MT-Facturation API Process Map (Chronological)

Last Updated: 2026-02-20 12:12 UTC

Base Prefix: /api/v1

This document maps all platform-supported API workflows in chronological business order, from setup to collections.

Command Syntax Defaults

- Base URL used in generated command syntax: http://192.168.10.9:8000
- Protected endpoint auth header syntax: Authorization: Bearer launch-smoke:admin,ops
- Write endpoints include Content-Type: application/json and placeholder -d
- Idempotent endpoints include header Idempotency-Key: demo-key-12345678

Global Rules (Apply to All Processes)

- Protected endpoints need auth (Authorization or X-Actor-Id).
- Public client endpoints are under /landing/*.
- Idempotency is required on critical writes:
- /landing/submit/new
- /landing/submit/plan-change
- /billing/runs
- /collections/payments
- /collections/invoices/{invoice_id}/approve-paid
- CIN is the client identity reference for landing self-service flows.
- Subscriber endpoints exist, but in normal operations subscriber lifecycle is handled by contract provisioning (no standalone subscriber calls needed).

P0) Environment and Access Check

Purpose: verify API availability and credentials before business operations.

1. GET /api/v1/health

```
curl "http://192.168.10.9:8000/api/v1/health"
```

2. GET /api/v1/me

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" "http://192.168.10.9:8000/api/v1/me"
```

3. Optional role check: GET /api/v1/admin/ping

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" "http://192.168.10.9:8000/api/v1/admin/ping"
```

P1) Catalog Foundation (Offer Setup)

Purpose: define services/offers before any contract can be created.

1. Create offer: POST /api/v1/offers

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/offers"
```

2. Validate catalog grouping:

- GET /api/v1/offer-categories

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offer-categories"
```

- optional legacy alias: GET /api/v1/offer-families

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offer-families"
```

3. Verify individual offer: GET /api/v1/offers/{offer_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offers/<offer_id>"
```

Output:

- reusable offer_id values for provisioning and landing flows.

P2) Offer Maintenance (Lifecycle of Offers)

Purpose: maintain commercial catalog over time.

1. Update offer: PUT /api/v1/offers/{offer_id}

```
curl -X PUT -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/offers/<offer_id>"
```

2. Re-check offer/grouping:

- GET /api/v1/offers/{offer_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offers/<offer_id>"
```

- GET /api/v1/offer-categories

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offer-categories"
```

3. Optional cleanup: DELETE /api/v1/offers/{offer_id}

```
curl -X DELETE -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/offers/<offer_id>"
```

Guardrail:

- delete is blocked if contract history exists (409 offer_delete_blocked).

P3) Client Master Data Setup and Maintenance

Purpose: create and maintain client records for internal operations.

1. Create client: POST /api/v1/customers

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/customers"
```

2. List/search clients: GET /api/v1/customers

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" "http://192.168.10.9:8000/api/v1/customers"
```

3. Get specific client: GET /api/v1/customers/{client_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/customers/<client_id>"
```

4. Update client: PUT /api/v1/customers/{client_id}

```
curl -X PUT -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/customers/<client_id>"
```

5. Optional delete (safe-only): DELETE /api/v1/customers/{client_id}

```
curl -X DELETE -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/customers/<client_id>"
```

Guardrails:

- CIN uniqueness enforced.
- deletion blocked if contract/invoice/payment/collection history exists.

P4) Internal New Contract Provisioning

Purpose: internal team creates a new active contract while subscriber assignment is auto-managed.

1. Ensure offer_id and client_id exist (P1 + P3).

2. Provision contract:

- POST /api/v1/contracts/provision

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/contracts/provision"
```

Recommended body pattern:

```
{
"offer_id": "OFFER_ID",
"contract_start_date": "2026-02-20",
"commitment_months": 12,
"client_id": "CLIENT_ID",
"subscriber": {
"service_identifier": "+212612300000"
},
"provisioning_intent": "auto"
}
```

3. Verify result:

- GET /api/v1/contracts/{contract_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/contracts/<contract_id>"
```

- GET /api/v1/contracts/{contract_id}/audit

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/contracts/<contract_id>/audit"
```

P5) Internal Manual Contract Creation (Controlled Path)

Purpose: create contract via explicit IDs (advanced/manual path remains supported).

1. POST /api/v1/contracts

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/contracts"
```

2. Verify:

- GET /api/v1/contracts/{contract_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/contracts/<contract_id>"
```

- GET /api/v1/contracts/{contract_id}/audit

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/contracts/<contract_id>/audit"
```

Note:

- Use this only when manual control is required; for normal operations prefer provisioning flow (P4).

P6) Contract Lifecycle Management (Internal)

Purpose: run day-2 contract operations after activation.

1. List contracts: GET /api/v1/contracts

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" "http://192.168.10.9:8000/api/v1/contracts"
```

2. Update status: PUT /api/v1/contracts/{contract_id}/status

```
curl -X PUT -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/contracts/<contract_id>/status"
```

3. Update offer directly: PUT /api/v1/contracts/{contract_id}/offer

```
curl -X PUT -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/contracts/<contract_id>/offer"
```

4. Audit trail read: GET /api/v1/contracts/{contract_id}/audit

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/contracts/<contract_id>/audit"
```

P7) Client Landing Bootstrap and Drafting

Purpose: initialize public client journey state.

1. Load landing metadata: GET /api/v1/landing/bootstrap

```
curl "http://192.168.10.9:8000/api/v1/landing/bootstrap"
```

2. Save draft: POST /api/v1/landing/drafts

```
curl -X POST -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/drafts"
```

3. Restore draft: GET /api/v1/landing/drafts/{draft_id}

```
curl "http://192.168.10.9:8000/api/v1/landing/drafts/<draft_id>"
```

4. Update draft: PUT /api/v1/landing/drafts/{draft_id}

```
curl -X PUT -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/drafts/<draft_id>"
```

P8) Client New Subscription (Public) -> Contract PDF

Purpose: end user subscribes from landing, system auto-creates/updates client+contract records.

1. (Optional UI step) load offers from bootstrap (P7).

2. Submit new subscription:

- POST /api/v1/landing/submit/new

```
curl -X POST -H "Idempotency-Key: demo-key-12345678" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/landing/submit/new"
```

- header: Idempotency-Key

3. Read document_download_url from response.

4. Download contract PDF via returned URL (GET .../landing/contracts/{contract_id}/document?token=...).

Fallback if URL missing or needs refresh:

1. POST /api/v1/landing/contracts/{contract_id}/document-link with CIN

```
curl -X POST -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/contracts/<contract_id>/document-link"
```

2. GET returned document_download_url

P9) Client Upgrade/Downgrade (Public) -> Contract PDF

Purpose: existing client changes offer inside same service family via CIN-only flow.

1. Verify CIN:

- POST /api/v1/landing/clients/verify-cin

```
curl -X POST -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/clients/verify-cin"
```

2. Fetch active subscriptions + eligible offers:

- GET /api/v1/landing/clients/{cin}/subscriptions?lookup_token=...

```
curl "http://192.168.10.9:8000/api/v1/landing/clients/<cin>/subscriptions?lookup_token="
```

3. Submit plan change:

- POST /api/v1/landing/submit/plan-change

```
curl -X POST -H "Idempotency-Key: demo-key-12345678" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/landing/submit/plan-change"
```

- header: Idempotency-Key

4. Download updated contract PDF from document_download_url.

Important:

- same endpoint handles upgrade and downgrade.
- target offer determines direction.
- selected contract_start_date becomes the effective upgraded/downgraded contract start date.

P10) Internal Contract Document Retrieval (Any Time)

Purpose: retrieve contract document after internal updates.

1. Generate secure link by CIN ownership:

- POST /api/v1/landing/contracts/{contract_id}/document-link

```
curl -X POST -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/contracts/<contract_id>/document-link"
```

2. Download:

- GET /api/v1/landing/contracts/{contract_id}/document?token=...

```
curl "http://192.168.10.9:8000/api/v1/landing/contracts/<contract_id>/document?token="
```

P11) Monthly Billing Run

Purpose: issue monthly invoices from active billable contracts.

1. Trigger billing:

- POST /api/v1/billing/runs

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Idempotency-Key: demo-key-12345678"
-H "Content-Type: application/json" -d "<JSON_BODY>" "http://192.168.10.9:8000/api/v1/billing/runs"
```

- header: Idempotency-Key

2. Store result metadata:

- billing_run_id, invoice_ids, totals.

P12) Invoice Operations (Internal)

Purpose: inspect and distribute invoices from operator side.

1. List invoices:

- GET /api/v1/invoices

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" "http://192.168.10.9:8000/api/v1/invoices"
```

- optional filters:
- client_id
- service (mobile|internet|landline)
- offer_id

2. Invoice detail:

- GET /api/v1/invoices/{invoice_id}

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/invoices/<invoice_id>"
```

3. Download PDF:

- GET /api/v1/invoices/{invoice_id}/pdf

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/invoices/<invoice_id>/pdf"
```

P13) Client Billing Self-Service (Public)

Purpose: client checks billing and downloads invoice PDFs.

1. Verify CIN:

- POST /api/v1/landing/clients/verify-cin

```
curl -X POST -H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/landing/clients/verify-cin"
```

2. Retrieve invoice list:

- GET /api/v1/landing/clients/{cin}/invoices?lookup_token=...

```
curl "http://192.168.10.9:8000/api/v1/landing/clients/<cin>/invoices?lookup_token="
```

3. Download invoice PDF:

- GET /api/v1/landing/invoices/{invoice_id}/document?token=...

```
curl "http://192.168.10.9:8000/api/v1/landing/invoices/<invoice_id>/document?token="
```

P14) Payment Posting and Invoice Settlement

Purpose: record payments and update invoice/collections state.

Standard payment

1. POST /api/v1/collections/payments (with Idempotency-Key)

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Idempotency-Key: demo-key-12345678"
-H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/collections/payments"
```

2. Read allocation result:

- invoice status after payment
- outstanding amount
- allocation state (partial/full)

Operator full-settlement shortcut

1. POST /api/v1/collections/invoices/{invoice_id}/approve-paid (with Idempotency-Key)

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Idempotency-Key: demo-key-12345678"
-H "Content-Type: application/json" -d "<JSON_BODY>"
"http://192.168.10.9:8000/api/v1/collections/invoices/<invoice_id>/approve-paid"
```

2. System creates full outstanding payment and closes settlement path.

P15) Collections Case Lifecycle and Dunning Actions

Purpose: manage overdue debt recovery lifecycle.

1. Collections snapshot: GET /api/v1/collections/overview

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/collections/overview"
```

2. Case listing and filtering:

- GET /api/v1/collections/cases

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/collections/cases"
```

3. Case actions history:

- GET /api/v1/collections/cases/{case_id}/actions

```
curl -H "Authorization: Bearer launch-smoke:admin,ops"
"http://192.168.10.9:8000/api/v1/collections/cases/<case_id>/actions"
```

4. Update case status:

- PUT /api/v1/collections/cases/{case_id}/status

```
curl -X PUT -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/collections/cases/<case_id>/status"
```

5. Add manual action:

- POST /api/v1/collections/cases/{case_id}/actions

```
curl -X POST -H "Authorization: Bearer launch-smoke:admin,ops" -H "Content-Type: application/json" -d
"<JSON_BODY>" "http://192.168.10.9:8000/api/v1/collections/cases/<case_id>/actions"
```

6. Optional payment reconciliation view:

- GET /api/v1/collections/payments

```
curl -H "Authorization: Bearer launch-smoke:admin,ops" -H "Idempotency-Key: demo-key-12345678"
"http://192.168.10.9:8000/api/v1/collections/payments"
```

Practical Handover Notes

- For teammate integration, follow P0 -> P1 -> P3 -> P4/P8/P9 -> P11 -> P12/P13 -> P14 -> P15.
- For contract change operations, prefer P9 (landing/submit/plan-change) over direct offer update if you need full client-journey behavior and regenerated documents.
- Keep idempotency keys unique per business operation attempt.