



EJERCICIOS

Análisis de Requisitos

Ejercicio 1

A continuación, se presentan una serie de requisitos para un sistema de reservas de hoteles. Para cada requisito, clasifícalo según su tipo y explica brevemente por qué es de ese tipo.

1. El sistema debe permitir a los usuarios registrarse y crear una cuenta proporcionando una dirección de correo electrónico y una contraseña.
2. El tiempo de respuesta del sistema para cargar la lista de hoteles disponibles no debe exceder los 2 segundos en horas pico.
3. La interfaz de usuario debe ser intuitiva y fácil de usar, accesible desde dispositivos móviles y navegadores web modernos.
4. Solo los usuarios con privilegios de administrador podrán agregar o eliminar hoteles del sistema.
5. Los usuarios que no tengan una cuenta creada deben poder utilizar el sistema para buscar hoteles, pero necesitarán registrarse para completar una reserva.
6. El sistema debe estar disponible el 99.99% del tiempo durante el año.
7. Los usuarios deben poder filtrar hoteles por ubicación, precio, clasificación por estrellas y comentarios de otros clientes.
8. El sistema debe cumplir con las regulaciones de protección de datos personales (GDPR) y garantizar que la información de los usuarios esté encriptada.
9. El proceso de reserva debe permitir al usuario seleccionar fechas de entrada y salida, el número de personas y las preferencias de habitación.
10. El sistema debe mostrar en la página principal un formulario de búsqueda donde se pueda seleccionar las fechas, el número de huéspedes y el tipo de habitación. El formulario debe incluir un botón de “Buscar”.
11. El sistema debe cumplir con las normativas internacionales de accesibilidad web (WCAG 2.1) para garantizar que usuarios con discapacidades puedan utilizarlo sin restricciones.
12. El sistema debe ser capaz de soportar al menos 10.000 usuarios concurrentes sin degradación del rendimiento.
13. Los usuarios desean poder ver el historial de sus reservas pasadas y futuras desde su perfil de cuenta.
14. Despues de realizar una reserva, el sistema debe mostrar una página de confirmación en la cual aparezca la opción de descargar o imprimir la reserva.
15. El sistema debe ser desarrollado utilizando metodologías ágiles, con iteraciones de 2 semanas, siguiendo las políticas internas del departamento de desarrollo.



16. Los usuarios deben poder cancelar una reserva hasta 24 horas antes de la fecha de entrada sin penalización.
17. El desarrollo del sistema debe completarse en un plazo de 12 meses, según los plazos establecidos por la dirección del proyecto.
18. El sistema debe realizar copias de seguridad automáticas de la base de datos cada 24 horas.
19. El sistema debe generar un informe mensual que muestre las estadísticas de ocupación, reservas canceladas y pagos recibidos, accesible solo para los gerentes del hotel.
20. Los usuarios deben recibir un correo de confirmación inmediatamente después de completar una reserva con todos los detalles de la misma, como fechas, precio y hotel.

Ejercicio 2

Una cadena de gimnasios llamada GO fit quiere desarrollar una plataforma de reservas en línea que permita a sus clientes reservar acceso a diferentes servicios del gimnasio. GO fit ofrece tanto acceso libre a las instalaciones (gimnasio, piscina, etc.) como clases de fitness programadas (yoga, spinning, pilates, etc.). La cadena tiene varios gimnasios en diferentes ciudades y cuenta con miles de clientes.

El sistema debe permitir a los clientes gestionar sus membresías, consultar horarios de clases, reservar plazas en clases y registrar su asistencia al gimnasio. Además, los administradores de cada gimnasio deben poder gestionar los horarios, las reservas, las listas de espera y visualizar el número de asistentes por clase.

Detalles adicionales:

- **Tipos de usuarios:**
 - **Clientes:** Los clientes deben poder registrarse en el sistema, ver el catálogo de clases y servicios, realizar reservas y cancelarlas si lo necesitan. Además, deben recibir notificaciones sobre cambios en las clases o recordatorios de sus reservas.
 - **Entrenadores:** Los entrenadores podrán ver las listas de asistentes a sus clases, registrar la asistencia y subir contenido exclusivo (vídeos de ejercicios o planes de entrenamiento) para los clientes.
 - **Administradores de gimnasio:** Los administradores tendrán acceso para gestionar los horarios de las clases, crear nuevas clases, limitar el número de asistentes y visualizar informes sobre el uso de las instalaciones.
- El sistema debe permitir que los clientes accedan desde cualquier dispositivo (móvil, tableta o computadora) y que sea fácil de usar.
- **Pagos y membresías:** Los clientes deben poder gestionar sus pagos y suscribirse a diferentes tipos de membresía (mensual, anual, clases ilimitadas, etc.).



- **Capacidad del gimnasio:** El sistema debe poder controlar la capacidad máxima de asistentes permitida para cumplir con las normativas locales de aforo.
- **Listas de espera:** Si una clase está llena, el sistema debe permitir a los clientes añadirse a una lista de espera y notificarles si se libera una plaza.
- **Multisede:** Los clientes con una membresía deben poder reservar en cualquier gimnasio de la cadena, pero algunas clases específicas solo estarán disponibles en ciertos gimnasios.
- **Estadísticas:** Los administradores deben poder ver estadísticas de uso, como el número de reservas realizadas por día, la asistencia promedio y la demanda de clases.
- **Cumplimiento legal:** El sistema debe cumplir con la normativa vigente de protección de datos personales y normativas locales de seguridad y salud.

Lee detenidamente el escenario y analiza las necesidades del sistema para los diferentes tipos de usuarios y las funcionalidades requeridas. A continuación, elabora una lista de requisitos que consideres necesarios para desarrollar el sistema software. Además, para cada requisito, indica qué tipo de requisito es y explica brevemente por qué lo consideras de ese tipo.

Ejercicio 3

Imagina que tu equipo de desarrollo ha sido contratado por una empresa de venta de entradas en línea llamada Enterticket. La empresa tiene un sitio web que permite a los usuarios comprar entradas para eventos como conciertos, deportes, teatro, etc. Actualmente, Enterticket quiere expandir su plataforma y mejorar la experiencia de los usuarios, así que ha pedido que se desarrollen nuevas funcionalidades para su sistema.

Enterticket ya tiene un sistema básico de compra de entradas, pero están interesados en implementar las siguientes mejoras:

1. Los usuarios podrán recibir notificaciones sobre eventos cercanos a su ubicación o de sus artistas favoritos.
2. Habrá un sistema de reseñas para que los usuarios puedan valorar y comentar sobre los eventos a los que han asistido.
3. Los usuarios podrán filtrar los eventos según su ubicación, fecha y tipo (conciertos, teatro, deportes, etc.).
4. Los usuarios podrán comprar entradas para varios eventos en una sola transacción.
5. Habrá una sección de recomendaciones personalizada basada en los eventos que los usuarios hayan comprado anteriormente.
6. El administrador del sistema podrá gestionar (agregar, editar, eliminar) los eventos disponibles en la plataforma.



A partir de las mejoras solicitadas, identifica al menos **cinco historias de usuario**. En la historia de usuario se deben incluir al menos los siguientes campos:

- ID
- Título
- Descripción, siguiendo el patrón.
- Criterios de validación, los cuales deben describir cómo se verificará que la funcionalidad cumple con los requisitos.
- Prioridad
- Estimación

Ejercicio 4

En un equipo de desarrollo que trabaja en un sistema de gestión de bibliotecas se quiere obtener una relación entre los objetivos y los requisitos del proyecto. Los objetivos del sistema son:

- **Facilitar la gestión de libros:** Permitir la adición, modificación y eliminación de libros en el catálogo de la biblioteca.
- **Gestión de usuarios:** Permitir el registro, actualización y eliminación de usuarios de la biblioteca.
- **Gestión de préstamos:** Permitir el préstamo y la devolución de libros de manera.
- **Generación de informes:** Proporcionar informes sobre los libros más prestados y el estado de los préstamos.

Mientras que los requisitos establecidos por la empresa y el cliente son los siguientes:

- **RF-01:** El sistema deberá permitir a los bibliotecarios agregar nuevos libros al catálogo.
- **RF-02:** El sistema deberá permitir la modificación de la información de los libros existentes.
- **RF-03:** El sistema deberá permitir la eliminación de libros del catálogo.
- **RF-04:** El sistema deberá permitir a los usuarios registrarse en la biblioteca.
- **RF-05:** El sistema deberá permitir a los bibliotecarios gestionar la información de los usuarios (actualizar y eliminar).
- **RF-06:** El sistema deberá permitir el préstamo de libros a los usuarios registrados.
- **RF-07:** El sistema deberá permitir la devolución de libros.
- **RF-08:** El sistema deberá generar reportes de libros más prestados.
- **RNF-01:** El sistema deberá ser accesible desde dispositivos móviles.
- **RNF-02:** El tiempo de respuesta del sistema deberá ser inferior a 2 segundos.
- **RNF-03:** El sistema deberá tener una disponibilidad del 99.9%.

Con base en los objetivos y requisitos proporcionados, elabora una matriz de trazabilidad que vincule los requisitos funcionales y no funcionales con los objetivos del sistema.

Requisitos:

Requisito funcional

Requisito no funcional

- Producto

- Organizacional

- Externas

Requisito de Usuario

Requisito de Interfaz

1. 1. Requisito funcional

2. Requisito no funcional → Producto
3. Requisito no funcional → Producto
4. Requisito funcional
5. Requisito de Usuario
6. Requisito no funcional → Producto
7. Requisito funcional
8. Requisito no funcional → Externas
9. Requisito funcional
10. Requisito de Interfaz
11. Requisito no funcional → Externas
12. Requisito no funcional → Producto
13. Requisito de Usuario
14. Requisito de Interfaz
15. Requisito no funcional → Organizacional
16. Requisito funcional
17. Requisito no funcional → Organizacional
18. Requisito no funcional → Producto
19. Requisito funcional
20. Requisito funcional

2. Requisito funcional

- Consultar horarios de clase
- Sistema permita crear usuarios
- Sistema permita guardar información bancaria de los dientes
- Solo los usuarios registrados pueden dar clase

Requisito no funcional

- Producto
 - Que pueda manejar 500 dientes a la vez
 - Que no se colapse
- Organizacional
 - Se desarrolle cumpliendo una metodología específica
- Ej: Android → Android Studio
- Externo
 - Debe cumplir la normativa de privacidad de datos
 - Debe cumplir la normativa de seguridad
 - Debe cumplir la normativa de salud

Requisito de Usuario

- Los usuarios necesitan ver el historial de acceso

Requisito de Interfaz

- Debe de mantener todas las características disponibles en un formulario
- La forma de ver los horarios de clases se muestran en calendario y se puedan reservar

3. 1.

- **ID:** Historia de usuario
- **Título:** Recibir notificaciones
- **Descripción:** Como usuario **quiero** recibir notificaciones sobre eventos cercanas a mi ubicación **para** estar al tanto de los eventos.
- **Criterios de validación:**
 - Comprobar que el sistema envía notificaciones
 - Verificar que el usuario recibe la notificación
- **Prioridad:** 1
- **Estimación:** 20

4.

	OBJ-1	OBJ-2	OBJ-3	OBJ-4
RF-1	X			
RF-2	X			
RF-3	X			
RF-4		X		
RF-5		X		
RF-6			X	
RF-7			X	
RF-8				X
RNF-1	X	X	X	
RNF-2	X	X	X	
RNF-3	X	X	X	