador de Cajero Automático

**Objetivo:** Diseña un algoritmo que simule un menú de cajero automático. El programa debe permitir al usuario consultar saldo, retirar dinero y salir.

- Reglas:
  1. El programa debe iniciar con un saldo de \$500.000.
  2. El menú principal se debe mostrar repetidamente mientras el usuario no elija la opción "Salir".
  3. Si el usuario elige "Retirar dinero", el programa debe pedir el monto a retirar. Se debe usar otro ciclo Mientras para validar que el monto ingresado no sea mayor al saldo disponible. Si lo es, debe mostrar un error y volver a pedir el monto.
  4. Una vez validado, se debe restar el monto del saldo y mostrar el nuevo saldo.

5. Si elige "Consultar saldo", solo debe mostrar el saldo actual.

### 3.3.1.2 Ejercicio 3 Juego de Adivinar el Número

**Objetivo:** Implementa un juego donde el programa "piensa" un número al azar entre 1 y 100, y el usuario debe adivinarlo.

- Reglas:
  1. El programa debe pedir un numero para adivinar.
  2. Usando un ciclo Mientras, el programa debe pedir al usuario que ingrese un número hasta que lo adivine.
  3. Después de cada intento, el programa debe dar una pista, indicando si el número secreto es "más alto" o "más bajo" que el número ingresado por el usuario.
  4. El programa también debe contar el número de intentos que le tomó al usuario adivinar.
  5. Cuando el usuario adivine, el programa debe felicitarlo y mostrarle cuántos intentos necesitó.

**3.3.1.3 Problema:** Un profesor necesita un programa para analizar las calificaciones de un grupo de 5 estudiantes. El programa debe hacer lo siguiente:

Usar un **ciclo Para** para pedir la calificación (de 0.0 a 5.0) de cada uno de los 5 estudiantes.

**Tiempo estimado:** 4 horas.

## 3.4 Identificar los conocimientos necesarios para elaborar algoritmos con cíclicas.

El ciclo "for" es una estructura de control de flujo repetitiva que se caracteriza por ser un **ciclo controlado por contador**. Esto significa que se sabe con exactitud cuántas veces se va a repetir una determinada tarea. Se define una variable de control (el contador) que se incrementa o decrementa de uno en uno, y las acciones dentro del ciclo se repiten hasta que esta variable alcanza un valor final predeterminado. Aunque la palabra "for" se traduce como "para", el autor sugiere que "desde" es una traducción más adecuada para su funcionamiento en pseudocódigo, ya que describe la repetición de acciones "desde" un valor inicial "hasta" un valor final.

Realice la siguiente lectura <https://dcodingames.com/el-ciclo-for/> y responda:

- ¿Qué tipos de datos puede tener la variable de control en un ciclo for?
- ¿Qué tipos de valores se pueden usar como límites (inicial y final) en un ciclo for?
- ¿Por qué no se debe modificar la variable de control dentro del cuerpo del ciclo for?
- ¿Qué sucede si se termina el encabezado del ciclo for con un punto y coma (;)?

### 3.4.1 Actividades de apropiación del conocimiento (PSeInt)

Visualice el siguiente video:

[https://www.youtube.com/watch?v=3G7AXyLvRuo&ab\\_channel=MarcosRios](https://www.youtube.com/watch?v=3G7AXyLvRuo&ab_channel=MarcosRios)

**3.4.1.4 Problema:** Un profesor necesita un programa para analizar las calificaciones de un grupo de 5 estudiantes. El programa debe hacer lo siguiente:

Usar un **ciclo Para** para pedir la calificación (de 0.0 a 5.0) de cada uno de los 5 estudiantes.

**Tiempo estimado:** 4 horas.

#### 4. ACTIVIDADES DE EVALUACIÓN (consolidación de conocimientos)

De forma individual diseñe el pseudocódigo, el diagrama de flujo y posteriormente implementa en PSeInt el algoritmo que solucione el siguiente problema:

##### Analizador de Calificaciones de Estudiantes

**Problema:** Un profesor necesita un programa para analizar las calificaciones de un grupo de 5 estudiantes. El programa debe hacer lo siguiente:

1. Usar un **ciclo Para** para pedir la calificación (de 0.0 a 5.0) de cada uno de los 5 estudiantes.
2. Dentro del ciclo, por cada calificación ingresada, debe usar una **estructura condicional (Si...Entonces...SiNo)** para determinar si el estudiante "Aprobó" (calificación  $\geq 3.0$ ) o "Reprobó" (calificación  $< 3.0$ ) y mostrar el mensaje inmediatamente.
3. El programa debe contar cuántos estudiantes aprobaron y cuántos reprobaron.
4. El programa debe calcular la suma total de todas las calificaciones (usando un acumulador).
5. Al finalizar el ciclo, el programa debe mostrar en pantalla:
  - El número total de estudiantes que aprobaron.
  - El número total de estudiantes que reprobaron.
  - El promedio de calificación de todo el grupo (Suma de calificaciones / 5).

#### 4.1 Actividad de Evaluación Final (Proyecto): Simulador de Venta de Boletos de Cine

En parejas, diseñe el pseudocódigo, el diagrama de flujo y la implementación en PSeInt para el siguiente proyecto.

**Problema:** Un cine local necesita un sistema para su taquilla que permita vender boletos y llevar un registro de las ventas del día.

##### Funcionalidades Requeridas:

1. **Menú Principal Interactivo:**
  - El programa debe mostrar un menú con tres opciones:
    1. Vender Entradas
    2. Ver Reporte de Ventas
    3. Salir
2. **Módulo "Vender Entradas":**
  - Al elegir esta opción, el sistema debe preguntar: "¿Cuántas entradas desea comprar?".
  - Para cada entrada, debe solicitar la **edad** del cliente.
  - El precio de la entrada se determina:
    - Menores de 12 años (niños): \$7.000
    - Entre 12 y 60 años (adultos): \$10.000
    - Mayores de 60 años (tercera edad): \$5.000 (descuento especial)
  - El sistema debe calcular el costo total de la transacción actual y mostrarlo.
3. **Módulo "Ver Reporte de Ventas":**
  - Al elegir esta opción, el sistema debe mostrar en pantalla:
    - Cantidad total de entradas vendidas (sumando todas las transacciones).
    - Cantidad de entradas vendidas por categoría (niños, adultos, tercera edad).
    - Dinero total recaudado.
4. **Módulo "Salir":**
  - Debe mostrar un mensaje de despedida como "Cerrando sistema. ¡Hasta pronto!" y finalizar el programa.

##### Variables Para Considerar:



- Necesitarás **acumuladores** para el total de dinero y para el total de entradas vendidas.
- Necesitarás **contadores** para cada categoría de entrada (niños, adultos, tercera edad).

**Tiempo estimado:** 8 horas.

## 5. GLOSARIO DE TERMINOS

**Estructura de Control** (Condicional): Instrucción que permite alterar el flujo de ejecución de un programa basándose en la evaluación (Verdadero o Falso) de una condición. La principal es Si...Entonces...SiNo (If...Then...Else).

**Estructura Cíclica** (Bucle o Iteración): Construcción que permite la ejecución repetida de un bloque de código. Las principales son Mientras (While) y Para (For).

**Condición:** Expresión lógica (que utiliza operadores relacionales como >, <, ==) que, al ser evaluada, produce un resultado booleano (Verdadero o Falso), determinando el flujo del programa.

**Bucle Mientras (While):** Ejecuta un bloque de código repetidamente *mientras* una condición especificada se mantenga verdadera. Su número de iteraciones es indeterminado.

**Bucle Para (For):** Ejecuta un bloque de código un número predefinido de veces. Utiliza una variable de control (contador) que se incrementa o decrementa en cada iteración.

**Contador:** Variable, típicamente de tipo entero, utilizada para llevar la cuenta del número de veces que se ha ejecutado un bucle.

**Acumulador:** Variable utilizada para almacenar la suma o el resultado consolidado de una serie de valores a lo largo de las iteraciones de un bucle.

**Bucle Infinito:** Un ciclo cuya condición de terminación nunca se cumple, provocando que el programa se ejecute indefinidamente y se bloquee. Es un error de lógica común.



## **PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL**

### **FORMATO GUÍA DE APRENDIZAJE**

#### **1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE**

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto Formativo (si aplica): DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL
- Fase del Proyecto (si aplica): Ejecución
- Actividad de Proyecto Formativo (si aplica): Desarrollar el software de acuerdo con el diseño y a la programación establecida
- Competencia: 220501096 DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y LA PROGRAMACIÓN ESTABLECIDA
- Resultados de Aprendizaje: Rap 3. CODIFICAR EL SOFTWARE EMPLEANDO EL LENGUAJE DE PROGRAMACIÓN SEGÚN REQUERIMIENTO Y DISEÑO
- Duración de la Guía de Aprendizaje (horas): 72 horas (24 sincrónicas + 48 asincrónicas)

#### **2. PRESENTACIÓN**

En la guía anterior tu aplicación dejó de ser un simple mensaje de texto y se convirtió en una página web real, con catálogo, categorías y vista de detalle. Sin embargo, hasta ahora tu sitio es como una vitrina: el usuario solo puede mirar. Si quieres agregar, modificar o eliminar un producto, todavía tienes que meterte al shell de Django o al panel de administración.

Piensa en cualquier app que uses a diario: cuando editas tu perfil de Instagram, publicas una historia, agregas una canción a una playlist de Spotify o te registras en una nueva cuenta, en todos esos casos estás llenando un formulario que envía información al servidor. Esa información se procesa, se valida y finalmente se guarda en una base de datos.



En esta guía vas a aprender a construir esos formularios con Django, para que tu aplicación de productos deje de ser una vitrina y se convierta en una herramienta completa donde el usuario pueda crear, editar y eliminar productos directamente desde el navegador, sin tocar el shell ni el panel de administración.

En esta guía vas a:

Entender cómo funciona un formulario HTML y la diferencia entre los métodos GET y POST.

Usar Django Forms y, en especial, ModelForm para crear formularios automáticamente a partir de tus modelos.

Crear una vista y un template para agregar nuevos productos (Create).

Crear una vista y un template para editar productos existentes (Update).

Crear una vista para eliminar productos (Delete), completando así el ciclo CRUD desde la web.

Al finalizar esta guía, cualquier persona podrá usar tu aplicación para administrar el catálogo completo de productos: agregar nuevos, corregir precios o existencias, y eliminar los que ya no se necesiten, todo desde el navegador. ¡Empecemos!

### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**

#### 3.1 Actividades de reflexión inicial:

Descripción de la actividad:

Piensa en las últimas veces que llenaste un formulario en línea (registro de cuenta, compra, encuesta). ¿Qué pasó cuando enviaste datos incorrectos? ¿La app te avisó del error? ¿Cómo crees que Django sabe qué datos son válidos y cuáles no? Escribe tu reflexión (mínimo 5 líneas).

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.



Estrategias o técnicas didácticas activas: Lluvia de ideas; discusión guiada; preguntas orientadoras; trabajo en parejas.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 1 hora.

### 3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje:

Descripción de la actividad:

Formularios HTML: GET y POST

Un formulario HTML es el elemento que permite que un usuario escriba información y la envíe al servidor, por ejemplo un nombre de usuario, un precio o una descripción. En HTML, un formulario básico se ve así:

```
<form method="POST">
  <input type="text" name="nombre">
  <button type="submit">Enviar</button>
</form>
```

El atributo method indica cómo se envían los datos al servidor:

GET: se usa normalmente para solicitar información (por ejemplo, mostrar una página). Los datos viajan visibles en la URL.

POST: se usa para enviar información que el servidor va a procesar o guardar, como los datos de un nuevo producto. Los datos NO se ven en la URL.

El token CSRF

Django protege tu aplicación de ataques donde un sitio externo intenta enviar formularios a tu servidor sin permiso. Por eso, todo formulario que use method="POST" en un template de Django debe incluir la etiqueta {% csrf\_token %} justo después de <form ...>:



```
<form method="POST">
  {% csrf_token %}
  ...
</form>
```

## Django Forms y ModelForm

Escribir a mano cada campo de un formulario HTML, y luego validar uno por uno los datos que llegan, sería muy repetitivo (¡y rompería el principio DRY!). Por eso Django ofrece la librería forms, que permite definir formularios como clases de Python.

La forma más rápida de crear un formulario para un modelo es usando ModelForm: Django genera automáticamente un campo de formulario por cada campo del modelo, junto con sus validaciones.

Crea un archivo forms.py dentro de tu aplicación productos:

```
from django import forms
from .models import Producto

class ProductoForm(forms.ModelForm):
    class Meta:
        model = Producto
        fields = '__all__'
```

La clase interna Meta le indica a Django:

model: a partir de qué modelo se debe construir el formulario (Producto).

fields: qué campos del modelo deben incluirse en el formulario. '\_\_all\_\_' significa "todos los campos".

Pasos para crear tu archivo de formularios

Dentro de la carpeta de tu aplicación (productos), crea un archivo llamado forms.py.

Importa la librería forms de Django y el modelo Producto.

Crea la clase ProductoForm que herede de forms.ModelForm.

Dentro de la clase, define la clase interna Meta indicando model = Producto y fields = '\_\_all\_\_'.

---

Una vista que procesa formularios

Una misma vista puede comportarse de dos formas distintas, según cómo se haya accedido a ella:



Si el usuario apenas entra a la página (método GET), la vista debe mostrar el formulario vacío.

Si el usuario ya llenó el formulario y lo envió (método POST), la vista debe tomar esos datos, validarlos y, si todo está correcto, guardarlos en la base de datos.

Django nos permite distinguir entre ambos casos revisando `request.method`:

```
from django.shortcuts import render, redirect
from .forms import ProductoForm

def crear_producto(request):
    if request.method == 'POST':
        form = ProductoForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('lista_productos')
    else:
        form = ProductoForm()

    return render(request, 'productos/crear.html', {'form': form})
```

¿Qué hace cada parte de esta vista?

`form = ProductoForm()`: crea un formulario vacío para mostrarlo la primera vez (GET).

`form = ProductoForm(request.POST)`: crea el formulario con los datos que el usuario envió.

`form.is_valid()`: revisa que los datos cumplan las reglas del modelo (por ejemplo, que precio sea un número).

`form.save()`: guarda un nuevo registro en la base de datos con esos datos.

`redirect('lista_productos')`: redirige al usuario a otra vista (en este caso, el catálogo) usando el name definido en `urls.py`.

El template del formulario

En el template, Django puede dibujar automáticamente todos los campos del formulario con `{{ form.as_p }}` (cada campo dentro de su propia etiqueta `<p>`):

```
{% extends 'productos/base.html' %}

{% block titulo %}Nuevo producto{% endblock %}

{% block content %}
```



```
<h2>Agregar nuevo producto</h2>
<form method="POST">
  {% csrf_token %}
  {{ form.as_p }}
  <button type="submit">Guardar</button>
</form>
{% endblock %}
```

Pasos para crear la vista y el template

En views.py, importa redirect y ProductoForm.

Crea la función crear\_producto siguiendo el ejemplo anterior.

En urls.py, agrega la ruta 'crear/' apuntando a crear\_producto, con name='crear\_producto'.

Crea el template crear.html que extienda de base.html, con el formulario, {% csrf\_token %} y {{ form.as\_p }}.

---

Editar un producto es muy parecido a crear uno, con una diferencia clave: el formulario debe aparecer ya lleno con los datos actuales del producto, y al guardar, debe actualizar ese registro en lugar de crear uno nuevo.

Para esto, ModelForm acepta un parámetro instance: si le pasamos un objeto ya existente, el formulario se llena automáticamente con sus datos, y al guardar, actualiza ese mismo objeto en vez de crear uno nuevo.

```
from django.shortcuts import render, redirect, get_object_or_404
from .models import Producto
from .forms import ProductoForm

def editar_producto(request, producto_id):
    producto = get_object_or_404(Producto, id=producto_id)

    if request.method == 'POST':
        form = ProductoForm(request.POST, instance=producto)
        if form.is_valid():
            form.save()
            return redirect('lista_productos')
    else:
        form = ProductoForm(instance=producto)

    return render(request, 'productos/editar.html', {'form': form, 'producto': producto})
```



Observa la diferencia con crear\_producto:

`get_object_or_404(Producto, id=producto_id)`: busca el producto que se quiere editar, igual que hiciste en la vista de detalle de la guía anterior.

`ProductoForm(instance=producto)`: en el método GET, llena el formulario con los datos actuales del producto.

`ProductoForm(request.POST, instance=producto)`: en el método POST, actualiza ese mismo producto con los nuevos datos enviados.

La ruta de edición necesita el id del producto, igual que la vista de detalle:

`path('editar/<int:producto_id>/', views.editar_producto, name='editar_producto')`,

El template `editar.html` puede ser casi idéntico a `crear.html`, cambiando únicamente el título y el texto del botón.

Pasos para crear la vista y el template

En `views.py`, crea la función `editar_producto` siguiendo el ejemplo anterior.

En `urls.py`, agrega la ruta `'editar/<int:producto_id>/'` con `name='editar_producto'`.

Crea el template `editar.html`, similar a `crear.html`, pero con el título “Editar producto”.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 4 horas.





### 3.3 Actividades de apropiación:

Descripción de la actividad:

Crea el archivo `forms.py` en tu aplicación productiva y define un `ModelForm` para tu modelo principal (el que usaste en las guías anteriores). Verifica que el formulario incluya todos los campos del modelo (nombre, precio, stock, imagen, categoría, marca, descripción, etc.).

Implementa la vista `crear_producto` en tu proyecto, usando tu `ModelForm`.

Agrega la ruta correspondiente en `urls.py`.

Crea el template `crear.html` con el formulario completo.

En `lista.html`, agrega un enlace o botón “Agregar producto” que lleve a esta nueva vista usando `{% url %}`.

Prueba el formulario: crea al menos dos productos nuevos desde el navegador y verifica que aparezcan en el catálogo.

Implementa la vista `editar_producto` y su ruta en `urls.py`.

Crea el template `editar.html`.

En `lista.html`, agrega un enlace “Editar” junto a cada producto que lleve a esta vista usando `{% url 'editar_producto' producto.id %}`.

Prueba editar al menos un producto y verifica que los cambios se reflejen en el catálogo.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.



Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Código fuente funcional del proyecto Django con las funcionalidades implementadas. Capturas de pantalla que evidencien el correcto funcionamiento en el navegador.

Instrumentos de evaluación: Lista de chequeo de funcionalidades implementadas. Revisión de código por pares.

Duración de la actividad: 6 horas.

### **3.4 Actividades de Transferencia el Conocimiento:**

Descripción de la actividad:

Implementa la vista `eliminar_producto` siguiendo el ejemplo anterior.

Agrega la ruta `'eliminar/<int:producto_id>/'` en `urls.py`, con `name='eliminar_producto'`.

Crea el template `eliminar.html` con el mensaje de confirmación.

En `lista.html`, agrega un enlace o botón “Eliminar” junto a cada producto que lleve a esta vista usando `{% url %}`.

Verifica el CRUD completo: crea, edita, consulta el detalle y elimina al menos un producto desde el navegador.

#### **PARTE TEÓRICA:**

Responde las siguientes preguntas:

¿Cuál es la diferencia entre los métodos GET y POST en un formulario HTML?

¿Para qué sirve la etiqueta `{% csrf_token %}` y qué pasaría si la quitaras de un formulario?

Explica con tus palabras qué hace un `ModelForm` y por qué resulta útil aplicar el principio DRY.

¿Qué diferencia hay entre la vista `crear_producto` y la vista `editar_producto`?

¿Por qué la operación de eliminar pide una confirmación (un formulario POST con botón) en lugar de borrar el producto directamente al hacer clic en un enlace?



Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Actividad práctica con capturas de pantalla. Respuestas escritas a las preguntas teóricas (entregadas en documento o plataforma del instructor).

Instrumentos de evaluación: Rúbrica de evaluación de desempeño. Cuestionario de preguntas abiertas.

Duración de la actividad: 3 horas.

#### 4. PLANTEAMIENTO DE EVIDENCIAS DE APRENDIZAJE PARA LA EVALUACIÓN EN EL PROCESO FORMATIVO.

| Fase del proyecto formativo | Actividad del proyecto formativo | Actividad de Aprendizaje | Evidencias de Aprendizaje | Criterios de Evaluación | Técnicas e Instrumentos de Evaluación |
|-----------------------------|----------------------------------|--------------------------|---------------------------|-------------------------|---------------------------------------|
|                             |                                  |                          |                           |                         |                                       |

Evidencia de Conocimiento: Respuestas a las preguntas teóricas de la actividad evaluativa (sección 3.4.2).

Evidencia de Desempeño: Implementación funcional de todas las actividades prácticas indicadas en las secciones 3.3 y 3.4.1, demostrada mediante capturas de pantalla y revisión del código fuente.



Evidencia de Producto: Proyecto Django actualizado con las funcionalidades de la guía correctamente implementadas, almacenado en repositorio Git.

## 5. GLOSARIO DE TÉRMINOS

Django: Framework web de alto nivel para Python que fomenta el desarrollo rápido y seguro.

Proyecto: Contenedor principal de una aplicación web, con configuraciones globales.

Aplicación (App): Módulo reutilizable que se encarga de una funcionalidad específica dentro de un proyecto.

MVT (Model-View-Template): Patrón de diseño de Django que organiza el código en tres capas lógicas: datos (Model), lógica (View) y presentación (Template).

DRY (Don't Repeat Yourself): Principio de desarrollo que busca reducir la duplicación de código.

ORM y CRUD: El ORM traduce código Python a consultas SQL; CRUD son las operaciones básicas de Crear, Leer, Actualizar y Eliminar datos.

ForeignKey: Campo de un modelo que crea una relación (muchos a uno) con otro modelo.

Template (Plantilla): Archivo, normalmente .html, que combina HTML con el Django Template Language (DTL) para presentar información al usuario.

get\_object\_or\_404: Función que busca un objeto en la base de datos por su identificador o muestra automáticamente un error 404 si no existe.

Formulario HTML: Elemento de una página web que permite a un usuario ingresar y enviar información al servidor.

Método GET: Método HTTP usado para solicitar información; los datos enviados son visibles en la URL.

Método POST: Método HTTP usado para enviar información que el servidor procesará o guardará; los datos no se muestran en la URL.

request.method: Atributo que indica con qué método HTTP (GET o POST) se hizo la petición a una vista.

CSRF / {% csrf\_token %}: Mecanismo de seguridad de Django que protege los formularios contra envíos no autorizados desde otros sitios.



Django Forms: Sistema de Django para definir formularios como clases de Python, con validación automática de datos.

ModelForm: Tipo especial de Django Form que genera automáticamente los campos de un formulario a partir de un modelo.

form.is\_valid(): Método que revisa si los datos enviados en un formulario cumplen las reglas y tipos definidos en el modelo.

form.save(): Método que guarda (crea o actualiza) en la base de datos los datos validados de un formulario.

redirect(): Función que envía al usuario a otra URL o vista, normalmente después de procesar un formulario.

## 6. REFERENTES BIBLIOGRÁFICOS

Django Software Foundation. (2024). Django documentation. <https://docs.djangoproject.com/>

Mozilla Developer Network. (2024). Django Web Framework (Python). <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>

Holovaty, A. & Kaplan-Moss, J. (2009). The Definitive Guide to Django. Apress.

Vincent, W. S. (2022). Django for Beginners. Leanpub.

Python Software Foundation. (2024). The Python Tutorial. <https://docs.python.org/3/tutorial/>

## 7. CONTROL DEL DOCUMENTO



|            | Nombre                    | Cargo      | Dependencia | Fecha      |
|------------|---------------------------|------------|-------------|------------|
| Autor (es) | Jheyson Fabian Villavisan | Instructor | CDM         | 25/06/2026 |

**8. CONTROL DE CAMBIOS** (diligenciar únicamente si realiza ajustes a la guía)

|            | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
|------------|--------|-------|-------------|-------|------------------|
| Autor (es) |        |       |             |       |                  |



## PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

### FORMATO GUÍA DE APRENDIZAJE

#### 1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto Formativo (si aplica): DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL
- Fase del Proyecto (si aplica): Ejecución
- Actividad de Proyecto Formativo (si aplica): Desarrollar el software de acuerdo con el diseño y a la programación establecida
- Competencia: 220501096 DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y LA PROGRAMACIÓN ESTABLECIDA
- Resultados de Aprendizaje: Rap 3. CODIFICAR EL SOFTWARE EMPLEANDO EL LENGUAJE DE PROGRAMACIÓN SEGÚN REQUERIMIENTO Y DISEÑO
- Duración de la Guía de Aprendizaje (horas): 72 horas (24 sincrónicas + 48 asincrónicas)

#### 2. PRESENTACIÓN

Tu aplicación de productos ya tiene catálogo, vista de detalle y formularios para crear, editar y eliminar productos. Pero hay un problema: en este momento, CUALQUIER persona que entre a tu sitio puede borrar todo el catálogo, sin necesidad de iniciar sesión ni demostrar quién es. Es como si la puerta de una tienda no tuviera llave y cualquiera pudiera entrar a reorganizar o llevarse la mercancía.

Piensa en cómo funcionan las apps que usas todos los días: para entrar a tu cuenta de Instagram, Spotify o tu plataforma de videojuegos, primero debes iniciar sesión con un usuario y una contraseña. Solo entonces la app sabe quién eres y qué puedes hacer: ver tu perfil, publicar contenido, cambiar tu configuración, etc. Eso se llama autenticación.



En esta guía vas a aprender a usar el sistema de autenticación que viene incluido en Django (django.contrib.auth) para que tu aplicación de productos tenga usuarios reales: cada persona podrá registrarse, iniciar sesión, cerrar sesión, y solo los usuarios autenticados podrán crear, editar o eliminar productos.

En esta guía vas a:

Conocer el sistema de autenticación de Django y el modelo User.

Crear un superusuario y explorar el panel de administración.

Construir un formulario de registro para que nuevos usuarios creen su cuenta.

Implementar las vistas de inicio de sesión (login) y cierre de sesión (logout).

Proteger las vistas de crear, editar y eliminar productos para que solo usuarios autenticados puedan usarlas.

Al finalizar esta guía, tu aplicación dejará de ser una puerta sin llave: cualquier visitante podrá ver el catálogo, pero solo los usuarios registrados y con sesión iniciada podrán modificarlo. ¡Vamos a ponerle seguridad a tu proyecto!

### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**

#### 3.1 Actividades de reflexión inicial:

Descripción de la actividad:

Reflexiona: cuando inicias sesión en Instagram o en tu correo, ¿cómo sabe la app que eres tú? ¿Qué pasa si alguien intenta ver tu perfil privado sin haber iniciado sesión? ¿Por qué crees que algunas páginas de tu proyecto deben estar protegidas? Comparte tus respuestas con el grupo.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.





Estrategias o técnicas didácticas activas: Lluvia de ideas; discusión guiada; preguntas orientadoras; trabajo en parejas.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 1 hora.

### **3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje:**

Descripción de la actividad:

Autenticación vs. autorización

Aunque suelen confundirse, son dos cosas distintas:

Autenticación: responde a la pregunta “¿quién eres?”. Es el proceso de identificar a un usuario, normalmente con un usuario y una contraseña.

Autorización: responde a la pregunta “¿qué puedes hacer?”. Una vez Django sabe quién eres, decide qué páginas o acciones puedes usar.

`django.contrib.auth`: la app de autenticación

Cuando creas un proyecto nuevo con Django, este ya incluye una aplicación llamada `django.contrib.auth` dentro de `INSTALLED_APPS` en `settings.py`. Esta app trae “de fábrica”:

Un modelo `User`, con campos como `username`, `password`, `email`, `first_name` y `last_name`.

Formularios listos para usar, como `UserCreationForm` (registro) y `AuthenticationForm` (inicio de sesión).

Funciones para iniciar sesión, cerrar sesión y verificar si un usuario está autenticado.

Un sistema de contraseñas seguro: Django nunca guarda la contraseña como texto plano, sino que la cifra (hashing) antes de guardarla.

El superusuario y el panel de administración



Django incluye un panel de administración (/admin/) al que solo pueden entrar usuarios con permisos especiales. Para crear el primer usuario con todos los permisos (superusuario), se usa un comando de manage.py:

```
python manage.py createsuperuser
```

Este comando te pedirá un nombre de usuario, un correo electrónico (opcional) y una contraseña. Con esos datos podrás iniciar sesión en /admin/ y administrar usuarios, productos y categorías desde la interfaz gráfica.

Pasos para crear tu superusuario

Abre la terminal en la carpeta de tu proyecto (donde está manage.py).

Ejecuta el comando `python manage.py createsuperuser`.

Ingresa un nombre de usuario, correo electrónico y contraseña (la contraseña no se mostrará en pantalla mientras la escribes, eso es normal).

Inicia el servidor con `python manage.py runserver` y visita la ruta /admin/.

Inicia sesión con el usuario y contraseña que acabas de crear.

---

El formulario UserCreationForm

Así como usaste `ModelForm` para crear formularios a partir de `Producto`, Django ofrece un formulario ya construido para registrar nuevos usuarios: `UserCreationForm`. Este formulario incluye los campos `username`, `password1` y `password2` (la contraseña se pide dos veces, para confirmarla), y valida automáticamente que la contraseña sea suficientemente segura.

Vista de registro (views.py):

```
from django.shortcuts import render, redirect
from django.contrib.auth.forms import UserCreationForm

def registro(request):
    if request.method == 'POST':
        form = UserCreationForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('login')
    else:
```



```
form = UserCreationForm()

return render(request, 'productos/registro.html', {'form': form})
```

Esta vista sigue exactamente el mismo patrón que crear\_producto de la guía anterior: si la petición es GET, muestra el formulario vacío; si es POST y los datos son válidos, guarda el nuevo usuario en la base de datos y redirige a la página de inicio de sesión.

Template de registro (registro.html):

```
{% extends 'productos/base.html' %}

{% block titulo %}Crear cuenta{% endblock %}

{% block content %}
<h2>Crear una cuenta nueva</h2>
<form method="POST">
  {% csrf_token %}
  {{ form.as_p }}
  <button type="submit">Registrarme</button>
</form>
{% endblock %}
```

Pasos para crear el registro

En views.py, importa UserCreationForm desde django.contrib.auth.forms.

Crea la función registro siguiendo el ejemplo anterior.

En urls.py, agrega la ruta 'registro/' apuntando a registro, con name='registro'.

Crea el template registro.html, similar a crear.html de la guía anterior.

---

Vistas de autenticación ya incluidas en Django

Django incluye vistas listas para iniciar y cerrar sesión, en django.contrib.auth.views: LoginView y LogoutView. Solo necesitas conectarlas en urls.py y crear el template del formulario de login (LogoutView no necesita template propio: simplemente cierra la sesión y redirige).

```
from django.contrib.auth import views as auth_views

urlpatterns = [
    ...
    path('login/', auth_views.LoginView.as_view(template_name='productos/login.html'),
```



```
name='login'),
    path('logout/', auth_views.LogoutView.as_view(next_page='lista_productos'),
name='logout'),
]
```

template\_name='productos/login.html': indica qué template usar para mostrar el formulario de inicio de sesión.

next\_page='lista\_productos': indica a dónde redirigir al usuario después de cerrar sesión.

Template de login (login.html):

```
{% extends 'productos/base.html' %}

{% block titulo %}Iniciar sesión{% endblock %}

{% block content %}
<h2>Iniciar sesión</h2>
<form method="POST">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Entrar</button>
</form>
{% endblock %}
```

Después de un login exitoso, Django redirige por defecto a la URL configurada en LOGIN\_REDIRECT\_URL (en settings.py). Agrega esta línea a tu settings.py:

```
LOGIN_REDIRECT_URL = 'lista_productos'
```

Saber quién inició sesión: request.user

En cualquier template, puedes usar la variable user para saber si hay alguien autenticado y mostrar contenido distinto según el caso. Esto es muy útil en base.html, para mostrar enlaces diferentes en el menú:

```
<nav>
    <a href="{% url 'index' %}">Inicio</a> |
    <a href="{% url 'lista_productos' %}">Catálogo</a> |
    {% if user.is_authenticated %}
        <span>Hola, {{ user.username }}</span> |
        <a href="{% url 'logout' %}">Cerrar sesión</a>
    {% else %}
        <a href="{% url 'login' %}">Iniciar sesión</a> |
        <a href="{% url 'registro' %}">Crear cuenta</a>
```



```
{% endif %}  
</nav>
```

`user.is_authenticated`: es True si hay un usuario con sesión iniciada, y False si es un visitante anónimo.

`user.username`: el nombre de usuario de la persona que inició sesión.

Pasos para crear el login y el logout

En `urls.py`, importa las vistas de autenticación: `from django.contrib.auth import views as auth_views`.

Agrega las rutas 'login/' y 'logout/' usando `LoginView` y `LogoutView`, como en el ejemplo.

Crea el template `login.html`.

En `settings.py`, agrega `LOGIN_REDIRECT_URL = 'lista_productos'`.

Modifica `base.html` para mostrar “Hola, usuario” y “Cerrar sesión” si hay sesión iniciada, o “Iniciar sesión” y “Crear cuenta” si no la hay.

**Ambiente requerido:** Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

**Estrategias o técnicas didácticas activas:** Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

**Materiales de formación** Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

**Material de apoyo:** Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

**Duración de la actividad:** 4 horas.

### 3.3 Actividades de apropiación:

Descripción de la actividad:



Crea un superusuario para tu proyecto y, desde el panel de administración (/admin/), verifica lo siguiente:

Que aparezca la sección “Users”, con el usuario que acabas de crear.

Que tus modelos (por ejemplo, Producto y Categoría) estén registrados y se puedan ver y editar desde ahí (recuerda admin.py de la guía de ForeignKey).

Implementa la vista registro y su ruta en urls.py.

Crea el template registro.html con el formulario de registro.

En base.html, agrega un enlace “Crear cuenta” usando {% url 'registro' %}.

Prueba el registro: crea al menos un usuario nuevo (diferente del superusuario) desde el navegador.

Implementa las rutas de login y logout en tu proyecto.

Crea el template login.html.

Actualiza base.html para mostrar el estado de la sesión (autenticado o no) y los enlaces correspondientes.

Prueba el flujo completo: regístrate, inicia sesión, verifica que el menú cambie, y luego cierra sesión.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.



Evidencias de aprendizaje: Código fuente funcional del proyecto Django con las funcionalidades implementadas. Capturas de pantalla que evidencien el correcto funcionamiento en el navegador.

Instrumentos de evaluación: Lista de chequeo de funcionalidades implementadas. Revisión de código por pares.

Duración de la actividad: 6 horas.

### **3.4 Actividades de Transferencia el Conocimiento:**

Descripción de la actividad:

Agrega `LOGIN_URL = 'login'` en `settings.py`.

Importa `login_required` desde `django.contrib.auth.decorators` en `views.py`.

Agrega el decorador `@login_required` encima de las vistas `crear_producto`, `editar_producto` y `eliminar_producto`.

Cierra sesión y verifica que, al intentar acceder a `'crear/'`, `'editar/<id>/'` o `'eliminar/<id>/'`, Django te redirige automáticamente al formulario de login.

Inicia sesión nuevamente y verifica que ahora sí puedes crear, editar y eliminar productos sin problema.

#### **PARTE TEÓRICA:**

Responde las siguientes preguntas:

Explica con tus palabras la diferencia entre autenticación y autorización, usando un ejemplo distinto al de la guía.

¿Qué formulario de Django se usa para registrar nuevos usuarios y qué campos incluye por defecto?

¿Qué hace la variable `user.is_authenticated` y dónde se puede usar?

¿Qué ocurre cuando un usuario sin sesión iniciada intenta acceder a una vista protegida con `@login_required`?

¿Por qué es importante que las contraseñas se guarden cifradas (hash) y no como texto plano en la base de datos?



Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Actividad práctica con capturas de pantalla. Respuestas escritas a las preguntas teóricas (entregadas en documento o plataforma del instructor).

Instrumentos de evaluación: Rúbrica de evaluación de desempeño. Cuestionario de preguntas abiertas.

Duración de la actividad: 3 horas.

#### 4. PLANTEAMIENTO DE EVIDENCIAS DE APRENDIZAJE PARA LA EVALUACIÓN EN EL PROCESO FORMATIVO.

| Fase del proyecto formativo | Actividad del proyecto formativo | Actividad de Aprendizaje | Evidencias de Aprendizaje | Criterios de Evaluación | Técnicas e Instrumentos de Evaluación |
|-----------------------------|----------------------------------|--------------------------|---------------------------|-------------------------|---------------------------------------|
|                             |                                  |                          |                           |                         |                                       |

Evidencia de Conocimiento: Respuestas a las preguntas teóricas de la actividad evaluativa (sección 3.4.2).

Evidencia de Desempeño: Implementación funcional de todas las actividades prácticas indicadas en las secciones 3.3 y 3.4.1, demostrada mediante capturas de pantalla y revisión del código fuente.





Evidencia de Producto: Proyecto Django actualizado con las funcionalidades de la guía correctamente implementadas, almacenado en repositorio Git.

## 5. GLOSARIO DE TÉRMINOS

Django: Framework web de alto nivel para Python que fomenta el desarrollo rápido y seguro.

Proyecto: Contenedor principal de una aplicación web, con configuraciones globales.

Aplicación (App): Módulo reutilizable que se encarga de una funcionalidad específica dentro de un proyecto.

MVT (Model-View-Template): Patrón de diseño de Django que organiza el código en tres capas lógicas: datos (Model), lógica (View) y presentación (Template).

DRY (Don't Repeat Yourself): Principio de desarrollo que busca reducir la duplicación de código.

ORM y CRUD: El ORM traduce código Python a consultas SQL; CRUD son las operaciones básicas de Crear, Leer, Actualizar y Eliminar datos.

Django Forms y ModelForm: Sistemas de Django para crear formularios; ModelForm genera los campos automáticamente a partir de un modelo.

Autenticación: Proceso mediante el cual una aplicación identifica quién es un usuario, normalmente con usuario y contraseña.

Autorización: Proceso mediante el cual una aplicación decide qué puede hacer un usuario ya identificado.

django.contrib.auth: Aplicación incluida en Django que provee el sistema de usuarios, contraseñas, login y permisos.

User: Modelo de Django que representa a un usuario, con campos como username, password y email.

Superusuario: Usuario con todos los permisos, creado con el comando `createsuperuser`, que puede acceder al panel de administración.

UserCreationForm: Formulario incluido en Django para registrar nuevos usuarios, con campos username, password1 y password2.



LoginView / LogoutView: Vistas incluidas en Django para iniciar y cerrar sesión, sin necesidad de escribirlas desde cero.

request.user: Objeto que representa al usuario que hace la petición; puede ser un usuario autenticado o un usuario anónimo.

user.is\_authenticated: Propiedad que indica si el usuario actual inició sesión (True) o es un visitante anónimo (False).

@login\_required: Decorador que obliga a que una vista solo pueda usarse si el usuario tiene una sesión iniciada.

LOGIN\_URL / LOGIN\_REDIRECT\_URL: Configuraciones en settings.py que indican a dónde enviar al usuario para iniciar sesión y a dónde redirigirlo después de un login exitoso.

Hash (cifrado de contraseñas): Proceso que transforma una contraseña en una cadena de texto irreconocible, para que no se almacene en texto plano.

## 6. REFERENTES BIBLIOGRÁFICOS

Django Software Foundation. (2024). Django documentation. <https://docs.djangoproject.com/>

Mozilla Developer Network. (2024). Django Web Framework (Python). <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>

Holovaty, A. & Kaplan-Moss, J. (2009). The Definitive Guide to Django. Apress.

Vincent, W. S. (2022). Django for Beginners. Leanpub.

Python Software Foundation. (2024). The Python Tutorial. <https://docs.python.org/3/tutorial/>



## 7. CONTROL DEL DOCUMENTO

|                   | Nombre                    | Cargo      | Dependencia | Fecha      |
|-------------------|---------------------------|------------|-------------|------------|
| <b>Autor (es)</b> | Jheyson Fabian Villavisan | Instructor | CDM         | 25/06/2026 |

## 8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

|                   | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
|-------------------|--------|-------|-------------|-------|------------------|
| <b>Autor (es)</b> |        |       |             |       |                  |



## **PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL**

### **FORMATO GUÍA DE APRENDIZAJE**

#### **1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE**

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto Formativo (si aplica): DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL
- Fase del Proyecto (si aplica): Ejecución
- Actividad de Proyecto Formativo (si aplica): Desarrollar el software de acuerdo con el diseño y a la programación establecida
- Competencia: 220501096 DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y LA PROGRAMACIÓN ESTABLECIDA
- Resultados de Aprendizaje: Rap 3. CODIFICAR EL SOFTWARE EMPLEANDO EL LENGUAJE DE PROGRAMACIÓN SEGÚN REQUERIMIENTO Y DISEÑO
- Duración de la Guía de Aprendizaje (horas): 72 horas (24 sincrónicas + 48 asincrónicas)

#### **2. PRESENTACIÓN**

Imagina que tu catálogo de productos crece y crece: pasas de tener 5 productos a tener 500. Si un usuario quiere encontrar “audífonos”, no tiene sentido que tenga que revisar página por página hasta encontrarlos. Necesita un buscador, como el que usas en Mercado Libre, Amazon o YouTube.

Además, cuando hay muchos resultados, mostrarlos todos de una sola vez haría que la página cargue muy lento y se vea saturada. Por eso, casi todas las apps dividen los resultados en páginas: piensa en cuando buscas algo en Google y ves “Página 1, 2, 3...” al final, o cuando haces scroll en Instagram y la app va cargando publicaciones poco a poco.



En esta guía vas a aprender a hacer que tu aplicación de productos sea mucho más usable: el usuario podrá buscar productos por nombre, filtrarlos por categoría, y navegar los resultados por páginas, sin que la aplicación se vuelva lenta ni desordenada.

En esta guía vas a:

Repasar qué es un QuerySet y cómo usar `filter()` para consultar datos específicos.

Crear un buscador de productos por nombre usando el operador `icontains`.

Agregar un filtro por categoría usando un menú desplegable.

Implementar paginación con la clase `Paginator` de Django, para mostrar los resultados en bloques.

Combinar búsqueda, filtro y paginación en una sola vista, conservando los criterios del usuario al cambiar de página.

Al finalizar esta guía, tu catálogo de productos funcionará como el de una tienda en línea real: con buscador, filtros y navegación por páginas. ¡Comencemos!

### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**

#### 3.1 Actividades de reflexión inicial:

Descripción de la actividad:

Entra a cualquier tienda en línea (Mercado Libre, Amazon, Falabella) y usa el buscador. ¿Cómo filtraste los resultados? ¿Pudiste filtrar por categoría, precio o marca? ¿Crees que los resultados estaban todos en una sola página o divididos? Reflexiona sobre cómo funciona esa búsqueda por detrás y escribe tus ideas.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Lluvia de ideas; discusión guiada; preguntas orientadoras; trabajo en parejas.



Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 1 hora.

### 3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje:

Descripción de la actividad:

Repaso: QuerySets y filter()

Recuerda que `Producto.objects.all()` devuelve un `QuerySet` con todos los productos. Para obtener solo algunos, Django ofrece el método `filter()`, que recibe condiciones en forma de `campo=valor`.

Por ejemplo, para buscar productos cuyo nombre contenga cierto texto (sin importar mayúsculas o minúsculas), se usa el lookup `icontains`:

```
# Busca productos cuyo nombre contenga la palabra 'audífonos'
Producto.objects.filter(nombre__icontains='audífonos')
```

`icontains`: "contains" (contiene) + "i" de insensitive (no distingue mayúsculas/minúsculas).

`nombre__icontains='texto'`: el doble guion bajo (`__`) conecta el nombre del campo con el tipo de búsqueda (lookup).

Leyendo lo que el usuario escribió: `request.GET`

Cuando un formulario usa `method="GET"` (como un buscador), los datos enviados aparecen en la URL como parámetros, por ejemplo `/productos/lista/?q=audífonos`. Django nos permite leer esos parámetros con `request.GET.get('q')`:

```
def lista_productos(request):
    query = request.GET.get('q')
    productos = Producto.objects.all()

    if query:
        productos = productos.filter(nombre__icontains=query)
```



```
contexto = {'productos': productos, 'query': query}
return render(request, 'productos/lista.html', contexto)
```

request.GET.get('q'): obtiene el valor del parámetro q en la URL. Si no existe, devuelve None.

if query: solo se filtra si el usuario realmente escribió algo en el buscador.

El formulario de búsqueda en el template

Un formulario de búsqueda usa method="GET" (no necesita {% csrf\_token %}), porque GET no modifica datos):

```
<form method="GET">
  <input type="text" name="q" value="{{ query|default:'' }}" placeholder="Buscar
producto...">
  <button type="submit">Buscar</button>
</form>
```

name="q": debe coincidir con el nombre del parámetro que lees con request.GET.get('q').

value="{{ query|default:'' }}": mantiene escrito en el buscador el último término buscado.

Pasos para crear el buscador

En views.py, modifica lista\_productos para leer request.GET.get('q').

Si query existe, filtra el QuerySet con .filter(nombre\_\_icontains=query).

Agrega query al contexto que se envía al template.

En lista.html, agrega el formulario de búsqueda con method="GET".

---

Filtrar usando relaciones (ForeignKey)

Recuerda que cada Producto tiene una categoría (ForeignKey hacia Categoría). Para filtrar productos por categoría, podemos usar el id de la categoría:

```
Producto.objects.filter(categoria_id=2)
```

Un menú desplegable (select) para elegir categoría

Para que el usuario elija una categoría, necesitamos enviarle al template la lista de categorías disponibles, y mostrarla en un <select>:



```
from .models import Producto, Categoria

def lista_productos(request):
    query = request.GET.get('q')
    categoria_id = request.GET.get('categoria')
    productos = Producto.objects.all()
    categorias = Categoria.objects.all()

    if query:
        productos = productos.filter(nombre__icontains=query)

    if categoria_id:
        productos = productos.filter(categoria_id=categoria_id)

    contexto = {
        'productos': productos,
        'categorias': categorias,
        'query': query,
        'categoria_id': categoria_id,
    }
    return render(request, 'productos/lista.html', contexto)
```

En el template, recorreremos categorías para generar las opciones del <select>, y marcamos como seleccionada la categoría que el usuario eligió:

```
<form method="GET">
  <input type="text" name="q" value="{{ query|default:'' }}" placeholder="Buscar
  producto...">

  <select name="categoria">
    <option value="">Todas las categorías</option>
    {% for cat in categorias %}
    <option value="{{ cat.id }}"
      {% if categoria_id == cat.id|stringformat:'s' %}selected{% endif %}>
      {{ cat.nombre }}
    </option>
    {% endfor %}
  </select>

  <button type="submit">Filtrar</button>
</form>

{% if categoria_id == cat.id|stringformat:'s' %}: compara el id de la categoría (convertido a
texto) con el valor recibido en la URL, para dejar la opción seleccionada.
```

Pasos para agregar el filtro por categoría





En views.py, lee request.GET.get('categoria').

Si categoria\_id existe, filtra el QuerySet con .filter(categoria\_id=categoria\_id).

Envía categorías (todas las categorías) y categoria\_id al contexto.

En lista.html, agrega el <select> con las categorías, dentro del mismo formulario de búsqueda.

---

La clase Paginator

Django incluye la clase Paginator, que toma un QuerySet (o cualquier lista) y un número de elementos por página, y se encarga de dividirlo en “páginas”:

```
from django.core.paginator import Paginator

def lista_productos(request):
    productos = Producto.objects.all()
    # ... aplicar filtros de búsqueda y categoría ...

    paginator = Paginator(productos, 5) # 5 productos por página
    numero_pagina = request.GET.get('page')
    page_obj = paginator.get_page(numero_pagina)

    contexto = {'page_obj': page_obj}
    return render(request, 'productos/lista.html', contexto)
```

Paginator(productos, 5): crea un paginador que reparte el QuerySet en grupos de 5 elementos.

request.GET.get('page'): lee el número de página desde la URL, por ejemplo ?page=2.

paginator.get\_page(numero\_pagina): devuelve un objeto Page con los productos de esa página. Si el número no es válido, devuelve la primera o la última página automáticamente, sin generar errores.

Usando page\_obj en el template

page\_obj se comporta como una lista (puedes recorrerla con {% for %}), pero además tiene información útil sobre la paginación:

page\_obj.number: el número de la página actual.

page\_obj.has\_previous() / page\_obj.has\_next(): indican si existe una página anterior o siguiente.



page\_obj.previous\_page\_number / page\_obj.next\_page\_number: el número de esas páginas.

page\_obj.paginator.num\_pages: el total de páginas.

```
{% for producto in page_obj %}
  <!-- fila de la tabla con producto.nombre, producto.precio, etc. -->
{% endfor %}

<div class="paginacion">
  {% if page_obj.has_previous %}
    <a href="?page={{ page_obj.previous_page_number }}">Anterior</a>
  {% endif %}

  <span>Página {{ page_obj.number }} de {{ page_obj.paginator.num_pages }}</span>

  {% if page_obj.has_next %}
    <a href="?page={{ page_obj.next_page_number }}">Siguiente</a>
  {% endif %}
</div>
```

Pasos para agregar paginación

En views.py, importa Paginator desde django.core.paginator.

Crea el paginador con el QuerySet ya filtrado y un tamaño de página (por ejemplo, 5).

Obtén page\_obj usando request.GET.get('page').

En lista.html, cambia el {% for %} para recorrer page\_obj en lugar de productos.

Agrega los enlaces “Anterior” y “Siguiente” usando los métodos de page\_obj.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.



Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 4 horas.

### 3.3 Actividades de apropiación:

Descripción de la actividad:

Agrega un buscador en tu catálogo de productos que permita filtrar por nombre. Prueba buscando un producto que exista (debe aparecer) y uno que no exista (debe mostrarse el mensaje de “no hay productos”, igual que en la guía de Templates).

Agrega el filtro por categoría a tu catálogo.

Verifica que el buscador por nombre y el filtro por categoría funcionen juntos (por ejemplo, buscar “audífonos” dentro de la categoría “Tecnología”).

Implementa la paginación en tu catálogo, mostrando un máximo de 5 productos por página.

Agrega al menos 8 productos a tu base de datos (puedes usar el formulario “Agregar producto” de la guía de Formularios) para poder ver más de una página.

Verifica que los enlaces “Anterior” y “Siguiente” funcionen correctamente, y que no aparezcan si no hay página anterior o siguiente.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.



Evidencias de aprendizaje: Código fuente funcional del proyecto Django con las funcionalidades implementadas. Capturas de pantalla que evidencien el correcto funcionamiento en el navegador.

Instrumentos de evaluación: Lista de chequeo de funcionalidades implementadas. Revisión de código por pares.

Duración de la actividad: 6 horas.

### **3.4 Actividades de Transferencia el Conocimiento:**

Descripción de la actividad:

Asegúrate de que la vista `lista_productos` aplique, en este orden: filtro por nombre (si hay `q`), filtro por categoría (si hay categoría) y, al final, la paginación.

Modifica los enlaces “Anterior” y “Siguiete” para que incluyan los parámetros `q` y categoría, además de `page`.

Prueba el flujo completo: busca un término, elige una categoría, y navega entre páginas verificando que el buscador y el filtro NO se pierdan.

Verifica qué ocurre si escribes manualmente en la URL un número de página que no existe (por ejemplo, `?page=999`): la aplicación no debe mostrar un error.

#### **PARTE TEÓRICA:**

Responde las siguientes preguntas:

¿Qué hace el `lookup icontains` y en qué se diferencia de usar simplemente `contains`?

¿Por qué el formulario de búsqueda usa `method="GET"` y no `method="POST"`?

¿Qué problema resuelve la clase `Paginator` y qué pasaría si mostraras 500 productos en una sola página sin paginación?

¿Qué información ofrece el objeto `page_obj` además de la lista de elementos de la página actual?

¿Por qué es importante conservar los parámetros `q` y categoría en los enlaces de paginación? ¿Qué pasaría si no se conservaran?



Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Actividad práctica con capturas de pantalla. Respuestas escritas a las preguntas teóricas (entregadas en documento o plataforma del instructor).

Instrumentos de evaluación: Rúbrica de evaluación de desempeño. Cuestionario de preguntas abiertas.

Duración de la actividad: 3 horas.

#### 4. PLANTEAMIENTO DE EVIDENCIAS DE APRENDIZAJE PARA LA EVALUACIÓN EN EL PROCESO FORMATIVO.

| Fase del proyecto formativo | Actividad del proyecto formativo | Actividad de Aprendizaje | Evidencias de Aprendizaje | Criterios de Evaluación | Técnicas e Instrumentos de Evaluación |
|-----------------------------|----------------------------------|--------------------------|---------------------------|-------------------------|---------------------------------------|
|                             |                                  |                          |                           |                         |                                       |

Evidencia de Conocimiento: Respuestas a las preguntas teóricas de la actividad evaluativa (sección 3.4.2).

Evidencia de Desempeño: Implementación funcional de todas las actividades prácticas indicadas en las secciones 3.3 y 3.4.1, demostrada mediante capturas de pantalla y revisión del código fuente.



Evidencia de Producto: Proyecto Django actualizado con las funcionalidades de la guía correctamente implementadas, almacenado en repositorio Git.

## 5. GLOSARIO DE TÉRMINOS

Django: Framework web de alto nivel para Python que fomenta el desarrollo rápido y seguro.

QuerySet: Colección de objetos obtenidos de la base de datos mediante el ORM de Django, sobre la cual se pueden aplicar filtros.

filter(): Método de un QuerySet que devuelve solo los objetos que cumplen una condición.

Lookup (icontains, exact, etc.): Forma especial de escribir una condición de filtro, conectada al nombre del campo con doble guion bajo (\_\_).

request.GET: Diccionario con los parámetros enviados en la URL mediante un formulario con method="GET".

Parámetro de consulta (query parameter): Valor enviado en la URL después de un signo de interrogación, por ejemplo ?q=audifonos.

ForeignKey: Campo de un modelo que crea una relación (muchos a uno) con otro modelo.

Paginator: Clase de Django que divide un QuerySet o lista en grupos (páginas) de un tamaño definido.

Page object (page\_obj): Objeto que representa una página específica generada por un Paginator, con información de navegación.

has\_next() / has\_previous(): Métodos de un page\_obj que indican si existe una página siguiente o anterior.

`{% if %} / {% else %} / {% endif %}`: Etiquetas DTL para mostrar contenido condicional dentro de un template.

`{% for %} / {% endfor %}`: Etiquetas DTL para recorrer una lista o QuerySet dentro de un template.

## 6. REFERENTES BIBLIOGRÁFICOS

Django Software Foundation. (2024). Django documentation.  
<https://docs.djangoproject.com/>



Mozilla Developer Network. (2024). Django Web Framework (Python).  
<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>

Holovaty, A. & Kaplan-Moss, J. (2009). The Definitive Guide to Django. Apress.

Vincent, W. S. (2022). Django for Beginners. Leanpub.

Python Software Foundation. (2024). The Python Tutorial.  
<https://docs.python.org/3/tutorial/>

## 7. CONTROL DEL DOCUMENTO

|                   | Nombre                    | Cargo      | Dependencia | Fecha      |
|-------------------|---------------------------|------------|-------------|------------|
| <b>Autor (es)</b> | Jheyson Fabian Villavisan | Instructor | CDM         | 25/06/2026 |

## 8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

|                   | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
|-------------------|--------|-------|-------------|-------|------------------|
| <b>Autor (es)</b> |        |       |             |       |                  |



## PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

### FORMATO GUÍA DE APRENDIZAJE

#### 1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en programación de Software
- Código del Programa de Formación: 233104-V2
- Nombre del Proyecto Formativo (si aplica): DESARROLLO DE APLICACIONES DE SOFTWARE PARA EL SECTOR EMPRESARIAL
- Fase del Proyecto (si aplica): Ejecución
- Actividad de Proyecto Formativo (si aplica): Desarrollar el software de acuerdo con el diseño y a la programación establecida
- Competencia: 220501096 DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y LA PROGRAMACIÓN ESTABLECIDA
- Resultados de Aprendizaje: Rap 3. CODIFICAR EL SOFTWARE EMPLEANDO EL LENGUAJE DE PROGRAMACIÓN SEGÚN REQUERIMIENTO Y DISEÑO
- Duración de la Guía de Aprendizaje (horas): 72 horas (24 sincrónicas + 48 asincrónicas)

#### 2. PRESENTACIÓN

Hasta ahora tu aplicación maneja productos y categorías: una categoría puede tener muchos productos, pero cada producto pertenece a una sola categoría. Esa es una relación uno a muchos, y la resolviste con ForeignKey en la guía de Migraciones y Relaciones.

Pero piensa en cómo funciona un carrito de compras en Mercado Libre, Amazon o Rappi: un pedido puede contener varios productos distintos (audífonos, un cargador y una mochila, por ejemplo), y al mismo tiempo, un mismo producto (esos audífonos) puede aparecer en muchos pedidos diferentes, hechos por distintos clientes. Aquí ya no es “uno a muchos”, sino muchos a muchos: muchos pedidos se relacionan con muchos productos.





En esta guía vas a aprender a modelar y construir esa relación: crearás un modelo Pedido, y usando un modelo intermedio podrás registrar qué productos (y en qué cantidad) tiene cada pedido. Al final, tu aplicación podrá crear pedidos, agregarles productos, calcular el total a pagar y mostrar el detalle completo de cada pedido.

En esta guía vas a:

Entender qué es una relación muchos a muchos y por qué se necesita un modelo intermedio cuando hay información extra (como la cantidad).

Crear los modelos Pedido y PedidoDetalle, y sus migraciones.

Construir una vista para crear pedidos y agregarles productos.

Mostrar el detalle de un pedido: productos, cantidades, subtotales y total.

Listar todos los pedidos y permitir eliminar un producto de un pedido.

Al finalizar esta guía, tu aplicación de productos habrá dado un salto enorme: ya no solo mostrará un catálogo, sino que permitirá simular pedidos completos, como una mini tienda en línea. ¡Vamos a construirlo!

### 3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**

#### 3.1 Actividades de reflexión inicial:

Descripción de la actividad:

Piensa en un pedido real de comida a domicilio: un solo pedido puede tener hamburguesa, papas y gaseosa. ¿Cómo crees que la app guarda esa información? ¿Un producto puede estar en varios pedidos distintos al mismo tiempo? Dibuja o describe cómo imaginas la relación entre Pedido y Producto en una base de datos.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.



Estrategias o técnicas didácticas activas: Lluvia de ideas; discusión guiada; preguntas orientadoras; trabajo en parejas.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 1 hora.

### **3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje:**

Descripción de la actividad:

Repaso: relación uno a muchos (ForeignKey)

En la guía de Migraciones y Relaciones, cada Producto tenía un campo categoria de tipo ForeignKey. Esto significa: una categoría puede tener muchos productos, pero cada producto apunta a UNA sola categoría.

¿Por qué un pedido y sus productos son “muchos a muchos”?

Un pedido puede tener varios productos.

Un producto puede estar en varios pedidos.

Django ofrece un campo llamado ManyToManyField justamente para este tipo de relaciones. Sin embargo, en nuestro caso necesitamos guardar información adicional sobre la relación: la cantidad de cada producto en el pedido. Por ejemplo, un pedido puede tener 2 audífonos y 1 mochila: el “2” y el “1” son datos de la RELACIÓN, no del producto ni del pedido por sí solos.

Cuando una relación muchos a muchos necesita información adicional, en lugar de usar ManyToManyField directamente, se crea un modelo intermedio (también llamado tabla intermedia o through model): un modelo que tiene un ForeignKey hacia cada uno de los modelos relacionados, más los campos extra que se necesiten.

Los modelos Pedido y PedidoDetalle



```
from django.db import models

class Pedido(models.Model):
    cliente = models.CharField(max_length=100)
    fecha = models.DateField(auto_now_add=True)

    def __str__(self):
        return f'Pedido #{self.id} - {self.cliente}'

class PedidoDetalle(models.Model):
    pedido = models.ForeignKey(Pedido, on_delete=models.CASCADE)
    producto = models.ForeignKey(Producto, on_delete=models.PROTECT)
    cantidad = models.PositiveIntegerField()

    def subtotal(self):
        return self.producto.precio * self.cantidad

    def __str__(self):
        return f'{self.cantidad} x {self.producto.nombre}'
```

Pedido: representa el pedido en sí, con el cliente y la fecha (auto\_now\_add=True guarda la fecha automáticamente al crearlo, igual que viste en la guía de Models).

PedidoDetalle: es el modelo intermedio. Cada fila representa “este producto, en esta cantidad, dentro de este pedido”.

on\_delete=models.CASCADE en pedido: si se elimina un Pedido, se eliminan automáticamente sus detalles.

on\_delete=models.PROTECT en producto: no se puede eliminar un Producto si todavía aparece en algún PedidoDetalle (recuerda las opciones de on\_delete de la guía de Migraciones).

subtotal(): un método que calcula precio × cantidad para esa línea del pedido.

Pasos para crear los modelos

En models.py, agrega las clases Pedido y PedidoDetalle como en el ejemplo.

Ejecuta python manage.py makemigrations para generar el archivo de migración.

Ejecuta python manage.py migrate para crear las tablas en la base de datos.

Registra ambos modelos en admin.py para poder revisarlos desde el panel de administración.



---

## Formularios para Pedido y PedidoDetalle

Igual que en la guía de Formularios, usamos ModelForm para construir los formularios automáticamente:

```
from django import forms
from .models import Pedido, PedidoDetalle

class PedidoForm(forms.ModelForm):
    class Meta:
        model = Pedido
        fields = ['cliente']

class PedidoDetalleForm(forms.ModelForm):
    class Meta:
        model = PedidoDetalle
        fields = ['producto', 'cantidad']
```

PedidoForm: solo pide el campo cliente (la fecha se llena sola, gracias a auto\_now\_add).

PedidoDetalleForm: pide producto (Django mostrará automáticamente un <select> con todos los productos, gracias al ForeignKey) y cantidad.

## Vista para crear un pedido

```
from django.shortcuts import render, redirect
from .forms import PedidoForm

def crear_pedido(request):
    if request.method == 'POST':
        form = PedidoForm(request.POST)
        if form.is_valid():
            pedido = form.save()
            return redirect('detalle_pedido', pedido_id=pedido.id)
        else:
            form = PedidoForm()

    return render(request, 'productos/crear_pedido.html', {'form': form})
```

Observa que form.save() devuelve el objeto recién creado (el pedido). Usamos pedido.id para redirigir directamente a la página de detalle de ESE pedido, donde el usuario podrá empezar a agregarle productos.



Vista para agregar un producto al pedido

```
from django.shortcuts import render, redirect, get_object_or_404
from .models import Pedido
from .forms import PedidoDetalleForm

def agregar_producto_pedido(request, pedido_id):
    pedido = get_object_or_404(Pedido, id=pedido_id)

    if request.method == 'POST':
        form = PedidoDetalleForm(request.POST)
        if form.is_valid():
            detalle = form.save(commit=False)
            detalle.pedido = pedido
            detalle.save()
            return redirect('detalle_pedido', pedido_id=pedido.id)
        else:
            form = PedidoDetalleForm()

    return render(request, 'productos/agregar_producto.html', {'form': form, 'pedido':
pedido})
```

`form.save(commit=False)`: crea el objeto `PedidoDetalle` EN MEMORIA, pero todavía no lo guarda en la base de datos. Esto nos da la oportunidad de completar datos antes de guardar.

`detalle.pedido = pedido`: asignamos a qué pedido pertenece este detalle (el formulario no pide este dato porque ya lo sabemos: viene en la URL).

`detalle.save()`: ahora sí se guarda en la base de datos, con el pedido ya asignado.

Pasos para crear estas vistas

Crea el archivo `forms.py` (o agrega a tu archivo existente) las clases `PedidoForm` y `PedidoDetalleForm`.

Crea la vista `crear_pedido` y su template `crear_pedido.html` (similar a `crear.html` de la guía de Formularios).

Crea la vista `agregar_producto_pedido` y su template `agregar_producto.html`.

En `urls.py`, agrega las rutas `'pedidos/crear/'` y `'pedidos/<int:pedido_id>/agregar/'`, con sus respectivos `name`.

---



related\_name y el “conjunto relacionado”

Cuando un modelo (PedidoDetalle) tiene un ForeignKey hacia otro (Pedido), Django nos permite, desde un objeto Pedido, acceder a TODOS los PedidoDetalle relacionados, usando NombreModelo\_set en minúsculas:

```
pedido = Pedido.objects.get(id=1)
detalles = pedido.pedidodetalle_set.all() # todos los PedidoDetalle de este pedido
```

Vista de detalle de pedido

```
from django.shortcuts import render, get_object_or_404
from .models import Pedido

def detalle_pedido(request, pedido_id):
    pedido = get_object_or_404(Pedido, id=pedido_id)
    detalles = pedido.pedidodetalle_set.all()

    total = 0
    for detalle in detalles:
        total += detalle.subtotal()

    contexto = {
        'pedido': pedido,
        'detalles': detalles,
        'total': total,
    }
    return render(request, 'productos/detalle_pedido.html', contexto)
```

Template detalle\_pedido.html

```
{% extends 'productos/base.html' %}

{% block titulo %}Detalle del pedido{% endblock %}

{% block content %}
<h2>{{ pedido }}</h2>
<p>Fecha: {{ pedido.fecha }}</p>

<table border="1">
  <tr>
    <th>Producto</th>
    <th>Cantidad</th>
    <th>Precio unitario</th>
    <th>Subtotal</th>
  </tr>
```



```
{% for detalle in detalles %}
<tr>
  <td>{{ detalle.producto.nombre }}</td>
  <td>{{ detalle.cantidad }}</td>
  <td>${{ detalle.producto.precio|floatformat:2 }}</td>
  <td>${{ detalle.subtotal|floatformat:2 }}</td>
</tr>
{% endfor %}
</table>

<h3>Total: ${{ total|floatformat:2 }}</h3>

<a href="{% url 'agregar_producto_pedido' pedido.id %}">Agregar producto</a>
{% endblock %}
```

Fíjate que `detalle.producto.nombre` y `detalle.producto.precio` acceden a campos del Producto relacionado, igual que `producto.categoria.nombre` en la guía de Templates: simplemente “se viaja” por el ForeignKey usando puntos.

Pasos para crear la vista de detalle

Crea la vista `detalle_pedido` siguiendo el ejemplo, calculando el total con un ciclo for en Python.

En `urls.py`, agrega la ruta `'pedidos/<int:pedido_id>/'` con `name='detalle_pedido'`.

Crea el template `detalle_pedido.html` mostrando la tabla de productos del pedido y el total.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Duración de la actividad: 4 horas.



### 3.3 Actividades de apropiación:

Descripción de la actividad:

Crea los modelos Pedido y PedidoDetalle en tu proyecto, genera y aplica las migraciones, y regístralos en el panel de administración. Verifica en /admin/ que ambas tablas existan y que puedas crear un pedido manualmente desde ahí.

Implementa crear\_pedido y agregar\_producto\_pedido junto con sus templates y rutas.

Crea un pedido nuevo desde el navegador y agrégale al menos tres productos distintos, con diferentes cantidades.

Implementa detalle\_pedido y su template.

Verifica que el total mostrado sea la suma correcta de todos los subtotales del pedido.

Agrega un producto más al pedido desde el enlace “Agregar producto” y confirma que el total se actualice.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.

Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Código fuente funcional del proyecto Django con las funcionalidades implementadas. Capturas de pantalla que evidencien el correcto funcionamiento en el navegador.





Instrumentos de evaluación: Lista de chequeo de funcionalidades implementadas. Revisión de código por pares.

Duración de la actividad: 6 horas.

### 3.4 Actividades de Transferencia el Conocimiento:

Descripción de la actividad:

Implementa la vista `lista_pedidos`, su ruta `'pedidos/'` (`name='lista_pedidos'`) y su template `lista_pedidos.html`, mostrando el cliente, la fecha y un enlace al detalle de cada pedido.

Implementa la vista `eliminar_detalle_pedido` y su ruta `'pedidos/detalle/<int:detalle_id>/eliminar/'` (`name='eliminar_detalle_pedido'`).

En `detalle_pedido.html`, agrega un enlace “Quitar” junto a cada producto del pedido.

Protege las vistas `crear_pedido`, `agregar_producto_pedido` y `eliminar_detalle_pedido` con `@login_required` (recuerda la guía de Autenticación).

Prueba el flujo completo: crea un pedido, agrégale varios productos, quita uno, y verifica que el total se recalcule correctamente.

#### PARTE TEÓRICA:

Responde las siguientes preguntas:

Explica con tus palabras por qué la relación entre Pedido y Producto es “muchos a muchos” y no “uno a muchos”.

¿Por qué se creó el modelo `PedidoDetalle` en lugar de usar directamente un `ManyToManyField` entre Pedido y Producto?

¿Qué hace `pedido.pedidodetalle_set.all()` y de dónde viene ese nombre?

¿Qué diferencia hay entre `form.save()` y `form.save(commit=False)`? ¿En qué situación usarías cada uno?

En el modelo `PedidoDetalle`, ¿qué pasaría si el campo `producto` tuviera `on_delete=models.CASCADE` en lugar de `PROTECT`? Explica con un ejemplo.

Ambiente requerido: Ambiente de trabajo individual o en parejas con acceso a computador con Python 3.x instalado, entorno virtual activo, proyecto Django configurado y conexión a internet para consultar documentación.



Estrategias o técnicas didácticas activas: Aprendizaje basado en proyectos (ABP); lectura analítica de documentación técnica; resolución de problemas por indagación; trabajo colaborativo entre pares; retroalimentación formativa.

Materiales de formación Computador con Python 3, Django y las librerías requeridas instaladas; editor de código (VS Code o similar); navegador web; terminal o consola del sistema.

Material de apoyo: Documentación oficial de Django (<https://docs.djangoproject.com/>); Documentación de Python (<https://docs.python.org/3/>); guías y videos del instructor; repositorio del proyecto en GitHub.

Evidencias de aprendizaje: Actividad práctica con capturas de pantalla. Respuestas escritas a las preguntas teóricas (entregadas en documento o plataforma del instructor).

Instrumentos de evaluación: Rúbrica de evaluación de desempeño. Cuestionario de preguntas abiertas.

Duración de la actividad: 3 horas.

#### 4. PLANTEAMIENTO DE EVIDENCIAS DE APRENDIZAJE PARA LA EVALUACIÓN EN EL PROCESO FORMATIVO.

| Fase del proyecto formativo | Actividad del proyecto formativo | Actividad de Aprendizaje | Evidencias de Aprendizaje | Criterios de Evaluación | Técnicas e Instrumentos de Evaluación |
|-----------------------------|----------------------------------|--------------------------|---------------------------|-------------------------|---------------------------------------|
|                             |                                  |                          |                           |                         |                                       |

Evidencia de Conocimiento: Respuestas a las preguntas teóricas de la actividad evaluativa (sección 3.4.2).

Evidencia de Desempeño: Implementación funcional de todas las actividades prácticas indicadas en las secciones 3.3 y 3.4.1, demostrada mediante capturas de pantalla y revisión del código fuente.

Evidencia de Producto: Proyecto Django actualizado con las funcionalidades de la guía correctamente implementadas, almacenado en repositorio Git.

#### 5. GLOSARIO DE TÉRMINOS

Django: Framework web de alto nivel para Python que fomenta el desarrollo rápido y seguro.



**ForeignKey:** Campo de un modelo que crea una relación (muchos a uno) con otro modelo.

**Relación muchos a muchos (M2M):** Relación en la que muchos registros de un modelo se asocian con muchos registros de otro modelo.

**ManyToManyField:** Campo de Django para crear relaciones muchos a muchos directamente, sin necesidad de un modelo intermedio.

**Modelo intermedio (through model):** Modelo adicional que conecta dos modelos en una relación muchos a muchos y permite guardar información extra sobre la relación.

**auto\_now\_add:** Opción de un campo de fecha que guarda automáticamente la fecha y hora actuales al crear el registro.

**related\_name / conjunto relacionado (\_set):** Forma de acceder, desde un objeto, a todos los registros relacionados mediante un ForeignKey hacia ese objeto.

**ModelForm:** Tipo especial de Django Form que genera automáticamente los campos de un formulario a partir de un modelo.

**form.save(commit=False):** Crea el objeto a partir de un formulario válido sin guardarlo todavía en la base de datos, para poder completarlo antes de guardar.

**get\_object\_or\_404:** Función que busca un objeto en la base de datos por su identificador o muestra automáticamente un error 404 si no existe.

**@login\_required:** Decorador que obliga a que una vista solo pueda usarse si el usuario tiene una sesión iniciada.

## 6. REFERENTES BIBLIOGRÁFICOS

Django Software Foundation. (2024). Django documentation.  
<https://docs.djangoproject.com/>

Mozilla Developer Network. (2024). Django Web Framework (Python).  
<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>

Holovaty, A. & Kaplan-Moss, J. (2009). The Definitive Guide to Django. Apress.

Vincent, W. S. (2022). Django for Beginners. Leanpub.

Python Software Foundation. (2024). The Python Tutorial.  
<https://docs.python.org/3/tutorial/>



## 7. CONTROL DEL DOCUMENTO

|                   | Nombre                    | Cargo      | Dependencia | Fecha      |
|-------------------|---------------------------|------------|-------------|------------|
| <b>Autor (es)</b> | Jheyson Fabian Villavisan | Instructor | CDM         | 25/06/2026 |

## 8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

|                   | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
|-------------------|--------|-------|-------------|-------|------------------|
| <b>Autor (es)</b> |        |       |             |       |                  |





|  |                    | 3450493          |               |                                     |                        |                          |          |                                 |                  |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
|-----------------------------------------------------------------------------------|--------------------|------------------|---------------|-------------------------------------|------------------------|--------------------------|----------|---------------------------------|------------------|-------------------------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|--------------|------------|-----------|--|
| Nombre Instructor                                                                 |                    | Tipo             | No Documento: | No. Ficha                           | Fecha Inicio trimestre |                          | REGIONAL |                                 | DISTRITO CAPITAL |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| PREYSON FABIAN VILLALBA RUTRAGO CARLO CARRELLO                                    |                    | C.C.             | 1017356593    | GRADO                               | DECIMO                 | Fecha fin trimestre      |          | CENTRO DE FORMACION:            |                  | CENTRO DE DISEÑO Y METROLOGIA |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| DADO DE LA RESERVA DE FORMACION                                                   |                    | Institución      |               | PROGRAMA DE FORMACION               |                        | INSTITUCION EDUCATIVA    |          | COLEGIO MAGDALENA ORTEGA NARIÑO |                  | DOCENTE ENLACE                |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| LUNES VIERNES                                                                     |                    | DURANA           |               | TÉCNICO EN PROGRAMACIÓN DE SOFTWARE |                        |                          |          | DAY SEYI ROCHA                  |                  |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| Competencia                                                                       |                    |                  |               |                                     |                        | Resultado de Aprendizaje |          |                                 |                  |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
|                                                                                   |                    |                  |               |                                     |                        |                          |          |                                 |                  |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| Nº.                                                                               | APRENDIZAJES       |                  | TIPO          | DOCUMENTO                           | ESTADO                 | ASISTO                   | TARDE    | FALLA S                         | CON SANGRA       | EN SANGRA                     | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA | FECHA SESION | TIPO FALLA |           |  |
| 1.                                                                                | PRETO MARIN        | ANAY LETITIE     | T.I           | 1033748139                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             | 7/2/2026     |            | 7/10/2/26    |            | 7/10/2/26    |            | 7/17/2/26    |            | 7/24/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26    |            | 7/27/2/26 |  |
| 2.                                                                                | DURAN CLESTAS      | INGRE TATIANA    | T.I           | 1032943629                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 3.                                                                                | MELO MORENO        | ABRALEY GABRIELA | T.I           | 1019607832                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 4.                                                                                | ESCORIAL CHILE     | CAMILA ANDREA    | T.I           | 1015363805                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 5.                                                                                | POVEDA VERRERA     | CAROL GABRIELA   | T.I           | 1141752026                          | FORMACION              | 0                        | 0        | 5                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 6.                                                                                | CHIZ BOTINA        | DESDORA ANDRERNA | T.I           | 1080582524                          | FORMACION              | 5                        | 0        | 0                               | 1                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 7.                                                                                | PEREZ CUEVO        | DESDORA ANDRERNA | PPT           | 10211962                            | FORMACION              | 4                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 8.                                                                                | LOZANO GARCIA      | EMERY NICOL      | T.I           | 1014241383                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 9.                                                                                | HERRERA BURDISO    | EVELYN SOFIA     | T.I           | 1014944212                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 10.                                                                               | ELIZABETH MARTINEZ | GABRIELA         | T.I           | 1141717812                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 11.                                                                               | FLORENI LOPEZ      | OSIEL JAJUAN     | T.I           | 1025145607                          | FORMACION              | 4                        | 0        | 1                               | 1                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 12.                                                                               | JOTA METO          | LOREN IBABELLA   | T.I           | 1011522380                          | FORMACION              | 5                        | 0        | 0                               | 0                | 0                             |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |
| 13.                                                                               | FOLIANA ORTIZ      | LUISA            | T.I           |                                     |                        |                          |          |                                 |                  |                               |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |              |            |           |  |

[illegible]



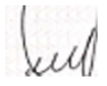
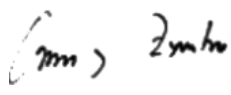
| ACTA No. 20                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                     |                                                    |                      |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|----------------------------------------------------|----------------------|
| NOMBRE DEL COMITÉ O DE LA REUNIÓN: Capacitación Pedagógica Equipo Articulación con la Educación Media.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                     |                                                    |                      |
| CIUDAD Y FECHA:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 1-Julio de 2026                     | HORA INICIO:<br>8:00 AM                            | HORA FIN:<br>14:00PM |
| LUGAR Y/O ENLACE:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Centro de Diseño y Metrología SENA. | DIRECCIÓN / REGIONAL / CENTRO: Diseño y Metrología |                      |
| AGENDA O PUNTOS PARA DESARROLLAR:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                     |                                                    |                      |
| <ul style="list-style-type: none"><li>• Generación de conceptos sobre metodológica, trabajo en equipo y pedagogía.</li><li>• Actividad Aula LEGO.</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                     |                                                    |                      |
| OBJETIVO(S) DE LA REUNIÓN:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                                     |                                                    |                      |
| <ul style="list-style-type: none"><li>• Establecer los parámetros para el desarrollo pedagógico y metodológico necesarios para la formación, así como los conceptos de trabajo en equipo y trabajo grupal.</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                     |                                                    |                      |
| DESARROLLO DE LA REUNIÓN                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                     |                                                    |                      |
| <ul style="list-style-type: none"><li>• Se realiza una presentación inicial por parte del dinamizador del programa Jhon altamar donde expresa diferentes conceptos referentes a la pedagogía, metodología y procesos de trabajo en grupo.</li><li>• Se generan conceptos base en la inclusión de personal académico en formación.</li><li>• Se determinan formas de trabajo como la metodología STEM, como proceso de desarrollo en el aula mediante micro proyectos.</li><li>• Se realiza un traslado al aula LEGO, donde se establecen grupos de trabajo y permite así socializar el trabajo STEM, en el aula como método pedagógico.</li><li>• En los grupos se realiza la actividad LEGO, donde se permite la construcción y programación de un prototipo robotizado LEGO el cual según la solicitud del programador realizara una actividad.</li><li>• Se realiza una retrospectiva de cómo se debe observar la formación y al aprendiz.</li></ul> |                                     |                                                    |                      |





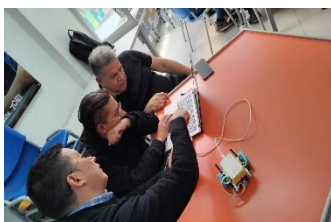
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                 |                                            |                                              |                                                                                       |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|--------------------------------------------|----------------------------------------------|---------------------------------------------------------------------------------------|
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                 |                                            |                                              |                                                                                       |
| <b>CONCLUSIONES</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                 |                                            |                                              |                                                                                       |
| <ul style="list-style-type: none"> <li>• Es de referir que aunque la experiencia determine múltiples procesos pedagógicos los cuales se asocian a la misma personalidad del formador, no siempre estas son correctas.</li> <li>• Es determinante observar las capacidades de cada quien para poder establecer nuevos procesos que generar educación y no memorización de proceso.</li> <li>• La metodología STEM, es un método claro y generacional para la comprensión de procesos técnicos según la metodología SENA.</li> </ul> |                                 |                                            |                                              |                                                                                       |
| <b>ESTABLECIMIENTO Y ACEPTACIÓN DE COMPROMISOS</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                 |                                            |                                              |                                                                                       |
| <b>ACTIVIDAD<br/>/DECISIÓN</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | <b>FECHA</b>                    | <b>RESPONSABLE</b>                         | <b>FIRMA O<br/>PARTICIPACIÓN<br/>VIRTUAL</b> |                                                                                       |
| Análisis particular de la pedagógica en la formación                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | N/A                             | Equipo Articulación con la Educación Media |                                              |                                                                                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                 |                                            |                                              |                                                                                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                 |                                            |                                              |                                                                                       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                 |                                            |                                              |                                                                                       |
| <b>DE: ASISTENTES Y APROBACIÓN DECISIONES</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                 |                                            |                                              |                                                                                       |
| <b>NOMBRE</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <b>DEPENDENCIA/<br/>EMPRESA</b> | <b>APRUEBA<br/>(SI/NO)</b>                 | <b>OBSERVACIÓN</b>                           | <b>FIRMA O<br/>PARTICIPACIÓN<br/>VIRTUAL</b>                                          |
| Jhon Alberto Altamar Rendon                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Apoyo Administrativo            | SI                                         | N/A                                          |  |
| Jheyson Fabian Villavisan Buitrago                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Instructor Articulación CDM     | SI                                         | N/A                                          |  |
| Helber Rodrigo Rojas Gacha                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Apoyo Administrativo            | SI                                         | N/A                                          |  |
| Cristian Johan Cuesta                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Instructor Articulación CDM     | SI                                         | N/A                                          |  |



|                                   |                             |    |     |                                                                                       |
|-----------------------------------|-----------------------------|----|-----|---------------------------------------------------------------------------------------|
| Guillermo Sánchez Isaza           | Instructor Articulación CDM | SI | N/A |    |
| William Mayorga G                 | Instructor Articulación CDM | SI | N/A |    |
| Nelson Oswaldo Santa Garcia       | Instructor Articulación CDM | SI | N/A |    |
| Fernando Andrés Salguero          | Instructor Articulación CDM | SI | N/A |    |
| Cristian Camilo Carrillo Cubillos | Instructor Articulación CDM | SI | N/A |    |
| Omar Oswaldo Zambrano             | Instructor Articulación CDM | SI | N/A |    |
| Washington Nieto Arce             | Instructor Articulación CDM | SI | N/A |    |
| Daniel Alberto Ardila             | Instructor Articulación CDM | SI | N/A |  |

De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.

## ANEXOS





REGISTRO DE ASISTENCIA / DÍA 01 DEL MES DE 07 DEL AÑO 202 26

| REGISTRO DE ASISTENCIA / DÍA 01 DEL MES DE 07 DEL AÑO 2022 |                              |               |        |             |             |                      |                        |               |                    |                               |
|------------------------------------------------------------|------------------------------|---------------|--------|-------------|-------------|----------------------|------------------------|---------------|--------------------|-------------------------------|
| OBJETIVO (S)                                               |                              |               |        |             |             |                      |                        |               |                    |                               |
| No                                                         | NOMBRES Y APELLIDOS          | No. DOCUMENTO | PLANTA | CONTRATISTA | OTRO ¿CUAL? | DEPENDENCIA/ EMPRESA | CORREO ELECTRÓNICO     | TELÉFONO/EXT. | AUTORIZA GRABACIÓN | FIRMA O PARTICIPACIÓN VIRTUAL |
| 1                                                          | WASHINGTON N.ieto Ortiz      | 99289525      |        | X           |             | CDM                  | unireto@senae.edu.co   | 313365093     | SI                 |                               |
| 2                                                          | Guillermo Solís Izazá        | 79300190      |        | X           |             | CDM                  | gsolis215@senae.edu.co | 3176752832    | SI                 |                               |
| 3                                                          | Fernando Andres Salgado Cruz | 4400484422    |        | X           |             | CDM                  | Fernandoc@senae.edu.co | 3143538744    | SI                 |                               |
| 4                                                          | Daniel Alberto Andia         | 1143361884    |        | X           |             | CDM                  | dardiana@senae.edu.co  | 3104003635    | SI                 |                               |
| 5                                                          | William Hatosuna             | 80265351      |        | X           |             | CDM                  | umatosuna@senae.edu.co | 3058861628    | SI                 |                               |
| 6                                                          | Helber D Pizarro             | 79208171      |        | X           |             | CDM                  | harpiz@senae.edu.co    | 3002828482    | SI                 |                               |
| 7                                                          | Johan Lloresta               | 1030614422    |        | X           |             | CDM                  | llorestab@senae.edu.co | 30333325      | SI                 |                               |
| 8                                                          | Nelson Santa Bruna           | 80217590      |        | X           |             | CDM                  | nsantab@senae.edu.co   | 3145202054    | SI                 |                               |
| 9                                                          | Ortiz Zumbare                | 79175531      |        | X           |             | CDM                  | ozumbare@senae.edu.co  | 318132052     | SI                 |                               |
| 10                                                         | Camilo Carrillo              | 1030639865    |        | X           |             | CDM                  | camilloc@senae.edu.co  | 3178926052    | SI                 |                               |
| 11                                                         | Jhon ALTAMAR                 | 79654213      |        | X           |             | CDM                  | jealtamar@senae.edu.co | 3158990546    | SI                 |                               |
| 12                                                         | Jheyma Villavieja            | 1012355635    |        | X           |             | CDM                  | jvillavieja            | 310484062     | SI                 |                               |
|                                                            |                              |               |        |             |             |                      |                        |               |                    |                               |
|                                                            |                              |               |        |             |             |                      |                        |               |                    |                               |
|                                                            |                              |               |        |             |             |                      |                        |               |                    |                               |

De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.



| ACTA No. 21                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                   |                                       |                               |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|---------------------------------------|-------------------------------|
| <b>NOMBRE DEL COMITÉ O DE LA REUNIÓN: GESTIÓN DE REPORTES, SOCIALIZACIÓN GUÍAS DE APRENDIZAJE, LEGALIZACIÓN HORAS INSTRUCTORES, ESTRATEGIAS PARA MANEJO DE PROYECTOS PRODUCTIVOS.</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                   |                                       |                               |
| <b>CIUDAD Y FECHA:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | <b>Bogotá, Julio 2 de 2026</b>                                                                    | <b>HORA INICIO:</b><br>2:00 p.m.      | <b>HORA FIN:</b><br>5:00 p.m. |
| <b>LUGAR Y/O ENLACE:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <b>Virtual por Teams:</b><br>REUNION EQUIPO DE ARTICULACION  <br>Reunión-Unirse   Microsoft Teams | <b>DIRECCIÓN / REGIONAL / CENTRO:</b> |                               |
| <b>AGENDA O PUNTOS PARA DESARROLLAR:</b> <ul style="list-style-type: none"><li>• Socialización de guías de aprendizaje para grados 10° y 11°.</li><li>• Observaciones sobre la asistencia y cumplimiento de compromisos contractuales.</li><li>• Socialización y directrices para el diligenciamiento del <b>Anexo 12</b> y el informe de ejecución semestral.</li><li>• Análisis de la deserción, retención y depuración de fichas en <b>Sena Sofia Plus</b>.</li><li>• Procedimiento para la evaluación de instructores y proyectos por parte de los aprendices.</li><li>• Requerimientos al equipo pedagógico y ajustes razonables.</li><li>• Seguimiento a capacitaciones (Curso inscrito y capacitación sobre acoso sexual).</li><li>• Estrategias de seguimiento al <b>Proyecto Productivo</b> y socialización de experiencias formativas.</li><li>• Legalización de horas contractuales y ajustes en la programación.</li></ul> |                                                                                                   |                                       |                               |
| <b>OBJETIVO(S) DE LA REUNIÓN:</b> <p>El propósito principal fue estandarizar los procesos de reporte de ejecución mediante el Anexo 12, coordinar las evaluaciones de fin de semestre en la plataforma Sofia Plus y definir estrategias colaborativas para el fortalecimiento de los Proyectos Productivos, asegurando el cumplimiento de las horas contractuales de los instructores.</p> <p>También socializar y validar las guías de aprendizaje desarrolladas por los instructores para el área de software.</p>                                                                                                                                                                                                                                                                                                                                                                                                                   |                                                                                                   |                                       |                               |
| <b>DESARROLLO DE LA REUNIÓN</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                                                                                   |                                       |                               |





- **Gestión de Reportes y Anexo 12:** Se presentó el Anexo 12 filtrado para el Centro de Diseño y Metrología. Este documento contiene datos críticos como horas registradas, porcentajes de retención y deserción. Se instruyó a los instructores para que utilicen estos datos en sus informes de ejecución. En caso de discrepancias entre las horas programadas y las registradas en Sofia Plus, se debe incluir una observación justificando la situación (por ejemplo, inicio tardío de fichas o eventos extemporáneos).
- **Depuración de Fichas:** Se enfatizó la necesidad de analizar las diferencias entre aprendices matriculados, en formación y en etapa productiva. El objetivo es que para el segundo semestre la información esté depurada y las cifras coincidan plenamente.
- **Evaluaciones en Sofia Plus:** Se informó sobre la inclusión de una nueva columna en los formatos: "Resultado de la evaluación realizada por los aprendices". Los instructores deben guiar a los estudiantes la próxima semana para evaluar a sus instructores (grados 10° y 11°) y el proyecto (solo grado 11°). Los resultados consolidados serán solicitados a la Coordinación Académica.
- **Legalización de Horas:** Se discutieron alternativas para los instructores que no alcanzan las 160 horas mensuales obligatorias. Las opciones incluyen la programación de cursos complementarios (como seguridad y salud en el trabajo) o el reconocimiento de hasta 40 horas por apoyo al Proyecto Productivo.
- **Estrategias Pedagógicas para Proyectos Productivos:** Varios instructores compartieron metodologías. Se destacó la asignación de la última hora de formación exclusivamente para el desarrollo de proyectos y el uso ético de IA para presentaciones. Asimismo, se propuso realizar "pre-sustentaciones" cruzadas entre diferentes instructores y colegios para preparar a los aprendices y eliminar sesgos en la evaluación.
- **Socialización de guías de aprendizaje:** Socialización de guías por parte de los diferentes grupos de instructores.

## CONCLUSIONES



- Se debe modificar el Anexo 12 de acuerdo con las indicaciones suministradas mejorando así la información de la ejecución semestral.
- Es obligatorio realizar las evaluaciones de instructor y proyecto en Sofia Plus antes del cierre del semestre.
- La revisión de las carpetas de Proyecto Productivo se realizará de manera individual y personalizada con cada instructor a partir de la próxima semana.
- Se fomentará el acompañamiento entre pares para las sustentaciones de proyectos, coordinando las visitas según la disponibilidad en los colegios.

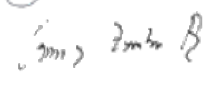
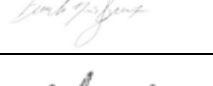
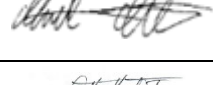
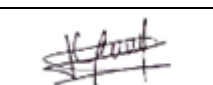
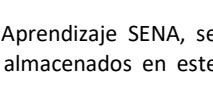
#### ESTABLECIMIENTO Y ACEPTACIÓN DE COMPROMISOS

| ACTIVIDAD /DECISIÓN                                                                                                                                                                                                                            | FECHA                                                             | RESPONSABLE                   | FIRMA O PARTICIPACIÓN VIRTUAL                                                         |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|-------------------------------|---------------------------------------------------------------------------------------|
| <p>Enviar el archivo del Anexo 12 filtrado a todo el equipo.</p> <p>Revisar y retroalimentar individualmente las carpetas de Proyecto Productivo.</p> <p>Confirmar asistencia a la capacitación presencial sobre acoso sexual (7 de julio)</p> | <p>02/07/2026</p> <p>A partir de 06/07/2026</p> <p>06/07/2026</p> | Jhon Altamar                  |  |
| Orientar a los aprendices en la evaluación de instructores y                                                                                                                                                                                   | 06 al 10 de /07/2026                                              | Omar Oswaldo Zambrano Arévalo |  |
|                                                                                                                                                                                                                                                |                                                                   | Nelson Oswaldo Santa García   |  |
|                                                                                                                                                                                                                                                |                                                                   | Fernando Andrés Salguero      |  |



|                                                                                                                                                                                                                                                                          |                      |                                    |                                                                                       |                               |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|------------------------------------|---------------------------------------------------------------------------------------|-------------------------------|
| proyectos en Sofia Plus<br><br>Entregar certificados de cursos realizados para el reporte del marco de articulación                                                                                                                                                      | 31/07/2026           | Daniel Alberto Ardila Carrasquilla |    |                               |
|                                                                                                                                                                                                                                                                          |                      | Guillermo Sánchez Isaza            |    |                               |
|                                                                                                                                                                                                                                                                          |                      | Washington Nieto Arce              |    |                               |
|                                                                                                                                                                                                                                                                          |                      | Giovanni de Jesús Montañez C       |    |                               |
|                                                                                                                                                                                                                                                                          |                      | William Mayorga Garzón             |    |                               |
|                                                                                                                                                                                                                                                                          |                      | Cristian Johan Cuesta Beltrán      |    |                               |
| Firmar documentos del Colegio Panamá (en formato digital para posterior gestión)<br><br>Orientar a los aprendices en la evaluación de instructores y proyectos en Sofia Plus<br><br>Entregar certificados de cursos realizados para el reporte del marco de articulación | 06/07/2026           | Jeyson Villavisán                  |   |                               |
|                                                                                                                                                                                                                                                                          | 06 al 10 de /07/2026 | Helber Rodrigo Rojas Gacha         |  |                               |
|                                                                                                                                                                                                                                                                          | 31/07/2026           |                                    |                                                                                       |                               |
| DE: ASISTENTES Y APROBACIÓN DECISIONES                                                                                                                                                                                                                                   |                      |                                    |                                                                                       |                               |
| NOMBRE                                                                                                                                                                                                                                                                   | DEPENDENCIA/ EMPRESA | APRUEBA (SI/NO)                    | OBSERVACIÓN                                                                           | FIRMA O PARTICIPACIÓN VIRTUAL |



|                                                   |                   |  |  |                                                                                       |
|---------------------------------------------------|-------------------|--|--|---------------------------------------------------------------------------------------|
| <b>Jhon Alberto<br/>Altamar Rendón</b>            | <b>SENA (CDM)</b> |  |  |    |
| <b>Omar Oswaldo<br/>Zambrano<br/>Arévalo</b>      | <b>SENA (CDM)</b> |  |  |    |
| <b>Jheyson Fabian<br/>Villavisán<br/>Buitrago</b> | <b>SENA (CDM)</b> |  |  |    |
| <b>Nelson Oswaldo<br/>Santa García</b>            | <b>SENA (CDM)</b> |  |  |    |
| <b>Fernando<br/>Andrés Salguero</b>               | <b>SENA (CDM)</b> |  |  |    |
| <b>Daniel Alberto<br/>Ardila<br/>Carrasquilla</b> | <b>SENA (CDM)</b> |  |  |    |
| <b>Guillermo<br/>Sánchez Isaza</b>                | <b>SENA (CDM)</b> |  |  |    |
| <b>Helber Rodrigo<br/>Rojas Gacha</b>             | <b>SENA (CDM)</b> |  |  |   |
| <b>William Mayorga<br/>Garzón</b>                 | <b>SENA (CDM)</b> |  |  |  |
| <b>Cristian Johan<br/>Cuesta Beltrán</b>          | <b>SENA (CDM)</b> |  |  |  |
| <b>Washington<br/>Nieto Arce</b>                  | <b>SENA (CDM)</b> |  |  |  |
| <b>Giovanni de<br/>J. Montañez</b>                | <b>SENA (CDM)</b> |  |  |  |

De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.





## ANEXOS

01:47:21

Búsqueda (Ctrl+Alt+E)

Chat Gente Participar Reaccionar Vista Aplicaciones Más Cámara Micro Compartir Salir

Reactivar audio del micrófono (Ctrl+Shift+M)

Guillermo Sanchez Isaza Jhon Alberto Altamar Rendon Omar Oswaldo Zambrano Arevalo Jheyson Fabian Villavisan Buitrago

Fernando Andres Salguero Cruz William Mayorga Garzon Helber Rojas Washington Nieto Arce

Nelson Oswaldo Santa Garcia Cristian Johan Cuesta Beltran Giovanni De Jesus Montañez Camacho Daniel Alberto Ardila Carrasquilla

OA JB WG HR CB

01:06:54

Búsqueda (Ctrl+Alt+E)

Chat Gente Participar Reaccionar Vista Aplicaciones Más Cámara Micro Compartir Salir

Reactivar audio del micrófono (Ctrl+Shift+M)

Archivos Inicio Insertar Diseño Disposición Referencias Correspondencia Revisar Vista Ayuda PDFelement ¿Qué desea hacer?

Actos 10 A Aa Fuente Párrafo Edición

Portapapeles Pegar Cortar Copiar Copiar formato

4.7. RETROALIMENTACIÓN EQUIPO PEDAGÓGICO: (DESARROLLO PROGRAMA DE FORMACIÓN).

1. Inscripción y matrícula de cupos nuevos
2. Comité de inicio en cada institución educativa
3. Viabilidad de ambientes de formación
4. Sensibilización Padres de familia proyecto productivo

4.8. ACCIONES ADELANTADAS EN EL MARCO DEL MANUAL DE ARTICULACIÓN VIGENTES.

| Actividad                                            | Fecha                           | Temas Abordados                  | Participantes                                                    | Tipo de Actividad que Soporta la Actividad |
|------------------------------------------------------|---------------------------------|----------------------------------|------------------------------------------------------------------|--------------------------------------------|
| Acta de Cierre proceso de ingreso grupos nuevos 2026 | Marzo 6 de 2026                 | Seguimiento proceso cupos nuevos | Equipo articulación con la media                                 | Acta 9                                     |
| Inducción (Grupos Nuevos)                            | 10 de Marzo de 2026             | Inducción SENA                   | Instructor Técnico y Aprendices cupos nuevos                     | Acta 02 Inducción MAONA                    |
| Entrenamiento: capacitación de instructores          | Junio 23 de 2026 Mayo 5 de 2026 | Acuerdo 009 de 2026              | Equipo de articulación con la media<br>Equipo de SST, Biblioteca | Acta N° 18 Acta N°16                       |

Jhon Alberto Altamar Rendon

1/2



Microsoft Teams interface showing a meeting in progress. The main window displays a document titled "Anexo 2. Formato Informe de Ejecución 1 Semestre 2020" with a table of events and results. The table has columns: NOMBRE INSTITUCIÓN (CUTIVA JUN DUE), NOMBRE DEL PROGRAMA, FICHA/GRUPO, TOTAL HORAS PLANEADAS, TOTAL HORAS REGISTRADAS EN SP, and OBSERVACIONES. The table contains data for various programs, including "Mantenimiento de Automatismos Industriales" and "Programación de Software".

| NOMBRE INSTITUCIÓN (CUTIVA JUN DUE) | NOMBRE DEL PROGRAMA                        | FICHA/GRUPO | TOTAL HORAS PLANEADAS | TOTAL HORAS REGISTRADAS EN SP | OBSERVACIONES |
|-------------------------------------|--------------------------------------------|-------------|-----------------------|-------------------------------|---------------|
| alena s de                          | Mantenimiento de Automatismos Industriales | 3191264     | 190                   |                               |               |
| alena s de                          | Mantenimiento de Automatismos Industriales | 3450436     | 190                   |                               |               |
| alena s de                          | Programación de Software                   | 3191165     | 190                   | 204:19                        |               |
| alena s de                          | Programación de Software                   | 3450403     | 190                   |                               |               |

The right sidebar shows a list of participants: OA, JB, WG, HR, and others. The bottom status bar indicates the meeting is in progress.

Microsoft Teams interface showing a meeting in progress. The main window displays a document titled "GOR-F-084 V02" with a table of errors in MySQL. The table has columns: Error, Descripción, and Solución. The table contains data for various errors, including "Error de conexión a la base de datos" and "Error de autenticación".

| Error                                | Descripción                                                                  | Solución                                                                    |
|--------------------------------------|------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| Error de conexión a la base de datos | MySQL maneja errores principalmente a través de verificaciones condicionales | Verificar la configuración de la base de datos y el código de la aplicación |
| Error de autenticación               | MySQL maneja errores principalmente a través de verificaciones condicionales | Verificar la configuración de la base de datos y el código de la aplicación |

The right sidebar shows a list of participants: JB, WG, GI, HR, OA, and others. The bottom status bar indicates the meeting is in progress.



Language changes detected (Español). Please restart Teams to see the change. Apply and restart

Activity 38:48

Take control Chat People Raise React View Controls Apps More Camera Mic Share Leave

Autoguardado Guardado en Este PC Omar Osvaldo Zambrano Arevalo

Archivo Inicio Insertar Dibujar Diseño Disposición Referencias Correspondencia Revisar Vista Ayuda Nitro PDF Pro Acrobat

Pegar Fuente Párrafo Estilos Edición Crear PDF y compartir con Outlook Solicitar firmas Fill and Sign Send for Signature Agreement Status Adobe Acrobat Sign

Portapapeles

Realice un taller práctico donde implemente los siguientes scripts:

1. **Conexión con MySQL Procedural:** transcriba un archivo `mysql_procedural.php` que establezca la conexión, verifique errores mediante `mysql_connect_error()` y cierre la conexión.

```
mysql_procedural.php

<?php
// Conectar
$conexion = mysql_connect("localhost", "usuario", "contraseña", "base_datos");

// Verificar conexión
if (!$conexion) {
    die("Error de conexión: " . mysql_connect_error());
}

// Realizar consulta
$resultado = mysql_query($conexion, "SELECT * FROM usuarios");

// Cerrar conexión
mysql_close($conexion);
}

php 8.2
```

Página 4 de 7 634 palabras Español (Colombia)

Omar Osvaldo Zambrano Arevalo

16°C Nublado

Concentración

ESP ES

11:39 a.m. 2/07/2026

JB WG GI HR OA

16°C Nublado

Buscar

ESP LAA

11:39 a.m. 2/07/2026



| ACTA No. 22                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                      |                                                                     |                              |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|------------------------------|
| <b>NOMBRE DEL COMITÉ O DE LA REUNIÓN:</b><br>Coordinación de Procesos Académicos y Administrativos: Matrículas, Proyectos y Recursos del Equipo de Articulación.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                                                      |                                                                     |                              |
| <b>CIUDAD Y FECHA:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Bogotá 16 de Julio de 2026                                                                           | <b>HORA INICIO:</b><br>9:30 am                                      | <b>HORA FIN:</b><br>11:30 am |
| <b>LUGAR Y/O ENLACE:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | <b>Virtual:</b><br> | <b>DIRECCIÓN / REGIONAL / CENTRO:</b> Centro de Diseño y Metrología |                              |
| <b>AGENDA O PUNTOS PARA DESARROLLAR:</b> <ul style="list-style-type: none"><li>• Seguimiento de entregables y compromisos del receso estudiantil.</li><li>• Control administrativo y cruce de bases de datos (Sena Sofia vs. Listados de Colegios).</li><li>• Análisis de estados de aprendices y deserción (Anexo 12).</li><li>• Estado de revisión de la Carpeta 10 e informes de Proyecto Productivo.</li><li>• Lineamientos para la entrega del Informe Semestral y evidencias de ejecución.</li><li>• Programación de sensibilizaciones de Proyecto Productivo para grado décimo.</li><li>• Acuerdos sobre plazos de entrega para cuentas de cobro y soportes</li></ul> |                                                                                                      |                                                                     |                              |
| <b>OBJETIVO(S) DE LA REUNIÓN:</b> <ul style="list-style-type: none"><li>• Evaluar el estado de cumplimiento de las obligaciones administrativas y pedagógicas de los instructores.</li><li>• Socializar fallas críticas detectadas en el proceso de matrícula y seguimiento de aprendices para establecer medidas preventivas.</li><li>• Concertar el cronograma final de entrega de informes semestrales y evidencias necesarias para el trámite de cuentas de cobro del mes de julio.</li></ul>                                                                                                                                                                            |                                                                                                      |                                                                     |                              |



## DESARROLLO DE LA REUNIÓN

### Desarrollo de la Reunión

- **Control Administrativo:** El coordinador John manifestó inconformidad por el incumplimiento en entregables acordados para el receso. Se presentó un caso crítico donde un aprendiz asistió a formación por año y medio sin estar matriculado en Sena Sofia. Se enfatizó la **obligatoriedad de utilizar exclusivamente los listados de Sena Sofia** para el llamado a lista y el control de novedades, evitando depender de la información de las instituciones educativas.
- **Seguimiento y Anexo 12:** Se analizó el informe de ejecución (Anexo 12), detectando casos de aprendices que aún figuran en estado de "Inducción" o con retiros voluntarios no tramitados. Se recordó la importancia de realizar **Comités de Evaluación y Seguimiento** al final de cada periodo y de documentar planes de mejoramiento para asegurar la trazabilidad pedagógica y evitar contingencias legales.
- **Estado de Entregables:** De los 12 instructores, solo 6 han sido revisados parcialmente; a la fecha, únicamente el instructor Jason presenta la documentación de Proyecto Productivo completa. Se detectó un error recurrente en el nombre del centro en el formato 005 de idea de negocio ("Centro de Diseño y Metrología" escrito incorrectamente).
- **Informe Semestral y Cuentas:** Sonia (supervisora) aclaró que el informe semestral requiere una revisión de múltiples niveles (coordinación, supervisión y subdirección), por lo que no se pueden aceptar entregas de último minuto. Las evidencias de ejecución (actas de voceros, inducción, planes concertados) son requisitos indispensables para el aval de la cuenta de cobro.
- **Petición de Instructores:** Los instructores Élber, Giovanni y Johan solicitaron una extensión del plazo para cargar evidencias, argumentando alta carga operativa al retomar clases presenciales y la complejidad de corregir documentos con los aprendices. El coordinador John recordó que los plazos originales fueron concertados y que el retraso de los instructores impacta sus propios tiempos personales de revisión



## CONCLUSIONES

### Conclusiones

- Se ratifica que el cumplimiento administrativo es un componente esencial de la labor del instructor y no puede ser opacado por la labor en aula.
- El equipo de coordinación acepta una extensión excepcional del plazo para las cuentas de cobro, siempre que se garantice la calidad de los documentos y no se generen retrocesos en la revisión.
- Se establece una reunión técnica para el día de mañana con el fin de unificar criterios sobre los errores detectados en los formatos de Proyecto Productivo

### ESTABLECIMIENTO Y ACEPTACIÓN DE COMPROMISOS

| ACTIVIDAD<br>/DECISIÓN                                                                               | FECHA                                      | RESPONSABLE                                                    | FIRMA O<br>PARTICIPACIÓN<br>VIRTUAL      |
|------------------------------------------------------------------------------------------------------|--------------------------------------------|----------------------------------------------------------------|------------------------------------------|
| Entrega final de evidencias y cuenta de cobro (con Carpeta 10 y Proyecto Productivo al día)          | Martes, 21 de julio (12:00 PM)             | Todos los equipos de Instructores de Articulación              | ANEXO ASISTENTES Y APROBACIÓN DECISIONES |
| Carga de evidencias pendientes del mes anterior (16 al 30 de junio)                                  | Inmediata                                  | Todos los equipos de Instructores de Articulación              | ANEXO ASISTENTES Y APROBACIÓN DECISIONES |
| Programación y ejecución de sensibilizaciones de Proyecto Productivo en grado 10 (acta 165 e inicio) | Según Cronograma y acuerdo en cada Colegio | Todos los equipos de Instructores de Articulación Con Grado 10 | ANEXO ASISTENTES Y APROBACIÓN DECISIONES |

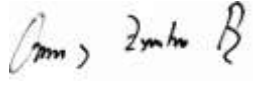
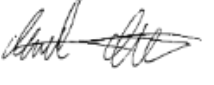


|                                                                                 |                                                                             |                                                   |                                          |
|---------------------------------------------------------------------------------|-----------------------------------------------------------------------------|---------------------------------------------------|------------------------------------------|
| Reunión virtual de observaciones técnicas sobre formatos de Proyecto Productivo | Viernes, 17 de julio (8:00 AM)                                              | Todos los equipos de Instructores de Articulación | ANEXO ASISTENTES Y APROBACIÓN DECISIONES |
| Revisión anticipada para instructores que entreguen antes del lunes             | Entre los días Viernes y Sábado 17 y 18 de Julio a resolución del Ingeniero | Ingeniero John Altamar                            | ANEXO ASISTENTES Y APROBACIÓN DECISIONES |
| Elaboración del acta de la reunión del 16 de julio                              | 16 DE JULIO                                                                 | Omar Zambrano                                     | <i>Omar, Zambrano B</i>                  |
|                                                                                 |                                                                             |                                                   |                                          |
|                                                                                 |                                                                             |                                                   |                                          |

**DE: ASISTENTES Y APROBACIÓN DECISIONES**

| NOMBRE                      | DEPENDENCIA/ EMPRESA   | APRUEBA (SI/NO) | OBSERVACIÓN | FIRMA O PARTICIPACIÓN VIRTUAL |
|-----------------------------|------------------------|-----------------|-------------|-------------------------------|
| Sonia Roció Rey             | Supervisor(a) contrato | SI              | -----       |                               |
| Jhon Alberto Altamar Rendon | Apoyo Administrativo   | SI              | -----       |                               |
| Jheyson Fabian Villavisan   | Instructor CDM SENA    | SI              | -----       | <i>Jheyson Villavisan</i>     |
| Helber Rodrigo Rojas Gacha  | Instructor CDM SENA    | SI              | -----       | <i>Helber Rojas</i>           |
| Cristian Johan Cuesta       | Instructor CDM SENA    | SI              | -----       | <i>Cristian Cuesta</i>        |
| Guillermo Sánchez Isaza     | Instructor CDM SENA    | SI              | -----       | <i>Guillermo Sánchez</i>      |
| William Mayorga G           | Instructor CDM SENA    | SI              | -----       | <i>William Mayorga</i>        |



|                                                                                                                                                                                                                                                                                                                                            |                        |    |       |                                                                                     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|----|-------|-------------------------------------------------------------------------------------|
| Nelson Oswaldo Santa García                                                                                                                                                                                                                                                                                                                | Instructor CDM<br>SENA | SI | ----- |  |
| Fernando Andrés Salguero                                                                                                                                                                                                                                                                                                                   | Instructor CDM<br>SENA | SI | ----- |  |
| Omar Oswaldo Zambrano                                                                                                                                                                                                                                                                                                                      | Instructor CDM<br>SENA | SI | ----- |  |
| Washington Nieto Arce                                                                                                                                                                                                                                                                                                                      | Instructor CDM<br>SENA | SI | ----- |  |
| Daniel Alberto Ardila                                                                                                                                                                                                                                                                                                                      | Instructor CDM<br>SENA | SI | ----- |  |
| Cristian Camilo Carrillo                                                                                                                                                                                                                                                                                                                   | Instructor CDM<br>SENA | SI | ----- |  |
| <p>De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.</p> |                        |    |       |                                                                                     |
| <b>ANEXOS</b>                                                                                                                                                                                                                                                                                                                              |                        |    |       |                                                                                     |