P@SSID · DOCUMENTACIÓN TÉCNICA

API del club

Reservas de cancha con video · v1.0

Para: el equipo técnico del club (primer cliente: Pádel Carbono-Providencia). **Todo en un lugar:** https://gameback.passid.com.mx/integration **Versión:** 1.0 · 2026-11-01 · **Especificación OpenAPI:** `openapi.json`, publicada en https://gameback.passid.com.mx/api/club/v1/openapi.json · **Documentación en línea:** https://gameback.passid.com.mx/api/club/v1/docs

1. ¿Para qué sirve?

Tu sistema de reservas puede apartar una cancha de P@ssID para uno o varios jugadores, con **cancha, horario y teléfono**, igual que lo haría tu recepción. Desde ese momento:

1. El jugador llega, **escanea el QR de la cancha** y entra con su WhatsApp (le llega un código de 6 dígitos).
2. Toca **«Iniciar grabación»**: cada set y cada repetición (botón físico o en pantalla) le llega por **WhatsApp**.
3. Si el club configuró su **webhook** en P@ssID, cada clip también se manda a tu sistema, firmado con HMAC-SHA256.

Quien no tiene reserva ni invitación ve «Pasa a recepción» y no recibe ningún código.

2. Acceso

Base	`https://gameback.passid.com.mx/api/club/v1`
Llave	`Authorization: Bearer cbk_…` (también se acepta `X-Api-Key: cbk_…`)
ID de cada solicitud	`X-Request-Id: <tu id>` — **obligatorio en TODAS las llamadas** (1–100 caracteres `A-Z a-z 0-9 . _ : -`)
Formato	JSON, UTF-8. Fechas ISO 8601 (`2026-11-02T18:00:00-06:00` o `…Z`). Respuestas en UTC.
Límite	120 llamadas por minuto por llave (si te pasas: `429`).

La llave la crea el dueño del club en *app.passid.com.mx → Mi club → Cámaras → «API del club»* (o P@ssID). Se muestra **una sola vez**; guárdala como secreto. **Vence a los 90 días** (el dueño puede elegir de 1 a 365) y el dueño o P@ssID la **revocan** en cualquier momento: deja de servir en el acto. Cada llave es de **un solo club** y sólo ve sus canchas.

3. El contrato: dos IDs que vinculan los dos sistemas

- **Tú mandas** `X-Request-Id` en cada solicitud. Nosotros lo regresamos en el header y en el cuerpo (`requestId`).
- **Toda respuesta exitosa es** `200` e incluye tu `requestId` y **nuestro** `reservationId`. Guárdalo junto a tu id.
- `externalId` (opcional, en el cuerpo de la reservación) es **el id de la reservación en tu sistema**; queda guardado y si lo vuelves a mandar regresamos la misma reservación (`existing: true`) en lugar de crear otra.
- **Reintentos seguros:** repetir un `POST` / `DELETE` con el **mismo** `X-Request-Id` regresa la respuesta original **sin volver a ejecutar** (header `X-Idempotent-Replay: true`). Así, si se cae tu conexión, reintenta con el mismo id sin miedo a duplicar.
- **Rastreo:** `GET /requests/{tuRequestId}` te dice qué pasó con cualquier solicitud (código, segundos, respuesta, reservación).

Errores: `{ "requestId": "...", "error": "mensaje en inglés" }` con `400` (datos), `401` (llave inválida, vencida o revocada), `404` (no existe en tu club), `409` (la cancha ya está ocupada a esa hora), `429` (límite).

4. Endpoints

Método	Ruta	Qué hace
GET	`/me`	Prueba la llave: tu club, vencimiento de la llave y canchas
GET	`/courts`	Canchas con video del club (id, slug, nombre, URL del QR, duración del turno, zona horaria)
GET	`/courts/{court}/availability?date=YYYY-MM-DD`	Horarios libres del día (`maxMinutes` = cuánto cabe desde ahí)
POST	`/reservations`	Aparta la cancha para 1 a 8 jugadores
GET	`/reservations?from=&to=`	Reservaciones del club (por defecto hoy + 7 días; máx. 62)
GET	`/reservations/{reservationId}`	Una reservación
DELETE	`/reservations/{reservationId}`	La cancela y libera la cancha (repetirlo también da 200)
GET	`/requests/{requestId}`	Estado de una solicitud tuya
GET	`/logs?from=&to=`	Bitácora: respuesta, código y segundos de cada llamada. **Sin parámetros = las llamadas de hoy** (hora de México). Máx. 31 días; se guarda 30 días

`{court}` acepta el **id** o el **slug** (`carbono-cancha-01`).

Crear una reservación

```
curl -X POST https://gameback.passid.com.mx/api/club/v1/reservations \
  -H "Authorization: Bearer cbk_TU_LLAVE" \
  -H "X-Request-Id: carb-2026-11-02-0001" \
  -H "Content-Type: application/json" \
  -d '{
        "court": "carbono-cancha-01",
        "startUtc": "2026-11-02T18:00:00-06:00",
        "minutes": 60,
        "phone": "3312345678", "name": "Pepe",
        "players": [{ "phone": "3398765432", "name": "Toño" }],
        "externalId": "CARB-000123",
        "note": "Pagó en recepción"
      }'
```

Respuesta `200`:

```
{
  "requestId": "carb-2026-11-02-0001",
  "reservationId": "5b0c7a1e-…",
  "externalId": "CARB-000123",
  "existing": false,
  "reservation": {
    "reservationId": "5b0c7a1e-…", "externalId": "CARB-000123",
    "court": { "id": "e80048cf-…", "slug": "carbono-cancha-01", "name": "Cancha 1",
               "url": "https://mi.passid.com.mx/court/carbono-cancha-01", "slotMinutes": 60,
"timezone": "America/Mexico_City" },
    "startUtc": "2026-11-03T00:00:00Z", "endUtc": "2026-11-03T01:00:00Z",
    "players": [ { "phone": "5213312345678", "name": "Pepe" }, { "phone": "5213398765432",
"name": "Toño" } ],
    "status": "upcoming", "source": "api", "createdBy": "API · Sistema de Carbono", "note":
"Pagó en recepción",
    "createdAt": "2026-11-01T17:20:11Z"
  }
}
```

- `phone` + `name` es el jugador principal; `players` agrega más (máximo 8 en total). Números de México de 10 dígitos se aceptan y se normalizan con lada (`521…`). **A todos** les llegan los clips y cualquiera maneja la grabación.
- `minutes` de 15 a 240 (por defecto, la duración del turno de la cancha).
- `status`: `upcoming` → `live` → `done`, o `cancelled`.

Bitácora del día

```
curl https://gameback.passid.com.mx/api/club/v1/logs \
  -H "Authorization: Bearer cbk_TU_LLAVE" -H "X-Request-Id: carb-logs-0001"
```

Cada llamada trae `requestId`, `at`, `method`, `path`, `status`, `seconds` (cuánto tardamos), `reservationId`, `request` y `response` (recortados a 4 KB). Para otro periodo: `?from=2026-11-01T00:00:00-06:00&to=2026-11-02T00:00:00-06:00`.

5. Webhook del club (clips de regreso)

Se configura en *Mi club → Cámaras → «Webhook del club»* (URL `https` + secreto). Por cada clip o repetición listos:

- `POST` JSON con `type: "clip.ready"`, `kind` (`clip` | `replay`), cancha, título, rango, `videos[]` (URL firmada válida 1 h) y `recipients[]` (nombre, teléfono y link de cada jugador). Con el `reservationId` de la API puedes cruzarlo por cancha y hora.
- Headers: `X-PassID-Event`, `X-PassID-Timestamp` (Unix) y `X-PassID-Signature: sha256=<hex>` = `HMAC-SHA256(secreto, "{timestamp}.{cuerpo}")`. Verifica la firma y que el timestamp sea reciente. 3 intentos (0 s, 5 s, 25 s).

6. MCP para agentes de IA

Si tu agente de IA habla **MCP** (Model Context Protocol: Claude, ChatGPT, Cursor, etc.), conéctalo sin programar llamadas HTTP:

URL	`https://gameback.passid.com.mx/mcp` (Streamable HTTP)
Autenticación	`Authorization: Bearer cbk_TU_LLAVE` (la misma llave de la API)
Herramientas	`club_info`, `list_courts`, `check_availability`, `create_reservation`, `get_reservation`, `list_reservations`, `cancel_reservation`, `request_status`, `call_log`

Cada herramienta es una llamada a esta API: aplica la misma llave, vigencia, revocación, límite y bitácora. `create_reservation` y `cancel_reservation` aceptan `request_id` (repítelo al reintentar); si mandas `external_id` y no `request_id`, se usa `mcp-<external_id>`, así un reintento nunca duplica. Ejemplo de configuración de un cliente MCP:

```
{ "mcpServers": { "passid-club": { "type": "http", "url": "https://gameback.passid.com.mx/mcp",
  "headers": { "Authorization": "Bearer cbk_TU_LLAVE" } } } }
```

7. Buenas prácticas

1. Genera un `X-Request-Id` único por operación de negocio (p. ej. tu id de reserva + intento) y **reintenta con el mismo**.
2. Manda siempre `externalId`: es tu llave de conciliación.
3. Consulta `availability` antes de reservar para no recibir `409`.
4. Si tu proceso duda de un resultado, pregunta `GET /requests/{id}` en lugar de volver a crear.

5. Guarda la llave como secreto y pide una nueva antes de los 90 días.