



Messaging API concepts

How messaging APIs are usually designed, so you can evaluate any provider's interface or plan your own. Wise Network does not currently operate a messaging API.

Educational guide. General information about messaging integrations. It does not describe a live Wise Network API, and examples are illustrative only.

Online version: <https://wisenetwork.co/support/api-planning/> · Content updated 28 October 2026

Before you read

The requests and responses on this page use the host name `api.wisenetwork.co` and invented field names to show common patterns. That host name is a placeholder: there is no Wise Network API behind it, the examples are not a Wise Network API contract, and no Wise Network endpoint accepts them.

HTTP and HTTPS basics

An HTTP API is a set of URLs (endpoints) that accept requests with a method such as GET (read), POST (create), PUT or PATCH (change) and DELETE (remove). The server answers with a status code and, usually, a body. HTTPS is HTTP inside a TLS-encrypted connection; credentials and message text must only ever travel over HTTPS. The standard reference for HTTP methods and status codes is [RFC 9110](#).

HTTP status code families

Range	Meaning	What the client should do
2xx	Success. 200 OK, 201 Created or 202 Accepted.	Store the identifiers returned.
4xx	The request was wrong or not allowed, for example 400, 401, 403, 404, 409, 422.	Fix the request. Do not retry it unchanged.
429	Too many requests: a rate limit was reached.	Wait, then retry more slowly.
5xx	A server-side or temporary failure.	Retry later with increasing delays, using an idempotency key.

Data formats: JSON and XML

JSON ([RFC 8259](#)) represents data as objects of named fields and is the most common format for new APIs. XML represents the same data as nested elements and is still used by many established telecom and enterprise systems, sometimes inside SOAP envelopes. Choose one per integration and follow the provider's schema exactly, including character encoding (UTF-8 is standard), date formats and number formats.

Illustrative example: not a Wise Network API request

```
POST /v1/messages HTTP/1.1
Host: api.wisenetwork.co
Authorization: Bearer <access-token>
Content-Type: application/json
Idempotency-Key: 7f6c2a10-example
```

```
{
  "from": "ExampleCo",
  "to": ["+12025550143"],
  "text": "Your code is 123456",
  "client_ref": "order-1001"
}
```

Illustrative example: the same data as XML, not a Wise Network API request

```
<?xml version="1.0" encoding="UTF-8"?>
<message>
  <from>ExampleCo</from>
  <to>+12025550143</to>
  <text>Your code is 123456</text>
  <clientRef>order-1001</clientRef>
</message>
```

Phone numbers are normally written in the international E.164 format: a plus sign, the country code and the national number, without spaces ([ITU-T E.164](#)). The number above is a fictional test number.

Authentication and access control

An API must know who is calling. Common methods, from simplest to strongest:

Common API authentication methods

Method	How it works	Points to check
Username and password (HTTP Basic)	Credentials sent in the Authorization header on each request.	HTTPS only. Rotate passwords. Never put them in URLs.
API key	A long random key sent in a header.	Store on the server, not in apps or web pages. Support several keys for rotation.
Short-lived token	A key or client credentials are exchanged for a token that expires, for example under OAuth 2.0.	Handle expiry and renewal; keep the long-term secret private.
Request signing	Each request carries a signature (for example HMAC-SHA256) over its content and time.	Protects against tampering and replay; needs clock synchronisation.
Source IP allow-list	Only listed public IP addresses may connect.	An extra layer, not a replacement for credentials. See IP access .

Whatever the method, keep secrets out of query strings: URLs are recorded in server logs, proxies and browser history.

Common messaging operations

Message submission

A submission names the sender ID, one or more destinations and the text, and may add a client reference, a validity period and a request for a delivery report. The response returns an identifier for each accepted message. Acceptance means the provider has taken the message for processing; it does not mean the handset received it.

Illustrative example: not a Wise Network API request

```
HTTP/1.1 202 Accepted
Content-Type: application/json

{
  "messages": [
    { "id": "msg-0001", "to": "+12025550143", "status": "accepted" }
  ]
}
```

Scheduling and cancellation

A scheduled message carries a send time. Always state the time zone, ideally as a full timestamp with offset in the [RFC 3339](#) format, for example 2026-11-02T09:00:00+02:00. Cancelling a scheduled message uses the identifier returned when it was created, and is only possible before sending starts.

Status enquiry

A status request returns the current state of a message or batch, using the identifier from the submission. Status enquiries are useful for reconciliation, but delivery reports or webhooks are better for tracking large volumes. See [delivery reports and errors](#).

Balance enquiry

Prepaid accounts may offer a balance request that returns the remaining credit and its currency or unit. Check how often it may be called and whether it includes messages still being processed.

Coverage and pricing lists

A coverage request or download lists the destination countries and networks an account can reach, often with the mobile country and network codes (MCC and MNC) and prices. Check its format (CSV, Excel or JSON), how often it changes and the date it applies from.

Message history and reports

History and report downloads list past messages for a period. Agree the time zone, the fields included, the file format, how long records are kept, paging for large results and whether message text is included or masked.

Bulk and personalized sending

Bulk sending submits the same text to many destinations, either as a list in one request or as an uploaded file. Personalized sending merges a template with values for each recipient, for example a name or a code. Before importing recipients, confirm consent, the file format and columns, the maximum size, validation of numbers and how rejected rows are reported.

Illustrative example: a personalization file, not a Wise Network file format

```
phone,name,code
+12025550143,Alex,4821
+12025550188,Sam,9917
```

Reliability: retries, idempotency and limits

- **Idempotency.** Send a unique key with each submission so a retried request is not sent twice.
- **Retries.** Retry only temporary failures (timeouts, 429 and 5xx), with increasing delays and a maximum number of attempts.
- **Rate limits.** Respect the agreed requests or messages per second; a Retry-After header, when present, says how long to wait ([RFC 6585](#) defines status 429).
- **Paging.** Large lists are returned in pages; follow the next-page reference rather than guessing offsets.
- **Versioning.** Use the version stated in the URL or header, and plan for deprecation notices.

Questions to ask any API provider

- Which operations exist, and is there a test environment with test credentials?
- How are credentials issued, rotated and revoked?
- What are the rate limits, message length limits and accepted encodings?
- Which statuses and error codes exist, and what does each mean?
- How are delivery reports delivered, signed and retried?

- How long are message records kept, and is message content stored?

To discuss an integration with Wise Network, [contact our messaging team](#).

Questions? Email support@wisenetnetwork.co. This PDF is generated from the web page above; the online version is the reference.