

Informe Consolidado: Técnicas de Elicitación y Modelado de Requisitos para Sistemas de Gestión y Autoservicio

Informe de Actividades - Ingeniería de Requisitos

Loor Medranda Marlon Taylor
Facultad de Ciencias de la Computación
Universidad Técnica Estatal de Quevedo
Quevedo, Ecuador
mloorm14@uteq.edu.ec

Resumen—Este documento consolida las actividades de investigación y modelado de requisitos desarrolladas durante el curso. Se presentan cuatro técnicas de elicitación de requisitos emergentes, adicionales a las tradicionales, enfocadas en entornos digitales y masivos: CrowdRE, Gamificación, Minería de Apps y Elicitación con IA. Posteriormente, se aplican técnicas de escenarios para describir flujos de información en el proyecto de “Sistema de Gestión Bibliotecaria”, basándose en la optimización de procesos de búsqueda y préstamo. Finalmente, se modela el caso “Autoservicio de Comida” mediante un diagrama de casos de uso y la especificación de historias de usuario, integrando componentes de IoT y facturación electrónica para mejorar la eficiencia operativa.

Index Terms—Elicitación, CrowdRE, Gamificación, Escenarios, Casos de Uso, Historias de Usuario, IoT.

I. INVESTIGACIÓN: TÉCNICAS DE ELICITACIÓN (SEMANA 11)

A continuación se detallan cuatro técnicas de elicitación de requisitos modernas, distintas a las tratadas en las diapositivas del curso, seleccionadas por su relevancia en el desarrollo de software actual (2020-2025).

I-A. Ingeniería de Requisitos Basada en la Multitud (CrowdRE)

CrowdRE es un enfoque participativo que busca obtener requisitos de una “multitud” de usuarios actuales o potenciales, en lugar de un grupo pequeño de stakeholders seleccionados. Aprovecha la ubicuidad de las plataformas digitales para recolectar feedback continuo. Estudios previos, como la revisión sistemática de Wang et al. (2019), destacan que CrowdRE es esencial para sistemas abiertos donde la heterogeneidad de usuarios es alta, permitiendo identificar necesidades ocultas mediante el análisis masivo de feedback [1].

I-B. Gamificación de la Elicitación

Esta técnica aplica mecánicas de juego (puntos, logros, avatares) en actividades de no-juego para motivar a los stakeholders a participar en el proceso de definición de requisitos.

Un avance significativo es el framework “Gamify4LexAmb” propuesto por Dar et al. (2024), el cual ha demostrado empíricamente que la gamificación no solo aumenta la participación, sino que reduce significativamente la ambigüedad léxica en los requisitos al incentivar a los usuarios a escribir descripciones más precisas para ganar recompensas [2].

I-C. Minería de Tiendas de Aplicaciones (App Store Mining)

Es una técnica automatizada que extrae requisitos analizando reseñas de usuarios y descripciones de aplicaciones similares en tiendas como Google Play. Wei et al. (2023) proponen el uso de herramientas de Procesamiento de Lenguaje Natural (NLP) para filtrar el “ruido” en los comentarios y extraer automáticamente “solicitudes de funciones” (feature requests) y reportes de fallos de la competencia. Esto permite construir un backlog inicial basado en datos reales del mercado antes de iniciar el desarrollo [3].

I-D. Elicitación Impulsada por IA (AI-Driven Elicitation)

El uso de Grandes Modelos de Lenguaje (LLMs) representa la vanguardia en elicitación. Herramientas basadas en modelos generativos pueden actuar como “analistas virtuales”. Investigaciones recientes de Arora et al. (2024) exploran la sinergia humano-IA, donde los agentes inteligentes generan borradores de historias de usuario y criterios de aceptación a partir de transcripciones de entrevistas, acelerando la fase de descubrimiento y permitiendo a los ingenieros centrarse en la validación y negociación de conflictos [4].

II. ESCENARIOS DE FLUJO DE INFORMACIÓN: PROYECTO ‘BIBLIOTECA’ (SEMANA 11)

Para el proyecto de “Sistema Integral de Gestión Bibliotecaria”, se han diseñado escenarios enfocados en resolver la ineficiencia de las búsquedas manuales y la falta de información sobre disponibilidad.

II-A. Escenario 1: Búsqueda y Localización Inteligente de Libros

Objetivo: Reducir el tiempo de búsqueda (actualmente 5-15 min) y asegurar la disponibilidad.

1. El estudiante accede a la App móvil e ingresa el título o tema (e.g., “Ingeniería de Software”).
2. El sistema consulta el catálogo en tiempo real y filtra los resultados, mostrando solo los ítems con estado “Disponible” en estantería.
3. El estudiante selecciona el libro. El sistema muestra la ubicación topográfica exacta (Bloque A, Estante 4, Nivel 2) y genera un mapa de ruta visual.
4. Al llegar a la estantería, el estudiante escanea el código QR del libro físico con la App para confirmar que es el ejemplar correcto.
5. El sistema valida la selección y habilita la opción de préstamo rápido.

II-B. Escenario 2: Préstamo Desatendido con Validación Digital

Objetivo: Automatizar el registro para eliminar errores manuales y colas.

1. El estudiante se acerca al Kiosco de Autoservicio y se identifica escaneando su credencial digital.
2. El sistema verifica que el estudiante no tenga multas pendientes ni haya excedido el límite de préstamos.
3. El estudiante coloca los libros sobre la plataforma RFID. El sistema lee las etiquetas de todos los libros simultáneamente.
4. El sistema muestra los títulos en pantalla y solicita confirmación.
5. Tras confirmar, el sistema cambia el estado de los libros a “Prestado”, desactiva la alarma de seguridad y envía el comprobante con la fecha de devolución al correo institucional del estudiante.

III. CASO DE ESTUDIO: AUTOSERVICIO DE COMIDA (SEMANA 13)

Para el sistema de autoservicio, la integración de hardware (balanzas IoT) con el software de gestión requiere especificaciones precisas. Como señalan Guerrero-Ulloa et al. (2023) en su revisión sobre metodologías para sistemas IoT, la gestión de requisitos no funcionales y la integración hardware-software son críticas para el éxito de estos proyectos [5]. Además, el uso de especificaciones detalladas facilita el desarrollo guiado por pruebas en entornos IoT [6].

III-A. Diagrama de Casos de Uso

A continuación se describen los actores y casos de uso principales, incluyendo la gestión de inventario (entradas y salidas) y la interacción con dispositivos IoT.

Actores:

- **Cliente:** Actor principal que se sirve y paga.
- **Cajero:** Apoya en el cobro y supervisión.
- **Administrador:** Gestiona el negocio e inventario.

- **Báscula IoT (Sistema):** Dispositivo inteligente que envía el peso.

Casos de Uso Principales:

- **Gestionar Identificación:** Registrar o identificar cliente (QR) para fidelización.
- **Pesar Alimentos (IoT):** Captura automática del peso y cálculo del costo.
- **Procesar Pago:** Cobro multicanal. Incluye “Emitir Factura” y “Descontar Stock (Salida)”.
- **Registrar Reabastecimiento:** El Administrador registra la entrada de nuevos insumos a la cocina (Entradas).
- **Generar Reportes:** Visualización de ventas y horas pico.

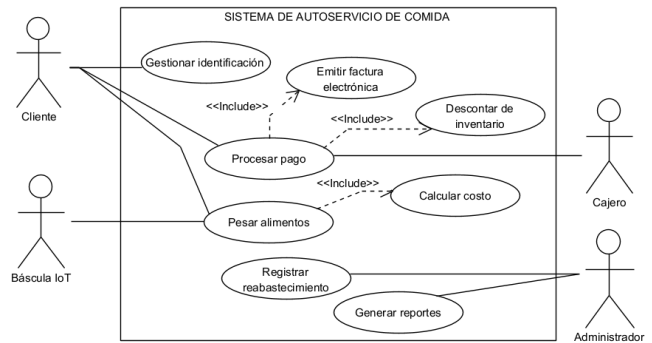


Figura 1. Diagrama de Casos de Uso del Sistema de Autoservicio.

III-B. Historias de Usuario

Se detallan 5 historias de usuario bajo el formato estándar.

HU-01: Identificación de Cliente Frecuente

Como cliente frecuente, **quiero** escanear mi código QR al inicio de la línea, **para** que el sistema me reconozca y aplique mis descuentos automáticamente.

- **Criterio:** El reconocimiento debe tardar menos de 2 segundos.

HU-02: Cálculo Automático (IoT)

Como cliente, **quiero** ver el precio de mi plato en la pantalla de la báscula mientras me sirvo, **para** controlar mi gasto antes de llegar a caja.

- **Criterio:** El precio se actualiza en tiempo real ($Peso \times PrecioUnitario$).

HU-03: Pago y Facturación Rápida

Como cajero, **quiero** generar la factura electrónica con un solo toque, **para** agilizar la fila en horas pico.

- **Criterio:** La factura debe llegar al correo del cliente en < 30 segundos.

HU-04: Control de Stock en Tiempo Real

Como administrador, **quiero** que el sistema descuente automáticamente los gramos vendidos del inventario, **para** saber cuándo reabastecer sin contar manualmente.

- **Criterio:** Se debe generar una alerta si un insumo baja del 10%.

HU-05: Analítica de Ventas

Como dueño, **quiero** ver un reporte de platos más vendidos y horas pico, **para** optimizar la producción de comida.

- *Criterio:* El reporte debe incluir gráficos de calor por horarios.

IV. CONCLUSIÓN

La investigación evidencia que técnicas modernas como CrowdRE y la gamificación son vitales para sistemas con gran volumen de usuarios, mientras que la especificación rigurosa es indispensable en proyectos con componentes de hardware (IoT), tal como se sustenta en la literatura reciente de Guerrero-Ulloa et al. [5], [6]. La combinación de estas estrategias permite desarrollar soluciones como el Autoservicio Inteligente con altos estándares de calidad.

REFERENCIAS

- [1] C. Wang, M. Daneva, M. van Sinderen, and P. Liang, "A Systematic Mapping Study on Crowdsourced Requirements Engineering using User Feedback," *Journal of Software: Evolution and Process*, vol. 31, no. 10, p. e2199, Oct. 2019. DOI: 10.1002/smr.2199.
- [2] H. Dar, R. Aziz, J. A. Khan, M. I. Lali, and N. A. Almujaally, "Gamify4LexAmb: a gamification-based approach to address lexical ambiguity in natural language requirements," *PeerJ Computer Science*, vol. 10, p. e2229, Sep. 2024. DOI: 10.7717/peerj-cs.2229.
- [3] J. Wei, A.-L. Courbis, T. Lambolais, and P. L. Bernard, "Enhancing Requirements Elicitation through App Stores Mining: Health Monitoring App Case Study," in *2023 IEEE 31st International Requirements Engineering Conference (RE)*, Hannover, Germany, 2023, pp. 396–400. DOI: 10.1109/RE57278.2023.00062.
- [4] C. Arora, J. Grundy, and M. Abdelrazek, "Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs," in *Generative AI for Effective Software Development*, A. Nguyen-Duc, P. Abrahamsson, and F. Khomh, Eds. Cham, Switzerland: Springer, 2024, pp. 129–148. DOI: 10.1007/978-3-031-55642-5_6.
- [5] G. Guerrero-Ulloa, C. Rodríguez-Domínguez, and M. J. Hornos, "Agile Methodologies Applied to the Development of Internet of Things (IoT)-Based Systems: A Review," *Sensors*, vol. 23, no. 2, p. 790, Jan. 2023. DOI: 10.3390/s23020790.
- [6] G. Guerrero-Ulloa, D. Carvajal-Suárez, P. Novais, M. J. Hornos, and C. Rodríguez-Domínguez, "Test-Driven Development Tool for IoT Systems," *IEEE Software*, vol. 42, no. 4, pp. 58–67, Jul. 2025 (Early Access 2024). DOI: 10.1109/MS.2024.3479880.