

# Ingestion Service — Unified Testing Guide

Step-by-step instructions to test all ingestion service features: normal uploads, correction file uploads, internal APIs, exposed customer API, and error handling.

## 1. Prerequisites

### 1.1 Required Software

- Java 17+
- Maven 3.8+
- MongoDB running locally on localhost:27017
- Postman (or curl) for API testing

### 1.2 Start MongoDB (if not running)

Platform	Command
Windows	<code>mongod</code>
Linux/Mac	<code>mongod --dbpath /data/db</code>
Docker	<code>docker run -d -p 27017:27017 mongo:7</code>

### 1.3 Start Ingestion Service

```
cd MyPolicy-Backend/ingestion-service mvn spring-boot:run
```

Verify: Service starts on <http://localhost:8082>

## 1.4 Sample Files

- **Normal upload:** Datasets/Auto\_Insurance.csv (or any .csv/.xls/.xlsx)
  - **Correction:** Auto\_Insurance\_correction.csv (policyNumber, panNumber, and corrected columns)
- 

## 2. Base URL & Conventions

Item	Value
Base URL	<a href="http://localhost:8082">http://localhost:8082</a>
Port	8082
API Key (exposed API)	mypolicy-customer-api-key-dev
File rules	Allowed: .csv, .xls, .xlsx   Max: 50 MB   Encoding: UTF-8

---

## 3. Part A: Internal APIs (No API Key)

Used by BFF, Processing Service, or internal tools. No X-API-Key required.

### 3.1 Test: Normal File Upload

**Purpose:** Upload a policy file for initial ingestion (INSERT flow).

**Postman:**

1. Method: POST

2. URL: `http://localhost:8082/api/v1/ingestion/upload`

3. Body: form-data — file [File], insurerId [Text]: HDFC\_LIFE, uploadedBy [Text]: test-user

**curl (Windows):**

```
curl -X POST http://localhost:8082/api/v1/ingestion/upload ^ -F  
"file=@Datasets/Auto_Insurance.csv" ^ -F "insurerId=HDFC_LIFE" ^ -F "uploadedBy=test-user"
```

**Expected Response:** [201 Created](#)

```
{ "jobId": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx", "status": "UPLOADED" }
```

Save the `jobId` for steps 3.4 and 3.5.

### 3.2 Test: Correction File Upload (via Parameter)

**Purpose:** Upload a delta/correction file for bulk updates (UPDATE flow).

Add form-data key: `fileType` [Text] → `correction`

Key	Type	Value
-----	------	-------

file	File	Auto_Insurance_correction.csv
insurerId	Text	HDFC_LIFE
uploadedBy	Text	data-ops-user
fileType	Text	correction

### 3.3 Test: Correction File Upload (via Filename Convention)

Same as 3.2, but detected automatically when filename contains `_correction` (e.g., `Auto_Insurance_correction.csv`). No fileType param needed.

### 3.4 Test: Get Job Status

**Method:** GET | **URL:** `http://localhost:8082/api/v1/ingestion/status/{ jobId }`

**Expected Response:** 200 OK with jobId, status, processedRecords, totalRecords, filePath, insurerId, fileType, createdAt, updatedAt.

### 3.5 Test: Update Progress (Processing Service)

**Prerequisite:** Job must be in PROCESSING state (see 3.6).

**Method:** PATCH | **URL:** `http://localhost:8082/api/v1/ingestion/{ jobId }/progress`

**Headers:** Content-Type: application/json

**Body:**

```
{ "processedRecordsDelta": 10 }
```

**Expected:** 204 No Content

### 3.6 Test: Update Status (Processing Service)

**Transitions:** UPLOADED → PROCESSING → COMPLETED | FAILED

**Method:** PATCH | **URL:** `http://localhost:8082/api/v1/ingestion/{jobId}/status`

**Body examples:**

```
{ "status": "PROCESSING" } { "status": "COMPLETED" } { "status": "FAILED", "failureReason":  
"Validation failed: invalid date in row 5" }
```

**Expected:** 204 No Content

## 4. Part B: Exposed / Public API (X-API-Key Required)

Used by external customers. Requires valid X-API-Key header.

### 4.1 Test: Public Upload (Normal)

**Headers:** X-API-Key: mypolicy-customer-api-key-dev

**URL:** POST `http://localhost:8082/api/public/v1/ingestion/upload`

**Body:** form-data — file, insurerId: HDFC\_LIFE, uploadedBy: customer-user-001

**Expected:** 201 Created (same as 3.1)

## 4.2 Test: Public Upload (Correction)

Same as 4.1, but add `fileType=correction` or use `*_correction.csv` filename.

## 4.3 Test: Public Job Status

**Method:** GET | **URL:** `http://localhost:8082/api/public/v1/ingestion/status/{ jobId }`

**Headers:** X-API-Key: mypolicy-customer-api-key-dev

## 4.4 Test: Auth Failure — Missing API Key

Same request as 4.1 but **do not** add X-API-Key header.

**Expected:** `401 Unauthorized` — `{"error": "Missing X-API-Key header"}`

## 4.5 Test: Auth Failure — Invalid API Key

Use X-API-Key: wrong-key-123

**Expected:** `401 Unauthorized` — `{"error": "Invalid API key"}`

# 5. Recommended Test Sequences

## Sequence A: Full Normal Ingestion Flow

1. Upload file (3.1) → Get jobId

2. Get status (3.4) → Verify UPLOADED, fileType=normal
3. Update status (3.6) → PROCESSING
4. Update progress (3.5) → processedRecordsDelta: 50
5. Get status (3.4) → Verify processedRecords updated
6. Update status (3.6) → COMPLETED
7. Get status (3.4) → Verify final COMPLETED

## **Sequence B: Correction File Flow**

1. Create correction CSV (Policy Number, PAN, mobileNumber, customerName, etc.)
2. Upload correction (3.2 or 3.3) → Get jobId
3. Get status (3.4) → Verify fileType=correction
4. Trigger Processing Service with isCorrection=true

## **Sequence C: Exposed API End-to-End**

1. Public upload (4.1) → Get jobId
2. Public status (4.3) → Verify UPLOADED
3. Optional: Test auth failures (4.4, 4.5)

---

## 6. Error Responses Reference

Code	Errors
400	File is empty or missing; Invalid file type; File size exceeds 50MB; Job not found; processedRecordsDelta must be positive; status is required
401	Missing X-API-Key header; Invalid API key
404	Job not found
409	Invalid transition; Cannot update progress (job must be PROCESSING); No transitions from terminal state
500/503	Database or internal errors

---

## 7. Postman Tips

- Environment: baseUrl = http://localhost:8082, jobId = (from response), apiKey = mypolicy-customer-api-key-dev
- Tests tab to save jobId: `pm.environment.set("jobId", pm.response.json().jobId);`

---

## 8. Configuration

```
server.port=8082 spring.data.mongodb.host=localhost spring.data.mongodb.port=27017  
spring.data.mongodb.database=mypolicy_ingestion_db  
ingestion.customer.api-keys=mypolicy-customer-api-key-dev ingestion.storage.path=storage/ingestion
```

---

## 9. Troubleshooting

- **503 / Database error:** Ensure MongoDB is running on 27017
- **401 on /api/public:** Add X-API-Key header with correct value
- **File not found:** Check ingestion.storage.path
- **Port in use:** Change server.port or stop conflicting process

**Document:** Ingestion Service Unified Testing Guide | MyPolicy Backend

**Version:** 1.0