

# Requerimientos no funcionales

## Ejercicio 1

Dada la siguiente lista de requisitos;

1. El software debe permitir la creación de perfiles de usuario.
2. El software debe permitir la reserva de citas en el taller.
3. El software debe permitir la consulta de información del vehículo.
4. El software debe enviar notificaciones de recordatorios de servicio.
5. El software debe generar informes de rendimiento del taller.
6. El software debe tener una disponibilidad del 99.9%.
7. El software debe ser fácil de actualizar sin causar interrupciones en el servicio.
8. El software debe cumplir con los estándares de accesibilidad.
9. El software debe tener una interfaz coherente y consistente.
10. El software debe tener un tiempo de carga rápido.
11. El software debe contar con un soporte al usuario eficiente.
12. El software debe contar con una arquitectura escalable.
13. El software debe ser fácil de integrar con otros sistemas.
14. El software debe contar con medidas de autenticación de usuarios seguras.
15. El software debe cumplir con las leyes de protección de datos y privacidad.
16. El software debe garantizar la confidencialidad de la información de los usuarios.
17. El software debe ser fácil de desinstalar en caso de ser necesario.
18. El software debe contar con un sistema de respaldo seguro y confiable.
19. El software debe cumplir con las leyes de protección de propiedad intelectual.

## Parte 1

Clasifíquelos en funcionales y no funcionales

Numero requisito	Descripción	Clasificación requisito
1	El software debe permitir la creación de perfiles de usuario.	FUNCIONAL
2	El software debe permitir la reserva de citas en el taller.	FUNCIONAL
3	El software debe permitir la consulta de información del vehículo.	FUNCIONAL

4	El software debe enviar notificaciones de recordatorios de servicio.	FUNCIONAL
5	El software debe generar informes de rendimiento del taller.	FUNCIONAL
6	El software debe tener una disponibilidad del 99.9%. Rendimiento/Performance	NO FUNCIONAL
7	El software debe ser fácil de actualizar sin causar interrupciones en el servicio.	NO FUNCIONAL
8	El software debe cumplir con los estándares de accesibilidad.	NO FUNCIONAL
9	El software debe tener una interfaz coherente y consistente.	NO FUNCIONAL
10	El software debe tener un tiempo de carga rápido.	NO FUNCIONAL
11	El software debe contar con un soporte al usuario eficiente.	NO FUNCIONAL
13	El software debe contar con una arquitectura escalable.	NO FUNCIONAL
14	El software debe ser fácil de integrar con otros sistemas.	NO FUNCIONAL
15	El software debe contar con medidas de autenticación de usuarios seguras.	NO FUNCIONAL
16	El software debe cumplir con las leyes de protección de datos y privacidad.	NO FUNCIONAL
17	El software debe garantizar la confidencialidad de la información de los usuarios.	NO FUNCIONAL
18	El software debe ser fácil de desinstalar en caso de ser necesario.	NO FUNCIONAL
19	El software debe contar con un sistema de respaldo seguro y confiable.	NO FUNCIONAL
20	El software debe cumplir con las leyes de protección de propiedad intelectual.	NO FUNCIONAL

## Parte 2

Defina un criterio de aceptación.

En caso de que sea un requisito funcional defina un caso de prueba según la plantilla especificada en: <https://www.geeksforgeeks.org/software-testing-test-case/?ref=gcse>

Para el caso de los requisitos no funcionales:

Numero requisito	Descripción	Tipo
6	El software debe tener una disponibilidad del 99.9%.	Rendimiento/Performance/Disponibilidad
Criterio: El software no debe tener una disponibilidad menor al 99.9%.		
7	El software debe ser fácil de actualizar sin causar interrupciones en el servicio.	Mantenibilidad/Soporte
Criterio: El software debe descargar actualizaciones en segundo plano y actualizar los cambios en tiempo real.		
8	El software debe cumplir con los estándares de accesibilidad.	Estándares
Criterio: El producto debe cumplir con el estándar interno de la empresa.		
9	El software debe tener una interfaz coherente y consistente.	UI test
Criterio: El producto tendrá los colores corporativos. El producto debe ser certificado según los estándares de marca corporativa por el departamento de comunicaciones.		
10	El software debe tener un tiempo de carga rápido.	Rendimiento/Performance
Criterio: El software tendrá un tiempo de carga menor a 3 segundos.		
11	El software debe contar con un soporte al usuario eficiente.	Mantenibilidad/Soporte
Criterio: El soporte deberá funcionar 24/7.		
13	El software debe contar con una arquitectura escalable.	Mantenibilidad/Soporte
Criterio: Si durante pruebas de carga simulando un aumento progresivo en el número de usuarios o la cantidad de datos procesados, se observa que el rendimiento del sistema se mantiene dentro de límites aceptables.		

14	El software debe ser fácil de integrar con otros sistemas.	Mantenibilidad/Soporte
Criterio: Si el software cumple con los estándares y normativas reconocidas por la industria, lo que hace mucho más fácil la integración del mismo con otros sistemas que también cumplen con estos estándares.		
15	El software debe contar con medidas de autenticación de usuarios seguras.	Seguridad
Criterio: El 100% de las veces solo los usuarios autorizados deberán acceder al software.		
16	El software debe cumplir con las leyes de protección de datos y privacidad.	Legal
Criterio: El software debe ser aprobado por el departamento legal.		
17	El software debe garantizar la confidencialidad de la información de los usuarios.	Seguridad
Criterio: El software deberá cumplir con la norma ISO 27001.		
18	El software debe ser fácil de desinstalar en caso de ser necesario.	UX test
Criterio: El software debe contar con un acceso rápido que permita desinstalar la aplicación en una acción.		
19	El software debe contar con un sistema de respaldo seguro y confiable.	Mantenibilidad/Soporte/Seguridad
Criterio: Respaldos periódicos generados a diario encriptados y con verificación de integridad.		
20	El software debe cumplir con las leyes de protección de propiedad intelectual.	Legal
Criterio: El software debe ser aprobado por el departamento legal.		

R1. El software debe permitir la creación de perfiles de usuario.

Module name	Perfiles de usuario
Test case ID	TC_01
Tester name	Ignacio
Test case description	Chequear la funcionalidad de creación de perfiles de usuario.
Prerequisites	Tener conexión a internet

	Tener un navegador instalado
Tester's name	Bruno, Gonzalo y Santiago
Environmental Information	OS: Windows Sistema: Laptop
Test Scenario	Chequear que luego de crear un perfil de usuario, el usuario pueda ver los perfiles.

Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
1	1. Ingresar el usuario 2. Ingresar la contraseña 3. Seleccionar el tipo de perfil 4. Clickear en crear perfil	usuario: <a href="#">ignacionosabe@queponer.com</a> contraseña: hola123	Nuevo perfil creado con éxito.	Nuevo perfil creado con éxito.	Pass	Sin comentarios

R2. El software debe permitir la reserva de citas en el taller.

Module name	Reserva cita taller
Test case ID	TC_02
Tester name	Joaquin
Test case description	Chequear la funcionalidad de reserva de citas en el taller.
Prerequisites	Tener un teléfono con internet. Tener un navegador instalado. Tener una cuenta en la aplicación, con todos los datos. Estar logueado.
Tester's name	Jose, Gonzalo y Joaquin
Environmental Information	OS: Android Sistema: Celular
Test Scenario	Chequear que luego de reservar, se registre la cita en taller.

Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
--------------	------------	------------	------------------	----------------	--------	----------

2	1. Ingresar el horario y fecha disponible. 2. Tapear en solicitar reserva	horario: 20:10 fecha: 30/12/2024	Cita agendada con éxito.	Cita agendada con éxito.	Pass	Sin comentarios
---	--	---	--------------------------	--------------------------	------	-----------------

R3. El software debe permitir la consulta de información del vehículo.

Module name	Información de vehículo
Test case ID	TC_03
Tester name	Bruno
Test case description	Chequear la funcionalidad de consultar la información del vehículo.
Prerequisites	Tener conexión a internet. Tener un navegador instalado.
Tester's name	Jose, Gonzalo y Santiago
Environmental Information	OS: Linux Sistema: Laptop
Test Scenario	Chequear que luego de seleccionar un vehículo se despliegue la información de este.

Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
3	1. Clickear un vehículo.	Ninguno	Nombre vehículo: ... Año vehículo: ... Marca: ... Modelo: ...	Nombre vehículo: ... Año vehículo: ... Marca: ... Modelo: ...	Pass	Selecciona cualquier vehículo aleatoriamente.

R4. El software debe enviar notificaciones de recordatorios de servicio.

Module name	Recordatorio
Test case ID	TC_04

Tester name	Ignacio
Test case description	Chequear la funcionalidad de envío de recordatorios de servicio.
Prerequisites	Tener conexión a internet Tener acceso de admin Loguearse en la aplicación
Tester's name	Bruno, Gonzalo y Santiago
Environmental Information	OS: Mac Sistema: Laptop
Test Scenario	Chequear que cuando el servicio manda recordatorios, les llegue a los usuarios.

Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
4	1. Escribir un mensaje. 2. Seleccionar un usuario 3. Clickear en mandar una notificación.	usuario: usuario_prueba_1@gmail.com	Notificación enviada.	Fallo al enviar notificación.	Error	El usuario tiene desactivada las notificaciones

R5. El software debe generar informes de rendimiento del taller.

Module name	Informes de rendimiento del taller
Test case ID	TC_05
Tester name	Ignacio
Test case description	Chequear la funcionalidad de generar informes de rendimiento del taller.
Prerequisites	Tener un teléfono con internet. Tener un navegador instalado. Tener acceso de admin Loguearse en la aplicación Tener datos precargados de la empresa.
Tester's name	Jose, Gonzalo y Joaquin
Environmental Information	OS: Android Sistema: Celular

Test Scenario	Chequear que se genere un informe de taller con todos los datos estipulados en la sección S21E del diccionario empresarial.
---------------	---

Test Case ID	Test Steps	Test Input	Expected Results	Actual Results	Status	Comments
5	1. Tapear en generar informe	Ninguno	Taller: ... Rendimiento promedio ... Inicio periodo: ... Fin periodo: ...	Taller: ... Rendimiento promedio ... Inicio periodo: ... Fin periodo: ...	Pass	Sin comentarios

### Parte 3

Identifique, en el caso que corresponda, el tipo de NFR

6	El software debe tener una disponibilidad del 99.9%.	RENDIMIENTO
7	El software debe ser fácil de actualizar sin causar interrupciones en el servicio.	MANTENIMIENTO
8	El software debe cumplir con los estándares de accesibilidad.	USABILIDAD
9	El software debe tener una interfaz coherente y consistente.	APARIENCIA
10	El software debe tener un tiempo de carga rápido.	RENDIMIENTO
11	El software debe contar con un soporte al usuario eficiente.	MANTENIMIENTO
13	El software debe contar con una arquitectura escalable.	MANTENIMIENTO
14	El software debe ser fácil de integrar con otros sistemas.	OPERACIONAL
15	El software debe contar con medidas de autenticación de usuarios seguras.	SEGURIDAD



16	El software debe cumplir con las leyes de protección de datos y privacidad.	LEGAL
17	El software debe garantizar la confidencialidad de la información de los usuarios.	SEGURIDAD
18	El software debe ser fácil de desinstalar en caso de ser necesario.	USABILIDAD
19	El software debe contar con un sistema de respaldo seguro y confiable.	SEGURIDAD
20	El software debe cumplir con las leyes de protección de propiedad intelectual.	LEGAL

## Ejercicio 2

Elija 4 requisitos del problema anterior y extiéndalos utilizando los conceptos de requirements creep y gold plating.

### Requerimiento 6

El software debe tener una disponibilidad del 100% y la máxima capacidad de rendimiento.

En este caso sería gold plating si se invierte excesivo tiempo y recursos en optimizar el rendimiento del software más allá de lo que es realmente necesario para cumplir con el requisito, además en una aplicación tener el rendimiento al 100% es imposible, siempre se puede tender a acercarse a este valor, pero nunca se llega.

### Requerimiento 10

El software debe tener un tiempo de carga de 0.01s.

En este caso podría ser gold plating si se dedica tiempo y esfuerzo a optimizar el tiempo de carga más allá de lo necesario para cumplir con el requisito. Podría ser innecesario optimizarlo tanto, las personas no lo notaran.

### Requerimiento 11

El software debe contar con un soporte al usuario con disposición 24/7 y un tiempo de respuesta inmediato, la cual incluya posibles soluciones al problema.

Este es requirements creep, ya que amplía las funcionalidades descritas en el requisito inicial. Además, las especificaciones de este nuevo, implican un aumento significativo de la carga de trabajo del equipo de soporte. Llevando esto a la implementación de un equipo y un sistema más complejo.

## Requerimiento 15

El software debe contar con medidas de autenticación, donde el token de seguridad expire cada 5 minutos.

Se podría caer en gold plating para este caso ya que sería un esfuerzo demasiado alto en mantener la seguridad del producto, en el que posiblemente por este requisito se termine descargando o actualizando.