

TA-informe

Luis Alberto Paca Pilataza

SEMANA 11

TÉCNICA DE ELICITACIÓN DE REQUISITOS

Las técnicas de elicitación de requisitos son un conjunto de prácticas esenciales en la ingeniería de software, cuyo propósito es descubrir, obtener y documentar los requerimientos de los usuarios y de las partes interesadas para un nuevo sistema o una mejora en uno existente. El objetivo principal es garantizar que el producto final satisfaga las necesidades reales del cliente y del negocio. El éxito de esta fase no solo depende del conocimiento técnico, sino también de la capacidad de comprender el contexto del problema y la perspectiva del cliente, lo que incluye sus actividades diarias y sus expectativas [1] [2].

Role Playing

Una de las técnicas más comunes para la elicitación de requisitos es la role playing, considerada fundamental en la práctica de la ingeniería de requisitos. Al asignar roles es un desafío, especialmente para analistas novatos, ya que el éxito depende en gran medida de las habilidades interpersonales y la experiencia [3]. otra técnica ampliamente utilizada es la Sesión de Desarrollo de Aplicaciones Conjuntas (JAD), que simula una práctica común en la industria al reunir a los desarrolladores y a los clientes para resolver conflictos y definir los requerimientos de un nuevo sistema [4].

El desarrollo de habilidades blandas es crucial para una elicitación de requisitos exitosa. Habilidades como la comunicación, la empatía, el razonamiento analítico y la resolución de conflictos son altamente valoradas en los ingenieros de requisitos [1]. La enseñanza de estas habilidades a los estudiantes de ingeniería de software es particularmente difícil debido a la naturaleza experiencial de la elicitación [3]. Por esta razón, se recomiendan actividades como los juegos de roles (role playing), la autoevaluación y la revisión por pares para que los estudiantes adquieran experiencia práctica, reflexionen sobre sus errores y mejoren sus habilidades de entrevista [5].

A pesar de la importancia de estas técnicas, el proceso de elicitación enfrenta varios desafíos. En entornos de aprendizaje remoto, por ejemplo, desarrollar habilidades sociales puede ser más difícil debido a la interacción limitada [1]. Además, los novatos tienden a cometer errores durante las entrevistas de requisitos, lo que puede llevar a una documentación de requisitos incompleta o ambigua, impactando negativamente las etapas futuras del proyecto [3]. Por lo tanto, es esencial que las técnicas utilizadas fomenten una comunicación clara y una comprensión completa para evitar estos problemas [5].

Delphi

El método Delphi se distingue por varias características clave. Es un proceso estructurado, organizado e iterativo que utiliza una serie de cuestionarios para destilar y correlacionar las opiniones de un panel de expertos. Entre rondas, los participantes reciben retroalimentación controlada, lo que les permite ver cómo sus opiniones se comparan con las del grupo, y el proceso continúa hasta que se alcanza un consenso confiable. La técnica garantiza el anonimato de las respuestas individuales, lo que ayuda a

evitar el sesgo causado por la influencia de personalidades dominantes o la presión del grupo, fomentando una comunicación indirecta que promueve la inteligencia colectiva [6] [7].

Este método ha sido aplicado eficazmente en el campo de la ingeniería de software y los sistemas de información para abordar problemas complejos. Por ejemplo, ha sido utilizado para identificar los elementos a considerar al asignar roles en proyectos de software, integrando herramientas que evalúan habilidades y rasgos de personalidad [8]. Además, se utiliza para promover la construcción de consenso en la planificación del transporte y en la definición de objetivos, criterios de evaluación y alternativas en diversos proyectos [9].

A pesar de su versatilidad y amplia aplicación, el método Delphi presenta algunas limitaciones. Una crítica frecuente es la falta de pruebas estadísticas estandarizadas, lo que puede afectar su fiabilidad y validez. Sin embargo, la incorporación de estadísticas no paramétricas puede mitigar esta deficiencia y mejorar el rigor del método como una herramienta de investigación [7]. La adaptabilidad de la técnica a diferentes contextos y su capacidad para permitir la participación asincrónica de expertos geográficamente dispersos son sus principales beneficios [8].

Peer Review

La validación de requisitos de software es una fase fundamental en el proceso de la ingeniería de requisitos, la cual se diferencia de la elicitación. Mientras que la elicitación se enfoca en el descubrimiento de las necesidades de los interesados, la validación busca asegurar que los requisitos obtenidos sean correctos, consistentes y completos [10] [11]. Esta actividad es esencial para garantizar que las características que se implementen en el producto final correspondan fielmente a lo que los usuarios y clientes realmente necesitan y esperan.

Una de las técnicas más importantes para llevar a cabo la validación de requisitos es la revisión por pares (Peer Review). Este proceso implica que un grupo de "pares" o colegas revisen de manera crítica la especificación de requisitos para identificar ambigüedades, inconsistencias o fallas. Esta revisión se considera un método de evaluación empírica de los requisitos, ayudando a asegurar la claridad y comprensibilidad del documento, aspectos que son evaluados de manera rigurosa para evitar errores en fases posteriores del desarrollo de software [12] [10]. La aplicación de estas revisiones es crucial en proyectos con desarrollo continuo para asegurar la calidad de los requisitos antes de que pasen a la fase de implementación [11].

Análisis de escenarios futuros o Foresight

Dentro de las técnicas de elicitación, el uso de escenarios es una herramienta efectiva para que los interesados descubran los requisitos. Un escenario es una secuencia de eventos con una estructura narrativa que permite modelar situaciones de uso del sistema [13]. Son un vehículo de comunicación que ayuda a los usuarios finales y a los expertos del dominio a comprender y validar los requisitos. Su uso se ha extendido en la práctica debido a su capacidad para facilitar el aprendizaje interdisciplinario y servir como un medio para la división de tareas en un proyecto [14].

El análisis de escenarios futuros / Foresight es una técnica de elicitación que utiliza el concepto de escenarios para explorar posibles futuros inciertos y anticipar los requisitos que puedan surgir en ellos [15]. Esta técnica es especialmente útil en contextos dinámicos donde los requisitos pueden cambiar con el tiempo. Se basa en la premisa de que un requisito influenciado por el futuro es aquel creado o

modificado a partir de un evento futuro. Un método para aplicar esto es la "Rueda de Futuros" (Futures Wheel), que modela las consecuencias de un evento futuro en el sistema[16].

La aplicación de métodos de Foresight, como la Rueda de Futuros, tiene la ventaja de proporcionar una visión más rica de los requisitos, ya que no se limita a las necesidades actuales del sistema. Al considerar los obstáculos y las situaciones no ideales, se obliga a los diseñadores a pensar en soluciones flexibles y robustas [14]. Sin embargo, esta técnica requiere un equilibrio, ya que si bien anticipar los cambios puede ser gratificante si la previsión es correcta, también puede generar costos innecesarios si no lo es. Para su implementación, se pueden utilizar las mismas prácticas de elicitación general, como entrevistas y grupos focales, para recopilar la información necesaria [16].

ESCENARIOS DE FLUJO

Escenario 1: Gestión de Temas y Anteproyectos

En la siguiente **Tabla 1** se describe el escenario de flujo en el que el estudiante selecciona un tema y registra su anteproyecto, el tutor revisa y el coordinador valida, notificando el sistema el resultado final.

Paso	Actor	Información	Acción / Proceso	Canal	Destino
1	Estudiante	Banco de temas disponible en el SGTT	Consulta los temas por categoría (investigación o tecnológico)	Plataforma SGTT (web)	Pantalla de selección de temas
2	Estudiante	Tema seleccionado	Registra su anteproyecto en el sistema	Plataforma SGTT	Registro de propuesta
3	Tutor	Anteproyecto del estudiante	Revisa la propuesta y añade observaciones	Plataforma SGTT	Retroalimentación almacenada
4	Coordinador	Anteproyecto revisado y observaciones	Valida el tema y aprueba/rechaza	Plataforma SGTT	Estado final de la propuesta
5	Sistema SGTT	Decisión del coordinador	Notifica automáticamente al estudiante y tutor	Correo electrónico / notificación interna	Buzón de mensajes de estudiante y tutor

Tabla 1: selección de tema del anteproyecto

Escenario 2: Gestión Documental

En la siguiente **Tabla 2** se describe el escenario se enfoca en el flujo donde el estudiante sube documentos, el tutor los revisa, el coordinador supervisa y el sistema asegura la trazabilidad mediante notificaciones automáticas.

Paso	Actor	Información	Acción / Proceso	Canal	Destino
1	Estudiante	Documento (anteproyecto, borrador, informe final)	Sube archivo al sistema	Plataforma SGTT (módulo documental)	Repositorio central del SGTT
2	Sistema SGTT	Documento cargado	Genera versión y fecha de registro automática	Plataforma SGTT	Historial de versiones
3	Tutor	Documento subido por estudiante	Revisa y comenta el documento	Plataforma SGTT	Registro de observaciones
4	Estudiante	Retroalimentación del tutor	Descarga o visualiza comentarios	Plataforma SGTT	Panel del estudiante
5	Coordinador	Historial de entregas	Supervisa y valida cumplimiento	Plataforma SGTT	Estado de revisión actualizado
6	Sistema SGTT	Aprobación final	Notifica a todos los actores involucrados	Notificación interna más correo	Estudiante y tutor

Tabla 2: subir documentos (anteproyecto o informe)

SEMANA 11

Construye un diagrama de casos de uso

"En el autoservicio de comida al peso, lo que hacemos es algo bastante dinámico. Primero, cuando llega un cliente, necesitamos tener un sistema que registre sus datos, sobre todo si es un cliente frecuente, para poder identificarlo rápidamente y hasta ofrecerle descuentos o promociones. Luego, lo más importante es el proceso en la línea de autoservicio: los clientes eligen sus alimentos y nosotros pesamos su plato. Aquí es donde necesitamos un sistema que calcule el costo automáticamente, basándose en el peso y los precios que tenemos asignados para cada categoría de comida.

Después de eso, pasamos a la facturación. Queremos que sea rápido y sin errores, así que necesitamos emitir comprobantes electrónicos y permitir pagos con tarjeta, efectivo o incluso apps. Algo clave es el inventario. Nos aseguramos de que siempre haya suficientes insumos en la cocina, pero sería ideal tener algo que registre automáticamente las entradas y salidas, para no quedarnos cortos en un día ocupado.

Por último, pensamos en los datos. Un sistema que nos genere reportes de las ventas, qué alimentos se consumen más y hasta las horas pico sería increíble para mejorar la operación. Todo esto debería

estar conectado para que tanto el cliente como nosotros tengamos una experiencia fluida y organizada."

En la siguiente **Figura 1** se muestra un caso de uso general sobre la gestión de un autoservicio, la interacción entre actores con el sistema.

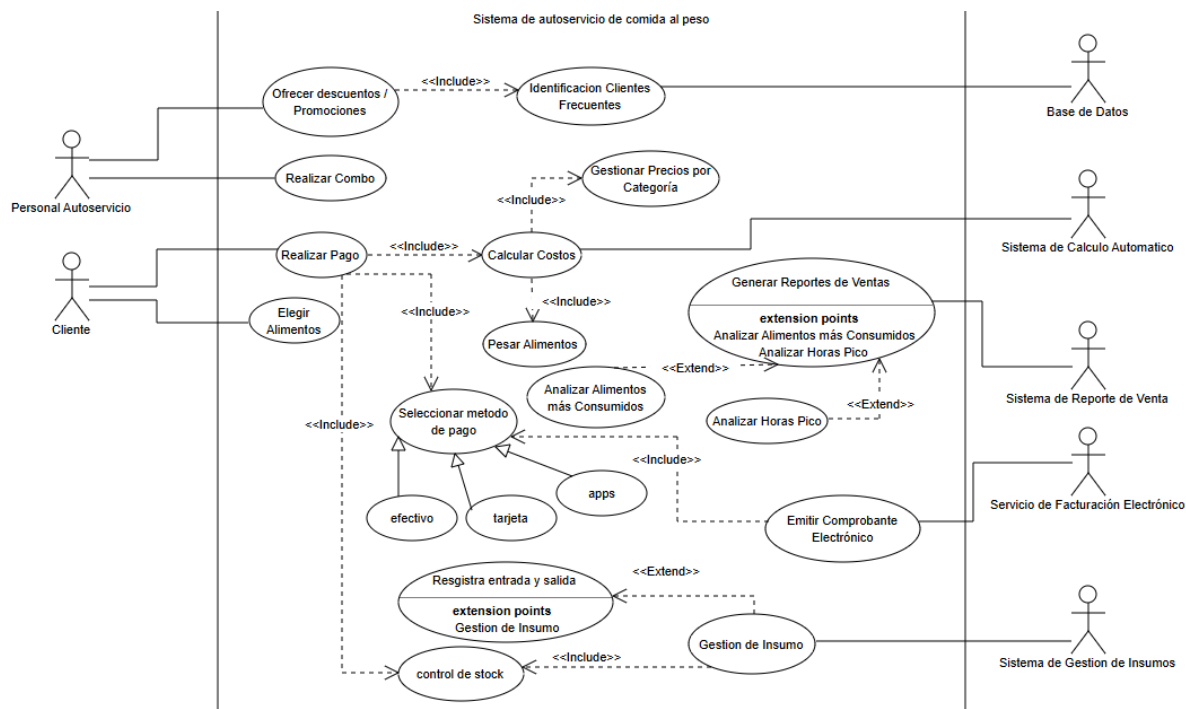


Figura 1: Caso de Uso de Autoservicio

Historias de usuario, acerca del caso de autoservicio

En las siguientes tablas se muestran las historias de usuario construidas a partir del análisis del sistema de autoservicio de comida al peso. Cada historia sigue la estructura CCC (Carta, Conversación y Confirmación), lo que permite especificar el rol del usuario, sus necesidades y los criterios de aceptación.

Historias de usuario: Registro de cliente frecuente		
CCC	ELEMENTOS	DESCRIPCIÓN
Carta	¿Quién?	Cliente frecuente del autoservicio.
	¿Qué?	Quiero que el sistema registre mis datos automáticamente al ingresar.
	¿Para qué?	Para ser identificado rápidamente y que se apliquen descuentos o promociones personalizadas.
Conversación	Descripción	El sistema debe reconocer automáticamente al cliente y mostrar sus beneficios o historial de compras.

Confirmación	Criterio de Aceptación	<ul style="list-style-type: none"> El sistema debe identificar al cliente por medio de su número de cliente o tarjeta de fidelidad. Mostrar historial de compras y promociones aplicables. Debe integrarse con el sistema de facturación y descuentos.
--------------	------------------------	---

Tabla 3: Registro de cliente frecuente

Historias de usuario: Pesar plato y calcular precio automáticamente		
CCC	ELEMENTOS	DESCRIPCIÓN
Carta	¿Quién?	Cliente del autoservicio.
	¿Qué?	Quiero que el sistema pese mi plato y calcule el precio automáticamente.
	¿Para qué?	Para saber cuánto debo pagar de manera rápida y precisa, sin errores manuales.
Conversación	Descripción	El sistema calcula el precio automáticamente según el peso y categoría de cada alimento.
Confirmación	Criterio de Aceptación	<ul style="list-style-type: none"> El cálculo debe ser preciso. Mostrar información de peso y precio por separado. Integración con el sistema de facturación.

Tabla 4: Pesar plato y calcular precio automáticamente

Historias de usuario: Emisión de comprobante electrónico		
CCC	ELEMENTOS	DESCRIPCIÓN
Carta	¿Quién?	Cliente del autoservicio.
	¿Qué?	Quiero que el sistema emita un comprobante electrónico después de pagar.
	¿Para qué?	Para tener un registro de mi compra y poder presentarlo si es necesario.
Conversación	Descripción	El sistema genera un comprobante electrónico con el detalle de los alimentos, cantidades y precios.
Confirmación	Criterio de Aceptación	<ul style="list-style-type: none"> Comprobante electrónico debe incluir fecha, hora, items y precios.

		<ul style="list-style-type: none"> • Debe poder enviarse por correo o mostrarse en pantalla. • Integración con el sistema de pago y facturación.
--	--	--

Tabla 5: Emisión de comprobante electrónico

Historias de usuario: Registro automático de inventario		
CCC	ELEMENTOS	DESCRIPCIÓN
Carta	¿Quién?	Personal de cocina o administrador del autoservicio.
	¿Qué?	Quiero que el sistema registre automáticamente las entradas y salidas de insumos.
	¿Para qué?	Para asegurarme de que siempre haya suficiente stock y evitar faltantes.
Conversación	Descripción	El sistema debe actualizar el inventario al pesarse y facturarse los alimentos, mostrando alertas si algún insumo está bajo.
Confirmación	Criterio de Aceptación	<ul style="list-style-type: none"> • Inventario actualizado en tiempo real según consumo y reposición. • Alertas automáticas si algún insumo está bajo. • Integración con el módulo de compras y planificación.

Tabla 6: Registro automático de inventario

Historias de usuario: Generación de reportes de ventas		
CCC	ELEMENTOS	DESCRIPCIÓN
Carta	¿Quién?	Sistema del autoservicio.
	¿Qué?	Quiero que el sistema genere reportes de ventas, alimentos más consumidos y horas pico.
	¿Para qué?	Para mejorar la operación y la toma de decisiones estratégicas.
Conversación	Descripción	El sistema analiza la información de ventas y genera reportes gráficos y tablas que faciliten la planificación
Confirmación	Criterio de Aceptación	<ul style="list-style-type: none"> • Reportes deben mostrar ventas por día, hora y categoría de alimento.

		<ul style="list-style-type: none"> • Mostrar tendencias y alimentos más consumidos. • Debe ser exportable y compatible con herramientas de análisis.
--	--	--

Tabla 7: Generación de reportes de ventas

BIBLIOGRAFIA

- [1] Gretchen Torres de Macedo, "View of Building soft skills through a role-play based approach for Requirements Engineering remote education", Accessed: Aug. 16, 2025. [Online]. Available: <https://journals-sol.sbc.org.br/index.php/jbcs/article/view/3071/2554>
- [2] K. Wieggers and J. Beatty, "Software Requirements, Third Edition".
- [3] "Sci-Hub | | 10.1007/s00766-020-00334-0." Accessed: Aug. 16, 2025. [Online]. Available: <https://sci-hub.se/https://doi.org/10.1007/s00766-020-00334-0>
- [4] G. Costain and B. Mckenna, "Experiencing the Elicitation of User Requirements and Recording them in Use Case Diagrams through Role-Play".
- [5] "Sci-Hub | | 10.1109/RE.2019.00015." Accessed: Aug. 16, 2025. [Online]. Available: <https://sci-hub.se/10.1109/RE.2019.00015>
- [6] A. Alarabiat and I. Ramos, "The Delphi Method in Information Systems Research," *The Electronic Journal of Business Research Methods*, vol. 17, no. 2, pp. 86–99, 2019, doi: 10.34190/JBRM.17.2.04.
- [7] B. Ju and T. Jin, "Incorporating nonparametric statistics into Delphi studies in library and information science," vol. 18, no. 3, 2013, Accessed: Aug. 16, 2025. [Online]. Available: <http://informationr.net/ir/18-3/paper589.html>
- [8] D. Varona and L. Fernando Capretz, "Using the DELPHI Method for Model for Role Assignment in the Software Industry," pp. 1–7, doi: 10.1109/IECON48115.2021.9589957.
- [9] M. Le Pira, G. Inturri, M. Ignaccolo, and A. Pluchino, "Modelling consensus building in Delphi practices for participated transport planning-review under responsibility of WORLD CONFERENCE ON TRANSPORT RESEARCH SOCIETY," 2017.
- [10] C. Gagliardi -, "Practicas y técnicas de elicitación y validación de requisitos en emprendimientos tecnológicos de software".
- [11] "Vista de Validación y verificación de requisitos de software en proyectos basados en desarrollo continuo usando técnicas de PLN." Accessed: Aug. 16, 2025. [Online]. Available: <https://acofipapers.org/index.php/eiei/article/view/2657/1794>
- [12] "Sci-Hub | | 10.1109/ESEM.2009.5314232." Accessed: Aug. 16, 2025. [Online]. Available: <https://sci-hub.se/10.1109/ESEM.2009.5314232>

- [13] "Sci-Hub | | 10.1007/s00766-009-0077-9." Accessed: Aug. 16, 2025. [Online]. Available: <https://sci-hub.se/10.1007/s00766-009-0077-9>
- [14] G. D. Gil, "HERRAMIENTA PARA IMPLEMENTAR LEL Y ESCENARIOS (TILS)," 2002.
- [15] K. Cordova-Pozo and E. A. J. A. Rouwette, "Types of scenario planning and their effectiveness: A review of reviews," *Futures*, vol. 149, p. 103153, May 2023, doi: 10.1016/J.FUTURES.2023.103153.
- [16] "Sci-Hub | | 10.4018/jismd.2012040101." Accessed: Aug. 16, 2025. [Online]. Available: <https://sci-hub.se/10.4018/jismd.2012040101>