

COMPETENCIA A EVALUAR: **DEFINICIÓN DE REQUERIMIENTOS DE SOFTWARE**

Aksel Deneken Maldonado A01711966

Resumen integrador de los conceptos atendidos durante el curso.

La competencia de requisitos de software se aplicó desde el inicio del proyecto, comenzando con la identificación y redacción de los requisitos funcionales y no funcionales. Aquí seguimos la teoría vista en clase, se estructuraron utilizando los elementos clave: nombre del requisito, descripción clara, actor principal, prioridad y dependencias.

Desde la perspectiva teórica, los requisitos funcionales se enfocan en describir qué debe hacer el sistema, es decir, las acciones que el usuario podrá realizar. En cambio, los requisitos no funcionales abordan cómo debe comportarse el sistema, incluyendo los aspectos de rendimiento, usabilidad y seguridad. En el proyecto se integraron ambos tipos para lograr una solución técnica completa, usable y adaptada al contexto del socio.

Además, los casos de prueba fueron implementados por mí, tomando como base la metodología de casos por escenarios explicada en clase y el PDF proporcionado por la profesora. Esta metodología parte de la definición del flujo básico de una funcionalidad, seguido por los flujos alternos posibles. A partir de ahí, se construyeron tablas que detallan cada escenario con sus condiciones iniciales y los resultados esperados. Esta estructura garantiza que los requisitos definidos sean verificables, medibles y correctamente implementados dentro del sistema.

A continuación se incluyen imágenes con ejemplos reales utilizados durante el proyecto, que evidencian la aplicación directa de esta competencia ya que no tomo apuntes:

Caso de prueba:

7. Visualización de Inventario por Cliente.

Flujo Básico:

1. El usuario con rol cliente inicia sesión en el sistema.
2. Accede a la sección de "Inventario".
3. El sistema consulta la base de datos y filtra los productos correspondientes a ese cliente.
4. Se muestra una tabla con el nombre del producto, precio, cantidad disponible, descripción y cantidad vendida.

Flujos Alternos:

A1. Cliente sin productos asociados

- En el paso 3, si el cliente no tiene productos registrados, el sistema muestra un mensaje: "No se ha cargado inventario para tu cuenta".

A2. Falla al consultar la base de datos

- En el paso 3, si ocurre un error al intentar acceder a los datos del inventario, se muestra un mensaje de error general: "Hubo un problema al cargar el inventario".

Escenario	Flujo Base	Alternos
Escenario 1- Cliente visualiza su inventario correctamente	Básico	-
Escenario 2- Cliente sin productos asociados	Básico	A1
Escenario 3- Falla al consultar la base de datos	Básico	A2

Test ID	Escenario	Rol de Cliente	Inventario asociado	Resultado esperado
TC24	Visualización exitosa	V	V	Personal asignado correctamente y mensaje de confirmación mostrado
TC25	Cliente sin productos asociados	V	I	Mensaje: "No se ha cargado inventario para tu cuenta"
TC26	Error al consultar base de datos	V	V	Mensaje: "Hubo un problema al cargar el inventario"

Requisitos Funcionales:

ID	Nombre del Requisito	Descripción	Actor Principal	Prioridad	Dependencias
RF01	Inicio de sesión y recuperación	Permite que los usuarios ingresen al sistema de forma segura con usuario y contraseña. Incluye la opción de recuperar la contraseña en caso de olvido. El acceso se limita según el rol asignado.	Usuario	Alta	Ninguna
RF02	Gestión de usuarios	Permite crear, editar y eliminar cuentas de acceso al sistema. Se pueden asignar diferentes roles (administrador, staff o cliente) con distintos niveles de permisos y visibilidad dentro de la plataforma.	Administrador	Alta	Ninguna
RF03	Gestión de clientes	Permite registrar y actualizar la información de las empresas que contratan los servicios. Se incluye el nombre de la empresa, datos de contacto, persona responsable y cualquier información adicional relevante para su seguimiento.	Staff / Administrador	Alta	RF02

Repositorio:

<https://github.com/saucedor/OBD/blob/main>

Identificación de requisitos de software:

Para definir los requisitos del sistema, usamos lo que vimos en clase: entender bien qué necesitaba el cliente, quiénes iban a usar la plataforma y qué tenía que poder hacer cada uno. Lo primero fue hablar con el socio para aterrizar sus ideas, porque al inicio ni ellos tenían tan claro todo lo que querían. Con base en esas conversaciones, fuimos armando una lista de requisitos funcionales y no funcionales, priorizando lo más importante para ellos.

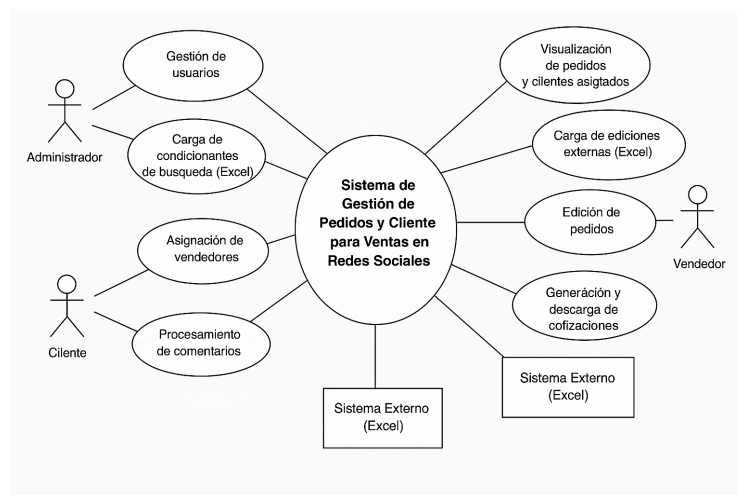
Los requisitos funcionales los hicimos en formato tabla con su ID, nombre, descripción, actor principal, prioridad y dependencias. Esto lo hicimos porque así se entiende fácil qué hace cada función, quién la usa y qué tan importante es. Los no funcionales salieron más de lo que sabíamos que tenía que tener la app para

funcionar bien: que fuera clara, segura, fácil de usar, que cada quien solo viera lo que le toca, etc.

Algo importante es que estos requisitos no salieron en una sola versión. A lo largo del proyecto fuimos ajustando varias cosas después de hablar con el socio, porque ellos también iban entendiendo mejor lo que necesitaban. Por eso estos requisitos tuvieron varias versiones, y probablemente se sigan ajustando conforme se use más la plataforma. En la Evidencia 5 se puede ver la tabla más reciente que armamos con todos los cambios integrados.

Detalle de casos de uso con diagramas

Los casos de uso se definieron en equipo, con base en los requisitos funcionales que ya habíamos trabajado. Aunque esta parte no la desarrollé directamente yo, sí fue una decisión conjunta. Todos opinamos sobre qué debía entrar, cómo se iba a organizar y qué actores debían interactuar con cada funcionalidad. A partir de eso se armó el diagrama final que se incluye como evidencia. Este diagrama representa cómo el usuario interactúa con el sistema desde su rol y qué funcionalidades tiene disponibles.



Documento con la identificación de todos los tipos de requisitos de software

Aquí se puede ver el documento más reciente con nuestros requisitos funcionales:
[OBD/Documentación Avance 4 OBD.pdf at main · saucedor/OBD](#)

Documento con la versión más reciente de los casos de uso/casos de prueba:
<https://github.com/saucedor/OBD/blob/main/Avance%205/Avance%205.pdf>

Plan de comunicación

Desde el inicio del proyecto sí se estableció un plan de comunicación, usando principalmente WhatsApp para coordinarnos rápido y Discord cuando necesitábamos hablar más a fondo o compartir pantalla. Aunque sí lo teníamos claro desde el principio, la verdad es que casi toda la comunicación terminó siendo presencial, porque la mayoría del tiempo estábamos juntos en el Tec, especialmente en los salones vacíos de EGADE. Ahí hablábamos de todo, tomábamos decisiones y resolvíamos dudas al momento, así que ese espacio se volvió nuestro punto central de trabajo.