

## 03 — Callbacks API (async action)

**Audience:** Tenant IT. **Prasyarat:** [01-overview.md](#), [02-authentication.md](#), dan paham konsep **custom action** (didefinisikan oleh tenant CS di konfigurasi EMBAN — lihat [#5 Action engine](#)).

Endpoint ini dipanggil Tenant IT **setelah** EMBAN memicu action asinkron ke backend Tenant IT. Alurnya searah: customer → EMBAN → Tenant IT → EMBAN → customer.

---

### 1. Model & terminologi singkat

Istilah	Artinya
<b>Action</b>	Spesifikasi kustom milik tenant yang mendefinisikan endpoint Tenant IT + field-field yang harus dikumpulkan dari customer + template jawaban. Disimpan sebagai row di <code>tenant_actions</code> .
<b>Action definition</b>	JSON schema action (lihat <code>src/services/action-schema.ts</code> ), mencakup <code>endpoint.url</code> , <code>endpoint.auth.secret_env</code> , dan <code>response.mode</code> (sync atau async).
<b>Submission</b>	Satu instansi eksekusi action. Row di <code>action_submissions</code> dengan id ULID. Status lifecycle: <code>collecting</code> → <code>submitted</code> → <code>awaiting_callback</code> → <code>callback_delivered</code> .
<b>Sync mode</b>	EMBAN menunggu response HTTP Tenant IT sampai <code>timeout_ms</code> , render langsung ke customer. <b>Tidak ada callback.</b>
<b>Async mode</b>	EMBAN balas <code>immediate_ack</code> ke customer, lalu menunggu Tenant IT POST ke callback endpoint untuk melanjutkan. <b>Mode inilah yang membutuhkan endpoint ini.</b>

Jika action Anda pakai `response.mode: "sync"`, dokumen ini tidak relevan — cukup reply body HTTP di tempat (lihat **#5 Action engine**).

---

## 2. Endpoint

POST `https://api.rentalai.id/api/v1/callbacks/:submission_id`

Content-Type: `application/json`

X-ASPRI-Timestamp: `<unix-millis>`

X-ASPRI-Signature: `sha256=<hex>`

Idempotency-Key: `<ulid | uuid>` # opsional tapi direkomendasikan

- `submission_id` diberikan ke Tenant IT di **outbound request pertama** (POST `<action.endpoint.url>`) dalam body JSON + header `X-ASPRI-Submission-Id` (lihat **#5**). Simpan itu, pakai di URL callback.
  - **Secret untuk HMAC** = value dari env var yang namanya disebut di `action.endpoint.auth.secret_env`. Disepakati saat tenant mendaftarkan action.
  - **Window status**: EMBAN hanya menerima callback saat submission masih berstatus `awaiting_callback`. Status lain → 409 `invalid_submission_state`.
- 

## 3. Request body

Skema Zod ada di `src/api/schemas/callback.ts`. Ringkasan field:

Field	Tipe	Wajib	Catatan
<code>status</code>	<code>"success"   "error"</code>	default <code>"success"</code>	Menentukan template mana yang dipilih otomatis.
<code>template_key</code>	<code>string</code>	optional	Override eksplisit — nama key di <code>response.error_templates</code> . Bekerja untuk success

Field	Tipe	Wajib	Catatan
			maupun error outcomes.
<code>data</code>	object	default <code>{}</code>	Payload yang diinterpolasi ke template (mustache-lite). Key bebas, nested paths OK.
<code>error_code</code>	string	optional, direkomendasikan saat <code>status="error"</code>	Dicari di <code>response.error_templates[&lt;code&gt;]</code> . Jatuh ke <code>.default</code> jika tidak ada.
<code>error_message</code>	string	optional	Tersedia sebagai <code>{{error_message}}</code> di template.
<code>files</code>	array ≤ 10	optional	Attachment untuk customer. Detail di §6.

Contoh minimal (success):

```
{
  "status": "success",
  "template_key": "booking_confirmed",
  "data": {
    "booking_id": "BK-2026-0412",
    "car_model": "Toyota Avanza",
    "pickup_date": "22 April 2026",
    "price_idr": "850.000"
```

```
}  
  
}
```

Contoh error:

```
{  
  
  "status": "error",  
  
  "error_code": "out_of_stock",  
  
  "error_message": "Unit sudah disewa tanggal tersebut",  
  
  "data": { "alternative_date": "24 April 2026" }  
}
```

---

## 4. Pemilihan template (fallback chain)

EMBAN memilih template dalam urutan berikut — yang pertama cocok dipakai:

1. `template_key` eksplisit → `response.error_templates[template_key]`
2. Jika `status="success"` → `response.success_template`, lalu `error_templates.default`
3. Jika `status="error"` → `error_templates[error_code]`, lalu `error_templates.default`
4. **Fallback terakhir** (no template matched): EMBAN menyusun pesan generik dari `data` / `error_message`. Tidak silent-drop.

Rekomendasi: tenant **selalu** menyediakan `error_templates.default` supaya fallback generik jarang dipakai.

---

## 5. Template rendering (mustache-lite)

EMBAN memakai parser mustache-lite sederhana (lihat `renderMustache` di `src/api/routes/callbacks.ts`):

- `{{field}}` → `data.field`
- `{{obj.nested.key}}` → `data.obj.nested.key`
- `{{error_code}}`, `{{error_message}}` tersedia top-level
- Value `null/undefined` → empty string (tidak error, tidak "undefined" literal)
- Non-string value → `JSON.stringify(value)`
- **Tidak ada** loop `{{#each}}`, conditional `{{#if}}`, partial `{{>}}`, atau filter Handlebars — sengaja sederhana. Kalau perlu logika, lakukan di sisi Tenant IT sebelum POST.

Contoh action definition yang cocok dengan request di §3:

```
{
  "response": {
    "mode": "async",
    "immediate_ack": {
      "id": "Booking sedang diproses, mohon tunggu..."
    },
    "error_templates": {
      "booking_confirmed": {
        "id": "Booking {{booking_id}} untuk {{car_model}} pada {{pickup_date}} sudah dikonfirmasi.
Total: Rp{{price_idr}}."
      },
      "out_of_stock": {
        "id": "Maaf, {{error_message}}. Tanggal terdekat tersedia: {{alternative_date}}."
      },
      "default": {
        "id": "Sistem: {{error_message}}"
      }
    }
  }
}
```

```
}  
  
}  
  
}
```

---

## 6. File attachment (Phase 5.5)

Field `files[]` memungkinkan Tenant IT melampirkan file (invoice PDF, foto unit, dll) yang akan dikirim ke customer di channel aslinya.

Schema per-file:

```
{  
  
  "url": "https://files.tenant.example/invoice/BK-2026-0412.pdf",  
  
  "filename": "invoice.pdf",  
  
  "mime_type": "application/pdf",  
  
  "expected_size_bytes": 524288  
}
```

### Aturan sisi EMBAN:

Aturan	Nilai default	Sumber config
Protokol	<b>HTTPS wajib</b>	hardcoded
Hostname	harus cocok dengan <b>allowlist per-tenant</b> ( <code>tenants.file_download_allowlist</code> )	dikonfigurasi tenant saat onboarding
Max size per file	10 MB	<code>FILE_DELIVERY_MAX_BYTES</code>
Max files per callback	10	schema

Aturan	Nilai default	Sumber config
MIME allowlist	PDF, PNG, JPG, MP3, MP4, MP3, docx, xlsx, zip (lihat <a href="#">file-download.ts</a> )	hardcoded
Content sniff	magic bytes dicocokkan dengan <code>mime_type</code> — mismatch → rejected	
TTL di disk EMBAN	24 jam (lalu auto-delete)	<code>FILE_DELIVERY_TTL_HOURS</code>
Connect timeout	10 detik	<code>FILE_DELIVERY_CONNECT_TIMEOUT_MS</code>
Read timeout	30 detik	<code>FILE_DELIVERY_READ_TIMEOUT_MS</code>

**Partial success:** text message (dari §5) selalu di-deliver dulu. Setiap file diproses independen — beberapa sukses, beberapa gagal, reply body membawa detail per file:

```
{
  "ok": true,
  "delivered": true,
  "customer_channel": "whatsapp",
  "submission_id": "01J...",
  "files": {
    "delivered": 1,
    "rejected": 1,
    "failed": 0,
    "details": [
      { "url": "...", "status": "delivered" },
```

```
{ "url": "...", "status": "rejected", "reason": "mime_not_allowed" }  
  
]  
  
}  
  
}
```

Status nilai: **delivered**, **rejected** (validation sisi EMBAN — hostname/MIME/size),  
**fetch\_failed** (network/4xx/5xx), **delivery\_failed** (channel adapter error).

---

## 7. Response ASPRI

200 OK — delivered

```
{  
  
  "ok": true,  
  
  "delivered": true,  
  
  "customer_channel": "whatsapp",  
  
  "submission_id": "01J1ZXK7..."  
  
}
```

200 OK — customer deleted / blocked

```
{  
  
  "ok": true,  
  
  "delivered": false,  
  
  "reason": "customer_deleted"  
  
}
```



Submission tetap ditandai `callback_delivered` (dengan `error_message=customer_deleted` di DB). Tenant IT tidak perlu retry — customer memang sudah hilang dari sistem.

#### 4xx/5xx — daftar error

HTTP	error_code	Artinya	Aksi Tenant IT
400	<code>schema_invalid</code>	body tidak lulus Zod validation	cek <code>details</code> , perbaiki payload
401	<code>missing_auth_headers / invalid_signature / timestamp_skew / unknown_signer</code>	HMAC gagal	lihat <code>02-authentication.md</code>
404	<code>submission_not_found</code>	<code>submission_id</code> tidak ada	cek typo / expired
409	<code>invalid_submission_state</code>	submission bukan <code>awaiting_callback</code> (mis. sudah delivered, atau timeout)	<b>jangan retry</b> — state tidak akan balik
409	<code>idempotency_key_conflict</code>	key yang sama dipakai untuk endpoint berbeda	ganti key
500	<code>action_missing / action_definition_corrupt / no_customer_linked / no_agent_for_tenant</code>	state rusak di sisi EMBAN	escalate ke tim EMBAN dengan <code>request_id</code>
503	<code>delivery_unavailable / delivery_failed</code>	channel adapter belum siap atau error saat kirim	retry dengan backoff eksponensial; kalau konsisten → escalate

## 8. Contoh lengkap — Node.js

```
import { createHmac, randomUUID } from "node:crypto";

const BASE = "https://api.rentalai.id";

const SECRET = process.env.ASPRI_ACTION_SECRET; // dari env yang SAMA dengan
action.endpoint.auth.secret_env

async function sendCallback(submissionId, body) {

  const raw = JSON.stringify(body);

  const ts = Date.now();

  const sig =

    "sha256=" +

    createHmac("sha256", SECRET).update(`${ts}.${raw}`).digest("hex");

  const res = await fetch(`${BASE}/api/v1/callbacks/${submissionId}`, {

    method: "POST",

    headers: {

      "Content-Type": "application/json",

      "X-ASPRI-Timestamp": String(ts),

      "X-ASPRI-Signature": sig,

      "Idempotency-Key": randomUUID(),

    },

    body: raw,

  });

  return { status: res.status, body: await res.json() };
```

```
}
```

```
// contoh pemakaian
```

```
await sendCallback("01J1ZXK7ABCDE", {
```

```
  status: "success",
```

```
  template_key: "booking_confirmed",
```

```
  data: {
```

```
    booking_id: "BK-2026-0412",
```

```
    car_model: "Toyota Avanza",
```

```
    pickup_date: "22 April 2026",
```

```
    price_idr: "850.000",
```

```
  },
```

```
  files: [
```

```
    {
```

```
      url: "https://files.tenant.example/invoice/BK-2026-0412.pdf",
```

```
      filename: "invoice-BK-2026-0412.pdf",
```

```
      mime_type: "application/pdf",
```

```
    },
```

```
  ],
```

```
});
```

---

## 9. Contoh lengkap — Python

```
import hmac, hashlib, json, time, uuid, os, requests

BASE = "https://api.rentalai.id"

SECRET = os.environ["ASPRI_ACTION_SECRET"].encode()

def send_callback(submission_id: str, body: dict):

    raw = json.dumps(body, separators=(",", ":"))

    ts = int(time.time() * 1000)

    sig = "sha256=" + hmac.new(

        SECRET, f"{ts}.{raw}".encode(), hashlib.sha256

    ).hexdigest()

    resp = requests.post(

        f"{BASE}/api/v1/callbacks/{submission_id}",

        data=raw,

        headers={

            "Content-Type": "application/json",

            "X-ASPRI-Timestamp": str(ts),

            "X-ASPRI-Signature": sig,

            "Idempotency-Key": str(uuid.uuid4()),

        },

        timeout=15,

    )
```

```
    return resp.status_code, resp.json()

# contoh pemakaian

code, body = send_callback(

    "01J1ZXK7ABCDE",

    {

        "status": "error",

        "error_code": "out_of_stock",

        "error_message": "Unit sudah disewa tanggal tersebut",

        "data": {"alternative_date": "24 April 2026"},

    },

)

print(code, body)
```

---

## 10. Retry, idempotency, dan timing

- **Retry:** ulang maks 4× dengan backoff 2s/4s/8s/16s untuk error transient (5xx, network). **Jangan retry** 4xx kecuali `schema_invalid` yang sudah diperbaiki.
  - **Idempotency-Key:** WAJIB kalau Anda mungkin retry. EMBAN cache response selama 24 jam — retry dengan key yang sama mengembalikan response cached tanpa double-deliver.
  - **Callback timeout:** tenant bisa set `response.callback_timeout_ms` di action definition. Kalau Tenant IT tidak callback dalam window itu, EMBAN akan mengirim pesan `callback_timeout_message` ke customer dan menandai submission timeout. Setelah itu callback late → 409 `invalid_submission_state`.
  - **Ordering:** tidak ada jaminan FIFO antar submission berbeda. Setiap submission ditangani independen.
-

## 11. Checklist pre-go-live

- ☐ Action definition di-register (via flow onboarding tenant) dengan `endpoint.auth.secret_env` valid + secret sudah ada di env EMBAN.
- ☐ Outbound handler di sisi Tenant IT menyimpan `submission_id` dari request pertama EMBAN.
- ☐ Callback sender pakai HMAC dengan secret yang sama.
- ☐ Semua `error_templates` punya entry `default`.
- ☐ File URL di HTTPS + hostname ada di `tenants.file_download_allowlist`.
- ☐ Retry + idempotency-key diimplementasikan.
- ☐ Smoke-test di staging: success path, error path, timeout path, file delivery success + rejected.

Lanjut: **#4 Ticketing API** — endpoint balasan dari sistem ticketing internal Tenant IT (CS role "forward to human").