

## **Sistemas de Información**

### **Objetivos Generales del Curso**

- 1. Comprender qué es un Sistema de Información Transaccional (SIT) y su importancia en los negocios.**
- 2. Aplicar las etapas del Ciclo de Vida de Desarrollo de Sistemas (SDLC) para crear un SIT funcional.**
- 3. Desarrollar habilidades prácticas para el diseño y conceptualización de sistemas orientados a la solución de problemas reales de negocios.**
- 4. Fomentar el trabajo en equipo y la capacidad de análisis para identificar las necesidades de un negocio familiar o conocido.**

### **Unidad 1: Introducción a los Sistemas de Información y los SIT**

**Objetivo:** Comprender los conceptos básicos de un sistema de información y los sistemas de información transaccional, y su papel en los negocios.

#### **Clase 1: ¿Qué es un Sistema de Información?**

- **Teoría:**
  - Definición de sistema de información.
  - Componentes de un sistema de información (hardware, software, datos, personas, y procesos).
  - Tipos de sistemas de información: Sistemas de información transaccionales (SIT), sistemas de apoyo a la toma de decisiones (DSS), sistemas expertos, etc.
- **Actividad:**
  - Lluvia de ideas: Los estudiantes identifican sistemas de información que usan en su vida diaria.
  - Discusión: En grupos, discuten cómo esos sistemas de información gestionan transacciones (por ejemplo, una app de entrega de comida o el sistema de caja de una tienda).

#### **Clase 2: Sistemas de Información Transaccional (SIT)**

- **Teoría:**

- **Definición de los SIT y su importancia para gestionar transacciones.**
- **Ejemplos de transacciones comunes: ventas, compras, inventarios, pagos.**
- **Funciones clave de un SIT: registro de transacciones, control de inventario, generación de reportes.**
- **Actividad:**
  - **Ejercicio: Los estudiantes deben identificar un negocio familiar o de conocidos (por ejemplo, tienda, restaurante, etc.) que utilice transacciones diarias y realizar un análisis preliminar de las transacciones que gestionan.**
  - **Entrega: Un párrafo con una breve descripción del negocio y las transacciones que gestionan.**

### Clase 3: Introducción a los Sistemas de Información Transaccionales (SIT) y el Ciclo de Vida del Desarrollo de Sistemas

**Objetivo:** Que los estudiantes comprendan la importancia de la fase de planificación en el desarrollo de un SIT y que sean capaces de identificar el problema de un negocio real, definiendo los objetivos y el alcance del sistema que desarrollarán.

#### 1. Inicio de la clase (10 minutos)

- **Pregunta inicial:** Comienza con una discusión abierta preguntando a los estudiantes si conocen o han trabajado en algún negocio que realice ventas o manejos de inventarios, como tiendas familiares, negocios de comida o servicios.
- **Introducción a los SIT:** Explica que los **Sistemas de Información Transaccionales** son aquellos que gestionan **transacciones diarias** dentro de un negocio. Estas transacciones pueden incluir:
  - Ventas y compras
  - Control de inventario
  - Registro de clientes y proveedores
  - Facturación o generación de recibos

**Ejemplo:** "Imaginen la tienda de su barrio: cada vez que un cliente compra algo, el dueño tiene que registrar esa venta, saber cuántos productos quedan en el inventario y generar una nota o factura."

**Presentación del tema:** Explica que la **planificación** es la primera y más importante fase del ciclo de vida del desarrollo de un sistema de información. En esta fase, los desarrolladores y analistas trabajan en identificar claramente el problema que el sistema resolverá y definen los objetivos del sistema, estableciendo un plan a seguir.

#### 2. Explicación de la fase de planificación (20 minutos)

- **Qué es la planificación:**
  - La fase de **planificación** se centra en identificar el **problema** que necesita solucionarse y definir los **objetivos y el alcance** del sistema.
  - Resalta que esta fase es crucial porque sienta las bases de lo que se desarrollará en las fases posteriores.
- **Componentes clave de la planificación:**

- **Identificación del problema:** Se trata de entender qué necesita mejorar o qué aspecto del negocio no está funcionando de manera eficiente. Ejemplo: un sistema de ventas que permita registrar cada transacción y controlar el inventario de manera automática.
- **Definición de objetivos:** Estos deben ser claros, medibles y alcanzables. Ejemplo: "El sistema permitirá registrar cada venta, actualizando automáticamente el inventario y generando reportes diarios de ventas".
- **Alcance del sistema:** Definir qué áreas del negocio serán cubiertas por el SIT. Por ejemplo, si el sistema solo gestionará ventas o si también incluirá control de inventario y proveedores.
- **Ejemplo práctico:**
  - **Negocio:** Tienda de abarrotes de un conocido.
  - **Problema:** La tienda tiene dificultades para saber cuántos productos tiene en inventario y cuántas ventas se realizan al día.
  - **Objetivo:** Implementar un sistema que registre todas las ventas y actualice el inventario en tiempo real.
  - **Alcance:** El sistema cubrirá la venta de productos y control del inventario, excluyendo la gestión de proveedores por el momento.

### 3. Actividad práctica: Identificación del problema y objetivos del sistema (20 minutos)

#### Instrucciones:

- Divide a los estudiantes en grupos de trabajo (o si lo prefieren, pueden trabajar de manera individual).
- **Identificación del problema:** Cada grupo debe analizar el negocio que seleccionaron previamente y definir el **problema principal** que su sistema de información resolverá. Deben enfocarse en una parte del negocio que sea crucial, como las ventas, control de inventarios, o la gestión de clientes.

#### Guía para la actividad:

- ¿Qué problema o desafío enfrenta el negocio en sus transacciones diarias?
- ¿Cómo afectaría este problema al negocio a largo plazo si no se resuelve?
- **Definición de objetivos:** A continuación, deben escribir al menos **dos objetivos** claros para el sistema, que respondan al problema identificado. Los objetivos

deben ser medibles, como mejorar el seguimiento de inventario o reducir los errores en las transacciones.

#### **Guía para la actividad:**

- ¿Qué esperan que el sistema logre en términos de mejora?
- ¿Cómo sabrán que han alcanzado esos objetivos (indicadores de éxito)?
- **Alcance:** Por último, deberán **definir el alcance** del sistema, es decir, qué áreas del negocio cubrirá el SIT (ventas, inventarios, facturación, etc.).

#### **Ejemplo para guiar la actividad:**

- **Negocio seleccionado:** Cafetería de la familia.
- **Problema identificado:** No tienen una manera organizada de registrar ventas ni de controlar inventario.
- **Objetivo 1:** Registrar cada venta de productos en tiempo real.
- **Objetivo 2:** Mantener actualizado el inventario al registrar automáticamente cada venta.
- **Alcance:** El sistema cubrirá las transacciones de ventas y el control de inventarios, pero no incluirá la gestión de empleados.

#### **4. Discusión y retroalimentación (10 minutos)**

- Pide a algunos grupos que compartan el problema y los objetivos que definieron para sus proyectos.
- **Retroalimentación:** Ofrece comentarios sobre la claridad de los problemas y objetivos definidos. Asegúrate de que los problemas sean específicos y los objetivos medibles. Corrige cualquier confusión o si un objetivo es demasiado amplio.

#### **5. Cierre de la clase (5 minutos)**

- **Asignación de tarea:** Pide a los estudiantes que terminen de pulir los problemas y objetivos si no alcanzaron a hacerlo en clase. Deben entregarlos por escrito en la siguiente sesión.
- **Recordatorio:** Comenta que en la siguiente clase se concentrarán en la **fase de análisis**, donde identificarán los requisitos funcionales y no funcionales del sistema.

## Clase 4: Fase de Análisis de Requisitos del Ciclo de Vida del Desarrollo de un Sistema de Información Transaccional (SIT)

**Objetivo General:** Que los estudiantes comprendan la fase de análisis de requisitos en el ciclo de vida del desarrollo de un SIT y que puedan identificar los requisitos funcionales y no funcionales para el sistema de información que desarrollarán para un negocio real.

---

### Desglose de la Clase

#### 1. Inicio de la clase (10 minutos)

- **Repaso breve:**
  - Comienza repasando lo que se discutió en la clase anterior sobre la fase de planificación. Pregunta a algunos estudiantes o equipos si ya han identificado el **problema** y definido los **objetivos y el alcance** del sistema de información para el negocio que seleccionaron.
  - **Transición:** Explica que una vez que tenemos claro el problema y los objetivos del sistema, el siguiente paso es profundizar en **qué funciones** necesitará el sistema para lograr esos objetivos. Esto se realiza en la fase de **análisis de requisitos**.

#### 2. Explicación de la fase de Análisis de Requisitos (20 minutos)

- **¿Qué es el análisis de requisitos?**
  - Es el proceso en el que se recopilan y documentan todas las necesidades y expectativas que debe cumplir el sistema de información para resolver el problema identificado en la fase de planificación.
  - **Requisitos funcionales:** Se refieren a **qué debe hacer el sistema** (funcionalidades específicas).
    - **Ejemplos:**
      - El sistema debe permitir registrar ventas diarias.
      - El sistema debe actualizar automáticamente el inventario cuando se realiza una venta.
      - El sistema debe generar reportes de ventas semanales.

- **Requisitos no funcionales:** Se refieren a **cómo** debe funcionar el sistema, es decir, características de calidad como seguridad, rendimiento y facilidad de uso.
  - **Ejemplos:**
    - El sistema debe ser accesible desde cualquier dispositivo móvil.
    - El sistema debe actualizar la base de datos en menos de 5 segundos.
    - El sistema debe tener un nivel básico de seguridad, protegiendo los datos de ventas y clientes.
- **Recolección de requisitos:**
  - Explica diferentes métodos para recolectar requisitos:
    - **Entrevistas:** Hablando directamente con los dueños o empleados del negocio.
    - **Observación:** Analizando cómo se realizan actualmente las transacciones en el negocio.
    - **Cuestionarios:** Preguntando a los usuarios qué esperaban de un sistema como este.
- **Importancia del análisis de requisitos:**
  - Resalta que la calidad de esta fase es crítica, ya que cualquier requisito no identificado podría causar problemas en el desarrollo o implementación del sistema.

### 3. Actividad práctica: Identificación de requisitos (20 minutos)

#### Instrucciones:

- Divide a los estudiantes en grupos (o si ya están trabajando en equipos, que continúen en los mismos).
- **Paso 1:** Pide a cada equipo que piense en el negocio que seleccionaron y, con base en los objetivos que ya definieron, **identifiquen al menos 5 requisitos funcionales** para su sistema.
  - **Ejemplo:**

- El sistema debe permitir registrar cada transacción de venta.
- El sistema debe generar un ticket para el cliente al realizar la venta.
- El sistema debe mostrar el stock actualizado en tiempo real.
- **Paso 2:** A continuación, deben pensar en **al menos 3 requisitos no funcionales**, relacionados con el rendimiento, seguridad o facilidad de uso del sistema.
  - **Ejemplo:**
    - El sistema debe funcionar en un entorno de baja conectividad.
    - El sistema no necesita de internet para funcionar
    - El sistema debe proteger la información de los clientes (nombres y correos electrónicos).
    - El sistema debe ser fácil de usar para empleados con poca experiencia tecnológica.

#### **Guía durante la actividad:**

- Camina entre los grupos para asegurarte de que estén identificando correctamente los requisitos funcionales y no funcionales. Haz preguntas para guiar su reflexión:
  - ¿Cómo debe manejar el sistema las ventas cuando hay múltiples clientes al mismo tiempo?
  - ¿Qué pasará si un producto se agota en el inventario?
  - ¿Cómo aseguraremos que el sistema funcione de manera rápida y sin errores?

**Entrega:** Al finalizar la actividad, cada grupo debe entregar una lista con **5 requisitos funcionales** y **3 requisitos no funcionales** para su sistema.

#### **4. Discusión y retroalimentación (10 minutos)**

- **Discusión grupal:**
  - Pide a algunos equipos que compartan los requisitos que identificaron para el negocio que están analizando.
  - Pregunta a la clase si creen que esos requisitos cubren todos los aspectos importantes del sistema o si hay algo que podría faltar.
- **Retroalimentación:**



- Da retroalimentación sobre los ejemplos presentados, señalando si los requisitos son lo suficientemente claros y específicos. Recuerda a los estudiantes que deben ser realistas y asegurarse de que los requisitos son alcanzables dentro del alcance del proyecto.

## 5. Cierre de la clase (5 minutos)

- **Resumen:**

- Recapitula los puntos más importantes de la fase de análisis de requisitos: qué son los requisitos funcionales y no funcionales, por qué es importante recolectarlos correctamente, y cómo afectan el desarrollo del sistema.

- **Tarea:**

- Pide a los estudiantes que terminen de afinar la lista de requisitos para su sistema y comiencen a pensar en cómo organizarán el **diseño** del sistema en la siguiente clase, que incluirá la estructura del sistema y su interfaz de usuario.

---

## Recursos Adicionales para la Clase

- **Materiales de Apoyo:**

- Ejemplos de requisitos funcionales y no funcionales.
- Gráficos que expliquen la diferencia entre funcional y no funcional.

- **Herramientas sugeridas:**

- Uso de plantillas o tablas para que los estudiantes puedan organizar mejor los requisitos que identifican.

Esta clase establecerá las bases para las fases de diseño y desarrollo, ya que los estudiantes ahora tendrán una idea clara de lo que su sistema debe hacer y cómo debe funcionar.

## Clase 5: Fase de Diseño del Sistema de Información Transaccional (SIT)

**Objetivo General:** Que los estudiantes comprendan el proceso de diseño de un sistema de información transaccional (SIT) y puedan crear una propuesta preliminar del diseño del sistema para el negocio que seleccionaron, incluyendo la estructura general, interfaces y flujo de datos.

---

## Desglose de la Clase

### 1. Inicio de la clase (10 minutos)

- **Repaso breve:**
  - Inicia repasando la fase de análisis de requisitos de la clase anterior. Pregunta a los estudiantes si pudieron identificar claramente los requisitos funcionales y no funcionales para el sistema de información que están diseñando.
  - **Transición:** Explica que, ahora que tienen claros los requisitos del sistema, el siguiente paso es **diseñarlo**. Esta fase define **cómo funcionará el sistema** y cómo se verá.

### 2. Explicación de la fase de Diseño del Sistema (20 minutos)

- **¿Qué es el diseño de un sistema?**
  - En esta fase, se traduce la funcionalidad requerida (identificada en la fase de análisis) en un plan concreto que describe cómo el sistema será construido. El diseño debe abarcar tanto el **diseño lógico** (estructura de datos, interacciones) como el **diseño físico** (interfaz de usuario, pantallas, etc.).
- **Componentes del diseño de un SIT:**
  - **1. Diseño de la arquitectura del sistema:** Cómo se estructurará el sistema y sus componentes. Aquí se define la estructura principal del sistema, incluyendo bases de datos, módulos principales, y flujos de trabajo.
    - **Ejemplo:** Un SIT puede tener tres módulos principales: ventas, inventario y reportes. Cada uno gestionará una parte específica del negocio.
  - **2. Diseño de la interfaz de usuario (UI):** Cómo los usuarios interactuarán con el sistema. Las interfaces deben ser intuitivas y fáciles de usar, especialmente si los empleados del negocio no tienen experiencia técnica.
    - **Ejemplo:** La pantalla de ventas debe permitir registrar fácilmente los productos vendidos, mostrando el precio y actualizando el inventario automáticamente.

- **3. Diseño del flujo de datos:** Cómo los datos se moverán entre los diferentes componentes del sistema. Aquí se describe cómo las transacciones (como una venta) impactarán el inventario, los reportes, etc.
  - **Ejemplo:** Cuando un producto es vendido, el sistema debe actualizar el inventario y reflejarlo en un reporte diario de ventas.
- **Ejemplo práctico:**
  - **Negocio:** Una cafetería.
  - **Arquitectura:** Tres módulos principales: gestión de ventas, control de inventario, reportes de ventas.
  - **Interfaz:** Una pantalla para ingresar los pedidos de los clientes, que automáticamente calcula el precio total y actualiza el inventario.
  - **Flujo de datos:** Cada vez que se realiza una venta, los productos vendidos se restan del inventario y el monto total se añade al reporte diario.

### 3. Actividad práctica: Diseño preliminar del sistema (20 minutos)

#### Instrucciones:

- Pide a los estudiantes que, en sus grupos, comiencen a diseñar el sistema para el negocio que seleccionaron. Deben abordar los siguientes elementos:
  - **1. Arquitectura del sistema:** Dividan su sistema en módulos o componentes principales. Por ejemplo, un sistema de ventas puede tener un módulo para registrar ventas, otro para manejar inventarios, y otro para generar reportes.
  - **2. Interfaz de usuario (UI):** Dibujen (puede ser a mano) un boceto de la pantalla principal del sistema. Esto incluye cómo se verá la interfaz para registrar ventas, manejar el inventario o generar reportes.
  - **3. Flujo de datos:** Describan cómo los datos se moverán a través del sistema. ¿Qué sucede cuando se realiza una venta? ¿Qué se actualiza en el inventario y los reportes?

#### Guía para la actividad:

- Pide a los estudiantes que respondan a preguntas clave mientras diseñan:
  - ¿Cómo es la pantalla de ventas? ¿Qué información debe mostrar?

- ¿Qué sucede en el sistema cuando se realiza una transacción de venta?
- ¿Cómo se organizan los módulos del sistema? ¿Qué interacciones hay entre ellos?
- **Entrega al final de la clase:** Cada grupo debe tener un boceto inicial de la interfaz de usuario y un esquema básico de cómo fluirán los datos dentro de su sistema.

#### 4. Cierre de la clase (10 minutos)

- **Discusión en grupo:** Pide a algunos grupos que presenten brevemente sus diseños preliminares. Invita a los demás a hacer preguntas o sugerencias.
- **Tarea:** Pide a los estudiantes que refinen sus diseños con base en el feedback que recibieron. Deben traer para la próxima clase una versión más detallada del diseño de la interfaz y una explicación clara del flujo de datos en su sistema.

---

#### Resultado esperado para la próxima clase:

Para la próxima clase, los estudiantes deben tener un diseño más completo de la interfaz y el flujo de datos de su sistema de información transaccional, lo cual servirá como base para las fases de desarrollo e implementación.

#### Clase 6: Fase de Desarrollo del Sistema de Información Transaccional (SIT) usando Python y Tkinter

**Objetivo General:** Que los estudiantes comprendan cómo desarrollar la interfaz y funcionalidades básicas de un sistema de información transaccional (SIT) utilizando Python y Tkinter, implementando las funciones principales definidas en el diseño.

---

#### Desglose de la Clase

##### 1. Inicio de la clase (10 minutos)

- **Repaso breve:**
  - Repasa lo aprendido en la clase anterior sobre el diseño del sistema, preguntando si todos los grupos definieron sus interfaces y el flujo de datos.
  - **Transición:** Explica que hoy pasarán a la fase de desarrollo, donde comenzarán a implementar las funcionalidades del sistema usando **Python y Tkinter** para la interfaz gráfica.

## 2. Introducción a Tkinter y su uso en el desarrollo de un SIT (20 minutos)

- **¿Qué es Tkinter?**
  - Tkinter es la biblioteca estándar de Python para crear interfaces gráficas. Permite a los usuarios interactuar con el sistema a través de ventanas, botones, cuadros de texto, etc.
  - Muestra un ejemplo simple de una ventana en Tkinter para que los estudiantes vean cómo se crea una interfaz gráfica básica.

```
# Ejemplo
import tkinter as tk

root = tk.Tk()
root.title("Sistema de Información")
root.geometry("400x300")

label = tk.Label(root, text="Bienvenido al Sistema de Ventas")
label.pack()

root.mainloop()
```

- **Componentes principales de una interfaz de SIT con Tkinter:**
  - **Ventanas:** Contenedores principales para la interfaz.
  - **Widgets:** Elementos como botones, etiquetas, cuadros de texto, que permiten la interacción del usuario.
  - **Eventos:** Las acciones que realiza el usuario, como hacer clic en un botón o ingresar datos.

## 3. Desarrollo de funcionalidades del sistema con Python y Tkinter (20 minutos)

- **Funcionalidades a implementar:**
  - **Registro de ventas:** Un formulario sencillo que permita ingresar productos vendidos, su cantidad y precio.
  - **Actualización de inventarios:** Cada vez que se registra una venta, el inventario del producto se debe reducir automáticamente.

- **Generación de reportes:** Mostrar un resumen de las ventas realizadas durante el día o la semana.
- **Pasos básicos para la implementación:**
  - **Paso 1:** Crear una interfaz para registrar ventas.
    - Usa **Tkinter Entry** (cuadros de texto) para que el usuario pueda ingresar el nombre del producto, la cantidad y el precio.

<pre># Ejemplo import tkinter as tk  def registrar_venta():     producto = entry_producto.get()     cantidad = int(entry_cantidad.get())     precio = float(entry_precio.get())     total = cantidad * precio     # Aquí va la lógica para registrar la venta y     actualizar el inventario.     label_resultado.config(text=f"Venta registrada: {producto}, Total: {total}")  root = tk.Tk() root.title("Sistema de Ventas")  label_producto = tk.Label(root, text="Producto:") label_producto.pack() entry_producto = tk.Entry(root) entry_producto.pack()  label_cantidad = tk.Label(root, text="Cantidad:") label_cantidad.pack() entry_cantidad = tk.Entry(root) entry_cantidad.pack()  label_precio = tk.Label(root, text="Precio:") label_precio.pack() entry_precio = tk.Entry(root) entry_precio.pack()  button_registrar = tk.Button(root, text="Registrar Venta", command=registrar_venta) button_registrar.pack()  label_resultado = tk.Label(root, text="") label_resultado.pack()  root.mainloop()</pre>	
---	--

- **Paso 2:** Actualizar el inventario automáticamente.
  - Explica cómo se puede implementar una lista o diccionario de productos en Python para gestionar el inventario.

<pre># Ejemplo de sistema de ventas con Tkinter usando id para productos import tkinter as tk</pre>	<pre># Ventana principal de Tkinter root = tk.Tk() root.title("Sistema de Ventas")</pre>
---	--

```

# Inventario con id para cada producto
inventario = {
    1: {"nombre": "Café", "cantidad": 50, "precio": 1.50},
    2: {"nombre": "Té", "cantidad": 30, "precio": 1.00},
    3: {"nombre": "Galletas", "cantidad": 100, "precio":
0.75}
}

# Función para registrar la venta
def registrar_venta():
    try:
        producto_id = int(entry_producto_id.get())
        cantidad = int(entry_cantidad.get())
    except ValueError:
        label_resultado.config(text="Error: ID o cantidad
inválido.")
        return

    if producto_id in inventario and
inventario[producto_id]["cantidad"] >= cantidad:
        inventario[producto_id]["cantidad"] -= cantidad
        total = cantidad * inventario[producto_id]["precio"]
        ventas.append((inventario[producto_id]["nombre"],
cantidad, total))
        label_resultado.config(text=f"Venta registrada:
{inventario[producto_id]['nombre']}, Total: {total:.2f}")
    else:
        label_resultado.config(text="Producto no
disponible o cantidad insuficiente.")

# Función para generar reporte de ventas
def generar_reporte():
    if ventas:
        reporte = "\n".join([f"{p}, Cantidad: {c}, Total: {t:.2f}"
for p, c, t in ventas])
        label_reporte.config(text=reporte)
    else:
        label_reporte.config(text="No hay ventas
registradas.")

```

```

root.geometry("400x300")
# Entradas y botones para la
interfaz
label_producto_id =
tk.Label(root, text="ID del
Producto:")
label_producto_id.pack()
entry_producto_id =
tk.Entry(root)
entry_producto_id.pack()

label_cantidad = tk.Label(root,
text="Cantidad:")
label_cantidad.pack()
entry_cantidad = tk.Entry(root)
entry_cantidad.pack()

button_venta = tk.Button(root,
text="Registrar Venta",
command=registrar_venta)
button_venta.pack()

label_resultado = tk.Label(root,
text="")
label_resultado.pack()

button_reporte =
tk.Button(root, text="Generar
Reporte",
command=generar_reporte)
button_reporte.pack()

label_reporte = tk.Label(root,
text="")
label_reporte.pack()

# Lista para registrar ventas
ventas = []

# Ejecutar la ventana principal
root.mainloop()

```

#### 4. Actividad práctica: Desarrollo de una funcionalidad básica (20 minutos)

##### Instrucciones:

- **Paso 1:** Pide a los estudiantes que en sus grupos implementen la interfaz para **registrar ventas** en su sistema usando Python y Tkinter.
  - Deben crear una ventana con **campos de texto** para ingresar el nombre del producto, la cantidad y el precio.
  - El sistema debe **calcular el total** de la venta y mostrar el resultado en la interfaz.
- **Paso 2:** Después de implementar el registro de ventas, deben agregar la funcionalidad para **actualizar el inventario** de manera automática.
  - Si el producto vendido está disponible en el inventario, deben restar la cantidad vendida y mostrar el nuevo stock.

##### Guía durante la actividad:

- Ayuda a los estudiantes que tengan dudas técnicas sobre la implementación.
- Si algunos grupos terminan rápido, sugiereles que implementen la funcionalidad para **generar reportes** de las ventas realizadas.

#### 5. Cierre y retroalimentación (10 minutos)

- **Discusión grupal:** Pide a los grupos que muestren lo que han implementado y expliquen cómo han resuelto la funcionalidad del registro de ventas.
- **Conclusión:** Resalta la importancia de la fase de desarrollo y la implementación de un sistema funcional que se ajuste a los requisitos y diseño previamente definidos.

##### Tarea para la siguiente clase:

- Los estudiantes deben avanzar en la implementación de las funcionalidades restantes, como la actualización de inventarios y generación de reportes, para presentar en la próxima sesión.

---

Este desglose permite que los estudiantes apliquen sus conocimientos de **Python** y **Tkinter** para desarrollar un sistema de información simple, mientras refuerzan conceptos clave de la programación orientada a sistemas de información.