

CircularMiners

CircularMiners Backend APIs

User Management

The APIs for user management includes functionality for user registration and login, for both mining company representatives and data requestors

Mine Representative

Employee of the mining company that responsible for managing data of a mine

GET Get Mining Company's Representative's Profile

`https://urchin-app-q36en.ondigitalocean.app/backend2/registration/minerepresentative/:id`

The API function for checking mining representative according to its assigned userID.
Here's a description of the API function :

Path Variables Example :

id : 1782bc72-7188-453f-acac-f3f0c73f2c35

PATH VARIABLES

id

uuid

POST Register New Mining Company's Representative

`https://urchin-app-q36en.ondigitalocean.app/backend2/registration/minerepresentative`

The API function for registering a new mining company representative for allowing his/her to submit information about their mining sites.

Here's a description of the API function :

Example :

```
{
  "mineRepresentativeName": "Sadrul Nashid",
  "mineRepresentativeEmail": "adminadmin@gmail.com",
  "mineRepresentativePassword": "p4s5w0rD",
  "mineRepresentativeCompanyname": "LUT",
  "mineRepresentativePhonenumber": "01199787635"
}
```

Body raw (json)

json

```
{
  "mineRepresentativeName": "string",
  "mineRepresentativeEmail": "string (email)",
  "mineRepresentativePassword": "string",
  "mineRepresentativeCompanyname": "string",
  "mineRepresentativePhonenumber": "string (phone number)"
}
```

Data Requestor

Users that want to get information about the sidestream data of mines

POST Register New Data Requestor

<https://urchin-app-q36en.ondigitalocean.app/backend2/registration/datarequestor>

The API function for registering a user, which intention is for getting mine data.

Here's a description of the API function :

Example :

```
{
  "dataRequestorName": "Jon Nashid",
  "dataRequestorEmail": "requestme@gmail.com",
  "dataRequestorPassword": "p4s5w0rD",
  "dataRequestorCompanyname": "LUT"
}
```

Body raw (json)

json

```
{
  "dataRequestorName": "string",
  "dataRequestorEmail": "string (email)",
  "dataRequestorPassword": "string (password)",
  "dataRequestorCompanyname": "string"
}
```

POST Login

<https://urchin-app-q36en.ondigitalocean.app/backend2/login>

The API function for checking user validity for accessing the user account

Example :

```
{
  "userEmail": "requestorme@gmail.com",
  "userPassword": "p4s5w0rD",
  "userType": "requestor"
}
```

Body raw (json)

json

```
{
  "userEmail": "string (email)",
  "userPassword": "string (password)",
  "userType": "string (requestor/...)"
}
```

Mineral

The APIs for adding mineral database for future usage

PUT Add new mineral

<https://urchin-app-q36en.ondigitalocean.app/backend2/mineral/addmineral/>

The API function for adding a new mineral to the mineral database allows users to submit information about a mineral and have it stored in the database for future reference. Here's a description of the API function

Example :

```
{
  "mineralChemicalFormula": "SiO2",
  "mineralChemicalName": "Silicon Dioxide",
  "mineralName": "Quartz"
}
```

Body raw (json)

json

```
{
  "mineralChemicalFormula": "string",
  "mineralChemicalName": "string",
  "mineralName": "string"
}
```

GET Get mineral list

<https://urchin-app-q36en.ondigitalocean.app/backend2/mineral/searchmineral/?mineralChemicalFormula=string&mineralChemicalName=string&mineralName=string>

The API function for getting mineral list or mineral information. Here's a description of the API function :

Mandatory params to be filled :

None, however it is extremely advisable to filled at least one column for speeding up the searching process

Request Params Examples :

mineralChemicalFormula : Fe

mineralChemicalName : Ferrum

mineralName : Iron

PARAMS

mineralChemicalFormula	string
mineralChemicalName	string
mineralName	string

Mine Management

The APIs function for adding new mining location to the database. The mine management functions only can be accessible by registered mine representative

POST Add new mine

<https://urchin-app-q36en.ondigitalocean.app/backend2/mine/addmine/>

The API function for adding a new mineral to the mineral database allows users to submit information about a mineral and have it stored in the database for future reference. Here's a description of the API function

Example :

```
{
  "mineName": "Sadrul Mine",
  "mineLocation": "Bangladesh",
  "mineDescription": "---",
  "mineRepresentativeId": "1782bc72-7188-453f-acac-f3f0c73f2c35"
}
```

Body raw (json)

json

```
{
  "mineName": "string",
  "mineLocation": "string (name of city)",
  "mineDescription": "string",
  "mineRepresentativeId": "uuid (referred from minerepresentation API)"
}
```

GET Get list of registered mine

[https://urchin-app-q36en.ondigitalocean.app/backend2/mine/searchmine?mineID=uuid&mineName=string&mineLocation=string&mineDescription=string&mineRepresentativeID=uuid\(referred from minerepresentation API\)](https://urchin-app-q36en.ondigitalocean.app/backend2/mine/searchmine?mineID=uuid&mineName=string&mineLocation=string&mineDescription=string&mineRepresentativeID=uuid(referred from minerepresentation API))

The API function for getting registered new mining locations allows users to retrieve information about the mining locations that have been registered in the mining address database.

Mandatory params: At least one of the params should be filled

Example :

mineID:541b5ab6-b3c1-47ff-9e6a-d908606b5f77

mineName:Sadrul Mine

mineLocation:Bangladesh

mineDescription:---

mineRepresentativeID:1782bc72-7188-453f-acac-f3f0c73f2c35

PARAMS

mineID	uuid
mineName	string
mineLocation	string
mineDescription	string
mineRepresentativeID	uuid(referred from minerepresentation API)

SideStream Data Management

The APIs for managing data of the piles mined from one of the registered mine sites

POST Create sidestream data (as a mine Representative)

<https://urchin-app-q36en.ondigitalocean.app/backend2/sidestream/:mineRepId/:mineId>

The API function for adding a new pile data to the sidestream database. This function can be done only by authorized mine representative.

Here's a description of the API function :

Example of Path Variables :

mineRepId : "1782bc72-7188-453f-acac-f3f0c73f2c35"

mineId : "4582bc82-8888-823f-acac-bcf0c73f2c45"

Example of Body JSON :

```
{
  "oreName": "soil",
  "description": "lot of soil",
  "weight": 30,
  "size": 300,
  "mineRepresentative": "1782bc72-7188-453f-acac-f3f0c73f2c35",
  .. .. .
```

```
"compositionMaterial":  
[  
  {  
    "mineralFormula": "H2O",  
    "mineralName": "water",  
    "mineralPercentage": 50  
  },  
  {  
    "mineralFormula": "H2O",  
    "mineralName": "water",  
    "mineralPercentage": 50  
  }  
]  
{
```

PATH VARIABLES

mineRepld uuid (from mine representative)

mineld uuid (from mine)

Body raw (json)

json

```
{  
  "oreName": "string",  
  "description": "string",  
  "weight": double,  
  "size": int,  
  "mineRepresentative": "uuid (from mine representative)",  
  "compositionMaterial": [  
    {  
      "mineralFormula": "string (from mineral)",  
      "mineralName": "string (from mineral)",  
      "mineralPercentage": int (0-100)  
    },  
    {  
      "mineralFormula": "string (from mineral)",  
      "mineralName": "string (from mineral)",  
      "mineralPercentage": int (0-100)  
    }  
  ]  
}
```

GET Search sidestream pile based on material name

https://urchin-app-q36en.ondigitalocean.app/backend2/sidestream/:oreName

The API function for searching sidestream data of the mines according to its main mineral content (ore). Here's a description of the API function

Example Path Variables :

oreName : soil

PATH VARIABLES

oreName	string
---------	--------

GET List of sidestream registered by a mine representative

https://urchin-app-q36en.ondigitalocean.app/backend2/sidestream/:mineRepld/:mineld

The API function for searching sidestream data of the mines according to the handler of the mine. Here's a description of the API function :

Example Path Variables :

mineRepld : 1782bc72-7188-453f-acac-f3f0c73f2c35

mineld : 541b5ab6-b3c1-47ff-9e6a-d908606b5f77

PATH VARIABLES

mineRepld	uuid
-----------	------

mineld	uuid
--------	------

GET Sidestream data for approved request access

https://urchin-app-q36en.ondigitalocean.app/backend2/sidestream/requestor/:dataRequestorId/:sidestreamId

The API function for searching a side stream detail data of an approved request access. Here's a description of the API function :

Example Path Variables :

dataRequestorId : 21734667-7a32-45f4-97aa-accffc62066d

sidestreamId : 3715affe-ed3f-4f4f-8de7-2ada5e67693f

PATH VARIABLES

dataRequestorId	uuid (from requestor)
sideStreamId	uuid (from requested sidestream data)

SideStream Data Request

The API function for requesting in-depth information about certain pile by data requestors. Mine representatives are given authority to permit or prevent the sidestream data requests.

GET Get Detail of a Request Access

`https://urchin-app-q36en.ondigitalocean.app/backend2/requestaccess/:mineRepId/:dataRequestorId`

The API function for searching detail of a request access.
Here's a description of the API function

Example Path Variables :

mineRepId : 1782bc72-7188-453f-acac-f3f0c73f2c35
dataRequestorId : 21734667-7a32-45f4-97aa-accffc62066d

PATH VARIABLES

mineRepId	uuid (from mine representative)
dataRequestorId	uuid (from requestor)

POST Data Requestor Request Access for a Sidestream Data

`https://urchin-app-q36en.ondigitalocean.app/backend2/requestaccess/:dataRequestorId/:sideStreamId`

The API function for submitting an requestor request for sidestream data access.
Here's a description of the API function

Example Path Variables :

dataRequestorId : 21734667-7a32-45f4-97aa-accffc62066d
sideStreamId : 1782bc72-7188-453f-acac-f3f0c73f2c35

PATH VARIABLES

dataRequestorId	uuid (from requestor)
sideStreamId	uuid (from sidestream)

Body raw (json)

```
json

{
  "requestAccessMessage": "plesss give me access"
}
```

GET All requestaccess as a mine rep

<https://urchin-app-q36en.ondigitalocean.app/backend2/requestaccess/miner/:mineRepId>

The API function for retrieving all data access of a mining representative, either it was approved, rejected, or are still under reviewed.

Here's a description of the API function

Example Path Variables :

mineRepId : 1782bc72-7188-453f-acac-f3f0c73f2c35

PATH VARIABLES

mineRepId	uuid (from mine representative)
------------------	---------------------------------

PUT Approving/Rejecting Data Request by Mine Representative

<https://urchin-app-q36en.ondigitalocean.app/backend2/requestaccess/:mineRepId>

The API function for mine representative for approving or rejecting data access requests.

Here's a description of the API function

Example Path Variables :

mineRepId : 1782bc72-7188-453f-acac-f3f0c73f2c35

Example :

```
{
  "dataRequestorId":"21734667-7a32-45f4-97aa-accffc62066d",
  "requestAccessStatus":"APPROVED",
  "sidestreamId":"3715affe-ed3f-4f4f-8de7-2ada5e67693f"
}
```

PATH VARIABLES

mineRepld uuid (from mine representative)

Body raw (json)

json

```
{
  "dataRequestorId" : "uuid (from requestor)",
  "requestAccessStatus" : "APPROVED/REJECTED",
  "sidestreamId" : "uuid (from sidestream)"
}
```

GET All requestaccess as a dataRequestor

<https://urchin-app-q36en.ondigitalocean.app/backend2/requestaccess/requestor/:dataRequestorId>

The API function for retrieving all data request from a requestor.
Here's a description of the API function

Example Path Variables :

dataRequestorId: 21734667-7a32-45f4-97aa-accffc62066d

PATH VARIABLES

dataRequestorId uuid (from data requestor)

GET mining data where access is APPROVED

<https://urchin-app-q36en.ondigitalocean.app/backend2/sidestream/requestor/:dataRequestorId/:sidestreamId>

The API function for retrieving side stream data of approved request access.

Here's a description of the API function

Example Path Variables :

dataRequestorId: 21734667-7a32-45f4-97aa-accffc62066d

sidestreamId: 3715affe-ed3f-4f4f-8de7-2ada5e67693f

PATH VARIABLES

dataRequestorId	uuid (from data requeuestor)
------------------------	------------------------------

sidestreamId	uuid (from data sidestream)
---------------------	-----------------------------

GET Testing API Connection

<https://urchin-app-q36en.ondigitalocean.app/backend2/test>

APIs for make sure the client is connected to the API server