

# Requerimientos Funcionales

1. El sistema debe mostrar en la página de inicio las opciones “Iniciar sesión” y “Registrarse” para todos los visitantes.
2. El sistema debe mostrar un formulario de registro accesible desde la página principal.
3. El sistema debe solicitar nombre y apellido como campos obligatorios.
4. El sistema debe solicitar un correo electrónico válido.
5. El sistema debe validar que el correo no esté previamente registrado.
6. El sistema debe solicitar una contraseña segura.
7. El sistema debe cifrar la contraseña antes de almacenarla.
8. El sistema debe almacenar los datos del usuario en la base de datos.
9. El sistema debe permitir que cualquier visitante se registre diligenciando un formulario con nombre, apellido, correo electrónico y contraseña.
10. El sistema debe mostrar un botón de “Registrarse” visible en la página principal.
11. El sistema debe redirigir al formulario de registro al seleccionar la opción “Registrarse”.
12. El sistema debe mostrar el formulario de registro con los campos solicitados.
13. El sistema debe marcar los campos obligatorios visualmente.
14. El sistema debe solicitar el nombre como campo obligatorio.
15. El sistema debe solicitar el apellido como campo obligatorio.
16. El sistema debe solicitar un correo electrónico como campo obligatorio.
17. El sistema debe validar el formato del correo electrónico.
18. El sistema debe verificar que el correo no exista previamente en la base de datos.
19. El sistema debe mostrar un mensaje si el correo ya está registrado.
20. El sistema debe solicitar una contraseña como campo obligatorio.
21. El sistema debe validar una longitud mínima de contraseña.

22. El sistema debe validar que la contraseña cumpla criterios de seguridad.
23. El sistema debe permitir mostrar u ocultar la contraseña ingresada.
24. El sistema debe validar que todos los campos estén completos antes de enviar el formulario.
25. El sistema debe impedir el envío del formulario con datos inválidos.
26. El sistema debe cifrar la contraseña antes de almacenarla.
27. El sistema debe almacenar los datos del usuario en la base de datos.
28. El sistema debe registrar la fecha y hora del registro.
29. El sistema debe asignar un estado inicial de “pendiente de verificación” al usuario.
30. El sistema debe validar que todos los campos obligatorios del formulario de registro estén completos antes de permitir enviar la información.
31. El sistema debe enviar un código de verificación al correo electrónico registrado por el usuario al finalizar el formulario de registro.
32. El sistema debe generar un código único de verificación.
33. El sistema debe asociar el código al usuario registrado.
34. El sistema debe definir un tiempo de expiración del código.
35. El sistema debe enviar el código al correo proporcionado.
36. El sistema debe mostrar un mensaje indicando que el correo fue enviado.
37. El sistema debe permitir reenviar el código de verificación.
38. El sistema debe limitar la cantidad de reenvíos permitidos.
39. El sistema debe mostrar un formulario para ingresar el código de verificación.
40. El sistema debe permitir que el usuario ingrese el código de verificación recibido por correo.
41. El sistema debe validar que el código ingresado coincida con el enviado.
42. El sistema debe validar que el código no esté expirado.
43. El sistema debe activar la cuenta del usuario si el código es válido.
44. El sistema debe mostrar un mensaje de activación exitosa.
45. El sistema debe impedir el inicio de sesión si la cuenta no está activada.
46. El sistema debe mostrar un formulario de inicio de sesión.

47. El sistema debe solicitar correo y contraseña.
48. El sistema debe validar que los campos no estén vacíos.
49. El sistema debe verificar que el usuario exista.
50. El sistema debe comparar la contraseña ingresada con la cifrada.
51. El sistema debe registrar intentos fallidos de inicio de sesión.
52. El sistema debe bloquear temporalmente la cuenta tras varios intentos fallidos.
53. El sistema debe crear una sesión segura al autenticarse.
54. El sistema debe redirigir al usuario según su rol.
55. El sistema debe mostrar un menú principal con la lista de servicios de la peluquería  
(por ejemplo: corte de cabello, peinado, manicure, coloración, pedicure, depilación)  
junto con su precio.
56. El sistema debe consultar los servicios activos en la base de datos.
57. El sistema debe mostrar el nombre del servicio.
58. El sistema debe mostrar el precio del servicio.
59. El sistema debe ocultar servicios inactivos.
60. El sistema debe ordenar los servicios por categoría o precio.
61. El sistema debe permitir al cliente agendar una cita seleccionando servicio, fecha,  
hora y profesional disponible, registrando la reserva en la agenda.
62. El sistema debe permitir seleccionar un servicio.
63. El sistema debe mostrar profesionales disponibles para el servicio.
64. El sistema debe mostrar fechas disponibles.
65. El sistema debe mostrar horarios disponibles según duración del servicio.
66. El sistema debe impedir seleccionar fechas pasadas.
67. El sistema debe validar disponibilidad antes de confirmar la reserva.
68. El sistema debe registrar la cita en la base de datos.
69. El sistema debe asignar un estado inicial a la cita.
70. El sistema debe mostrar un listado de citas del cliente.
71. El sistema debe mostrar fecha, hora, servicio y estado.
72. El sistema debe ordenar las citas por fecha.

73. El sistema debe permitir al cliente consultar el listado de sus citas reservadas, mostrando servicio, fecha, hora y estado de la reserva.
74. El sistema debe permitir al cliente cancelar una cita futura y actualizar el estado de la reserva en la agenda.
75. El sistema debe solicitar confirmación de cancelación.
76. El sistema debe actualizar el estado de la cita.
77. El sistema debe liberar el horario cancelado.
78. El sistema debe permitir al administrador registrar, modificar y eliminar servicios (nombre, precio, duración, descripción).
79. El sistema debe permitir al administrador gestionar los usuarios de tipo cliente (crear, editar, desactivar cuentas cuando sea necesario).
80. El sistema debe permitir crear servicios nuevos.
81. El sistema debe permitir editar servicios existentes.
82. El sistema debe validar datos del servicio.
83. El sistema debe impedir eliminar servicios con citas asociadas.
84. El sistema debe permitir desactivar servicios.
85. El sistema debe permitir al administrador consultar un reporte básico de citas (por fecha, servicio o profesional) para apoyar la gestión del negocio.
86. El sistema debe permitir listar usuarios clientes.
87. El sistema debe permitir editar información del usuario.
88. El sistema debe permitir desactivar cuentas de usuarios.
89. El sistema debe permitir filtrar reportes por fecha.
90. El sistema debe permitir filtrar por servicio.
91. El sistema debe permitir filtrar por profesional.
92. El sistema debe mostrar totales de cita.

## Requerimientos No Funcionales

1. El sistema debe estar disponible al menos el 95% del tiempo mensual, salvo períodos de mantenimiento programado.
2. La página de inicio debe cargar en un tiempo máximo de 5 segundos bajo una conexión a internet estándar.
3. Cualquier consulta o acción de usuario (inicio de sesión, ver servicios, reservar cita) debe responder en un tiempo máximo de 2-3 segundos en condiciones normales.
4. Las contraseñas de los usuarios deben almacenarse cifradas en la base de datos para proteger las credenciales de acceso.

5. El sistema debe restringir el acceso a la agenda, la gestión de servicios y los reportes sólo a usuarios autenticados con el rol correspondiente (cliente o administrador).
6. La aplicación debe cumplir con las normas de protección de datos personales aplicables (por ejemplo, uso adecuado de datos de clientes).
7. La interfaz debe ser intuitiva, con botones y mensajes claros, de forma que un usuario nuevo pueda registrarse y reservar una cita sin necesidad de capacitación previa.
8. La aplicación web debe ser responsive, adaptándose correctamente a pantallas de computador, Tablet y celular.
9. El sistema debe ser compatible, como mínimo, con los navegadores Chrome, Edge y Firefox en sus versiones soportadas.
10. El sistema debe registrar en un log interno los intentos de inicio de sesión fallidos y las operaciones críticas (alta/baja de servicios, cambios de rol, etc.) para fines de auditoría.
11. Los mensajes de error deben ser claros y no técnicos, indicando al usuario qué acción debe realizar (por ejemplo: "revise el correo o la contraseña").
12. El diseño del código debe seguir una estructura modular (separación entre interfaz, lógica y datos) para facilitar su mantenimiento y futuras ampliaciones.