

REMOTE MATCH API version 1

API approved by:

- PhenomeCentral
- GeneMatcher
- Decipher

OVERVIEW

Submit patient matching request:

HTTP POST to remote server: <base_remote_url>/mmapi/v1/match

For example: <https://phenomecentral.org/mmapi/v1/match>

Receive asynchronous response:

HTTP POST **from** remote server to: <base_origin_url>/mmapi/v1/matchResults

For example: <https://yourmatchmaker.org/mmapi/v1/matchResults>

Update previous request:

HTTP PUT to remote server: <base_remote_url>/mmapi/v1/match/<queryID>

For example: <https://phenomecentral.org/mmapi/v1/match/a32fa90vd>

Delete previous request:

HTTP DELETE to remote server: <base_remote_url>/mmapi/v1/match/<queryID>

For example: <https://phenomecentral.org/mmapi/v1/match/a32fa90vd>

SEARCH REQUEST

HTTP POST request to <base_remote_url>/mmapi/v1/match, with an application/json body with the following format:

Example

```
{
  "id" : <identifier>,
  "queryType" : "once"|"periodic",
  "label" : <identifier>,
}
```

```

"submitter" : {
  "name" : "First Last",
  "email" : <email address>,
  "institution" : "Some Hospital"
},

"gender" : "M"|"F",
"ageOfOnset" : <HPO code>,
"inheritanceMode" : <inheritance code>,

"disorders" : [
  "MIM:#####",
  "ORPHA#####",
  ...
],
"features" : [
  {
    "id" : <ICHPT code>,
    "observed" : "yes"|"no"|"unknown",
    "ageOfOnset" : "..."
  },
  ...
],
"genes" : [
  {
    "gene" : <gene name>|<ensembl gene ID>|<entrez gene ID>,
    "referenceName" : "1"|"2"|...|"X"|"Y",
    "start" : <number>,
    "end" : <number>,
    "referenceBases" : "A"|"ACG"|...,
    "alternateBases" : "A"|"ACG"|...,
    "zygosity" : <number>,
    "type" : <mutation type>,
    "assembly" : "NCBI36"|"GRCh37.p13"|"GRCh38.p1"
  },
  ...
]
}

```

ID

- **Mandatory**

- The internal identifier (obfuscated or not) that can be used by the originating system to reference the patient data.
- Transparent string, limited to 255 characters in utf-8.

Label

- [Optional](#)
- A name/identifier assigned by the user which can be used to reference the patient in a recognizable manner (in an email for example); it should not contain any *personally identifiable information*.
- Transparent string, limited to 255 characters in utf-8.

Query type

- [Optional](#)
- Accepted values:
 - `once`: only search once in the current database
 - `periodic`: repeat the search monthly until cancelled, reporting new and updated matches
- The default value is `once`
- If a system doesn't support the requested type, the `once` behavior is used

Submitter

- **Mandatory** if an email response is expected, [Optional](#) otherwise
- Consists of contact information of the person that submitted the search:
 - `email`: the email address where matches can be sent (**mandatory**); the values must conform to the [RFC 2822 address specification](#) mailbox format (no group)
 - `name`: the first and last name ([optional](#))
 - `institution`: human-readable institution name ([optional](#))
- **The contact information is for transmitting match results only, and may not be collected and/or used for any other purposes**

Gender

- [Optional](#)
- Accepted values: "M", "F"
- Any other value is treated as "unknown"

Age of onset

- [Optional](#)
- An HPO term identifier (HP:#####) associated with an age interval [as defined by the HPO](#)
 - "HP:0003577" (Congenital onset)
 - "HP:0011460" (Embryonal onset)
 - "HP:0011461" (Fetal onset)
 - "HP:0003623" (Neonatal onset)

- "HP:0003593" (Infantile onset)
- "HP:0011463" (Childhood onset)
- "HP:0003621" (Juvenile onset)
- "HP:0003581" (Adult onset)
 - "HP:0011462" (Young adult onset)
 - "HP:0003596" (Middle age onset)
 - "HP:0003584" (Late onset)

Inheritance Mode

- [Optional](#)
- Accepted values:
 - ad - Autosomal dominant
 - ar - Autosomal recessive
 - xd - X-linked dominant
 - xr - X-linked recessive
 - y1 - Y-linked
 - mi - Mitochondrial
 - ic - Isolated cases
 - un - Uncertain

Disorders

- [Optional](#)
- Is a list of OMIM (MIM:#####) or OrphaNet (ORPHA#####) identifiers, can be empty
NOTE: we may want to support other sources later.

Features

- It is **mandatory** to have at least one of these two: features, genes (having both is preferred)
- Is a **list of features** described by:
 - id: an ICHPT term identifier
 - observed: "yes" | "no" | "unknown"
 - ageOfOnset: same as the global age of onset described above ([optional](#); system which do not support this type of information per symptom should ignore it)
- More metadata can be later added to each feature if necessary.
- By default we shouldn't sent any features with the observed status (or value) "unknown"

Genes

- It is **mandatory** to have at least one of these two: features, genes (having both is preferred)
- Is a **list of possible causes** described by:
 - gene:
 - <gene symbol> from the [HGNC database](#) OR

- <ensembl gene ID> OR
 - <entrez gene ID>
- referenceName: "1", "2", ..., "22", "X", "Y"
- start: <number> (0-indexed)
- end: <number> (0-indexed exclusive)
- NOTE: The location (referenceName, start, end) is **optional**
- referenceBases: "A" | "ACG" | ..., VCF-style reference of at least one base (**optional**)
- alternateBases: "A" | "ACG" | ..., VCF-style alternate allele of at least one base (**optional**)
- zygosity: <number> (1 for heterozygous or hemizygous, 2 for homozygous; **optional**)
- type: the (**optional**) type of mutation, as a means to describe the broad category of cDNA effect predicted to result from a mutation to improve matchmaking, without disclosing the actual mutation:
 - TRUNCATING (e.g. stopgain, stoploss, startloss, frameshift indel)
 - ALTERING (e.g. missense, non-frameshift indel)
 - SPLICING
 - UTR (UTR3, UTR5)
 - INTRONIC
 - PROXIMAL (e.g. upstream, downstream)
 - OTHER (e.g. motif disruption, synonymous)
- assembly: reference assembly identifier, including patch number if relevant, of the form: <assembly>[.<patch>] (**mandatory**)
 - example valid values: "NCBI36", "GRCh37", "GRCh37.p13", "GRCh38", "GRCh38.p1"
 - If the patch is not provided, the assembly is assumed to represent the initial (unpatched) release of that assembly.

SEARCH RESULTS RESPONSE

Either a synchronous application/json response to a /match request, an asynchronous application/json HTTP POST request to <baseOriginURL>/mmapi/v1/matchResults, or a human-readable email sent to the user's email address.

The response to the search request looks like:

```
{
  "queryID" : <identifier>,
  "responseType" : "inline"|"asynchronous"|"email",
  "results" : [
    {
      "label" : <identifier>,
      "submitter" : {...},
      "gender" : "M"|"F",
      "ageOfOnset" : <number of years>|<number of months>|<age code>,
      "inheritanceMode" : <inheritance code>,
      "disorders" : [...],
      "features" : [...],
      "genes" : [...]
    },
    ...
  ]
}
```

Query identifier

- **Mandatory**
- Helps match the results to the original query for asynchronous results, and allows the submitter to manage the search submission
- This does not have to be the same as the id sent in the request since it represents how the remote host stores queries
- Transparent string, limited to 255 characters in utf-8.

Response type

- **Optional**
- inline responses are sent in the same response (the default value if the results property exists)
- asynchronous responses will be sent by the remote server at a later time, in a separate request to the origin server (the default value if the results property is missing)

- email responses will be sent by email directly to the contact email, in a human readable format

Results

- **Absent** for asynchronous results
- **Mandatory** for inline results, but can be empty
- Is a **list of matches**, where each match has the same format as the one described above for the query

Asynchronous responses

- Are sent through a HTTPS request to the originating server
- Same format as the synchronous response, but placed in an array and wrapped in an object, so that multiple responses can be sent at the same time

```
{ "responses":  
  [  
    {  
      "queryID" : <identifier>,  
      "results" : [...]  
    },  
    {  
      "queryID" : <identifier>,  
      "results" : [...]  
    },  
    ...  
  ]  
}
```

Email responses

The format of email responses is not restricted, and is left up to each site to implement in a user-friendly way.

SEARCH REQUEST UPDATE

HTTP PUT request to <baseRemoteURL>/mmapi/v1/match/<queryID>, with an application/json body with the same format as a search request:

Example

```
{  
  "id" : <identifier>,  
  ...  
}
```

A search request update is exactly the same as the search request with two differences:

- The HTTP method used is a PUT (as opposed to a POST).
- The URL includes the “queryID” that was returned in the search result response when the search was originally submitted is required.
- The “id” has to match the original “id” in the search request.
- All other information in the search request is replaced with a search request update.

The search request update returns a search results response.

SEARCH REQUEST DELETE

HTTP DELETE request to <baseRemoteURL>/mmapi/v1/match/<queryID>, with an application/json body with the following format:

Example

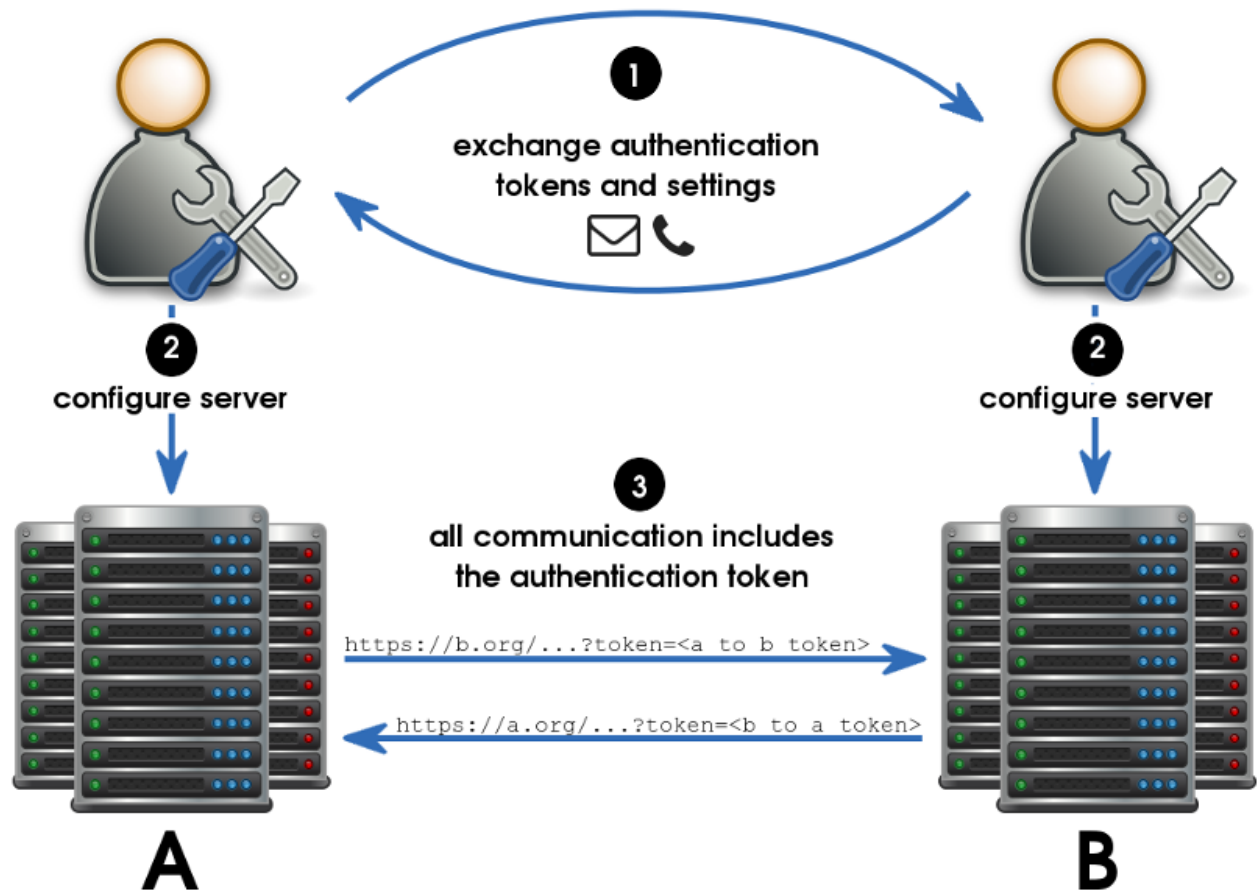
```
{  
  "id" : <identifier>  
}
```

A search request delete:

- The HTTP method used is a DELETE.
- The URL includes the “queryID” that was returned in the search result response when the search was originally submitted is required.
- The “id” has to match the original “id” in the search request.

The search request delete returns an OK (200) status to indicate that the search was deleted, nothing more.

SYSTEM REGISTRATION AND AUTHENTICATION



Allowing an external site (A) to submit match requests to a local site (B) requires manual acceptance and configuration of the A system into the B system. Once A is accepted by B, then **all requests sent by A are to be considered trusted**, which means that A must ensure that no fraudulent requests are going to be sent from it.

It is not mandatory for A to automatically also accept B's requests, this has to be negotiated between the two parties; the same protocol is then followed with A and B reversed.

- 1.a. When requesting access to B, the administrators of A must send to the administrators of B:
- a human readable **name** identifying A, to be presented to data owners on B when a search submitted by A matches patients in B and B wants to notify the data owners of a match (optional behavior that can be implemented by B)
 - an **authentication token** to be used by B in its requests to A, if any, to authenticate B to A, for example when sending back asynchronous match results; if the search agreement is mutual, this is also the token that B will use when submitting match requests to A

- a **base URL** to be used for requests, including scheme (https://), domain, eventual port, and path prefix (trailing / is optional); **B** will append /mmapi/v1/matchResults to this path when sending back asynchronous results, or, if **B** is to also be allowed to send queries to **A**, /mmapi/v1/match
- the preferred **response type** expected by **A**; if **B** does not support this type, then the two parties should negotiate what works best for them
 - the responseType is defined in the search response JSON format, with the possible values of "inline" | "asynchronous" | "email"

1.b. Upon acceptance of **A** as a trusted source of queries, the administrators of **B** must respond with:

- a suggested human readable **name** and **description** identifying **B**, to be presented to users of **A** as a possible remote site to search; **A** could ignore these and use their preferred name and description, but for consistency across systems **B**'s preference should be used
- an **authentication token** that must be used by **A** in the search requests
- a **base URL** to be used for requests, including scheme (https://), hostname, eventual port, and path prefix (trailing / is optional); **A** will append /mmapi/v1/match to this path when sending queries

For data security, HTTPS is mandatory, with a valid, globally acceptable certificate!

Since authentication tokens are the only means of identifying a site, this token should be unique to each remote site accepted by **B** to correctly determine where a query comes from. There is no imposed format for the token, except that it must be less than 255 characters long. A random 40-chars SHA1 key is recommended.

2.a. Configuring **B** to accept **A** as a trusted origin of remote searches requires that **B** stores somehow:

- the URL prefix for HTTP requests sent to **A** (for asynchronous responses)
- the token that **B** will send in its HTTP requests to **A**
- the token that **B** expects in the requests from **A**
- the response format preferred by **A**
- the name that identifies **A**

2.b. Configuring **A** to support **B** as a possible destination of remote searches requires that **A** stores somehow:

- the URL prefix for HTTP requests sent to **B**
- the token that must be sent to **B**
- the token that **A** expects in the HTTP requests from **B**
- the name that identifies **B**
- the description of **B**

3. Every HTTP request that **A** submits to **B** must contain the token that **B** told **A** to use in an URL parameter called key. For example:

```
https://matchmaker.phenomecentral.org/rest/remoteMatcher/mmapi/v1/match?key=854a439d278df4283bf5498ab020336cdc416a7d
```

In this case:

- `https://matchmaker.phenomecentral.org/rest/remoteMatcher` is the base URL
- `/mmapi/v1/match` is the API method for submitting queries
- `854a439d278df4283bf5498ab020336cdc416a7d` is the authentication token that **B** told **A** to use in all match requests

If the token is missing from a request, or if the token is not recognized, then the query is refused and a 401 Unauthorized response is given.