



INGENIERIA DE SOFTWARE II 2025

Clase 1

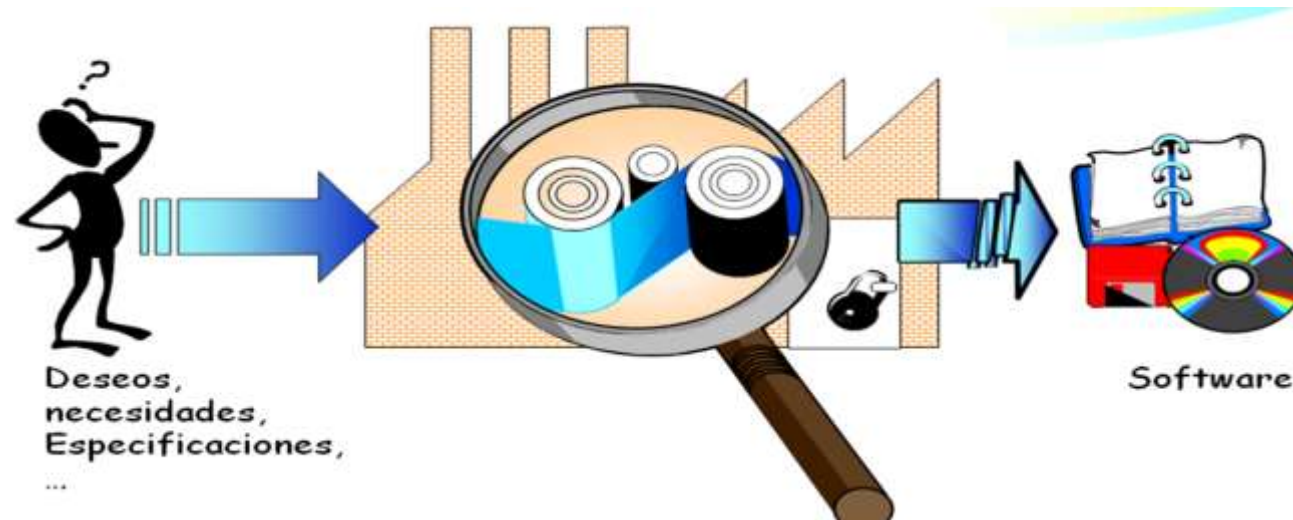
Ingeniería de Software II - Temas

- 1- Gestión o Administración de Proyectos.
- 2- Diseño e Implementación de Software.
- 3- Verificación y Validación.
- 4- Mantenimiento de Software.
- 5- Gestión de Configuración.
- 6- Conceptos de Auditoría y Peritaje.



¿Qué es un proceso de software? - Repaso

- ❖ Es un conjunto de actividades y resultados asociados que producen un producto de software.



¿Modelo ? Es una representación abstracta de un proceso del software

Actividades fundamentales de los modelos de Proceso

- Especificación del software
 - *Técnicas de elicitación*
 - *Especificación de requerimientos*
- Desarrollo del software
- Validación del software
- Evolución del software

El problema de la comunicación - Repaso

❖ Les proponemos ver el video y analizar los inconvenientes que se producen y qué sería necesario para poder resolverlo:

❖ Requerimientos

<https://www.youtube.com/watch?v=93SgXeu-SeY>



El problema de la comunicación y los requisitos



La solicitud del usuario



Lo que entendió el líder del proyecto



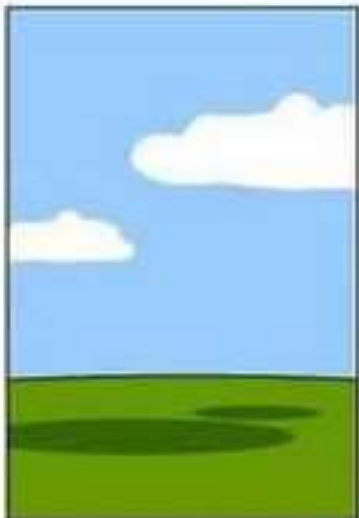
El diseño del analista de sistemas



El enfoque del programador



La recomendación del consultor externo



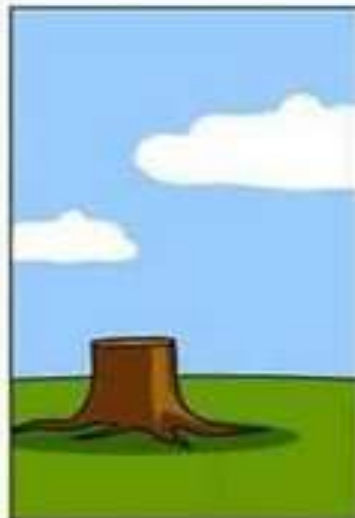
La documentación del proyecto



La implantación en producción



El presupuesto del proyecto



El soporte operativo



Lo que el usuario realmente necesitaba

❖ ¿Cómo comunicar?

La **comunicación** es el proceso más importante de la interacción humana, una necesidad personal y la forma más completa de crear la realidad. Las personas comunicamos constantemente, tanto con las palabras (comunicación verbal) como con nuestra actitud y comportamiento (comunicación no verbal)

Peter Heinemann dice que la comunicación es un proceso de interacción social a través de símbolos y sistemas de mensajes que se producen como parte de la actividad humana.



Elicitación de Requisitos

Repaso

Elicitación de Requisitos



- ❖ Es el proceso de adquirir (“eliciting”) [sonsacar] todo el conocimiento relevante necesario para producir un modelo de los requerimientos de un dominio de problema

- ❖ **Objetivos:**
 - Conocer el dominio del problema para poder comunicarse con clientes y usuarios y entender sus necesidades.
 - Conocer el sistema actual (manual o informatizado).
 - Identificar las necesidades, tanto explícitas como implícitas, de clientes y usuarios y sus expectativas sobre el sistema a desarrollar.

Muestreo de la documentación, los formularios y los datos existentes

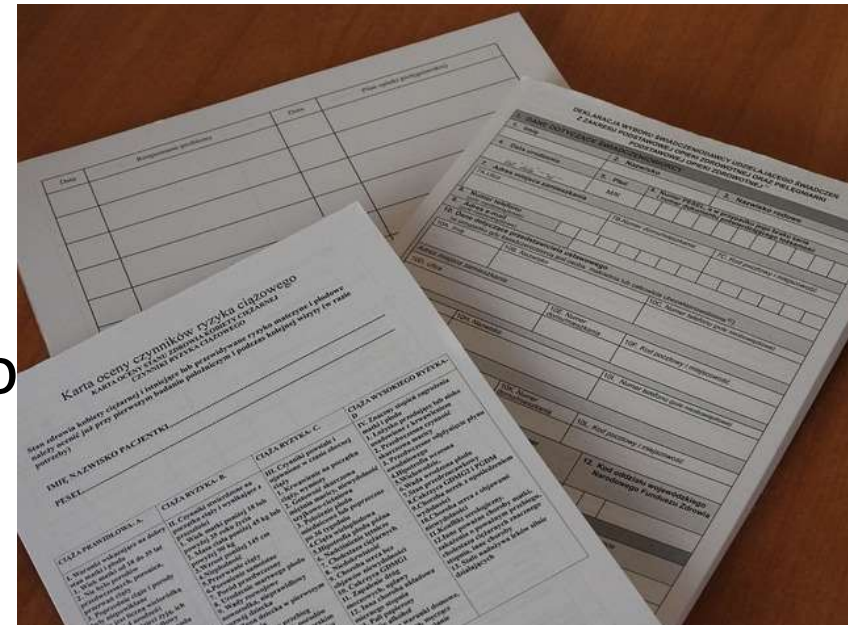


Recolección de hechos a partir de la documentación existente.

¿Qué tipo de documentos pueden enseñar algo acerca del sistema?

- Organigrama (identificar el propietario, usuarios claves).
- Memos, notas internas, minutas, registros contables.
- Solicitudes de proyectos de sistemas de información anteriores.

Permiten conocer el historial que origina el proyecto



Observación del ambiente de trabajo

El analista se convierte en observador de las personas y actividades con el objeto de aprender acerca del sistema.

Lineamientos de la observación:

- Determinar quién y cuándo será observado.
- Obtener el permiso de la persona y explicar el porqué será observado.
- Mantener bajo perfil.
- Tomar nota de lo observado.
- Revisar las notas con la persona apropiada.
- No interrumpir a la persona en su trabajo.



Visitas al sitio

- ✓ Investigar el dominio del problema.
- ✓ Patrones de soluciones (mismo problema en otra organización).
- ✓ Revistas especializadas.
- ✓ Buscar problemas similares en internet.
- ✓ Consultar otras organizaciones.



Tipos de Preguntas

Abiertas

Cerradas

Lenta



Velocidad de conclusión



Rápida

Alta



Naturaleza exploratoria



Poca

Alta



Amplitud y profundidad



Poca

Fácil



Facilidad de preparación



Difícil

Difícil



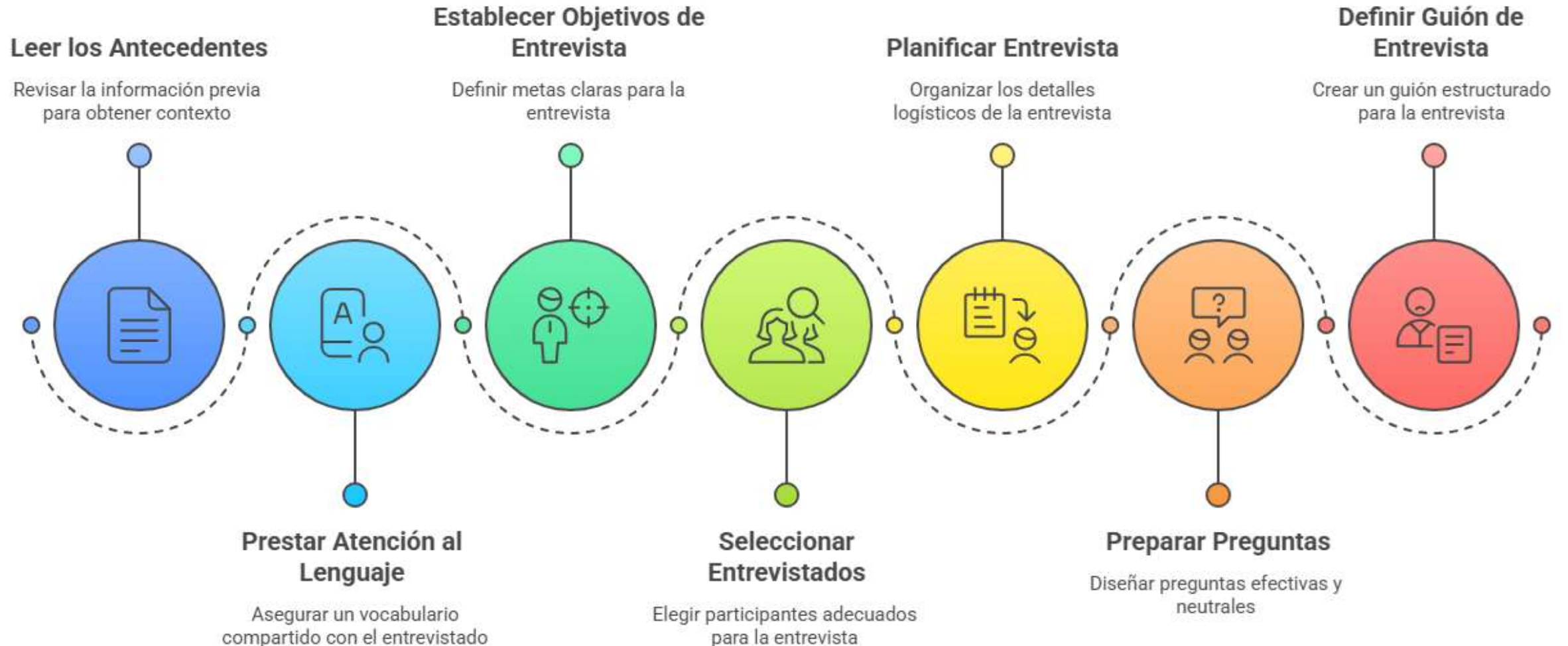
Facilidad de Análisis



Fácil

Entrevistas

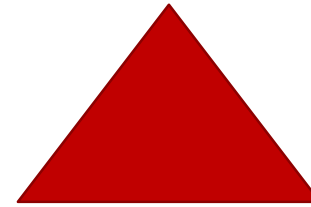
Proceso de Preparación y Conducción de Entrevistas



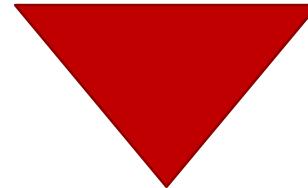
Organización de una entrevista



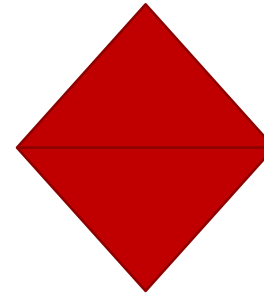
Piramidal (Inductivo)



Embudo (Deductivo)



Diamante (Comb. de las anteriores)



Entrevistas

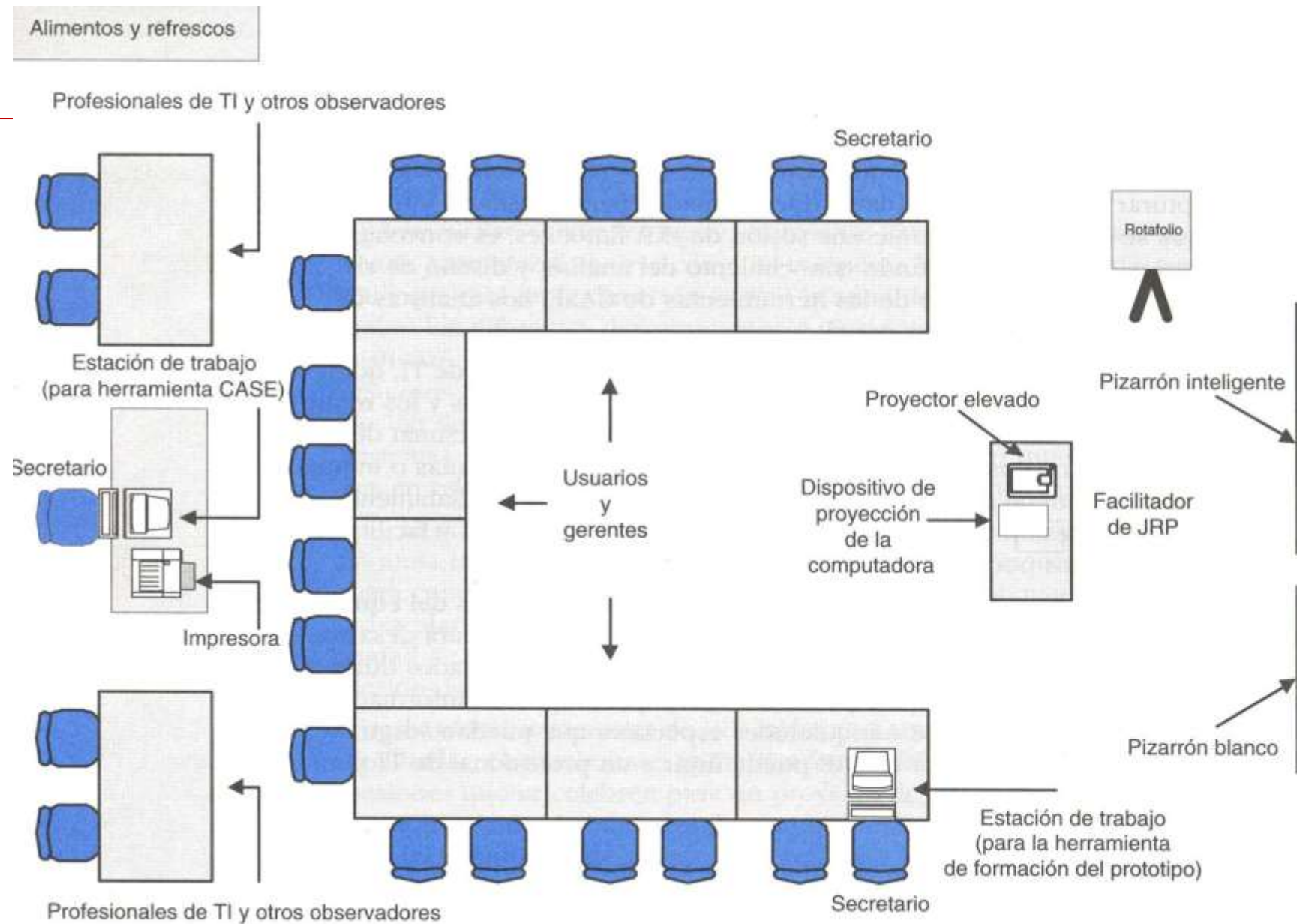
Ejemplo de guión

Entrevistado:	Jeff Bentley, Gerente de cuentas por cobrar	
Fecha:	19 de enero de 2003	
Hora:	1:30 p. m.	
Lugar:	Sala 223, Edificio de administración	
Tema:	Política actual de investigación de crédito	
Tiempo asignado	Pregunta u objetivo del administrador	Respuesta del entrevistado
1 a 2 min.	Objetivo Comienza la entrevista: <ul style="list-style-type: none">• Nos presentamos• Gracias Sr. Bentley por su valioso tiempo• Enunciar el propósito de la entrevista: obtener una comprensión de las políticas existentes de investigación de crédito.	
5 min.	Pregunta 1 ¿Qué condiciones determinan si se aprueba una solicitud de crédito del cliente? Seguimiento	
5 min.	Pregunta 2 ¿Cuáles son las posibles decisiones o acciones que podrían tomarse una vez que estas condiciones han sido evaluadas? Seguimiento	
3 min.	Pregunta 3 ¿Cómo se notifica a los clientes cuando no se aprueba su solicitud de crédito? Seguimiento	

Entrevistas

Ejemplo de guión

1 min.	Pregunta 4 Después que se aprueba una nueva solicitud de crédito y se coloca en el archivo que contiene las solicitudes que pueden llenarse, un cliente puede pedir que se haga una modificación a la solicitud. ¿Tendría que pasar ésta nuevamente por la aprobación de crédito si el costo total de la nueva solicitud sobrepasa al costo original? Seguimiento	
1 min.	Pregunta 5 ¿Quiénes son las personas que realizan las investigaciones de crédito? Seguimiento	
1 a 3 min.	Pregunta 6 ¿Puedo obtener el permiso para hablar con estas personas para aprender específicamente cómo llevan a cabo el proceso de investigación de crédito? Seguimiento Si así es: ¿Cuál sería el momento apropiado para reunirme con cada uno de ellos?	
1 min.	Objetivo Termino de la entrevista: <ul style="list-style-type: none"> Agradezca al Sr. Bentley por su cooperación y asegúrele que estará recibiendo una copia de lo que se obtuvo durante la entrevista. 	
21 minutos	Tiempo asignado para preguntas y objetivos	
9 minutos	Tiempo asignado para preguntas de seguimiento y redirección	
30 minutos	Tiempo asignado para la entrevista (1:30 p.m. a 2:00 p.m.)	
Comentarios generales y notas:		



Brainstorming

» Técnica para generar ideas al alentar a los participantes para que ofrezcan tantas ideas como sea posible en un corto tiempo sin ningún análisis hasta que se hayan agotado las ideas.

» Se promueve el desarrollo de ideas creativas para obtener soluciones.





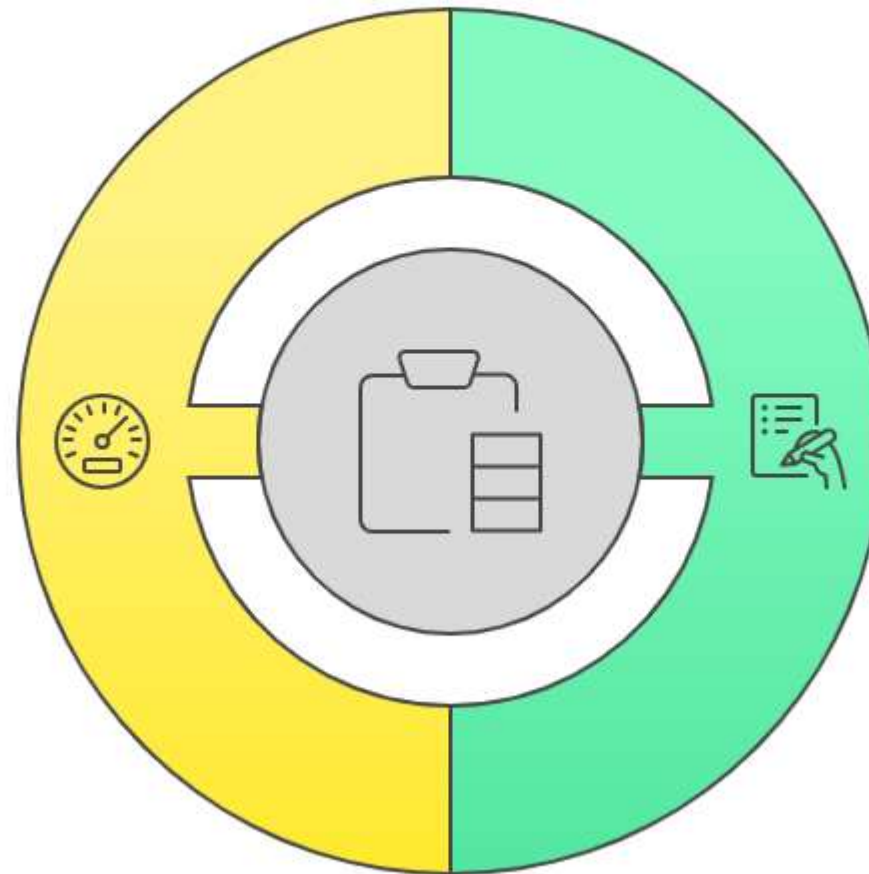
Requerimientos

Repaso

Tipos de requerimientos

Tipos de requerimientos del Sistema

Requerimientos No Funcionales
Describen atributos deseables como el rendimiento y la usabilidad.



Requerimientos Funcionales
Definen el comportamiento y las tareas del sistema.

Especificación del sistema



- ❖ Documento, también denominado **ConOps** y normalizado en el estándar **IEEE Std. 1362-1998**.
- ❖ Documento dirigido a los usuarios, que describe las características de un sistema propuesto, desde el punto de vista del usuario.
- ❖ La Descripción del Sistema es el medio de comunicación que recoge la visión general, cualitativa y cuantitativa de las características del sistema; compartido por la parte cliente y desarrolladora.

ConOps IEEE
Std 1362

Descripción del sistema - IEEE 1362



- ❖ Ofrece un formato y contenidos para la confección de las descripciones de sistema en los desarrollos y modificaciones de sistemas.
- ❖ El estándar no especifica técnicas exactas, sino que proporciona las líneas generales que deben respetarse. Es una guía de referencia.
- ❖ El estándar identifica los elementos que al menos debe incluir una Descripción del sistema. El usuario puede incorporar otros elementos, agregando cláusulas y sub-cláusulas.

Descripción del Sistema

❖ Requerimientos del Software

Documento, también denominado SRS (ERS) y normalizado en el estándar IEEE Std. 830-1998.

Un documento **SRS** es la especificación de las funciones que realiza un determinado producto de software, programa o conjunto de programas en un determinado entorno.

El documento de especificación de requisitos puede desarrollarlo personal representativo de la parte desarrolladora, o de la parte cliente; si bien es aconsejable la intervención de ambas partes

SRS IEEE Std
830

IEEE Std 830-1998
(Revision of
IEEE Std 830-1993)

IEEE Recommended Practice for Software Requirements Specifications

❖ Alcance

Brindar una colección de buenas prácticas para escribir especificaciones de requerimientos de software (SRS). Se describen los contenidos y las cualidades de una buena especificación de requerimientos.

Características del SRS

❖ Naturaleza del SRS

El SRS es una especificación para un producto de software particular. El SRS es escrito por uno o más representantes del equipo de desarrollo y uno o mas representantes de la parte cliente o ambos.

❖ Ambiente del SRS

El software puede contener toda la funcionalidad del proyecto o puede ser parte de un sistema más grande. En el último caso habrá un SRS que declarará las interfaces entre el sistema y su software desarrollado, y pondrá qué función externa y requerimientos de funcionalidad tiene con el software desarrollado.

Características de un buen SRS



❖ **Correcto**

Un SRS es correcto si, y sólo si, cada requisito declarado se encuentra en el software.

❖ **No ambiguo**

Un SRS es inequívoco si, y sólo si, cada requisito declarado tiene sólo una interpretación.

Características de un buen SRS

❖ **Completo**

Un SRS está completo si, y sólo si, se reconoce cualquier requisito externo impuesto por una especificación del sistema.

❖ **Consistente**

La consistencia se refiere a la consistencia interior. Si un SRS no está de acuerdo con algún documento del nivel superior, como una especificación de requerimientos de sistema, entonces no es consistente.

Características de un buen SRS

❖ **Priorizado**

Un SRS es priorizado por la importancia de sus requerimientos particulares.

❖ **Comprobable**

Un SRS es comprobable si, y sólo si, cada requisito declarado es comprobable. Un requisito es comprobable si, y sólo si, existe algún proceso con que una persona o máquina puede verificar que el producto del software reúne el requisito. En general cualquier requisito ambiguo no es comprobable.

Características de un buen SRS

❖ **Modificable**

Un SRS es modificable si, y sólo si, su estructura y estilo son tales que puede hacerse cualquier cambio a los requerimientos fácilmente, completamente y de forma consistente mientras conserva la estructura y estilo

❖ **Trazabilidad**

Claridad del origen de cada requerimiento y su trazabilidad hacia los requerimientos para futuros desarrollos. Hacia adelante y hacia atrás

Consideraciones para un buen SRS

❖ Preparación conjunta del SRS

El SRS se debe preparar en conjunto con las partes intervinientes para lograr un buen acuerdo entre las partes.

❖ Evolución de SRS

El SRS debe evolucionar conjuntamente con el software, registrando los cambios, los responsables y aceptación de los mismos.

❖ Prototipos

El uso de prototipos se utiliza frecuentemente para la definición de requerimientos .

Consideraciones para un buen SRS

❖ **Diseño incorporado en el SRS**

El SRS puede incorporar los atributos o funciones externos al sistema, en particular las que describen el diseño para interactuar entre los subsistemas.

❖ **Requerimientos incorporados en el SRS**

Los detalles particulares de los requerimientos son anexados como documentos externos (CU, Plan de proyecto, Plan de aseguramiento de la calidad, etc.).

FICHA DEL DOCUMENTO

CONTENIDO

1 [INTRODUCCIÓN](#)

1.1 Propósito

1.2 Alcance

1.3 Referencias

2 [DESCRIPCIÓN GENERAL](#)

2.1 Perspectiva del producto

2.2 Funcionalidad del producto

2.3 Características de los usuarios

2.4 Evolución previsible del sistema

3 [REQUISITOS NO FUNCIONALES](#)

3.1 Requisitos de rendimiento

3.2 Seguridad

3.3 Portabilidad

4 [MANTENIMIENTO](#)

5 [APÉNDICES](#)

Sección 1 del SRS

Introducción



❖ 1.1 Propósito

Se define el propósito del documento y se especifica a quién va dirigido el documento

❖ 1.2 Alcance o ámbito del sistema

Se da un nombre al futuro sistema. Se explica lo que el sistema hará y lo que no hará. Se describen los *beneficios*, *objetivos* y *metas* que se espera alcanzar con el futuro sistema

❖ 1.3 Referencias

Se presenta una lista completa de todas las referencias de los documentos mencionados o utilizados para escribir el SRS.

Identificar cada documento por el título, número de reporte, fecha y publicación. Y las fuentes de las referencias de donde se obtuvieron.

Sección 2 del SRS

Descripción General



- ❖ Esta sección del SRS debe describir los factores generales que afectan el producto y sus requerimientos. No declara los requerimientos específicos. Los que se definen en detalle en Sección 3 del SRS.
- ❖ Esta sección normalmente consiste en:
 - Perspectiva del producto**
 - Funcionalidades del producto**
 - Características de los usuarios**
 - Evoluciones previsibles del sistema**



Sección 2 del SRS

Descripción General



❖ 2.1. Perspectiva del producto

Si el producto es independiente y totalmente autónomo, debe declararse que así es.

Si el SRS define un producto que es un componente de un sistema más grande entonces se debe relacionar los requerimientos de ese sistema más grande a la funcionalidad del software y debe identificar las interfaces entre ese sistema y el software.

Sección 2 del SRS

Descripción General



❖ 2.2. Funciones del sistema

Se debe presentar un resumen de las funciones del futuro sistema.

Las funciones deberán mostrarse de forma organizada, y pueden utilizarse gráficos, siempre que reflejen las relaciones entre funciones y no el diseño del sistema.

Sección 2 del SRS

Descripción General



❖ 2.3. Características del Usuario

Se deben describir las características generales de los usuarios intencionales del producto que incluye nivel educativo, experiencia, y la especialización técnica.

❖ 2.4. Evoluciones previsibles del sistema

Se identifican requerimientos que serán implementados en futuras versiones

- ❖ Debe contener todos los requerimientos no funcionales del software a un nivel de detalle para permitirles a los diseñadores diseñar el sistema, y a los auditores probar que el sistema satisface esos requerimientos.

❖ 3.3.1 Requerimientos de rendimiento

Requerimientos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, nro de terminales, nro esperado de usuarios simultáneamente conectados, etc.

Todos estos requerimientos deben ser mensurables. Por ejemplo, indicando “el 95% de las transacciones deben realizarse en menos de 1 segundo”, en lugar de “los operadores no deben esperar a que se complete la transacción”.

Sección 3 del SRS

Requerimientos no funcionales

❖ 3.3.2 Seguridad

Especificación de elementos que protegerán al software de accesos, usos y sabotajes maliciosos, así como de modificaciones o destrucciones maliciosas o accidentales. Los requerimientos pueden especificar:

Empleo de técnicas criptográficas, Registro de ficheros con “logs” de actividad, Asignación de determinadas funcionalidades a determinados módulos, Restricciones de comunicación entre determinados módulos, Comprobaciones de integridad de información crítica.

Sección 3 del SRS

Requerimientos no funcionales



❖ 3.3.3 Portabilidad

Especificación de atributos que debe presentar el software para facilitar su traslado a otras plataformas u entornos.

Pueden incluirse:

% componentes dependientes del servidor. % código dependiente del servidor. Uso de un determinado lenguaje por su portabilidad. Uso de un determinado compilador o plataforma de desarrollo. Uso de un determinado sistema operativo.

Sección 4 del SRS

Mantenimiento



- ❖ Identificación del tipo de mantenimiento necesario del sistema.
- ❖ Especificación de quien debe realizar las tareas de mantenimiento, por ejemplo usuarios, o un desarrollador.
- ❖ Especificación de cuando debe realizarse las tareas de mantenimiento. Por ejemplo, generación de estadísticas de acceso semanales y mensuales.

Sección 5 del SRS

Apéndices



Pueden contener todo tipo de información relevante para la SRS pero que, propiamente, no forme parte de la SRS.

Por ejemplo:

Casos de Uso

Historias de usuario



Comunicación del proyecto

- ❖ ¿Qué pasa cuando realizamos presentaciones acompañadas de un software tipo power point o prezi?

ménos
más

Regla de presentación

Guy Kawasaki, experto en marketing y publicidad, recomienda atenerse a la regla del “**10/20/30**”: la presentación debe contar con no más de 10 diapositivas, no sobrepasar los 20 minutos de duración y no contener tipografías con cuerpo menor a 30. Busca en todo momento cómo simplificar la información y aliviar la diapositiva de elementos innecesarios.

Comunicación - Consejos



No leas la
presentación al
pie de la letra.



No mires la
pantalla cuando
expongas.



Usa la zona de
notas de las
diapositivas.



Colócate en el
centro del
escenario.

Comunicación - Consejos



Escribe menos,
una sola idea por
diapositiva.



No animes el
texto de las
diapositivas.

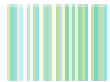


Tamaño de la
letra, usa un
tamaño que sea
legible a cualquier
distancia, se
recomienda usar
un tamaño de
fuente de 32.



Usa un tipo de
letra que se lea
bien, que no use
adornos o
remates, como es
el caso de la
Times New
Roman o Courier.

Comunicación - Consejos



Usa fondos claros y sencillos.



Usa titulares, no apartados.



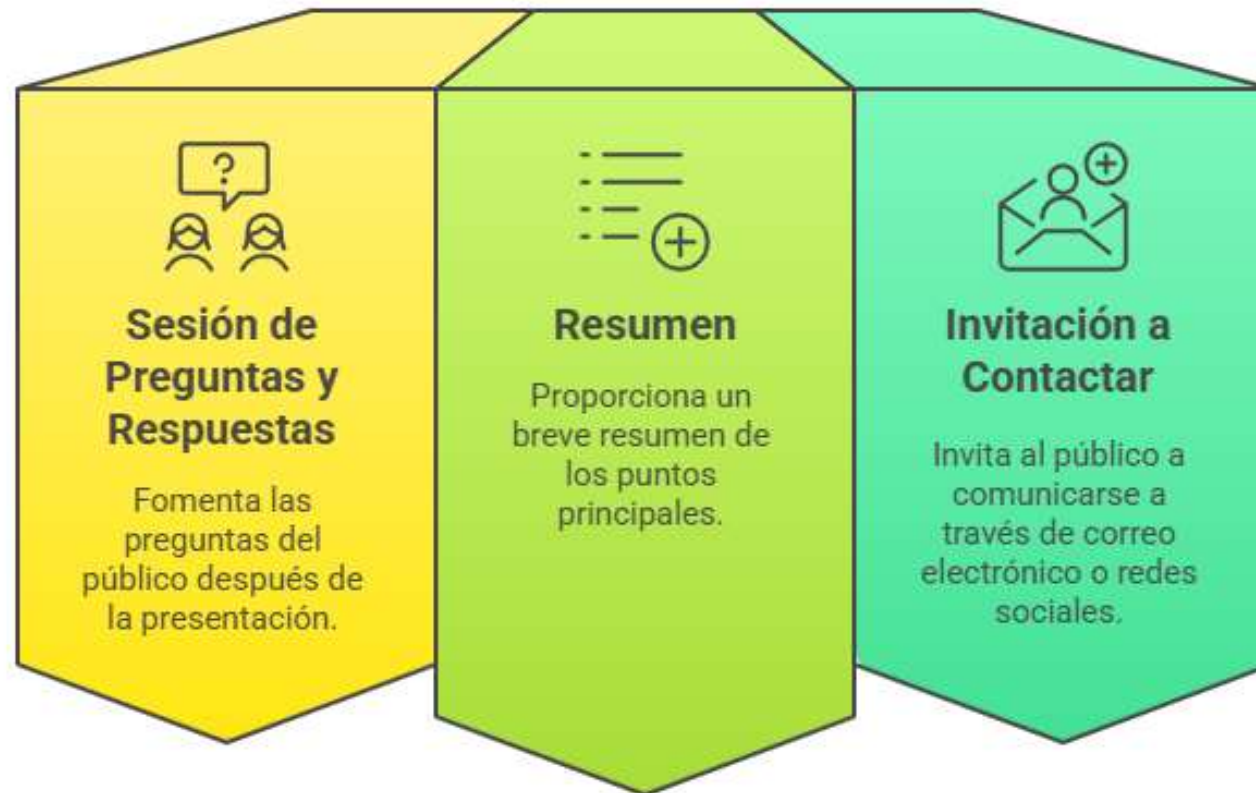
Cuidado con los *bullets* (flechas, puntos, asteriscos, símbolos que se ponen delante de una frase).



Diseña las diapositivas para ser entendidas en 3 segundos.

Comunicación - Consejos

Ideas para Cerrar la Presentación



Comunicación

❖ Presentaciones detalles estéticos

Las ayudas visuales como gráficos, mapas, dibujos o fotografías, entre otras, se deben utilizar para permitir a la audiencia visualizar conceptos que de otra forma resultarían difíciles de entender

El formato de las diapositivas debe ser el de un esquema, en el que cada párrafo representará una idea.

Colores

Textos

Imágenes

Transiciones
y
animaciones

Formato de
las
diapositivas

Colores muy oscuros para el texto y pasteles pálidos para el fondo, como amarillo claro, aunque algunos prefieren texto claro sobre fondo oscuro.

Mantenga el formato de fuente (tipo de letra, tamaño, títulos, etc.). Se debe usar un tipo de letra claro y fácil de leer como Arial, Tahoma o Verdana

Use transiciones naturales, como el texto que cae o aparece desde la izquierda. No se exceda con las transiciones y animaciones de texto, pues no todas las diapositivas requieren efectos especiales

Algunos tips más en general para el ppt que se pide



La presentación debe ser llamativa desde lo visual, debe ofrecer algo distintivo, disruptivo o novedoso.

Debe invitar a ser vista por quien recibirá el mensaje.

El contenido debe ser afirmativo en tiempo presente, por ejemplo del estilo "Esta app permite...."

Cosas que no deben faltar:

- » Nombre del proyecto
- » Nombre de la empresa de desarrollo
- » Objetivo
- » Características o funcionalidades principales y/o diferenciadoras
- » Técnicas de recolección de datos que se emplean
- » Contacto



Video tipo elevator pitch

❖ ¿Cómo comunicar un proyecto o idea?

Les proponemos ver esta situación y luego realizaremos un análisis para ver las características y formas de comunicar

Elevator Pitch – 20 segundos!!

https://www.youtube.com/watch?v=2b3xG_YjgvI



❖ ¿Cómo comunicar un proyecto o idea?

✓ ¿Qué es lo primero que debe hacerse para arrancar la presentación de la idea/proyecto?

✓ ¿Cómo se ofrece el proyecto?

✓ ¿Cómo sugiere el video que se consiga el financiamiento?

✓ ¿Cómo se sugiere continuar la presentación del proyecto?

❖ ¿Cómo hacer un Elevator pitch?

<https://www.youtube.com/watch?v=uv357YzY7-k>



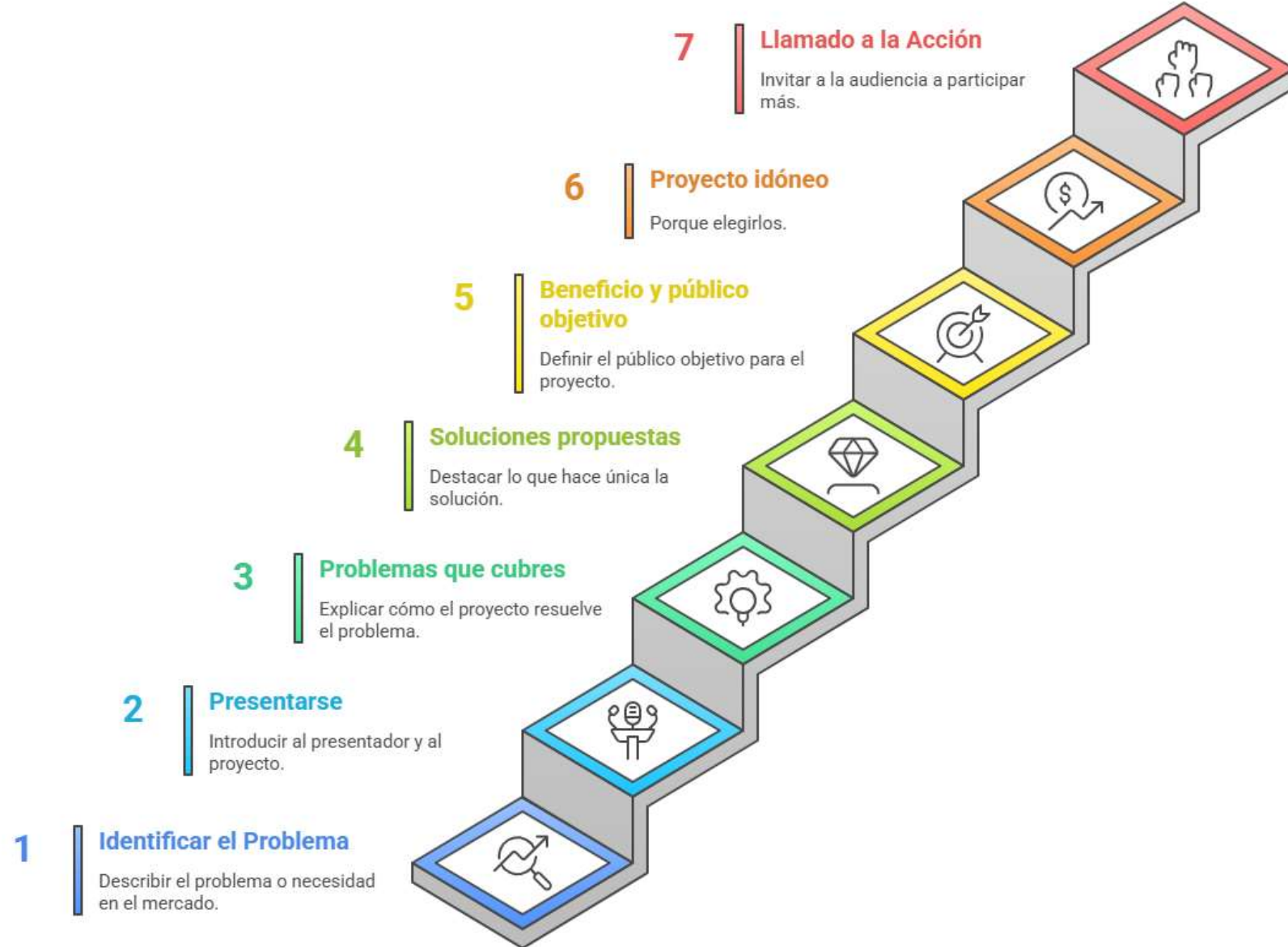
Un “elevator pitch” es un mensaje corto de aproximadamente un minuto de duración para contar tu proyecto de manera diferenciadora

Pasos del video del Elevator pitch

- 1 • Afirmación o pregunta sorprendente para llamar la atención
- 2 • Presentación: contar quién eres
- 3 • Problemas o necesidades que cubres
- 4 • ¿Qué soluciones aportas?
- 5 • ¿Qué beneficio principal obtiene la gente contigo?
- 6 • ¿Por qué tu proyecto es el idóneo?
- 7 • Termina con una llamada a la acción final

Pasos

Pasos para un Pitch de Proyecto Exitoso



Tips para elaborar un guión de elevator pitch

Consejos para el Elevator Pitch



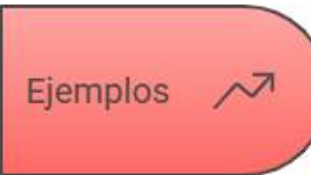
Conocer a tu audiencia y adaptar el mensaje.

Organizar la información de forma lógica.



Evitar tecnicismos para facilitar la comprensión.

Incorporar cifras y casos prácticos para credibilidad.



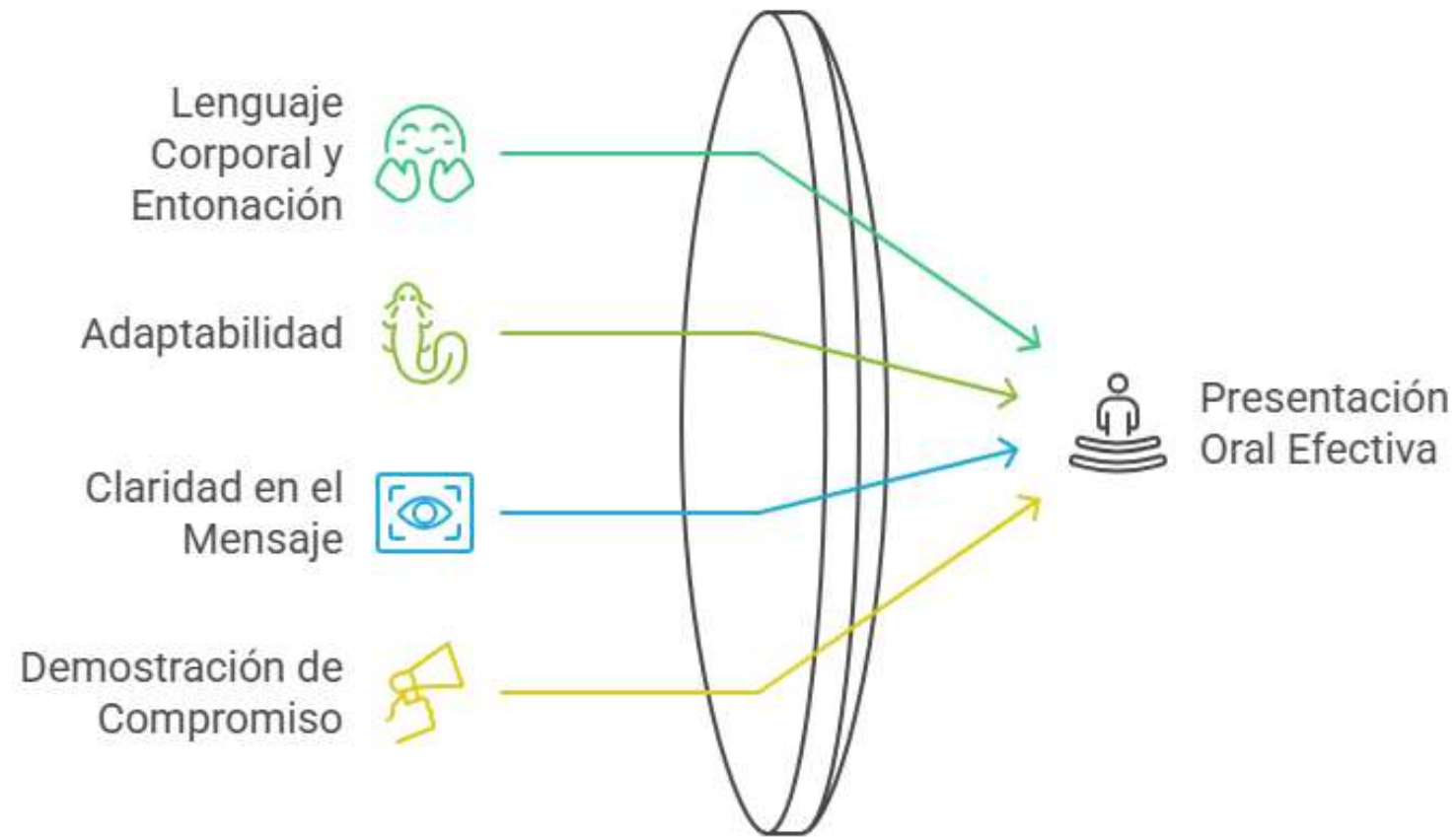
Practicar y ajustar el discurso según la audiencia.

Mantener la presentación dentro del tiempo estimado.



Presentación oral en vivo, consejos

Claves para una Presentación Exitosa



Algunos tips más para video grabado

Deben prestar atención a la comunicación NO verbal:

- La / las personas que aparecen en el video deben sonreír
- Utilizar los dedos para contar ideas o puntos importantes
- Mostrar y/o mover las manos en referencia a lo que se está contando

Referido a lo verbal:

- Deben poner diferentes énfasis en la voz para lograr comunicar efectivamente y resaltar cosas.
- Analizar cuándo modificar la voz y cuando utilizar gestos no verbales