Web API

In addition to the web interface, one can access Cpf1-Database via web API (Application Programming Interface). All data is returned in JSON (JavaScript Object Notation) format.

1. Retrieving an organism list

Request

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><a href="http://www.rgenome.net/cpf1-database/organisms/">http://www.rgenome.net/cpf1-database/organisms/</a></td>
</tr>
</tbody>
</table>

Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>organism_type</td>
<td>If specified, only the organisms with this type of organism are selected. Currently 1 for Vertebrates, 2 for Plants, 3 for Insects, and 4 for Others.</td>
</tr>
</tbody>
</table>

Response

Response

```json
{  "organisms": [    {"type": "organism type", "id": <organism id>, "name": <organism name>},    ...  ]}
```

Example (http://www.rgenome.net/cpf1-database/organisms/)

```json
[  "organisms": [    {"type": "vertebrate", "id": 1, "name": "Homo sapiens (GRCh38/hg38) - Human"},    {"type": "plant", "id": 2, "name": "Arabidopsis thaliana (TAIR10)", "type": "plant", "id": 3, "name": "Musa acuminata (MA1) - Banana"},    {"type": "plant", "id": 4, "name": "Vitis vinifera (IGGP_12X) - European grapevine"},    {"type": "vertebrate", "id": 5, "name": "Rattus norvegicus (Rnor 6.0) - Rat"},    {"type": "vertebrate", "id": 6, "name": "Musa acuminata (MA1) - Banana"},    {"type": "vertebrate", "id": 7, "name": "Solanum lycopersicum (SL 2.5) - Tomato"},    {"type": "vertebrate", "id": 8, "name": "Danio rerio (GRCz10) - Zebrafish"},    {"type": "insect", "id": 9, "name": "Drosophila melanogaster (BDGP6) - Fruit fly"},    {"type": "other", "id": 10, "name": "Caenorhabditis elegans (WBcel235)"},    {"type": "plant", "id": 11, "name": "Glycine max (JGI v1.0) - Soybean"},    {"type": "vertebrate", "id": 12, "name": "Sus scrofa (Ensembl v10.2) - Pig"}]}
```

2. Retrieving a gene list

Request

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><a href="http://www.rgenome.net/cpf1-database/organisms/">http://www.rgenome.net/cpf1-database/organisms/</a>&lt;organism_id&gt;/genes/</td>
</tr>
</tbody>
</table>

Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>page</td>
<td>Page number to retrieve.</td>
</tr>
<tr>
<td>query</td>
<td>Searching query for genes. If specified, only genes with query in symbol or Ensemble id are returned.</td>
</tr>
<tr>
<td>all</td>
<td>Flag to retrieve all genes without pagination. 1 is true, and 0 is false.</td>
</tr>
</tbody>
</table>

Response

Response
Example (http://www.rgenome.net/cpf1-database/organisms/1/genes/?page=1&query=c4bp)

{"page_count": 1, "gene_count": 2, "genes": [{"symbol": "C4BPA", "ensembl_id": "ENSG00000123838", "id": 18037, "biotype": "Protein coding", "description": "complement component 4 binding protein, alpha [Source:HGNC Symbol;Acc:HGNC:1325]"}, {"symbol": "C4BPB", "ensembl_id": "ENSG00000123843", "id": 15648, "biotype": "Protein coding", "description": "complement component 4 binding protein, beta [Source:HGNC Symbol;Acc:HGNC:1328]"}]

3. Retrieving a target list

Request

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><a href="http://www.rgenome.net/cpf1-database/genes/">http://www.rgenome.net/cpf1-database/genes/</a>&lt;gene_id&gt;/targets/</td>
</tr>
</tbody>
</table>

Parameter Description

- `filter_mis0`: Filtering query for the count of 0 mismatches, specified in number. Trailing ‘+’ or ‘-’ sign to the number specifies ranges of count, e.g. mismatch count of equals to 2 or above would be ‘2+’.
- `filter_mis1`: Filtering query for the count of 1 mismatch, specified in number. Trailing ‘+’ or ‘-’ sign to the number specifies ranges of count.
- `filter_mis2`: Filtering query for the count of 2 mismatches, specified in number. Trailing ‘+’ or ‘-’ sign to the number specifies ranges of count.
- `filter_maxcoverage`: Flag for selection of the targets with maximum transcript coverage. 1 is true, 0 is false.
- `filter_gc`: Flag for filtering GC contents. If set, the targets with GC contents below 20 or above 80 are neglected. 1 is true, 0 is false.
- `filter_trepeats`: Flag for filtering Thymine repeats. If set, the targets with more than 4 Thymines are neglected. 1 is true, 0 is false.
- `filter_oof`: Filtering query for the Microhomology-associated out-of-frame score, specified in number. If set, only the targets above this criterion are selected.
- `filter_cdsposition`: Filtering query for the relative CDS position, specified in number. If set, only the targets below this criterion are selected.
- `max_count`: If specified, the optimal targets up to this value among the filtered targets are selected. The targets are selected to have minimal off-target numbers and also maximum Microhomology-associated out-of-frame score.
Response

```json
{
  "targets": [
    {
      "id": <target ID in gene>,
      "chromosome": <chromosome name>,
      "position": <position of target>,
      "gc_contents": <GC contents of target>,
      "strand": <strand information of target>,
      "oof_score": <Microhomology-associated out-of-frame score>,
      "sequence": <sequence of target>,
      "coverage": <exon coverage>,
      "cds_percentages": {
        <Ensembl ID of transcript>: <relative position in CDS>,
        ...
      },
      "offtarget_counts": [
        <count of off-targets with mismatch 0>,
        <count of off-targets with mismatch 1>,
        <count of off-targets with mismatch 2>
      ]
    },
    ...
  ],
  "target_counts": {
    <Ensembl ID of transcript>: {
      "total_count": <count of all targets in the transcript>,
      "filtered_count": <count of filtered targets in the transcript>
    }
  }
}
```

Example ([http://www.rgenome.net/cpf1/database/genes/101/targets/?filter_mis0=1&filter_mis1=0&filter_mis2=0](http://www.rgenome.net/cpf1/database/genes/101/targets/?filter_mis0=1&filter_mis1=0&filter_mis2=0))

```json
{"targets": [{"gc_contents": 56.5217391304, "strand": "+", "oof_score": 65.070908394, "sequence": "TTTAGCATGGCTCAGACAGTCACTCAG", "coverage": 1, "position": 22281450, "offtarget_counts": [1, 0, 0], "id": 1, "chromosome": "chr14", "cds_percentages": {"ENST00000390465": 21.4899713467}}, "target_counts": {"ENST00000390465": {"total_count": 5, "filtered_count": 1}}]}
```

4. Retrieving an off-target list

Request

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><a href="http://www.rgenome.net/cpf1/database/genes/">http://www.rgenome.net/cpf1/database/genes/</a>&lt;gene_id&gt;/targets/&lt;target_id&gt;/offtargets/</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>page</td>
<td>Page number to retrieve.</td>
</tr>
<tr>
<td>count</td>
<td>If specified, only the off-targets with the mismatch count of this value are selected.</td>
</tr>
<tr>
<td>all</td>
<td>Flag to retrieve all off-targets without pagination. If the number of off-targets is more than 1000, then this option is neglected. 1 is true, and 0 is false.</td>
</tr>
</tbody>
</table>
Example (http://www.rgenome.net/cpf1-database/genes/101/targets/2/offtargets/)

{"page_count": 1, "offtargets": [{"chromosome": "chr14", "sequence": "TTTGTTCTGGTACAAGCAGCCTCCCAG", "region": "CDS", "strand": "+", "position": 22272371, "mismatch_count": 0}, {"chromosome": "chr14", "sequence": "TTTATTCTGGTACAAGCAGCCTCCCAG", "region": "CDS", "strand": "+", "position": 22281557, "mismatch_count": 0}], "offtarget_count": 2}