

1) Funcionalidades del Empleado (usuario)

A. Cuenta y acceso

- **E1. Iniciar sesión** con usuario/contraseña. (RF1.4)
- **E2. Gestión de vehículos:**
 - Añadir vehículo / matrícula
 - Dar de baja vehículo (RF2.8)

B. Reservas

- **E3. Solicitar reserva de plaza** para un día. (RF2.1)
- **E4. Ver disponibilidad por día** (calendario/consulta). (RF2.2, RF4.3)
- **E5. Ver historial y reservas activas** (pasadas + futuras). (RF2.4)
- **E6. Cancelar reservas futuras.** (RF2.5)
- **E7. Restricción opcional:** impedir solicitar si ya tiene una reserva ese día. (RF2.6)
- **E8. Ver estado de sus reservas:** pendiente, confirmada, usada, cancelada, expirada. (RF2.7)

C. Interfaz (empleado)

- **E9. Página de login.** (RF4.1)
 - **E10. Dashboard personal** con próximas reservas y avisos. (RF4.2)
 - **E11. Formulario de solicitud/cancelación.** (RF4.4)
 - **E12. Notificaciones visuales** (éxito/error). (RF4.5)
-

2) Funcionalidades del Administrador (admin)

A. Gestión de empleados y roles

- **A1. Alta de empleados** (crear usuarios). (RF1.1)
- **A2. Editar datos de empleado.** (RF1.2)
- **A3. Eliminar usuarios.** (RF1.3)
- **A4. Asignar roles y permisos** (“empleado”, “administrador”). (RF1.5)
- **A5. Gestión completa de usuarios** (listado, filtros, estado activo/inactivo). (RF5.4)

B. Reservas del sistema

- **A6. Ver todas las reservas** del sistema. (RF5.1)
- **A7. Cambiar ajustes globales:** cupos diarios, horarios, reglas, plazas por planta, etc. (RF5.2)
- **A8. Exportar información** (PDF/CSV). (RF5.5)

C. Algoritmo de asignación

- **A9. Ejecutar algoritmo automáticamente** cuando un empleado solicita matrícula/reserva. (RF3.1)
- **A10. Asignar plaza si hay disponibilidad.** (RF3.2)
- **A11. Lista de espera** si no hay plazas. (RF3.3)
- **A12. Garantizar prioridad según reglas** (rotación, no-shows, uso reciente...). (RF3.4)
- **A13. Lanzar asignación diaria manualmente** (según el enunciado del TFG).
(relacionado con simulador/operación diaria)

D. Estadísticas e informes

- **A14. Ver estadísticas:** ocupación, no-shows, etc. (RF5.3)

- **A15. Calcular ocupación diaria/horaria.** (RF7.1)
- **A16. Exportar estadísticas a PDF/CSV.** (RF7.2)
- **A17. Relación reservas vs ocupación** (y que el empleado vea sus propios datos). (RF7.3)

E. Simulador de accesos (operativa)

- **A18. Simular llegada de un vehículo.** (RF6.1)
 - **A19. Verificar matrícula con reserva válida.** (RF6.2)
 - **A20. Mostrar resultado:** plaza asignada / denegado / no hay reserva / reserva ya usada. (RF6.3)
 - **A21. Registrar entrada/salida en base de datos.** (RF6.4)
 - **A22. Notificar incidencias al admin.** (RF6.5)
 - **A23. Informes del simulador** (uso, incidencias, estadísticas). (RF6.6)
 - **A24. Ver registros del simulador.** (RF5.6)
-

3) Requisitos no funcionales (aplican a todo el sistema)

- Rendimiento: respuestas < 1s (general) y simulador < 300ms.
- Seguridad: hashing fuerte (BCrypt/Argon2), roles estrictos, protección SQLi/XSS/CSRF.
- UX: interfaz accesible, sencilla, responsive, navegadores modernos.
- Mantenibilidad: modular, documentado, MVC o similar, ampliable (plazas/días/reglas).
- Calidad: tests unitarios e integración.
- Disponibilidad: mínimo 95% + recuperación automática ante errores.
- Integración: API REST estándar.

4) criterios de asignación justa + usabilidad real de plazas

4.1 Criterios de equidad (fairness) para el algoritmo

- **F1. Historial de reservas (C1):** a más reservas históricas, menor prioridad (rotación).
- **F2. No-shows (C2):** penalización fuerte si reserva y no entra (optimiza uso).
- **F3. Uso reciente (C3):** si aparcó ayer, baja hoy; si lleva tiempo sin aparcar, sube.
- **F4. Antigüedad / orden de llegada** (si lo quieres como regla adicional).
- **F5. Empates (C5):** sorteo controlado (seed/registro) para evitar sesgos y poder auditar.

Extra que queda muy bien en memoria:

- **F6. Límite de “ventana”:** prioridad calculada sobre últimas N semanas (evitas castigar para siempre).
- **F7. Transparencia:** el usuario puede ver “por qué” está en lista de espera (explicación de puntuación resumida).

4.2 Criterios de verificación de usabilidad de plazas (operativos)

Esto responde a tu frase “verificación de usabilidad de plazas”.

- **U1. Validación por acceso real:** una reserva cuenta como “usada” solo si hay entrada registrada (simulador).
- **U2. Expiración automática:** si no entra antes de X minutos desde la hora límite → pasa a “expirada” y se libera.
- **U3. Reasignación dinámica:** si expira, reasignar al siguiente de lista de espera (si aplica).
- **U4. Consistencia reserva-ocupación:**

- Ocupación real = entradas - salidas
- Comparar contra reservas confirmadas → detectar desvíos
- **U5. Detección de abuso:** muchos no-shows seguidos → bloqueo temporal o reducción de prioridad.
- **U6. Reglas por tipo de plaza** (si hay plantas o zonas): plazas reservadas por planta, movilidad reducida, etc.