

SpeedyIndex

API Documentation v2

<https://api.speedyindex.com>

RESPONSE CODES

| | |
|---|----------------------------------|
| 0 | Success |
| 1 | Insufficient balance / Not found |
| 2 | Validation error |
| 3 | Rate limit / Server error |

HTTP STATUS CODES

| | |
|-----|----------------------|
| 200 | OK |
| 401 | Unauthorized |
| 402 | Insufficient balance |
| 413 | Payload too large |
| 429 | Rate limit exceeded |

AUTHENTICATION

All endpoints require `Authorization: YOUR_API_KEY` header.

POST endpoints also require `Content-Type: application/json` .

Rate limit: 120 requests per minute per IP address

ACCOUNT

GET /v2/account

Check the balance

Returns your current balance across all services.

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|---------|--------|--|
| balance | object | balance information (tokens, indexer, checker) |

EXAMPLE

REQUEST

```
curl -H "Authorization: YOUR_API_KEY" https://api.speedyindex.com/v2/account
```

RESPONSE

```
{
  "code": 0,
  "balance": {
    "tokens": 5000,
    "indexer": 10014495,
    "checker": 100732
  }
}
```

ACCOUNT

POST /v2/account/invoice/create

Create an invoice to purchase tokens

Available packages: \$5 (5 000), \$20 (30 000), \$50 (120 000), \$100 (250 000), \$250 (700 000), \$500 (1 500 000 tokens).
PayPal: +10% fee. Yookassa: USD→RUB (1 USD = 100 RUB), requires email.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|---------------------|--------|---|
| <code>price</code> | number | package price in USD (5, 20, 50, 100, 250, 500) |
| <code>method</code> | string | payment method ('crypto', 'paypal', 'yookassa') |
| <code>email</code> | string | (required for yookassa) email for payment receipt |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|----------------------------|--------|--------------------------|
| <code>code</code> | number | status code |
| <code>result</code> | string | payment invoice URL |
| <code>tokens_amount</code> | number | tokens to be credited |
| <code>price</code> | number | final price (USD or RUB) |
| <code>currency</code> | string | 'USD' or 'RUB' |

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"price":50,"method":"crypto"}' https://api.speedyindex.com/v2/account/invoice/create
```

RESPONSE

```
{
  "code": 0,
  "result": "https://pay.cryptocloud.plus/LJQ18AI1?lang=en",
  "tokens_amount": 120000,
  "price": 50,
  "currency": "USD"
}
```

TASKS (INDEXING & CHECKING)

POST /v2/task/{SEARCH_ENGINE}/{TASK_TYPE}/create

Create a new indexing or checking task

Currently only the `pay_per_indexed: true` mode is available for Google indexing tasks. Standard Google indexing (without `pay_per_indexed`) is temporarily unavailable.

SEARCH_ENGINE: google, yandex, bing. TASK_TYPE: indexer, checker.

Note: bing supports checker only.

Google indexer: pass `pay_per_indexed: true` — 100 tokens/URL, 100% refund for unindexed links after 7 days.

Yandex indexer: Pay per Indexed is always enabled by default.

Drip Feed (google/indexer only): set `drip_feed: true` and `drip_feed_days` (2–30) to spread URL submissions across multiple days. URLs are split into equal daily batches. Number of URLs must be \geq `drip_feed_days`. Compatible with `pay_per_indexed`.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|------------------------------|---|---|
| <code>title</code> | string (optional) | task name |
| <code>urls</code> | string[] | array of URLs (max 10,000) |
| <code>pay_per_indexed</code> | boolean (optional) | true for Pay per Indexed (google/indexer; always true for yandex/indexer). 100 tokens/URL |
| <code>drip_feed</code> | boolean (optional) | true to create a Drip Feed task (google/indexer only) |
| <code>drip_feed_days</code> | number (optional, required if <code>drip_feed=true</code>) | number of days to spread submissions (2–30, must be \leq urls count) |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|------------------------------|--------------------|--|
| <code>code</code> | number | status code (0: success, 1: insufficient balance, 2: validation error) |
| <code>task_id</code> | string | unique task identifier |
| <code>type</code> | string | task type (e.g. 'google/indexer') |
| <code>pay_per_indexed</code> | boolean (optional) | true if Pay per Indexed task |
| <code>drip_feed</code> | boolean (optional) | true if Drip Feed task |
| <code>drip_feed_days</code> | number (optional) | number of drip feed days |

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"title":"my task","urls":["https://example.com/page1","https://example.com/page2"],"pay_per_indexed":true}' https://api.speedyindex.com/v2/task/google/indexer/create
```

RESPONSE

```
{
  "code": 0,
  "task_id": "6609d023a3188540f09fec6c",
  "type": "google/indexer",
  "pay_per_indexed": true
}
```

TASKS (INDEXING & CHECKING)

GET /v2/task/{SEARCH_ENGINE}/{TASK_TYPE}/list/{PAGE}

Get list of tasks

Each page contains up to 1000 tasks. Numbering starts from 0.
Sorted from newest to oldest.

Filter (google/indexer only): use ?filter= query parameter to filter by task type:

- ?filter=standard — only standard submit tasks
- ?filter=pay_per_indexed — only Pay per Indexed tasks
- ?filter=drip_feed — only Drip Feed tasks (includes both standard and PPI drip feeds)
- no filter — standard + Pay per Indexed tasks (default, without Drip Feed)

Drip Feed sub-filter: combine with &pay_per_indexed=true or &pay_per_indexed=false to filter drip feed tasks by type.

Drip Feed response includes: days, days_passed, is_stopped, pay_per_indexed, child_task_ids.

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|-----------|--------|--|
| code | number | status code |
| page | number | current page number (0-based) |
| last_page | number | last page number |
| result | Task[] | array of task objects (PPI tasks include extra fields) |

EXAMPLE

REQUEST

```
curl -H "Authorization: YOUR_API_KEY" https://api.speedyindex.com/v2/task/google/indexer/list/0
```

RESPONSE

```
{
  "code": 0,
  "page": 0,
  "last_page": 0,
  "result": [
    {
      "id": "65f8c7315752853b9171860a",
      "size": 690,
      "processed_count": 690,
      "indexed_count": 279,
      "title": "my_urls.txt",
      "is_completed": true,
      "type": "google/indexer",
      "created_at": "2024-03-18T22:58:56.901Z"
    }
  ]
}
```

TASKS (INDEXING & CHECKING)

POST /v2/task/{SEARCH_ENGINE}/{TASK_TYPE}/status

Get status of tasks

Provide either task_id or task_ids (one is required).

Drip Feed (google/indexer): when task_id is a Drip Feed task, the response includes days, days_passed, is_stopped, pay_per_indexed, child_task_ids, and a children array with per-day child task details.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|-----------|----------|--|
| task_id | string | single task ID (required if task_ids not provided) |
| task_ids | string[] | array of task IDs, max 1000 (required if task_id not provided) |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|-----------------|----------------|---|
| code | number | status code |
| result | Task or Task[] | task object(s) with status |
| pay_per_indexed | boolean | true for PPI tasks ⚡ |
| status | string | task status ⚡ |
| tokens_charged | number | tokens charged ⚡ |
| tokens_refunded | number | tokens refunded ⚡ |
| check_after | string | ISO 8601 timestamp, check date ⚡ |
| children | Task[] | child tasks (Drip Feed single task_id only) 🔄 |

⚡ = Pay per Indexed tasks only | 🔄 = Drip Feed tasks only

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"task_id":"65f8c7305759855b9171860a"}' https://api.speedyindex.com/v2/task/google/indexer/status
```

RESPONSE

```
{
  "code": 0,
  "result": {
    "id": "65f8c7305759855b9171860a",
    "size": 690,
    "processed_count": 690,
    "indexed_count": 279,
    "is_completed": true,
    "title": "my_urls.txt",
    "type": "google/indexer",
    "created_at": "2024-03-18T22:58:56.901Z",
    "pay_per_indexed": true,
    "status": "completed",
    "tokens_charged": 69000,
    "tokens_refunded": 41100,
    "check_after": "2024-03-25T22:58:56.901Z"
  }
}
```

TASKS (INDEXING & CHECKING)

POST /v2/task/{SEARCH_ENGINE}/{TASK_TYPE}/report

Download task report

Returns indexed and unindexed link lists for the task.

Drip Feed (google/indexer): when task_id is a Drip Feed task, indexed/unindexed links are aggregated across all child tasks.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|----------------------|--------|------------------------|
| <code>task_id</code> | string | unique task identifier |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|------------------------------|--------------------|----------------------------|
| <code>id</code> | string | task identifier |
| <code>size</code> | number | total URLs |
| <code>processed_count</code> | number | processed URLs |
| <code>indexed_links</code> | string[] | indexed URLs |
| <code>unindexed_links</code> | string[] | unindexed URLs |
| <code>type</code> | string | task type |
| <code>title</code> | string | task title |
| <code>created_at</code> | string | ISO 8601 timestamp |
| <code>pay_per_indexed</code> | boolean (optional) | PPI flag ⚡ |
| <code>tokens_charged</code> | number (optional) | tokens charged ⚡ |
| <code>tokens_refunded</code> | number (optional) | tokens refunded ⚡ |
| <code>drip_feed</code> | boolean (optional) | true for Drip Feed tasks 🔄 |
| <code>days</code> | number (optional) | total drip feed days 🔄 |
| <code>days_passed</code> | number (optional) | days elapsed 🔄 |

⚡ = Pay per Indexed tasks only | 🔄 = Drip Feed tasks only

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"task_id":"653278d8a2b987d72b36f2a2"}' https://api.speedyindex.com/v2/task/google/indexer/report
```

RESPONSE

```
{
  "code": 0,
  "result": {
    "id": "653278d8a2b987d72b36f2a2",
    "size": 3,
    "processed_count": 3,
    "indexed_links": [
      "https://example.com/page1"
    ],
    "unindexed_links": [
      "https://example.com/page2"
    ],
    "title": "test",
    "type": "google/indexer",
    "created_at": "2023-10-20T12:55:52.031Z",
    "pay_per_indexed": true,
    "tokens_charged": 300,
    "tokens_refunded": 100
  }
}
```

TASKS (INDEXING & CHECKING)

POST /v2/task/{SEARCH_ENGINE}/{TASK_TYPE}/fullreport

Download full task report with titles

Same as report, but indexed links include page titles.

Drip Feed (google/indexer): when task_id is a Drip Feed task, data is aggregated across all child tasks.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|----------------------|--------|------------------------|
| <code>task_id</code> | string | unique task identifier |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|------------------------------|--------------------|---|
| <code>id</code> | string | task identifier |
| <code>size</code> | number | total URLs |
| <code>processed_count</code> | number | processed URLs |
| <code>indexed_links</code> | object[] | [{url, title}] |
| <code>unindexed_links</code> | object[] | [{url, error_code}]; PPI / Drip Feed: [{url}] |
| <code>type</code> | string | task type |
| <code>title</code> | string | task title |
| <code>created_at</code> | string | ISO 8601 timestamp |
| <code>pay_per_indexed</code> | boolean (optional) | PPI flag ⚡ |
| <code>tokens_charged</code> | number (optional) | tokens charged ⚡ |
| <code>tokens_refunded</code> | number (optional) | tokens refunded ⚡ |
| <code>drip_feed</code> | boolean (optional) | true for Drip Feed tasks 🔄 |
| <code>days</code> | number (optional) | total drip feed days 🔄 |
| <code>days_passed</code> | number (optional) | days elapsed 🔄 |

⚡ = Pay per Indexed tasks only | 🔄 = Drip Feed tasks only

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"task_id":"653278d8a2b987d72b36f2a2"}' https://api.speedyindex.com/v2/task/google/indexer/fullreport
```

RESPONSE

```
{
  "code": 0,
  "result": {
    "id": "653278d8a2b987d72b36f2a2",
    "size": 3,
    "processed_count": 3,
    "indexed_links": [
      {
        "url": "https://example.com/page1",
        "title": "Page Title"
      }
    ],
    "unindexed_links": [
      {
        "url": "https://example.com/page2"
      }
    ],
    "title": "test",
    "type": "google/indexer",
    "created_at": "2023-10-20T12:55:52.031Z",
    "pay_per_indexed": true,
    "tokens_charged": 300,
    "tokens_refunded": 100
  }
}
```

BACKLINK CHECKER

POST /v2/task/backlink/checker/create

Create a backlink check task

Check donor pages for backlinks to your target domains.

Provide donor URLs and target domains to check.

Target domains limit: 50 per task.

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|-----------------------------|-------------------|--|
| <code>title</code> | string (optional) | task name |
| <code>urls</code> | string[] | donor URLs to check (max 10,000) |
| <code>target_domains</code> | string[] | target domains (e.g. ['mydomain.com']), max 50 |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|----------------------|--------|--|
| <code>code</code> | number | status code (0: success, 1: insufficient balance, 2: validation error) |
| <code>task_id</code> | string | unique task identifier |
| <code>type</code> | string | 'backlink/checker' |

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"title":"backlink check","urls":["https://donor1.com/page","https://donor2.com/page"],"target_domains":["mydomain.com']}' https://api.speedyindex.com/v2/task/backlink/checker/create
```

RESPONSE

```
{
  "code": 0,
  "task_id": "680abc123def456",
  "type": "backlink/checker"
}
```

BACKLINK CHECKER

POST /v2/task/backlink/checker/status

Get status of backlink check tasks

Provide either task_id or task_ids (one is required).

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|-----------|----------|--|
| task_id | string | single task ID (required if task_ids not provided) |
| task_ids | string[] | array of task IDs, max 1000 (required if task_id not provided) |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|--------|----------------|---|
| code | number | status code |
| result | Task or Task[] | task object(s) with size, processed_count, target_domains, is_completed |

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"task_id":"680abc123def456"}' https://api.speedyindex.com/v2/task/backlink/checker/status
```

RESPONSE

```
{
  "code": 0,
  "result": {
    "id": "680abc123def456",
    "size": 100,
    "processed_count": 100,
    "title": "backlink check",
    "target_domains": [
      "mydomain.com"
    ],
    "is_completed": true,
    "type": "backlink/checker",
    "created_at": "2024-10-20T12:55:52.031Z"
  }
}
```

BACKLINK CHECKER

POST /v2/task/backlink/checker/report

Download backlink check report

Detailed backlink analysis per donor URL: DA, spam score, indexing status, target links.
Supports pagination (max 100 per page).

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|----------------------|-------------------|----------------------------|
| <code>task_id</code> | string | unique task identifier |
| <code>page</code> | number (optional) | page number, starts from 0 |
| <code>limit</code> | number (optional) | items per page, max 100 |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|------------------------------|----------|------------------------|
| <code>id</code> | string | task identifier |
| <code>size</code> | number | total donor URLs |
| <code>processed_count</code> | number | processed count |
| <code>donors</code> | object[] | donor analysis results |
| <code>page</code> | number | current page |
| <code>last_page</code> | number | last page number |

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"task_id":"680abc123def456","page":0,"limit":50}' http://api.speedyindex.com/v2/task/backlink/checker/report
```

RESPONSE

```
{
  "code": 0,
  "result": {
    "id": "680abc123def456",
    "size": 2,
    "processed_count": 2,
    "is_completed": true,
    "page": 0,
    "last_page": 0,
    "donors": [
      {
        "donor_url": "https://donor1.com/page",
        "processed": true,
        "http_status": 200,
        "moz_domain_authority": 45,
        "moz_spam_score": 3,
        "target_links": [
          {
            "url": "https://mydomain.com/article",
            "anchor": "read more",
            "nofollow": false
          }
        ]
      }
    ]
  }
}
```

SINGLE URL INDEXING

POST /v2/{SEARCH_ENGINE}/url

Index a single link

Submit a single URL for indexing.

SEARCH_ENGINE: google, yandex.

Optionally pass `pay_per_indexed: true` (Google only) — costs 100 tokens/URL with 100% refund for unindexed links after 7 days.

Without this flag, standard indexing credits are used (1 credit/URL).

REQUEST BODY

| PARAMETER | TYPE | DESCRIPTION |
|------------------------------|--------------------|---|
| <code>url</code> | string | URL to index |
| <code>pay_per_indexed</code> | boolean (optional) | true for Pay per Indexed mode (Google only). 100 tokens/URL, 100% refund if not indexed |

RESPONSE

| FIELD | TYPE | DESCRIPTION |
|------------------------------|--------------------|---|
| <code>code</code> | number | status code (0: success, 1: insufficient balance, 2: invalid URL) |
| <code>task_id</code> | string (optional) | task identifier (returned when <code>pay_per_indexed: true</code>) ⚡ |
| <code>pay_per_indexed</code> | boolean (optional) | true if Pay per Indexed mode ⚡ |

⚡ = Pay per Indexed only

EXAMPLE

REQUEST

```
curl -X POST -H "Authorization: YOUR_API_KEY" -H "Content-Type: application/json" -d '{"url":"https://example.com/page","pay_per_indexed":true}' https://api.speedyindex.com/v2/google/url
```

RESPONSE

```
{
  "code": 0,
  "task_id": "6654a1b2c3d4e5f6a7b8c9d0",
  "pay_per_indexed": true
}
```