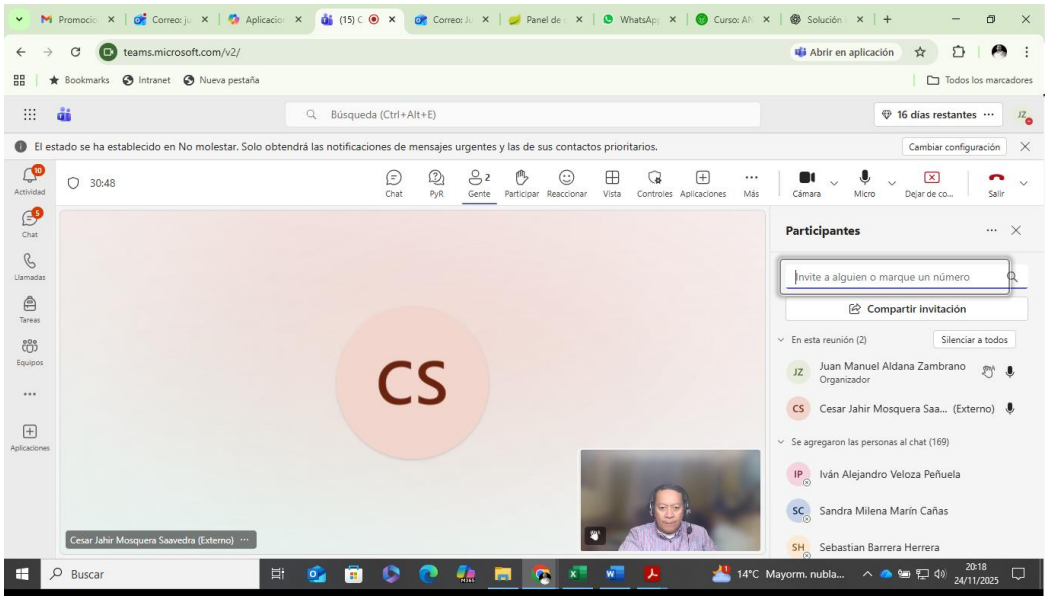
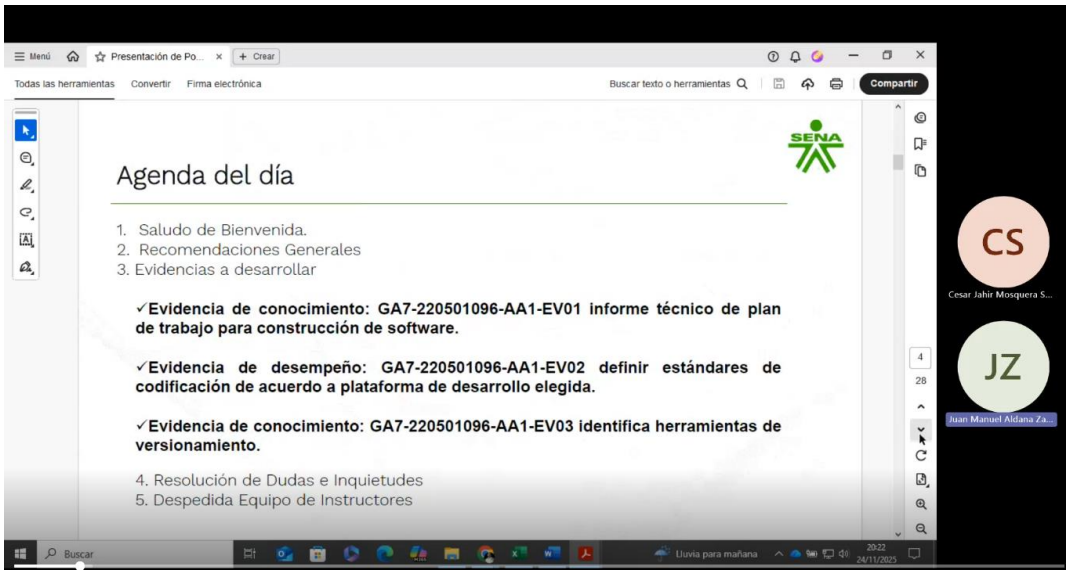
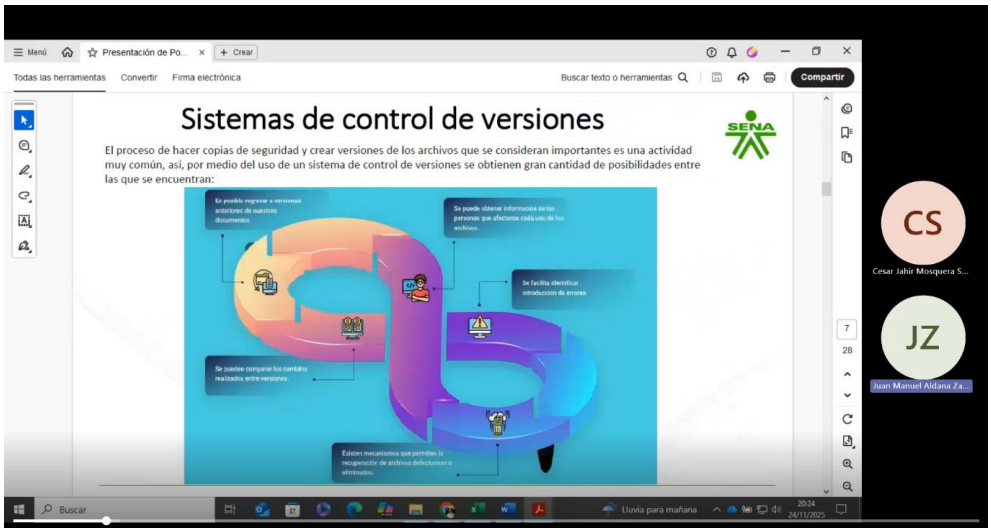
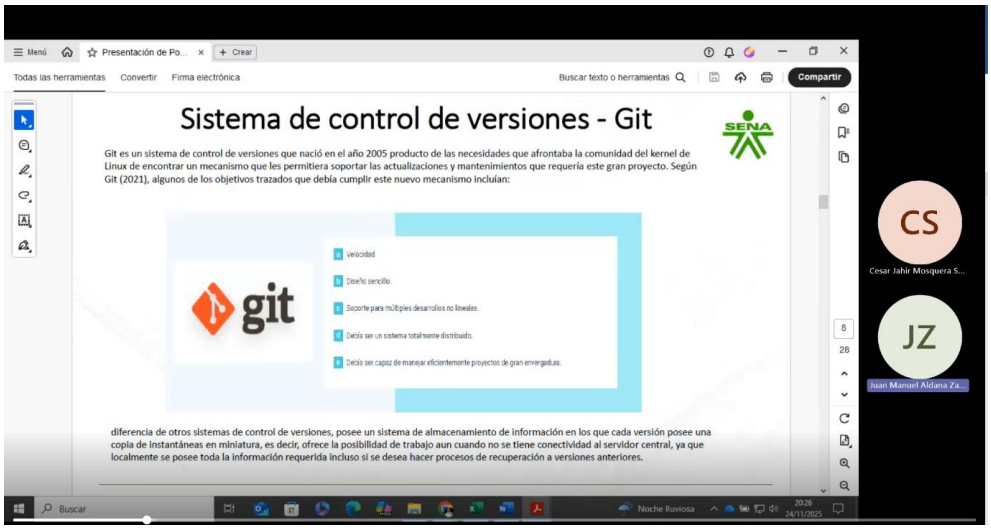
Ficha **3118312**. Análisis y Desarrollo de Software (ADSO).

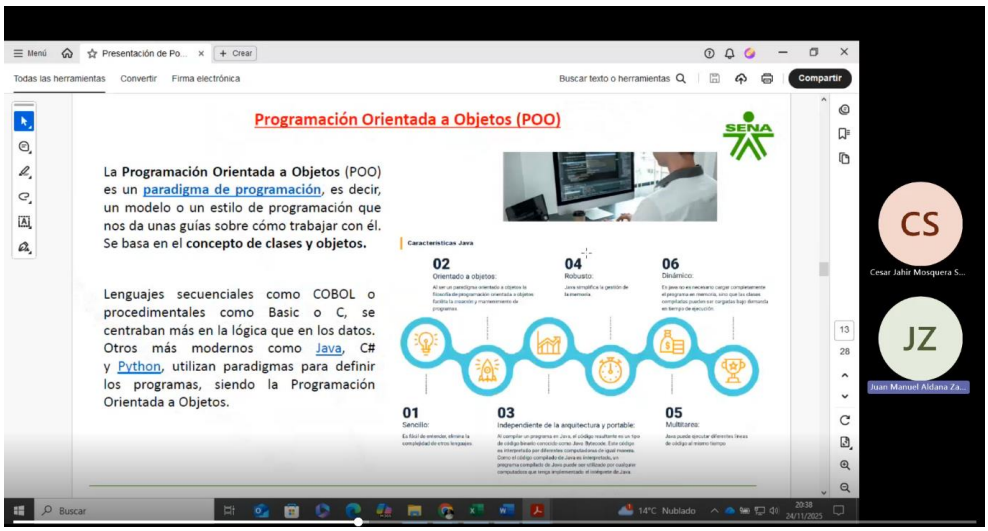
Grabación y resumen conferencia web – GA7-220501096-AA1-EV01, EV02 Y EV03. 24/11/2025. https://youtu.be/m5iirP7cDH4

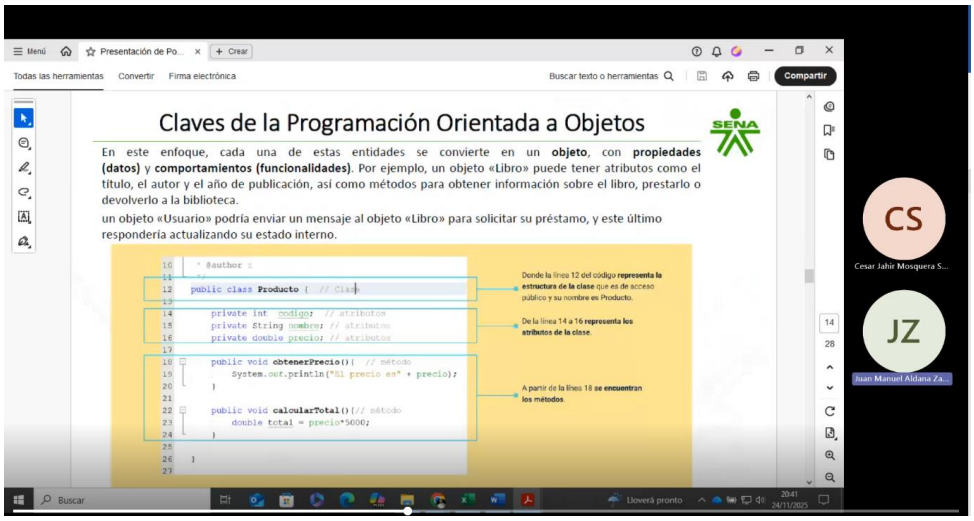


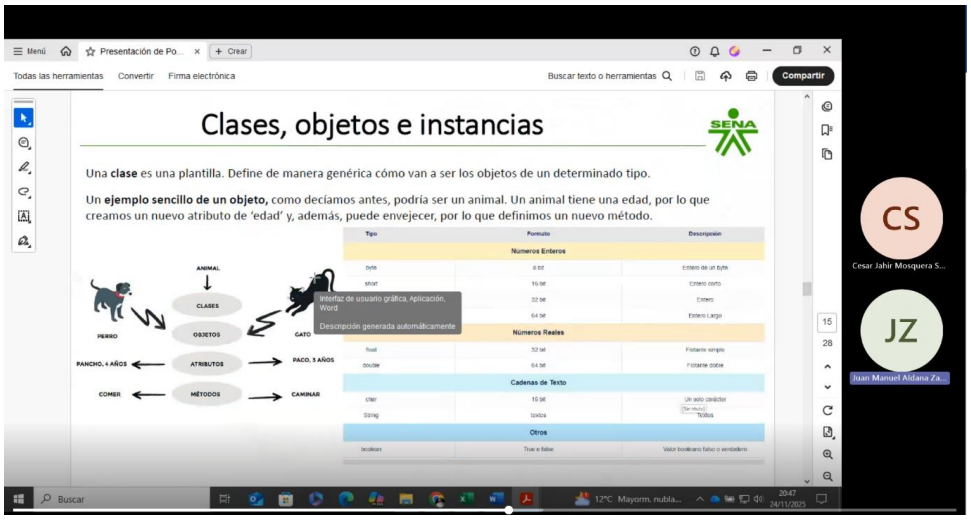



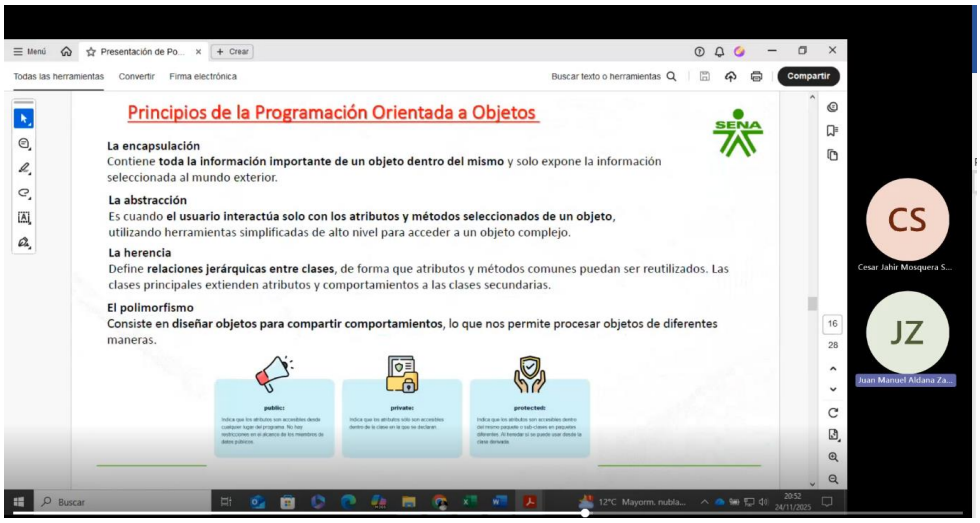

Sistemas de control de versiones
El proceso de hacer copias de seguridad y crear versiones de los archivos que se consideran importantes es una actividad muy común, así, por medio del uso de un sistema de control de versiones se obtienen gran cantidad de posibilidades entre las que se encuentran:
Es posible regresar a versiones anteriores de nuestros documentos.
Se puede obtener información de las personas que efectuaron cada uso de los archivos.
Se facilita identificar introducción de errores
Se pueden comparar los cambios realizados entre versiones.
Existen mecanismos que permiten la recuperación de archivos defectuosos o eliminados.
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...


Sistema de control de versiones - Git
Git es un sistema de control de versiones que nació en el año 2005 producto de las necesidades que afrontaba la comunidad del kernel de Linux de encontrar un mecanismo que les permitiera soportar las actualizaciones y mantenimientos que requería este gran proyecto. Según Git (2021), algunos de los objetivos trazados que debía cumplir este nuevo mecanismo incluían:
Velocidad
Diseño sencillo
Soporte para múltiples desarrollos no lineales.
Debía ser un sistema totalmente distribuido.
Debía ser capaz de manejar eficientemente proyectos de gran envergadura.
diferencia de otros sistemas de control de versiones, posee un sistema de almacenamiento de información en los que cada versión posee una copia de instantáneas en miniatura, es decir, ofrece la posibilidad de trabajo aun cuando no se tiene conectividad al servidor central, ya que localmente se posee toda la información requerida incluso si se desea hacer procesos de recuperación a versiones anteriores.
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...


Programación Orientada a Objetos (POO)
La Programación Orientada a Objetos (POO) es un paradigma de programación, es decir, un modelo o un estilo de programación que nos da unas guías sobre cómo trabajar con él. Se basa en el concepto de clases y objetos.
Lenguajes secuenciales como COBOL o procedimentales como Basic o C, se centraban más en la lógica que en los datos. Otros más modernos como Java, C# y Python, utilizan paradigmas para definir los programas, siendo la Programación Orientada a Objetos.
Características Java
02
Orientado a objetos:
04
Robusto:
06
Dinámico:
01
Sencillo:
03
Independiente de la arquitectura y portable:
05
Multitarea:
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...


Claves de la Programación Orientada a Objetos
En este enfoque, cada una de estas entidades se convierte en un objeto, con propiedades (datos) y comportamientos (funcionalidades). Por ejemplo, un objeto «Libro» puede tener atributos como el título, el autor y el año de publicación, así como métodos para obtener información sobre el libro, prestarlo o devolverlo a la biblioteca.
un objeto «Usuario» podría enviar un mensaje al objeto «Libro» para solicitar su préstamo, y este último respondería actualizando su estado interno.
public class Producto {
private int codigo;
private String nombre;
private double precio;
public void obtenerPrecio(){
System.out.println("El precio es" + precio);
public void calcularTotal(){
double total = precio*5000;
Donde la línea 12 del código representa la estructura de la clase que es de acceso público y su nombre es Producto.
De la línea 14 a 16 representa los atributos de la clase.
A partir de la línea 18 se encuentran los métodos.
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...


Clases, objetos e instancias
Una clase es una plantilla. Define de manera genérica cómo van a ser los objetos de un determinado tipo.
Un ejemplo sencillo de un objeto, como decíamos antes, podría ser un animal. Un animal tiene una edad, por lo que creamos un nuevo atributo de 'edad' y, además, puede envejecer, por lo que definimos un nuevo método.
ANIMAL
CLASES
OBJETOS
ATRIBUTOS
MÉTODOS
PERRO
GATO
PANCHO, 4 AÑOS
PACO, 5 AÑOS
COMER
CAMINAR
Interfaz de usuario gráfica, Aplicación, Word
Descripción generada automáticamente
Tipo
Formato
Descripción
Números Enteros
Números Reales
Cadenas de Texto
Otros
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...


Principios de la Programación Orientada a Objetos
La encapsulación
Contiene toda la información importante de un objeto dentro del mismo y solo expone la información seleccionada al mundo exterior.
La abstracción
Es cuando el usuario interactúa solo con los atributos y métodos seleccionados de un objeto, utilizando herramientas simplificadas de alto nivel para acceder a un objeto complejo.
La herencia
Define relaciones jerárquicas entre clases, de forma que atributos y métodos comunes puedan ser reutilizados. Las clases principales extienden atributos y comportamientos a las clases secundarias.
El polimorfismo
Consiste en diseñar objetos para compartir comportamientos, lo que nos permite procesar objetos de diferentes maneras.
public:
private:
protected:
CS
Cesar Jahir Mosquera S...
JZ
Juan Manuel Aldana Za...