

# LOS 10 MANDAMIENTOS DE UN INGENIERO DE REQUISITOS

UNIVERSIDAD DE HUELVA, ETSI

Trabajo realizado por:

Ismael Da Palma Fernández

Cristian Delgado Cruz

Juan Jiménez Serrano

# íNDICE

1. Introducción
2. Listado de los Mandamientos
3. Los Mandamientos
4. Conclusión
5. Bibliografía

# INTRODUCCIÓN

Nuestro trabajo consiste en seleccionar los diez mandamientos adecuados que un ingeniero de requisitos debe de tener en cuenta para minimizar los riesgos y maximizar las oportunidades de éxito en el desarrollo de software.

Además de la explicación en profundidad de cada una y el porqué de su elección.

# LOS MANDAMIENTOS

COMUNICACIÓN CON  
EL CLIENTE

CENTRALIZACIÓN DE  
ESFUERZOS

DOCUMENTACIÓN  
DE CAMBIOS

COMPRENSIÓN  
DEL DOMINIO  
DEL PROBLEMA

DOMINIO DE LAS  
HERRAMIENTAS

COORDINACIÓN DE  
EQUIPO

CONCIENCIARSE  
DE LOS LÍMITES

USO DE PLANTILLAS  
ESTANDARIZADAS

REALIZACIÓN DE  
PRUEBAS

COMPARAR  
INFORMACIÓN  
EXISTENTE

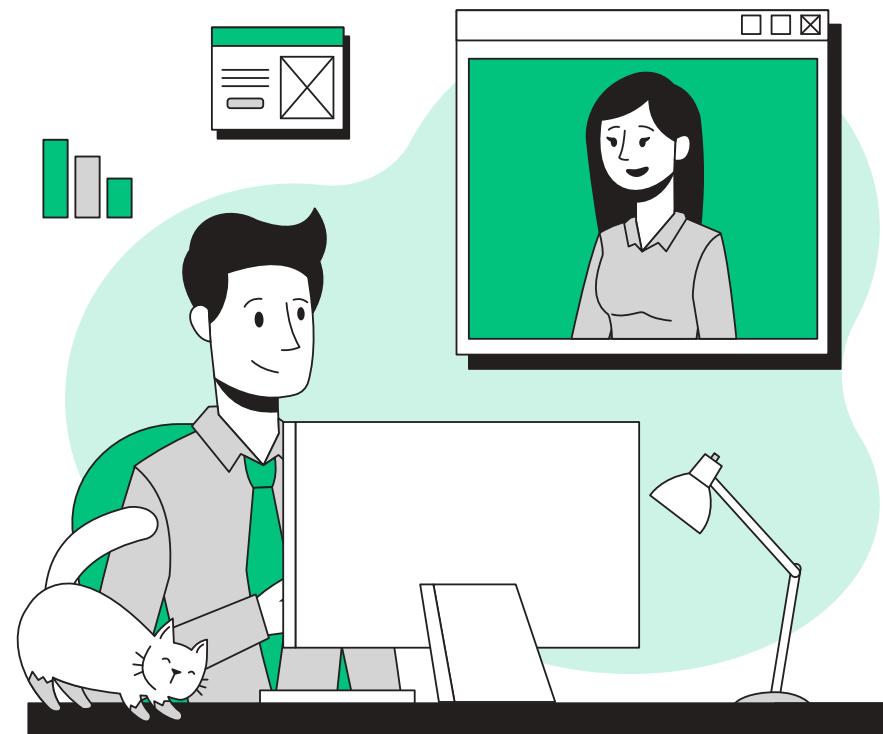
# 1. COMUNICACIÓN CON EL CLIENTE

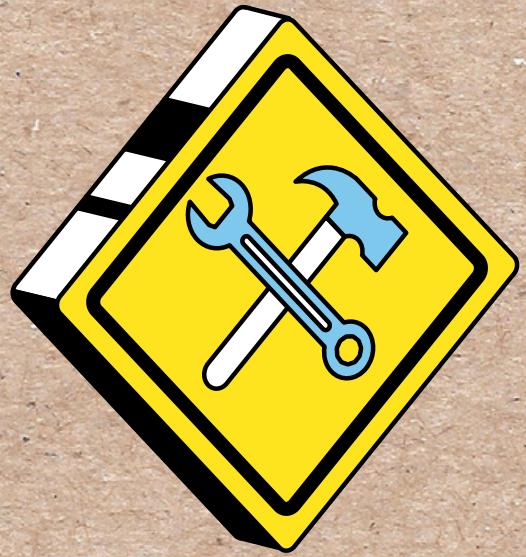


Una comunicación clara y fluida con el cliente es esencial para entender y satisfacer sus necesidades.



Sin llegar a molestar al cliente más de lo necesario, sería algo bueno que el cliente fuera consciente de como mejora y avanza el proyecto en todo momento, para poder minimizar todos los problemas de diseño.





## 2. DOMINIO DE LAS HERRAMIENTAS

Un buen dominio de las herramientas que se utilizan durante el proyecto es un rasgo muy positivo para evitar errores.

Instruir a todo el equipo en las mismas herramientas puede resultar costoso a nivel de tiempo y dinero, pero son recursos que se amortizan con el tiempo al no hacer errores que puedan ralentizar el proyecto.



### 3. CENTRALIZACIÓN DE ESFUERZOS



La centralización de esfuerzos consiste en reunir todos los recursos disponibles para conseguir un objetivo común de forma eficiente, en este caso, la satisfacción del cliente y los usuarios finales.



Con esto nos referimos a cumplir con las necesidades esenciales del proyecto y dejar de lado los añadidos, hasta que el objetivo general se haya cumplido correctamente.



## 4. COORDINACIÓN DE EQUIPO



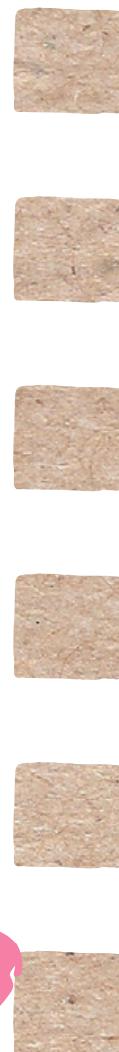
La coordinación del equipo asegura una visión integral de los requisitos y su implementación, además de evitar la redundancia e incoherencia en el trabajo.



Incluye varios aspectos, como la definición clara de roles y responsabilidades, la comunicación efectiva, la asignación de tareas, la planificación y el seguimiento del progreso del trabajo.

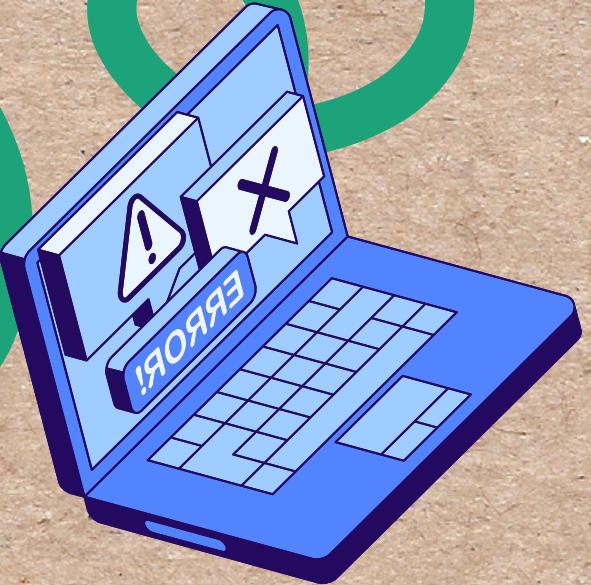


## 5. COMPARAR INFORMACIÓN EXISTENTE



Comparar información existente ayuda a identificar similitudes y usarla como referencias o evitar errores o ineficiencias en el nuestro, por ejemplo basarnos en el post-mortem de otros proyectos similares para evitar los mismos errores y así evitar entrar en zonas críticas o alcanzar un final no deseado.





## 6. DOCUMENTACIÓN DE CAMBIOS



La documentación de cambios es útil para realizar mantenimiento del software a largo plazo. Si surge un problema, el equipo pueden referirse a esta para comprender qué se ha modificado recientemente y cómo se ha afectado el software.



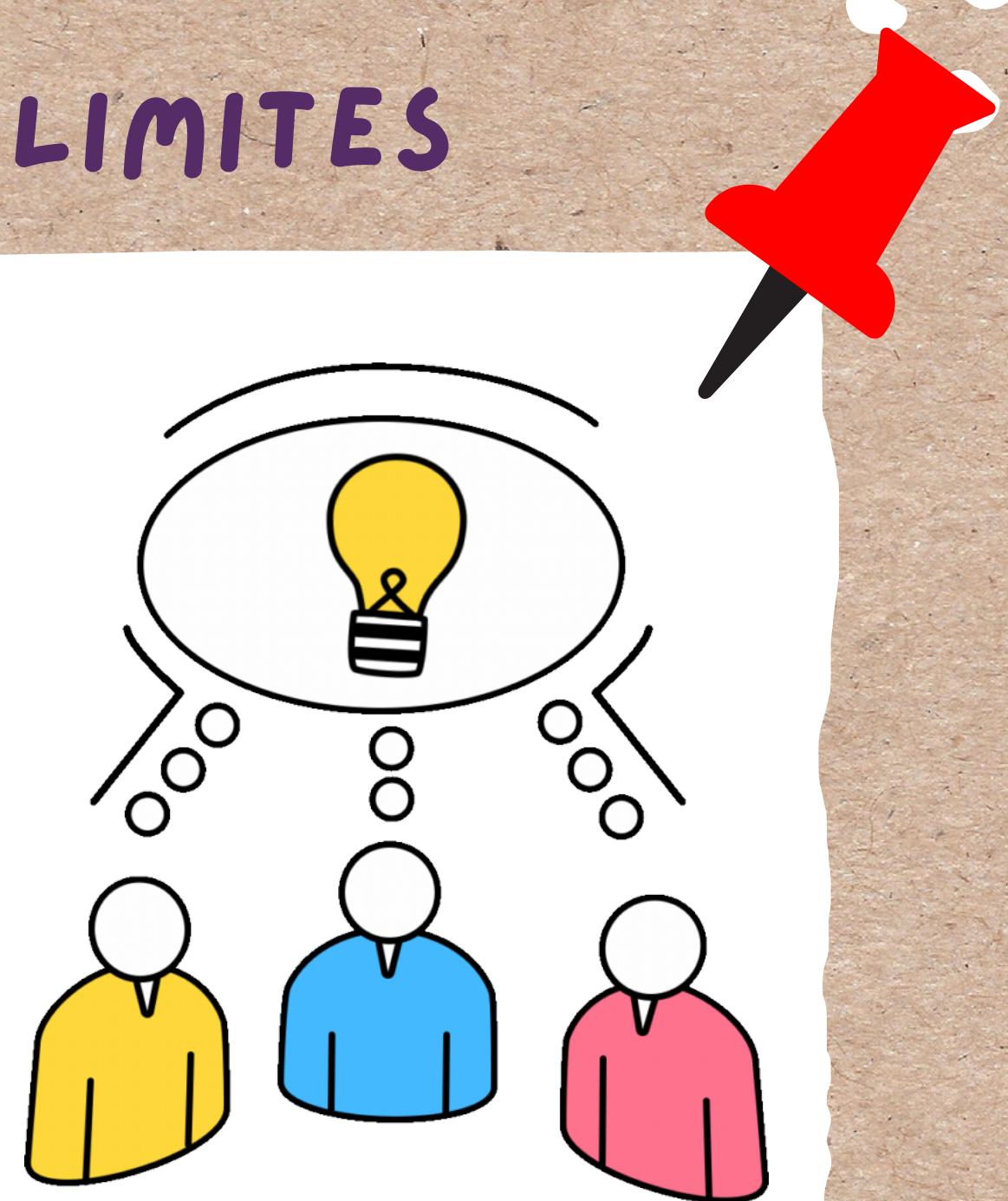
También es útil para hacer un seguimiento del progreso del proyecto, lo que permite ver qué se ha hecho y cuánto se ha avanzado.

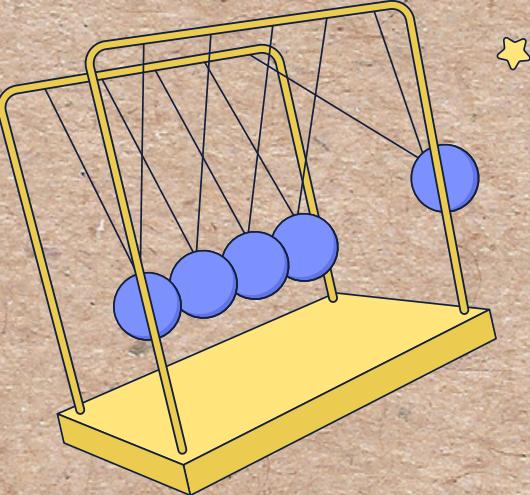




## 7. CONCIENCIARSE DE LOS LIMITES

Concienciarse de los límites ayuda a definir requisitos realistas y alcanzables, evitando la implementación o intento de implementación de objetivos, inefficientes, innecesarios o imposibles entre otros a nuestro proyecto.

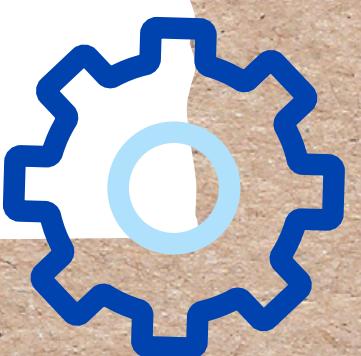




## 8. COMPRENSIÓN DEL DOMINIO DEL PROBLEMA

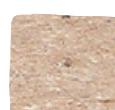


La comprensión del dominio del problema es fundamental para especificar los requisitos adecuadamente, una mala compresion del mismo puede llevarnos a generar requisitos incongruentes con otros, o generar requisitons innecesarios o inutil, o incluso generar requisitos que mas adelante puedan llegar a ser una molestia para el usuario final, o incluso el proyecto en si.





## 9. USO DE PLANTILLAS ESTANDARIZADAS



El uso de plantillas estandarizadas puede facilitar la creación de requisitos y ayudarnos a comprender incluso partes del problema no planteadas o cuestionadas, debido a la falta de experiencia o de perspectiva, además de evitar gastar mas tiempo de lo necesario en las partes mas simples en la creación del propio proyecto.





## 10. REALIZACIÓN DE PRUEBAS.



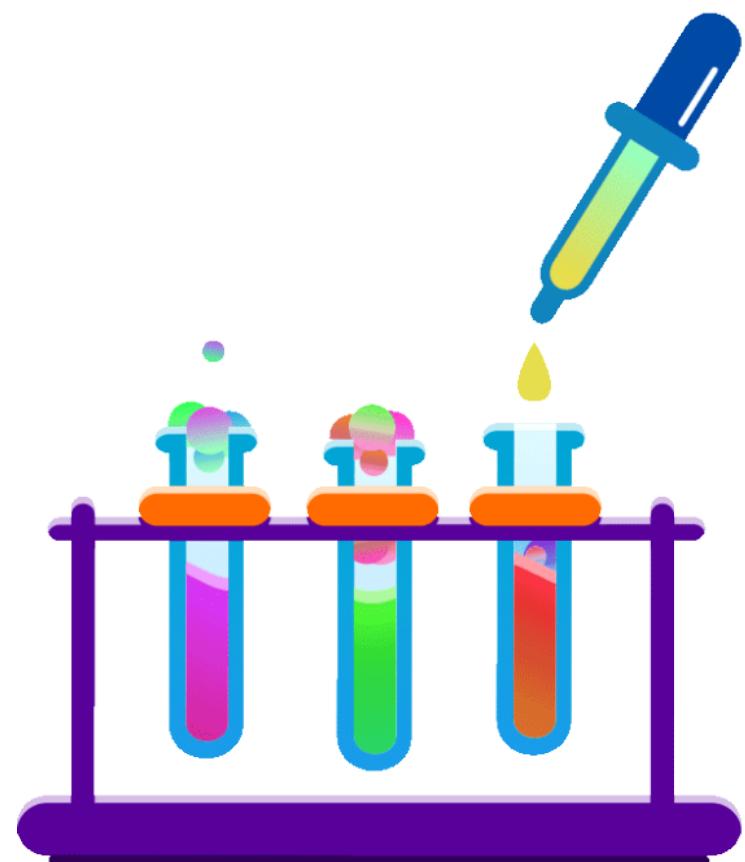
Las pruebas ayudan a identificar errores y defectos en el software, identificar y corregir estos errores antes del lanzamiento del software puede evitar problemas costosos a largo plazo.



Por otra parte también ayudan a mejorar la calidad del software y a asegurarse de que el software funcione de acuerdo con las necesidades del cliente



En algunos casos, las pruebas son necesarias para cumplir con regulaciones y estándares específicos en la industria.



# CONCLUSIONES

## 1º Conclusión

El seguimiento de las reglas adecuadas en la ingeniería de requisitos es esencial para el éxito del proyecto.

## 2º Conclusión

La parte más importante de todo proyecto es la comunicación y la utilización de metodologías fáciles y entendibles por todo el mundo.

## 3º Conclusión

Un factor en cuenta muy importante sobre todo es la actitud ante los problemas, todo trabajo puede llegar a tener algún bache, pero una buena actitud puede ayudar a solucionar todos los problemas.

# BIBLIOGRAFIA

- 1º [VisureResolutions](#)
- 2º [Asana](#)
- 3º [MicroFocus \(PDF\)](#)
- 4º [Slideshare \(Presentación\)](#)

**GRACIAS**

¿Hay Alguna Pregunta?

