

Sistema Web de Gestión de Centros Deportivos y Reservas

Resumen de Especificaciones y Requisitos (PDF)

Este documento resume, depura y organiza las especificaciones y requisitos del proyecto, extraídos de la Especificación Técnica – Sistema Web de Gestión de Centros Deportivos y Reservas. Incluye alcance, arquitectura, requisitos funcionales y no funcionales, modelo de datos, API y consideraciones de seguridad.

1. Alcance del proyecto

- Aplicación web para gestionar usuarios (jugadores, administradores de centros, entrenadores, árbitros) y espacios deportivos.
- Reservas en tiempo real con prevención de doble reserva y soporte de pagos en línea.
- Panel de administración para centros y backoffice global con estadísticas, promociones y gestión de personal.
- Soporte multilenguaje (al menos español e inglés) y experiencia responsive.

Fuera de alcance inicial (próximas fases)

- Integración con wearables.
- Sincronización con calendarios externos (Google Calendar, etc.).
- Matchmaking avanzado entre jugadores.

2. Arquitectura y tecnologías

- Frontend: React (SPA), preferentemente con TypeScript y Tailwind CSS. Enrutamiento con React Router; estado con Context + React Query/SWR.
- Backend: Node.js + Express, arquitectura en capas (rutas, controladores, servicios, middlewares). Stateless para escalado horizontal.
- Base de datos: PostgreSQL gestionado en Supabase (Auth JWT, Storage, Realtime, RLS).
- Pagos: Stripe (Payment Intents o Checkout) con confirmación vía webhooks y reembolsos.
- Notificaciones: correo (p. ej. SendGrid) y opcional SMS/push. Posible uso de Supabase Edge Functions.
- Despliegue: Frontend (Vercel/Netlify); Backend (Fly/Heroku/Docker). CI/CD y variables de entorno seguras.

3. Requisitos funcionales

- Gestión de usuarios y roles: registro/authenticación con Supabase Auth; perfiles con roles (player, center_admin, coach, referee, admin).

- Centros e instalaciones: CRUD de centros e instalaciones (tipo, precio/hora, estado), horarios base y bloqueos puntuales.
- Disponibilidad y reservas: consulta de slots por fecha/instalación; creación/cancelación con reglas; historial y próximas reservas.
- Prevención de doble reserva: verificación de disponibilidad y constraint único en BD (facility_id, fecha, hora_inicio).
- Pagos en línea: cálculo de importe, inicio de cobro, confirmación y actualización de estado; reembolsos cuando proceda.
- Cancelaciones: política de plazo (p. ej., no permitir el mismo día); reembolso total/parcial según reglas.
- Notificaciones: confirmación/cancelación por email y en-app; soporte realtime (WebSockets) para cambios.
- Promociones: cupones globales o por centro (porcentaje/importe fijo, vigencia, usos).
- Panel Admin Centro: reservas del centro, CRUD de instalaciones/personal, estadísticas locales.
- Backoffice: métricas globales, gestión de usuarios, promociones y centros.

4. Requisitos no funcionales

- Seguridad: HTTPS, validación de JWT, CORS restringido, RLS en Supabase, autorizaciones por rol, rate limiting, Helmet.
- Calidad: TypeScript opcional, linting (ESLint), formateo (Prettier), pruebas unitarias e2e (Jest/RTL, Cypress).
- Rendimiento: API stateless, caché en cliente (React Query/SWR), DB indexada; posibilidad de réplicas de lectura.
- Disponibilidad y escalado: escalado horizontal del backend; backups automáticos y migraciones versionadas.
- Accesibilidad y UX: ARIA, navegación por teclado, diseño responsive mobile-first.
- Internacionalización: i18next con carga dinámica; detección de idioma y selector.
- Observabilidad: logging estructurado y métricas; monitorización de errores.
- Cumplimiento: SCA (PSD2) y PCI-DSS delegados en Stripe; protección de datos y privacidad.

5. Modelo de datos (resumen)

- profiles (id UUID, nombre, rol, center_id, foto_url, timestamps).
- centers (id, nombre, dirección, admin_user_id, horario_apertura/cierre, activos).
- facilities (id, center_id, nombre, tipo, capacidad, precio_hora, facilitator_id, activo).
- bookings (id, facility_id, user_id, fecha, hora_inicio, estado, price_paid, payment_id, created_at, cancelled_at).
- payments (id, booking_id, monto, moneda, status, metodo, transaccion_id, timestamps).

- promotions (id, codigo, tipo_descuento, valor_descuento, centro_id, vigencia, uso_maximo, usos_realizados, activo).

6. Endpoints principales (indicativos)

- Auth: POST /auth/register, POST /auth/login (opcional, vía Supabase Auth desde frontend).
- Usuarios: GET /usuarios/me, PUT /usuarios/me, (admin) GET/PUT /admin/usuarios/:id.
- Centros: GET /centros, GET /centros/:id, POST/PUT /centros/:id (según permisos).
- Instalaciones: GET /centros/:id/instalaciones, POST /centros/:id/instalaciones, PUT /instalaciones/:id, DELETE /instalaciones/:id.
- Disponibilidad: GET /instalaciones/:id/disponibilidad?fecha=YYYY-MM-DD.
- Reservas: POST /reservas, GET /reservas?usuarioId=..., GET /reservas?centroId=..., DELETE /reservas/:id.
- Pagos: POST /pagos/checkout | /pagos/payment-intent, POST /webhook/stripe, GET /pagos/:id.
- Admin Centro: GET /admin-centro/resumen, GET /admin-centro/reservas, POST /admin-centro/personal.
- Backoffice: GET /admin/estadisticas, POST /admin/promociones, GET /admin/centros.

7. Políticas de negocio y reglas

- Prevención de doble reserva mediante verificación previa y restricción única en BD.
- Política de cancelación (configurable): p. ej., no permitir cancelar el mismo día; reembolso según caso.
- Estados de reserva: PENDING_PAYMENT, CONFIRMED, CANCELLED, COMPLETED; limpieza de pendientes no pagadas.
- Permisos por rol y centro: jugadores ven/gestionan sus reservas; admins centro gestionan lo de su ámbito; admin global todo.

8. Seguridad y control de acceso (RLS)

- RLS activado en tablas sensibles (bookings, profiles, etc.).
- Políticas: jugadores solo sus filas; admins centro, reservas de sus instalaciones; admin global vía backend (service key).
- Validación de JWT en middleware; autorización por rol antes de cada operación crítica.

9. Integraciones y notificaciones

- Stripe para cobros y reembolsos; confirmación por webhooks.
- Email transaccional (p. ej. SendGrid).
- Realtime via Supabase para actualizaciones de reservas y notificaciones en-app.

- Opcional: SMS/WhatsApp; push web; Edge Functions para tareas asíncronas.

10. Roadmap (futuras ampliaciones)

- Integración con calendarios externos (Google Calendar, etc.).
- Matchmaking de jugadores y recomendación de partidos.
- Integración con wearables y analítica avanzada.
- Membresías/suscripciones y multimoneda.