



**Universidad Nacional
Autónoma de México**



Facultad de Ingeniería

Guión para realizar el video.

Ingeniería en software

Grupo: 4

Equipo: 5

Semestre: 2026 – 1

Profesor: Orlando Zaldívar Zamorategui

Fecha de entrega:

13 de Noviembre 2025

**Líder del equipo: Domínguez Ruiz Samantha Atziry -
322273672**


Tiempo	Texto
0:00 – 0:10	[Imágenes relacionadas con tecnología] La ingeniería de software no comienza escribiendo código, sino entendiendo lo que el cliente necesita.
0:10 – 0:25	En este tutorial se explicará el proceso de obtención de requerimientos, su importancia en el desarrollo de software y las principales técnicas utilizadas para lograr un producto exitoso.
0:25 – 0:35	El objetivo es comprender cómo una buena gestión de requerimientos garantiza la calidad, funcionalidad y satisfacción del usuario final.
0:35 – 0:40	La pantalla pasa a la siguiente diapositiva. Ya sea una transición de diapositiva o en la edición de video. Se muestra el título.
0:40 – 0:55	Los requerimientos de software son la descripción de las funciones, características y restricciones que un sistema debe cumplir. Klaus Pohl, en 2010, los comparó con los planos de una construcción: si están mal definidos, el proyecto corre el riesgo de fracasar.
0:55 – 1:05	En esta etapa se busca comprender las necesidades reales del usuario antes de diseñar o programar. Se presentan los tipos de requerimientos con otra diapositiva.
1:05 – 1:10	Existen dos tipos principales de requerimientos:
1:10 – 1:30	Un texto en algún lado de la pantalla “Requerimientos funcionales” Son las acciones o servicios que el sistema debe realizar. Ejemplo uno: El usuario debe poder iniciar sesión con su correo y contraseña. Ejemplo dos: El sistema debe generar reportes mensuales en formato PDF. Ejemplo tres: El administrador debe poder agregar o eliminar usuarios.
1:30 – 1:55	“Requerimientos no funcionales” Se refieren a las cualidades o restricciones

	del sistema, es decir, cómo realiza sus funciones. Ejemplo uno: La página principal debe cargar en menos de dos segundos. Ejemplo dos: Las contraseñas deben almacenarse cifradas usando el algoritmo SHA-256. Ejemplo tres: El sistema debe tener una disponibilidad del 99.9%.
1:55 – 2:10	<i>Elizabeth Hull “Aquí hay que mostrar una foto de la portada del libro”, en 2017, menciona que esta clasificación es esencial para que todos los miembros del equipo comprendan de forma clara las expectativas del proyecto y los criterios de calidad del producto final.</i>
2:10 – 2:25	Una correcta gestión de requerimientos garantiza el éxito de un proyecto. James Crowder, en 2022, señala que los errores cometidos en esta etapa son los más costosos de corregir, ya que pueden afectar todo el proceso posterior.
2:25 – 2:40	Cuando los requerimientos están completos, claros y documentados, el equipo trabaja con una visión común. Entre los principales beneficios destacan: Permitir la alineación entre usuarios, clientes y desarrolladores. Servir como guía para el diseño, la implementación y las pruebas. Facilitar la validación del producto final. Y reducir riesgos innecesarios durante el desarrollo.
2:40 – 2:45	<i>A partir de aquí se tiene que proyectar algo relacionado para “técnicas para obtener requerimientos”.</i>
2:45 – 2:55	<i>El proceso de ingeniería de requerimientos incluye la identificación, análisis, documentación, validación y gestión de las necesidades del usuario.</i>
2:55 – 3:10	<i>Según Edgar Serna (2019) y Phillip Laplante (2022), no existe una técnica única. Cada proyecto requiere una combinación de métodos que se adapten a su contexto.</i>
3:10 – 3:20	<i>Las palabras en negritas deben</i>

	<i>proyectarse al mismo tiempo que aparece una imagen en el video relacionada con la técnica. Primero, las entrevistas. Se realizan reuniones entre analistas y usuarios para recopilar información directa.</i>
3:20 – 3:30	Su principal ventaja es que permiten profundizar en los detalles y aclarar dudas, aunque pueden requerir mucho tiempo si hay varios involucrados.
3:30 – 3:40	<i>Los cuestionarios / encuestas permiten recopilar información de un gran número de usuarios de forma rápida. Son económicos y fáciles de analizar, aunque las respuestas pueden ser generales o poco precisas.</i>
3:40 – 3:50	<i>La observación directa consiste en analizar cómo los usuarios realizan sus actividades diarias. Esta técnica ayuda a descubrir necesidades no expresadas, aunque puede influir en el comportamiento natural del usuario.</i>
3:50 – 4:00	<i>El análisis de documentación permite conocer los procesos actuales revisando manuales, formularios o reportes antiguos. Es útil cuando no se dispone de tiempo con los usuarios, aunque la información puede estar desactualizada.</i>
4:00 – 4:10	<i>Los talleres / workshops reúnen a usuarios, analistas y desarrolladores para discutir y priorizar requerimientos. Fomentan la colaboración, pero requieren buena organización para evitar confusiones.</i>
4:10 – 4:20	<i>El prototipado consiste en construir versiones preliminares del sistema para que los usuarios visualicen su funcionamiento. Permite validar requerimientos y detectar errores antes del desarrollo real, aunque puede generar falsas expectativas si no se aclara que es solo un modelo.</i>
4:20 – 4:30	<i>La tormenta de ideas promueve la creatividad del equipo y ayuda a generar nuevas funciones o mejoras. Es importante filtrar las ideas</i>

	<i>posteriormente para conservar sólo las viables.</i>
4:30 – 4:40	<i>Las historias de usuario se utilizan en metodologías ágiles. Describen una funcionalidad desde el punto de vista del usuario final. Por ejemplo: “Como cliente, quiero recibir una notificación cuando mi pedido sea enviado.” Son fáciles de entender, aunque pueden omitir detalles técnicos.</i>
4:40 – 4:50	<i>Mostrar el diagrama de flujo relacionando cada técnica con las etapas de la ingeniería de requerimientos. Estas técnicas se pueden organizar dentro de las fases del proceso de ingeniería de requerimientos: Identificación, análisis, documentación, validación y gestión.</i>
4:50 – 4:55	<i>El diagrama del tutorial mostrará cómo cada técnica se relaciona con estas etapas para obtener resultados más precisos.</i>
4:55 – 5:00	<i>Aquí es la despedida del video Los requerimientos de software son el fundamento sobre el cual se construye cualquier sistema. Un mal levantamiento puede ocasionar fallas graves, sobrecostos y pérdida de tiempo. Por el contrario, una buena gestión de requerimientos garantiza que el producto final sea funcional, confiable y de alta calidad.</i>

Ingeniería de requerimientos.



```
graph TD; A[Ingeniería de requerimientos.] --> B[1. Identificación. Entradas: Entrevistas, Encuestas, Observación directa]; B --> C[2. Análisis y priorización Entradas: Talleres de trabajo, Tormenta de ideas, Análisis de documentos]; C --> D[3. Documentación Entradas: Especificación de RF y RNF, Historias de usuario]; D --> E[4. Validación Entradas: Prototipado, Retroalimentación cliente]; E --> F[5. Gestión y mantenimiento Entradas: Control de cambios, Revisión de alcance];
```

1. Identificación.

Entran:

- Entrevistas
- Encuestas
- Observación directa

2. Análisis y priorización

Entran:

- Talleres de trabajo
- Tormenta de ideas
- Análisis de documentos

3. Documentación

Entran:

- Especificación de RF y RNF
- Historias de usuario

4. Validación

Entran:

- Prototipado
- Retroalimentación cliente

5. Gestión y mantenimiento

Entran:

- Control de cambios
- Revisión de alcance

Referencias:

Klaus Pohl. *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer, 2010.

Elizabeth Hull, Ken Jackson, Jeremy Dick. *Requirements Engineering*. Springer, 2017.

Karl Wieggers, Candace Hokanson. *Software Requirements Essentials*. Pearson, 2023.

James A. Crowder, Curtis W. Hoff. *Requirements Engineering: Laying a Firm Foundation*. Springer, 2022.

Phillip A. Laplante, Mohamad Kassab. *Requirements Engineering for Software and Systems*. Routledge, 2022.

Isabel María del Águila Cano. *Fundamentos de Ingeniería de los Requisitos*. Editorial UAL, 2016.

Ralph R. Young. *The Requirements Engineering Handbook*. Artech House, 2004.

Alistair Sutcliffe. *User-Centred Requirements Engineering*. Springer, 2002.

Edgar Serna M. *Gestión de la Ingeniería de Requisitos Integrando Principios del Pensamiento Complejo*. 2019.

Carlos Eduardo Vázquez, Guilherme Simões. *Ingeniería de Requisitos: Software Orientado al Negocio*. Unimart, 2018.