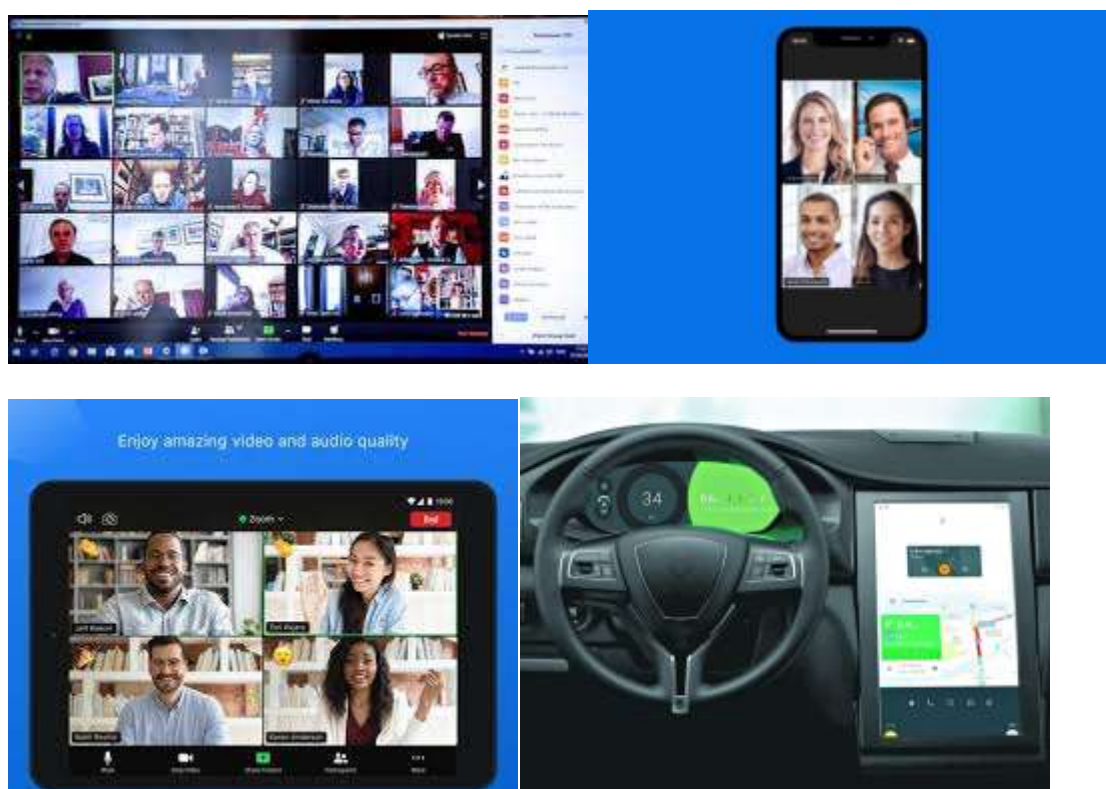




## Ejercicio 6

### Competencias a desarrollar:

- Identifica los requisitos funcionales del sistema a desarrollar a partir del problema planteado.
- Elabora el análisis y el diseño del programa utilizando un diagrama de clases en UML.
- Reconoce elementos polimórficos en el diseño de la solución.
- Utiliza interfaces para definir comportamientos de las clases.



### Ejercicio:

Una nueva normalidad. Las clases suceden a distancia ... El trabajo sucede a distancia ...

La empresa Zoom ha sido tan exitosa, que ahora hay celulares, computadoras, tablets ... incluso carros que tienen ya incorporada la aplicación directamente desde fábrica, en forma “incrustada” (o *embedded*<sup>1</sup>). Son denominados dispositivos Zoom.

<sup>1</sup> <https://internetofthingsagenda.techtarget.com/definition/embedded-system>

Cada dispositivo Zoom tiene su funcionalidad propia, e incorpora también la funcionalidad para sostener videoconferencias Zoom.

Las videoconferencias Zoom tienen:

- Identificación de la videoconferencia (un número entero)
- Contraseña de la videoconferencia (un número entero)
- Anfitrión: identificación del usuario administrador de la reunión
- Varios usuarios que están conectados a la videoconferencia

El Anfitrión tiene las opciones de iniciar o terminar la videoconferencia. Mientras la videoconferencia no está iniciada, ningún usuario puede conectarse a la misma.

Los usuarios son identificados por un nombre y se conectan desde un dispositivo (computadora, celular, Tablet o carro) de la preferencia del usuario.

En la videoconferencia aparecen los usuarios que se han conectado. Además, aparece si tiene activado el micrófono y la cámara de video de su dispositivo ZOOM. Cada usuario puede activar/desactivar tanto su micrófono, como su cámara. El Anfitrión puede activar/desactivar el micrófono o cámara de cualquier usuario conectado. También puede verse desde que tipo de dispositivo se ha conectado.

Los usuarios pueden chatear mensajes de texto. Cuando un usuario envía un mensaje queda registrado en el chat, indicando quien lo envía y ese mensaje. El chat aparece un espacio de la videoconferencia y muestra siempre los mensajes más recientes.

Si el usuario desea mandar un mensaje privado a otro usuario, no aparece en el chat general. El usuario tiene la opción de ver los mensajes privados que ha recibido. Siempre aparece quien se los ha enviado.

Los usuarios también pueden desconectarse de la videoconferencia en cualquier momento.

---

Nota: los dispositivos Zoom tiene su funcionalidad propia:

- Un celular se prende, apaga, hace y recibe llamadas telefónicas;
- un carro se puede arrancar, frenar, conducir, apagar.
- Una Tablet y una computadora se comportan de manera similar, ya que a ambas se pueden prender, apagar, instalar aplicaciones y correr aplicaciones.

Para cada una de estas operaciones propias de los dispositivos Zoom, solo debe aparecer un mensaje indicando que están haciendo esa acción.

En general, si el dispositivo no está prendido no se puede sostener una videoconferencia Zoom.

### **Evaluación:**

Requisitos Funcionales **(5 puntos)**:

- Identifica correctamente todo lo que debe hacer el programa según la situación planteada.

Análisis **(27 puntos)**:

- Identificación de Clases, atributos y métodos **(9 puntos)**:

- (3 puntos) Se identifican correctamente las clases que se necesitan para resolver el problema. El número de clases identificadas es suficiente para darle solución a la situación planteada.
- (5 puntos) Se define la interfaz que servirá para incluir el comportamiento de Zoom en cada dispositivo.
- (1 punto) Se explica correctamente la forma en que cada clase implementará.
- Atributos de las clases (10 puntos):
  - (6 puntos) Se identifican correctamente todos los atributos de cada una de las clases empleadas. Solo los necesarios para resolver el problema planteado.
  - (4 puntos) Se explica correctamente y de forma lógica el propósito de cada uno de los atributos.
- Métodos de las clases (8 puntos):
  - (3 puntos) Se identifican correctamente los métodos necesarios para resolver la situación planteada.
  - (3 puntos) Queda claro cuales son los métodos que implementan la interfaz.
  - (2 puntos) Se explica correctamente y de forma lógica el propósito de cada uno de los métodos de las clases.

#### Diseño (18 puntos):

- Diagrama de Clases en UML
  - (7 puntos) Se representa correctamente la relación de las clases entre sí, y con la interfaz.
  - (7 puntos) Todos los atributos y métodos tienen correctamente representada la visibilidad que poseen.
  - (4 puntos) Los métodos tienen todos los parámetros y valor de retorno correctamente representados.

#### Implementación (50 puntos)

- (15 puntos) Están correctamente implementados los requisitos funcionales identificados. Se da solución con ellos al problema planteado.
- (10 puntos) El sistema es completamente reactivo a entradas incorrectas, mostrando errores amigables con el usuario.
- (10 puntos) Está correctamente implementado el polimorfismo basado en interfaces. Se usan los atributos de la clase padre de forma correcta así como sus métodos.
- (10 puntos) Se usa correctamente la interface Comparable de java para organizar la lista de usuarios conectados por orden alfabético.
- (5 puntos) Está totalmente documentado usando el estándar JavaDoc.

Nota: Para que sea calificada la implementación deben cumplirse los siguientes requisitos:

- El sistema no tiene errores de sintaxis ni en tiempo de compilación que impidan la revisión del mismo.
- La implementación debe coincidir en un 95% con el diseño. Los cambios que surjan deben ser mínimos y deben estar documentados en el código.