Prerequisites

Remote machine 1: VM with Ubuntu, CapRover and fixed IP*

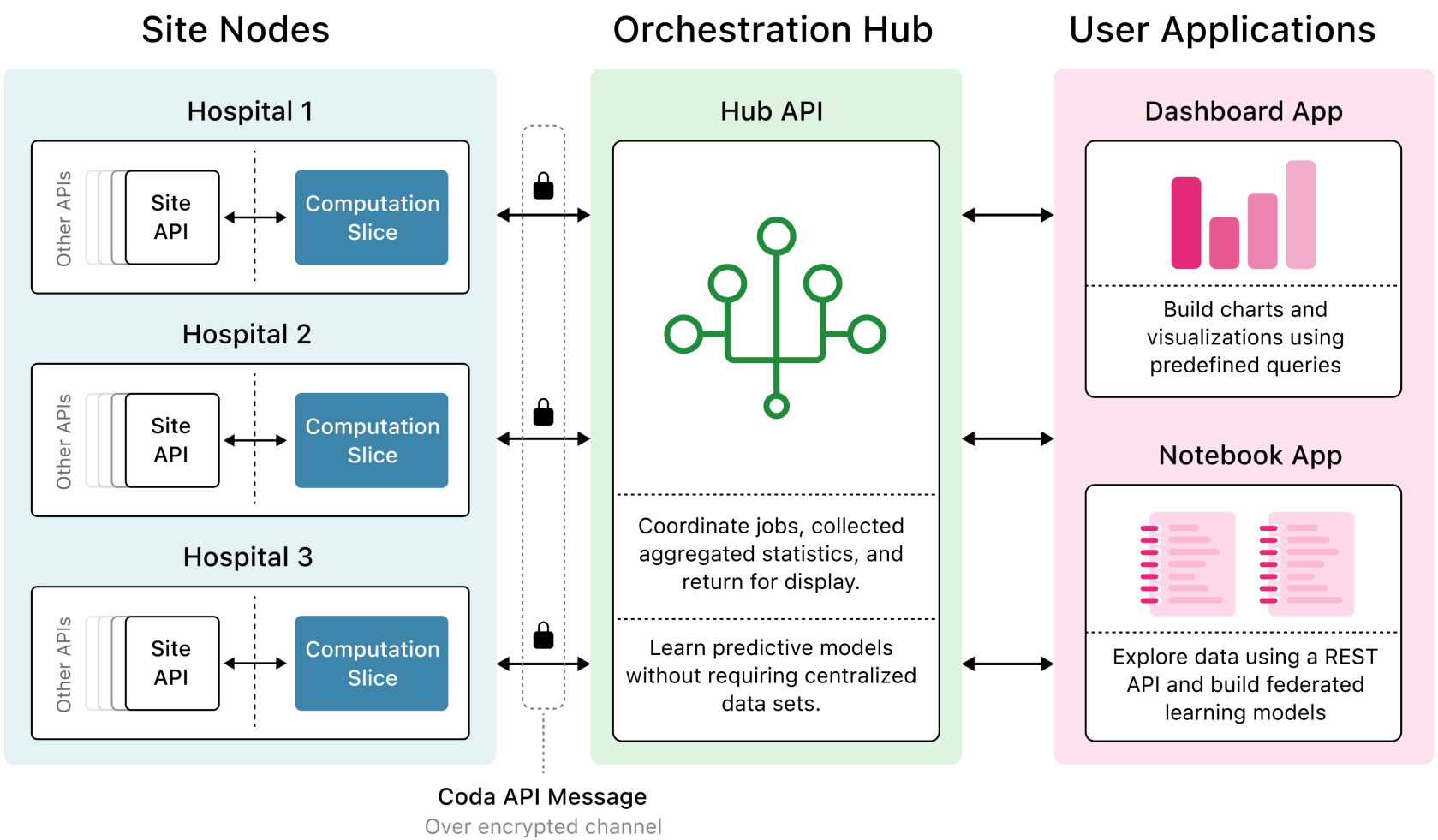
Remote machine 2: VM with Ubuntu, CapRover and fixed IP*

(Your) local machine: GitHub and NodeJS >= 18 installed

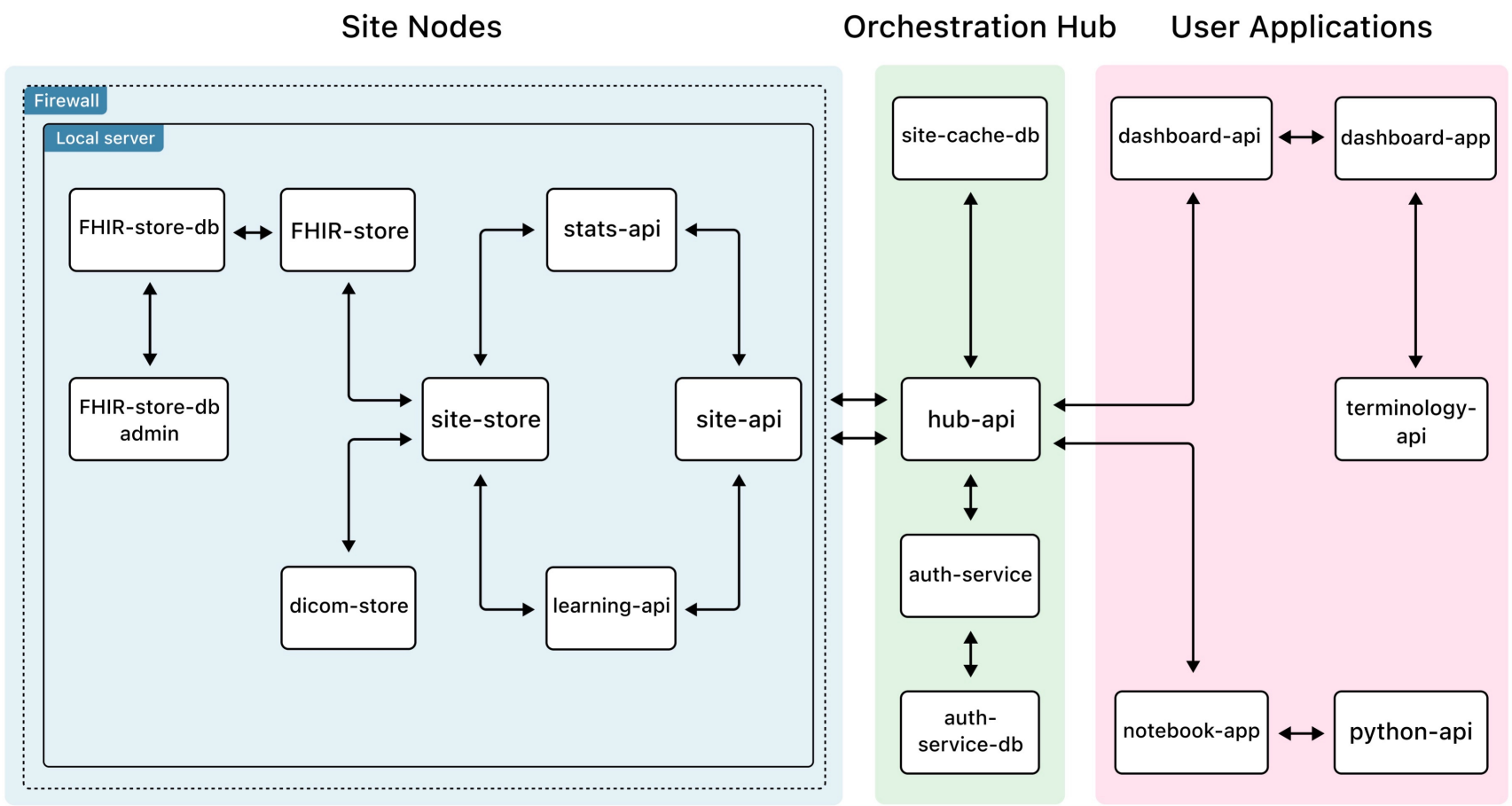
A domain name: with access to DNS configuration settings

***For example, using one-click deploy on DigitalOcean:**
https://caprover.com/docs/get-started.html

Conceptual overview



Technical overview

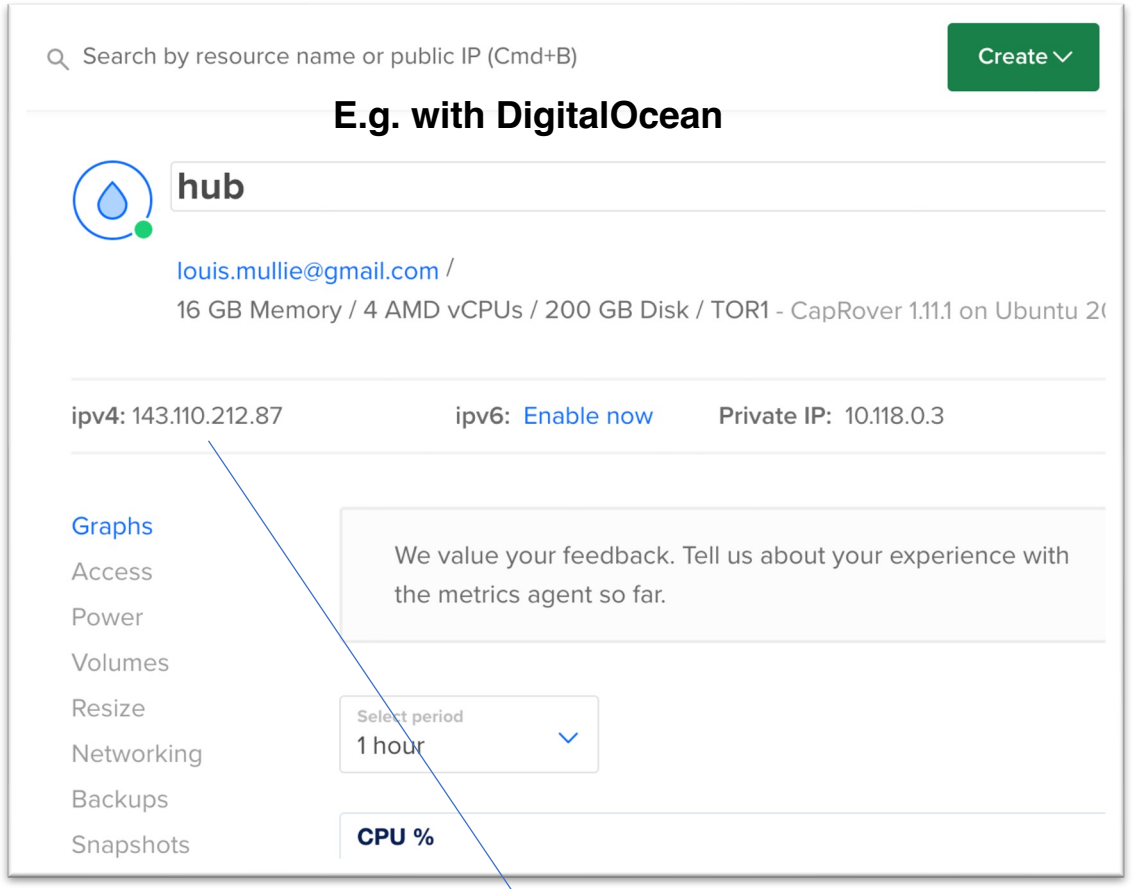


Each white box corresponds to a GitHub repository:
https://github.com/orgs/coda-platform/repositories

1. Configure CapRover

The process will be demonstrated for the **hub**
and is identical for the all nodes.

1.1 Obtain the IP of the machine you will be using for the hub



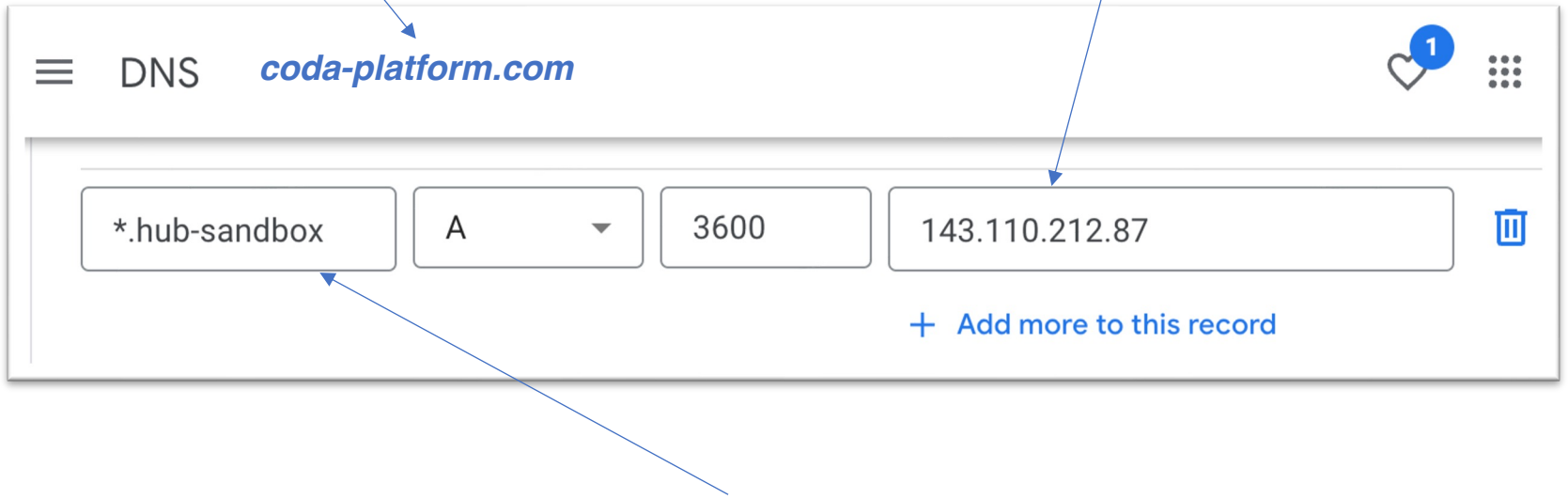
Note the IP address of your hub server.
In this example, it is:

`143.110.212.87`

1.2 Configure DNS on your domain name

Your domain name

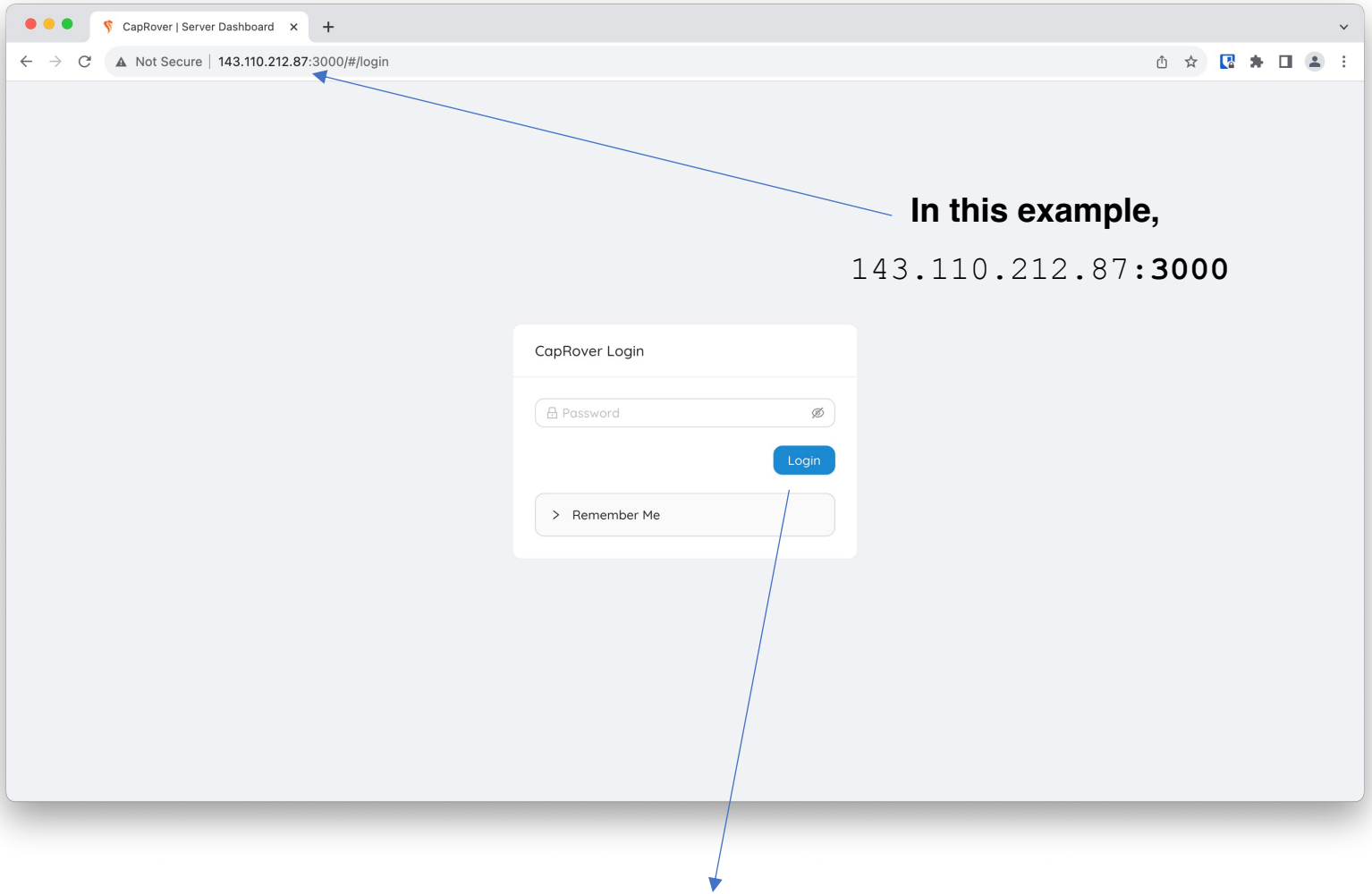
IP address of your hub VM



Subdomain of your choice for hub
In this example, hub will live at:

`*.hub-sandbox``.coda-platform.com`

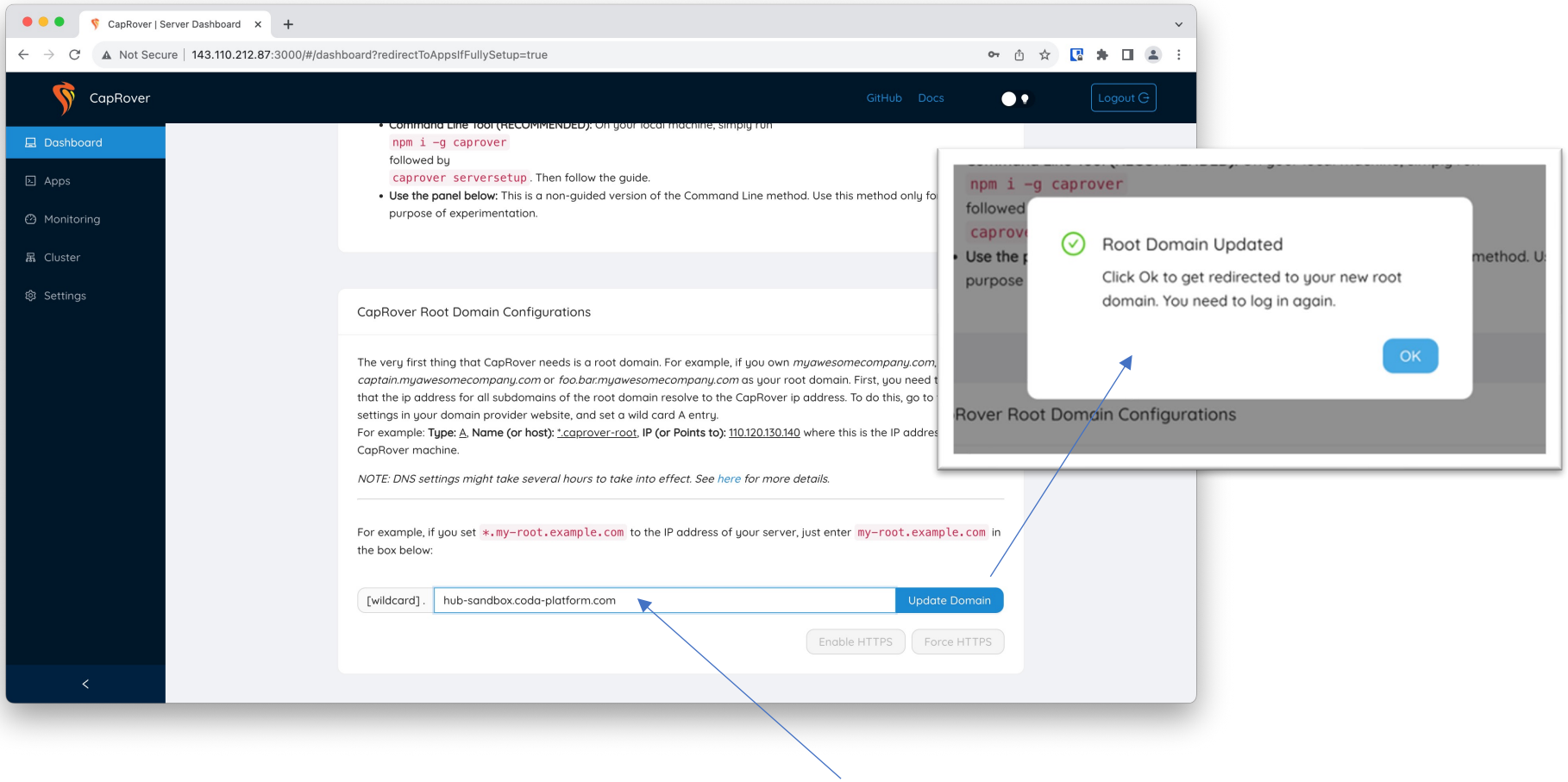
1.3 Navigate to http://<server IP>:3000



Confirm that CapRover is functional by viewing login screen above.

Login using default password: `captain42`

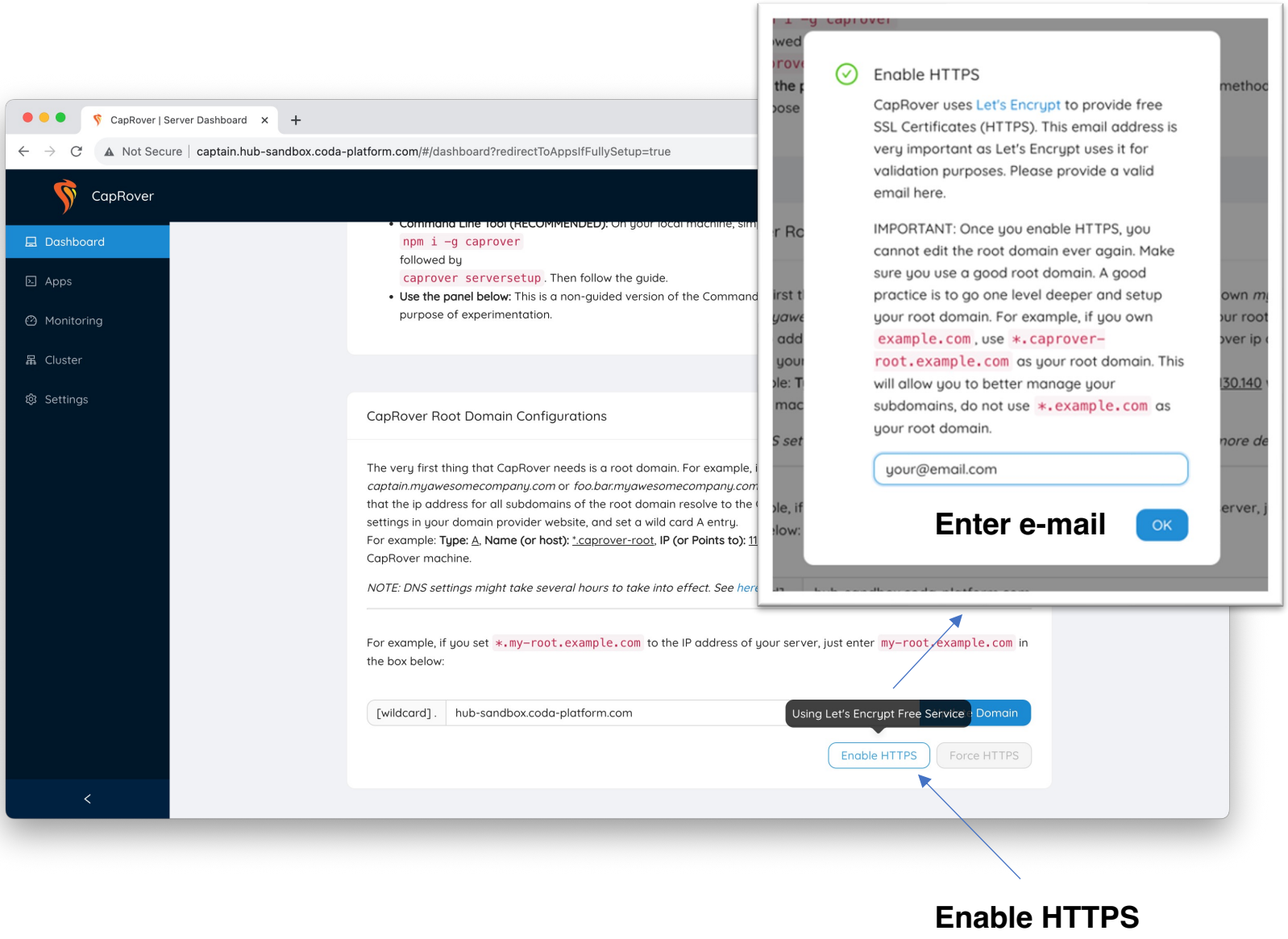
1.4 Configure CapRover root domain



Enter root domain that will be used for “hub” server
In this example, we previously configured DNS for:

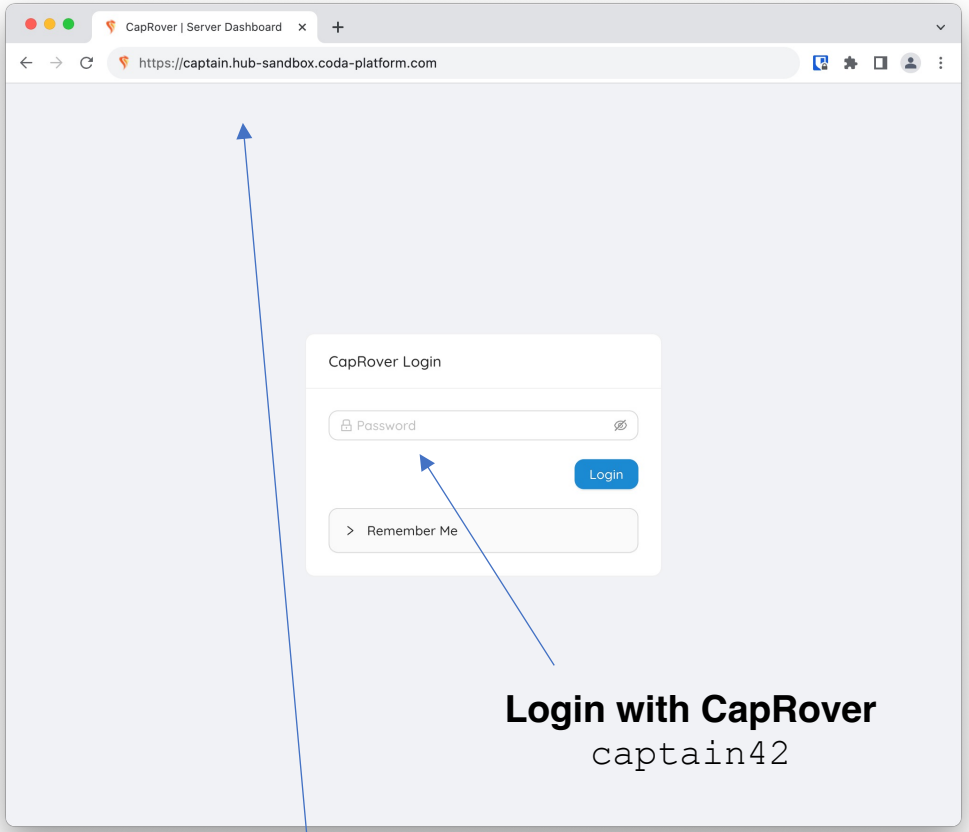
`hub-sandbox.coda-platform.com`

1.5 Enable HTTPS



1.6 Test root domain with HTTPS

`https://captain.`**`your-hub-subdomain.your-domain.com`**

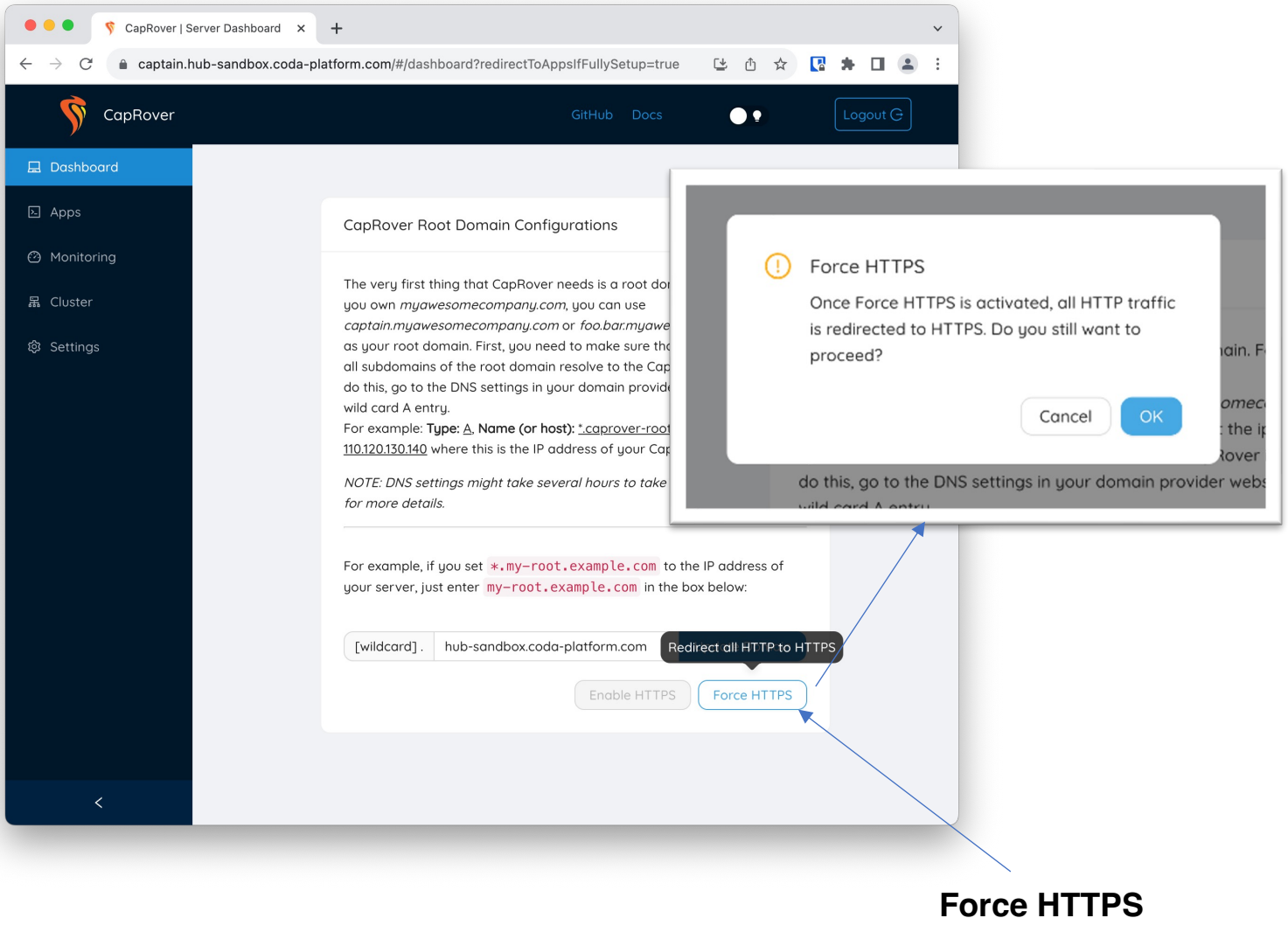


Navigate to the address as described above

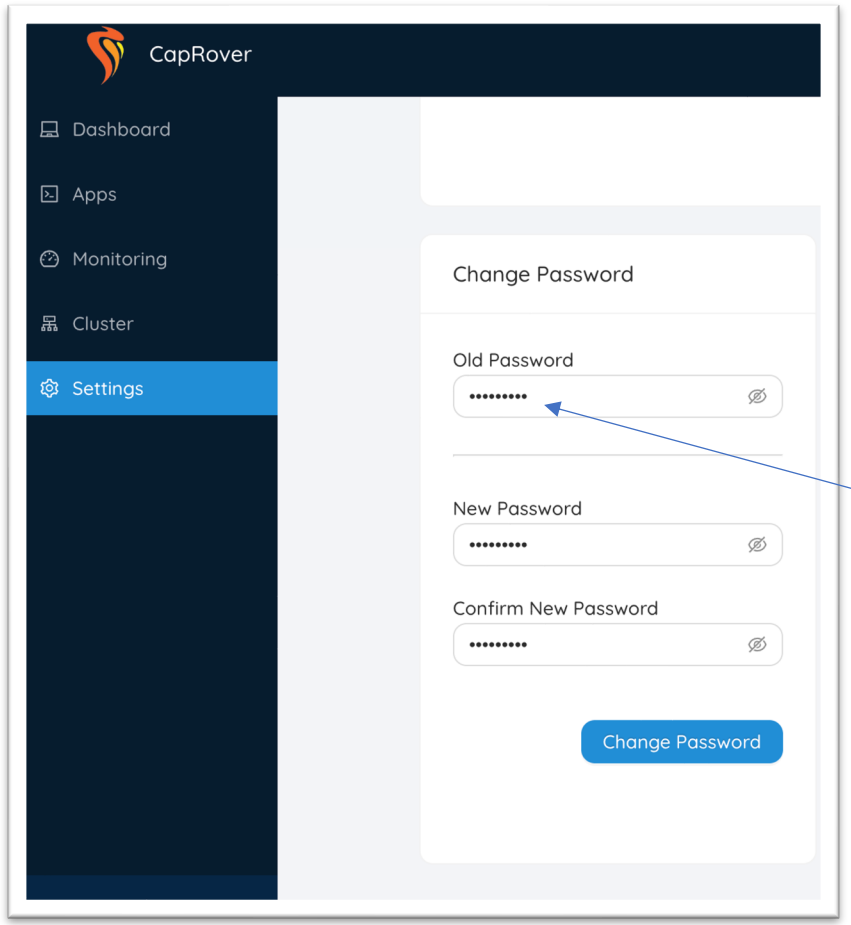
In this example, it corresponds to:

`https://captain.hub-sandbox.coda-platform.com`

1.7 Force HTTPS on the root domain



1.8 Change the default password



Default password
`captain42`

2. Configure CapRover on a site machine

Repeat steps 1-9 using a a different server (IP) and subdomain.

In this example, we'll be using the following for site #1:

Site 1 IP address: `165.227.38.118`
Site 1 root domain: `site1-sandbox.coda-platform.com`
Site 1 DNS A record: `*.site1-sandbox.coda-platform.com A 165.227.38.118`

3. Configure additional machines for multi-site

Repeat previous steps as for site 1, modifying configuration accordingly.

Make sure you update the following configuration variables when deploying additional sites.

```
CAPROVER_NAME, CAPROVER_URL,
CODA_SITE_API_HOSPITAL_CODE,
CODA_SITE_API_AUTH_USERNAME,
CODA_SITE_API_AUTH_PASSWORD
```

Overview at this stage

Your configuration should look similar to this
(substituting your IPs, domain name, and subdomains).

VM	IP in A record	Root domain	CapRover control panel
0	143.110.212.87	hub-sandbox.coda-platform.com	captain.hub-sandbox.coda-platform.com
1	165.227.38.118	site1-sandbox.coda-platform.com	captain.site1-sandbox.coda-platform.com
…	…	…	…
4		site4-sandbox.coda-platform.com	captain.site4-sandbox.coda-platform.com

4. Deploy the hub

Clone the hub deployment automation scripts:
https://github.com/coda-platform/hub-deployer

Then, navigate to the directory where it was cloned in your terminal.

Once inside this directory, follow the steps on the next slide.

Configure environment in `.env` file →

```
CAPROVER_URL=https://captain.hub-sandbox.coda-platform.com
CAPROVER_PASSWORD=captain42
CAPROVER_NAME=hub-sandbox
CAPROVER_ROOT=coda-platform.com

CODA_AUTH_SERVICE_URL=https://auth-service.hub-sandbox.coda-platform.com
CODA_HUB_API_URL=https://hub-api.hub-sandbox.coda-platform.com
CODA_DASHBOARD_API_URL=https://dashboard-api.hub-sandbox.coda-platform.com
CODA_DASHBOARD_APP_URL=https://dashboard-app.hub-sandbox.coda-platform.com
```

Run `npm install` →

```
(base) louismullie@Louis-Antoines-MacBook-Pro-2 hub-deployer % npm install

up to date, audited 10 packages in 216ms

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
(base) louismullie@Louis-Antoines-MacBook-Pro-2 hub-deployer %
```

Run `caprover login` →

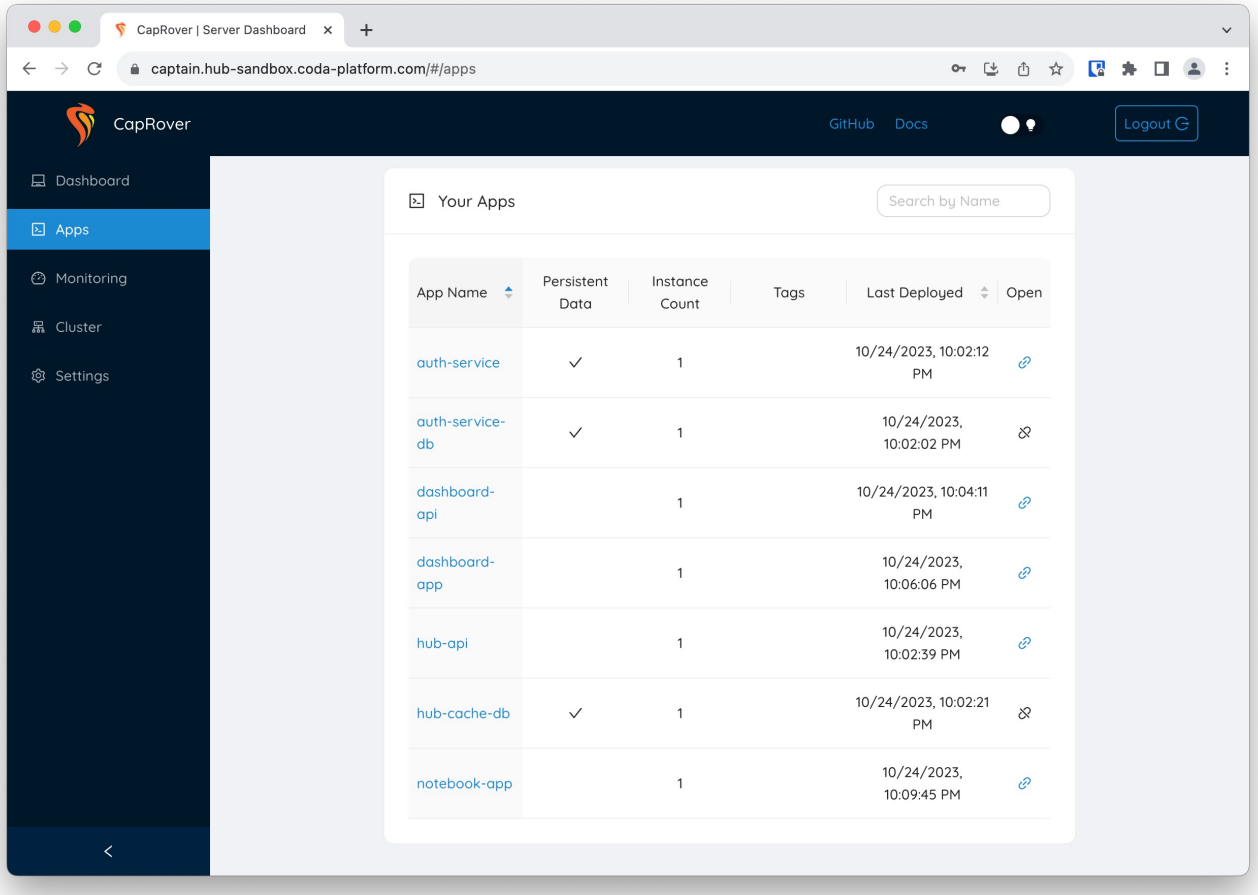
Using password you created for site machine during CapRover setup.

```
(base) louismullie@Louis-Antoines-MacBook-Pro-2 hub-deployer % caprover login


Login to a CapRover machine...

? CapRover machine URL address, it is "[http[s]://][captain.]your-captain-root.d
omain": https://captain.hub-sandbox.coda-platform.com
? CapRover machine password: [hidden]
? CapRover machine name, with whom the login credentials are stored locally: hub
-sandbox

Logged in successfully.
Authorization token is now saved as hub-sandbox at https://captain.hub-sandbox.coda-platform.com.
```

Once done, run: `npm run deploy`

Open the **hub** CapRover control panel

`https://captain.`**`your-hub-subdomain.your-domain.com`**



Your dashboard should show the exact same application names and configurations.

5. Deploy a site

Clone the site deployment automation scripts:
https://github.com/coda-platform/site-deployer

Then, navigate to the directory where it was cloned in your terminal.

Once inside this directory, follow the steps on the next slide.

Configure environment in `.env` file →

```
CAPROVER_URL=https://captain.site1-sandbox.coda-platform.com
CAPROVER_PASSWORD=captain42
CAPROVER_NAME=site1-sandbox

CODA_AUTH_SERVICE_URL=https://auth-service.hub-sandbox.coda-platform.com
CODA_HUB_API_URL=https://hub-api.hub-sandbox.coda-platform.com

CODA_STATS_API_URL=http://srv-captain--stats-api:8082
CODA_LEARNING_API_URL=http://srv-captain--learning-api:8084
CODA_FHIR_STORE_URL=http://srv-captain--fhir-store:8888
CODA_DICOM_STORE_URL=http://srv-captain--dicom-store:8042
```

Run `npm install` →

```
(base) louismullie@Louis-Antoines-MacBook-Pro-2 site-deployer % npm install

up to date, audited 10 packages in 226ms

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
(base) louismullie@Louis-Antoines-MacBook-Pro-2 site-deployer %
```

Run `caprover login` →

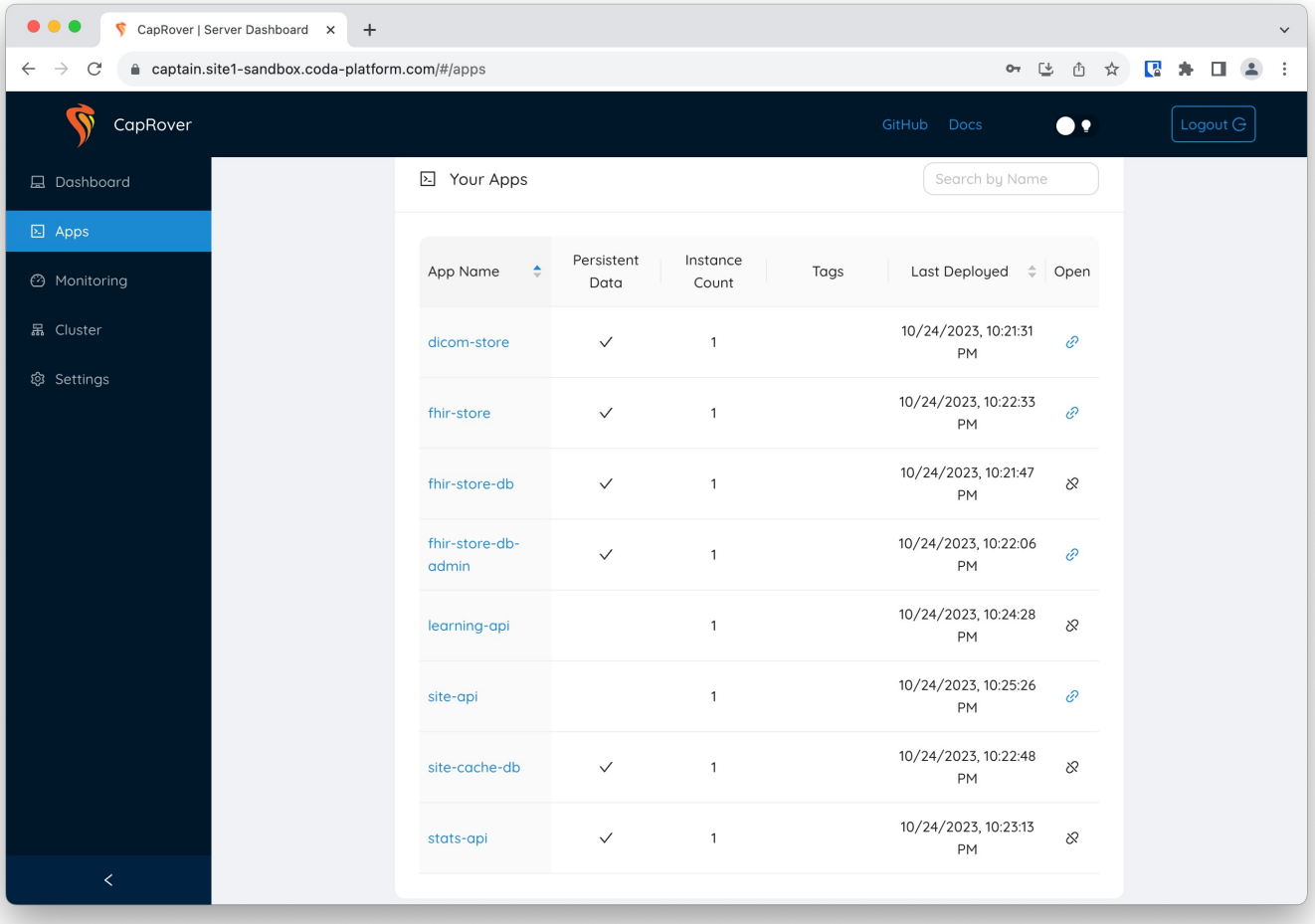
Using password you created for hub machine during CapRover setup.

```
(base) louismullie@Louis-Antoines-MacBook-Pro-2 site-deployer % caprover login


Login to a CapRover machine...

? CapRover machine URL address, it is "[http[s]://][captain.]your-captain-root.domain": https://captain.site1-sandbox.coda-platform.com
? CapRover machine password: [hidden]
? CapRover machine name, with whom the login credentials are stored locally: site1-sandbox

Logged in successfully.
Authorization token is now saved as site1-sandbox at https://captain.site1-sandbox.coda-platform.com.

(base) louismullie@Louis-Antoines-MacBook-Pro-2 site-deployer %
```

Once done, run: `npm run deploy`

Open the **site** CapRover control panel

`https://captain.`**`your-site-subdomain.your-domain.com`**



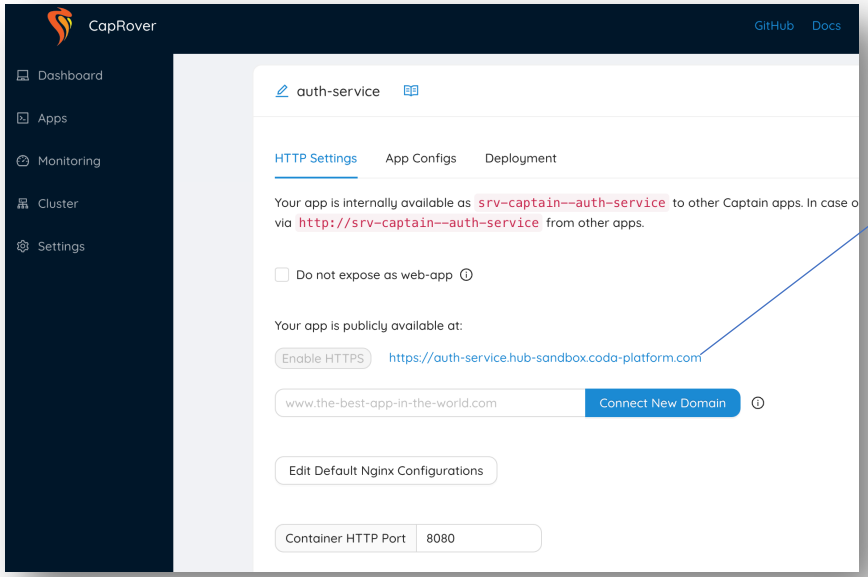
Your dashboard should show the exact same application names and configurations.

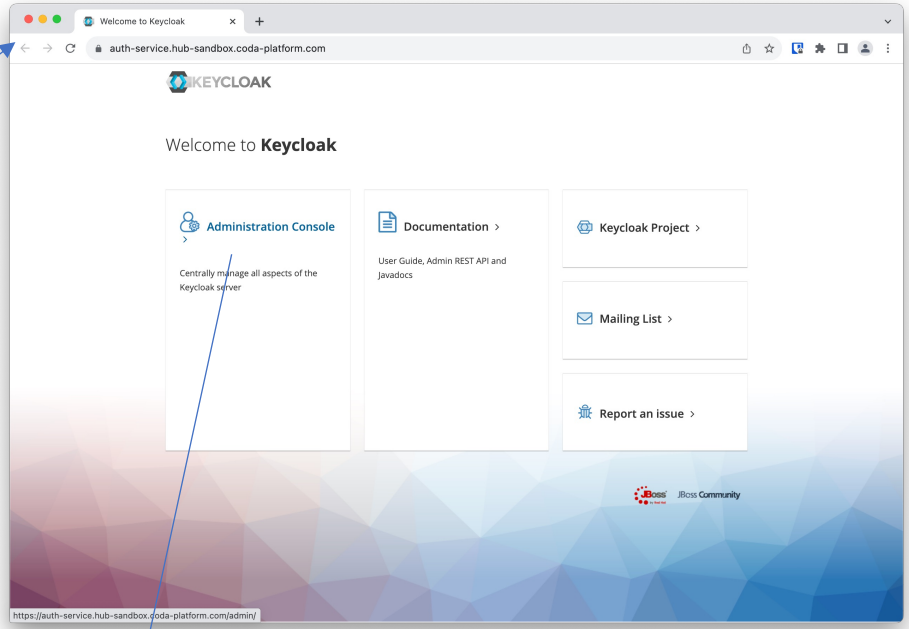
6. Configure Keycloak

Open the hub CapRover control panel

`https://captain.`**`your-hub-subdomain.your-domain.com`**

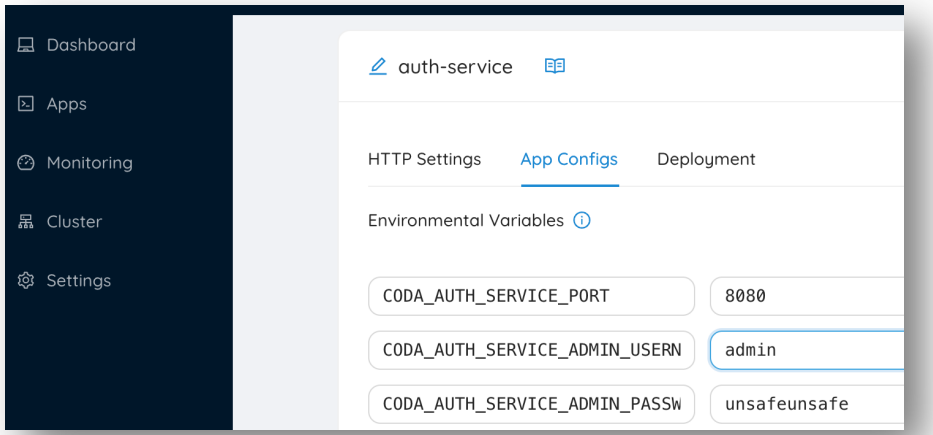
Open auth-service app to get link to Keycloak

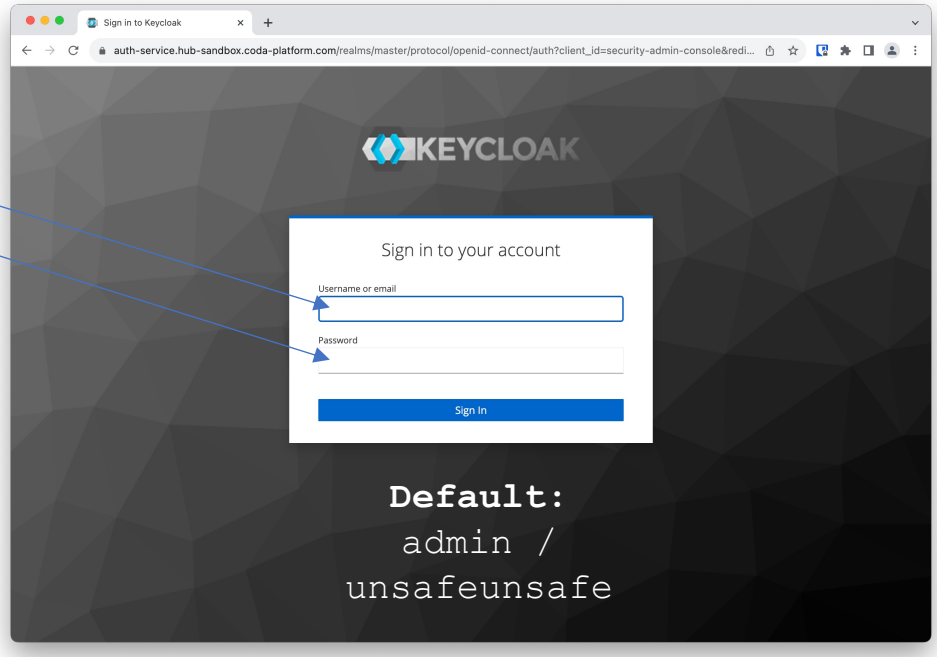




Admin username and password defined in app environment

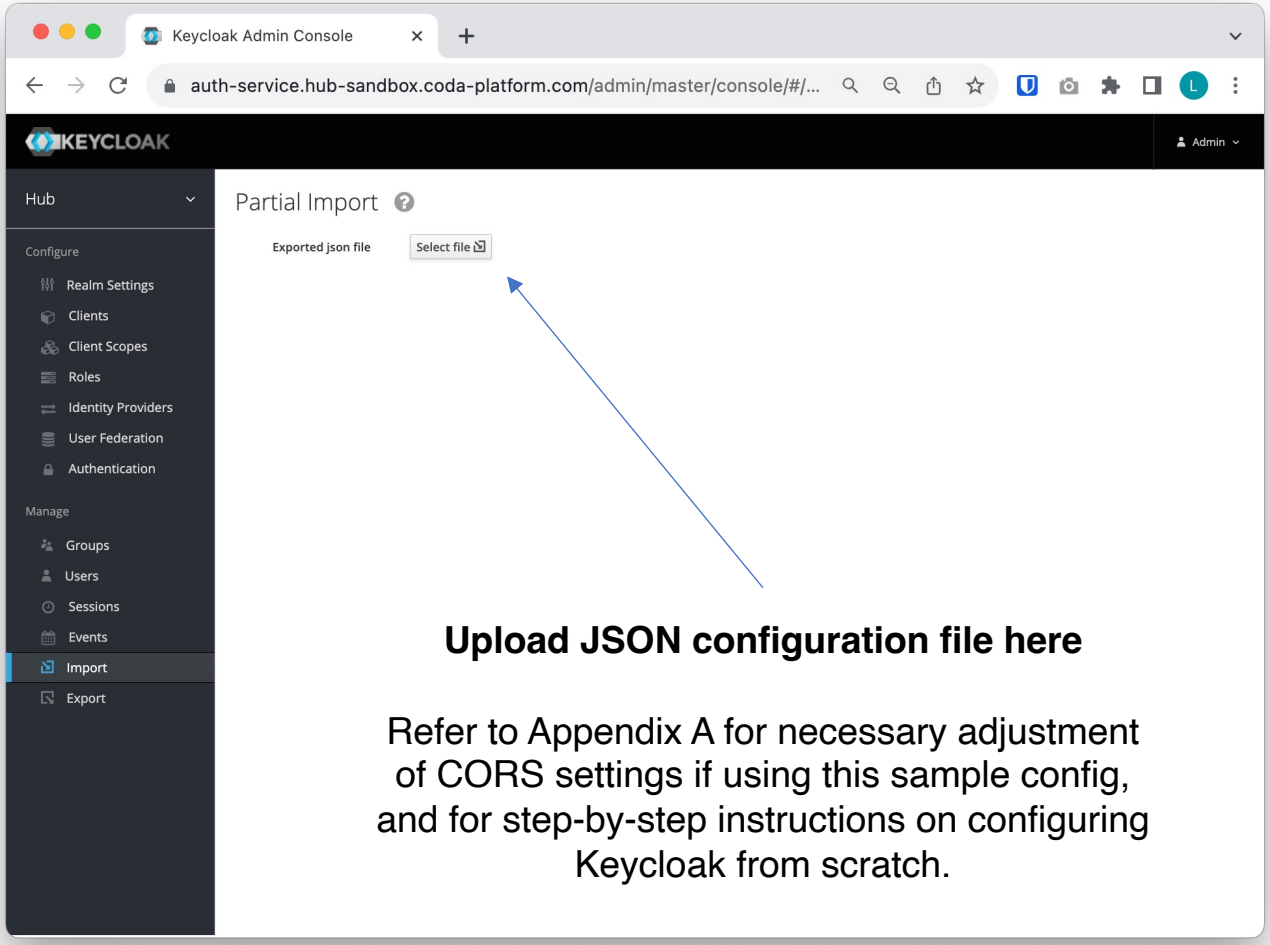
```
CODA_AUTH_SERVICE_ADMIN_USERNAME
CODA_AUTH_SERVICE_ADMIN_PASSWORD
```





Configure Keycloak using a default / example configuration

https://github.com/coda-platform/auth-service/blob/main/config.example.json



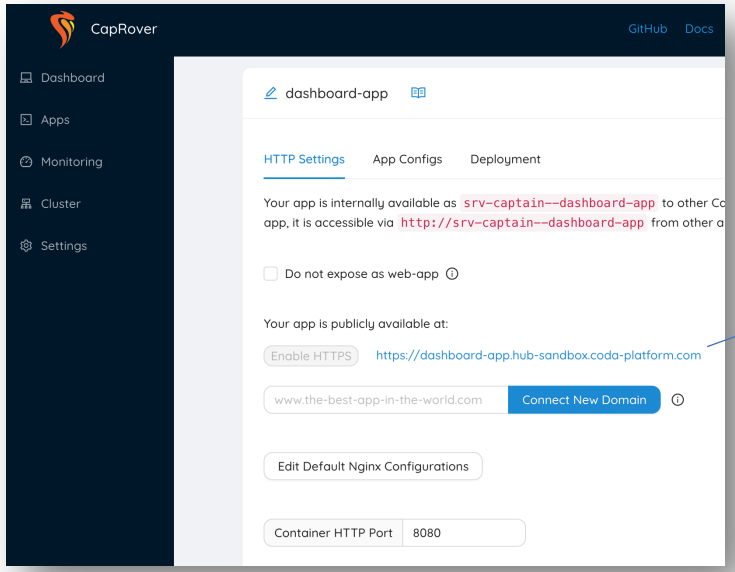
Note: the “sandbox” configuration is insecure, and for testing purposes only.

7. Test dashboard login

Open the hub CapRover control panel

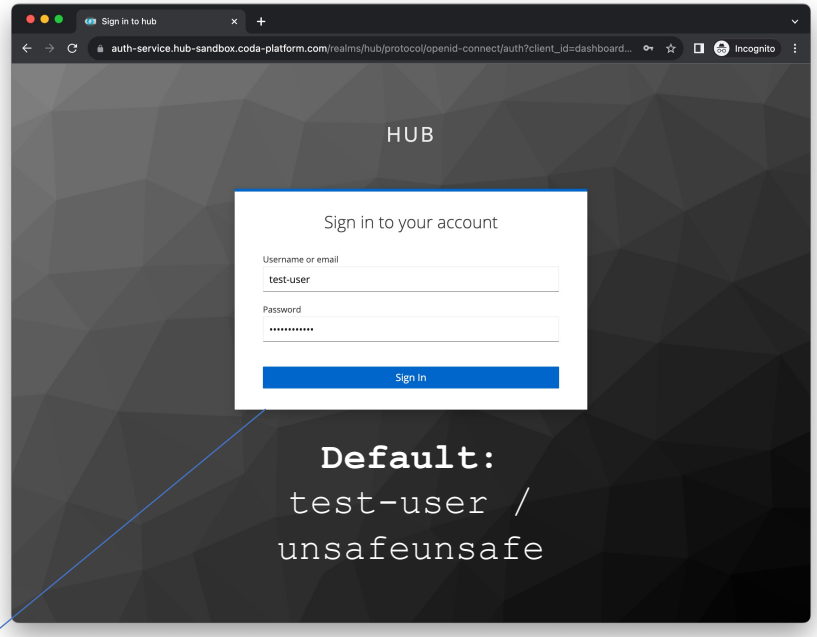
`https://captain.your-hub-subdomain.your-domain.com`

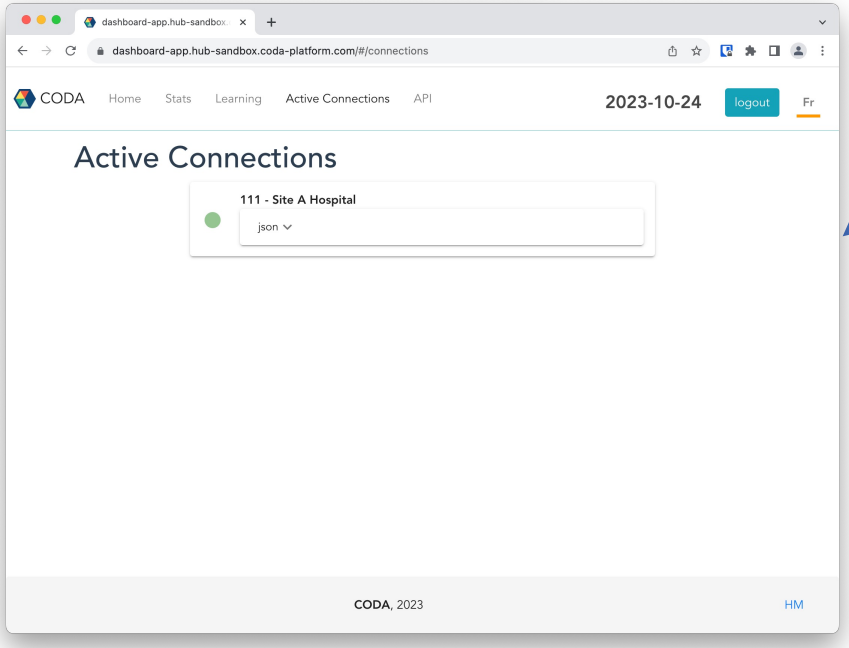
Open `dashboard-app` app to get link to dashboard



Login using `test-user` credentials, i.e.

`CODA_DASHBOARD_APP_AUTH_CLIENT_ID`
`CODA_DASHBOARD_API_AUTH_CLIENT_SECRET`





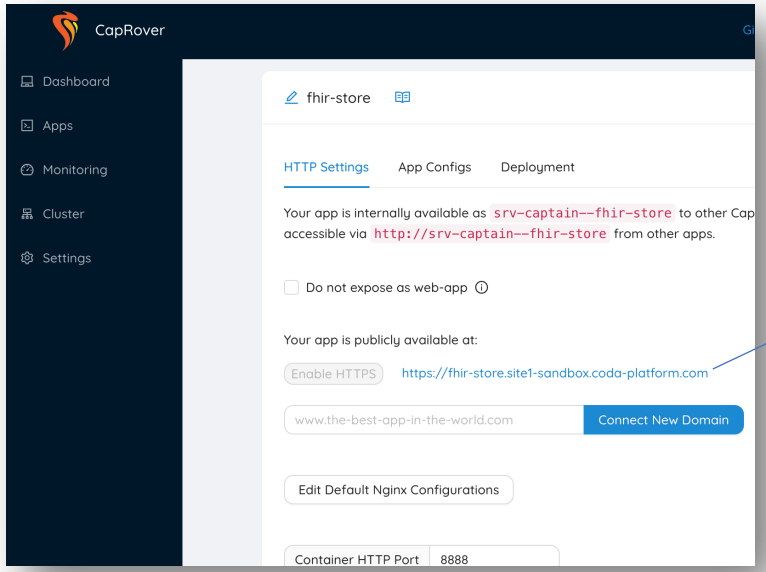
Confirm that able to successfully login to the dashboard. If Keycloak is properly configured, you should see that the site you created has already connected to the hub.

8. Load dummy FHIR data

Open the site CapRover control panel

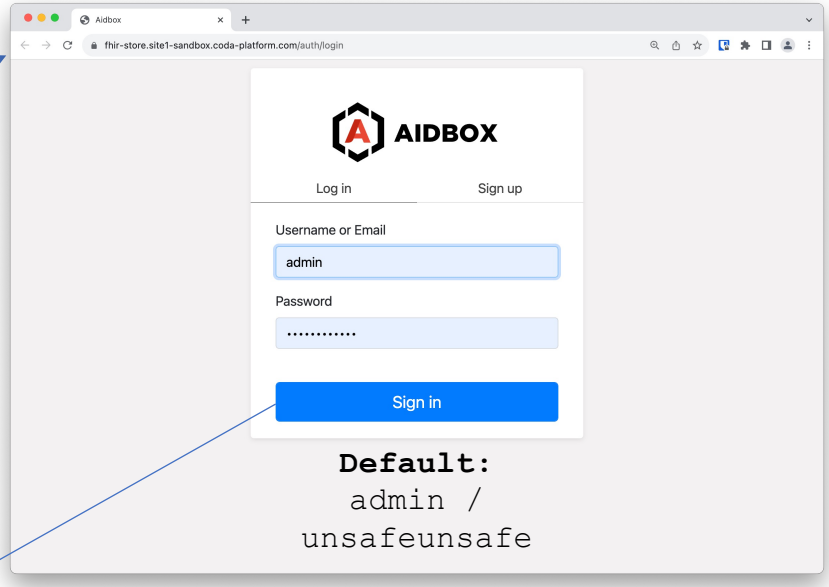
`https://captain.`**`your-site-subdomain.your-domain.com`**

Open `fhir-store` app to get link to AidBox

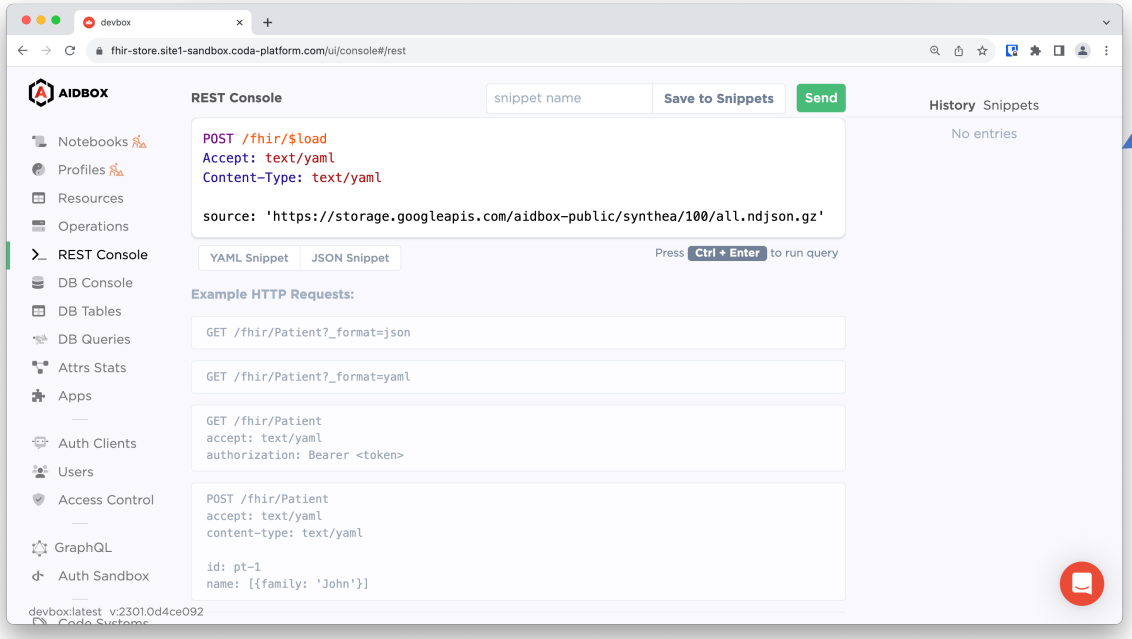


Login using FHIR store credentials, i.e.

`CODA_FHIR_STORE_DB_USERNAME`
`CODA_FHIR_STORE_DB_PASSWORD`



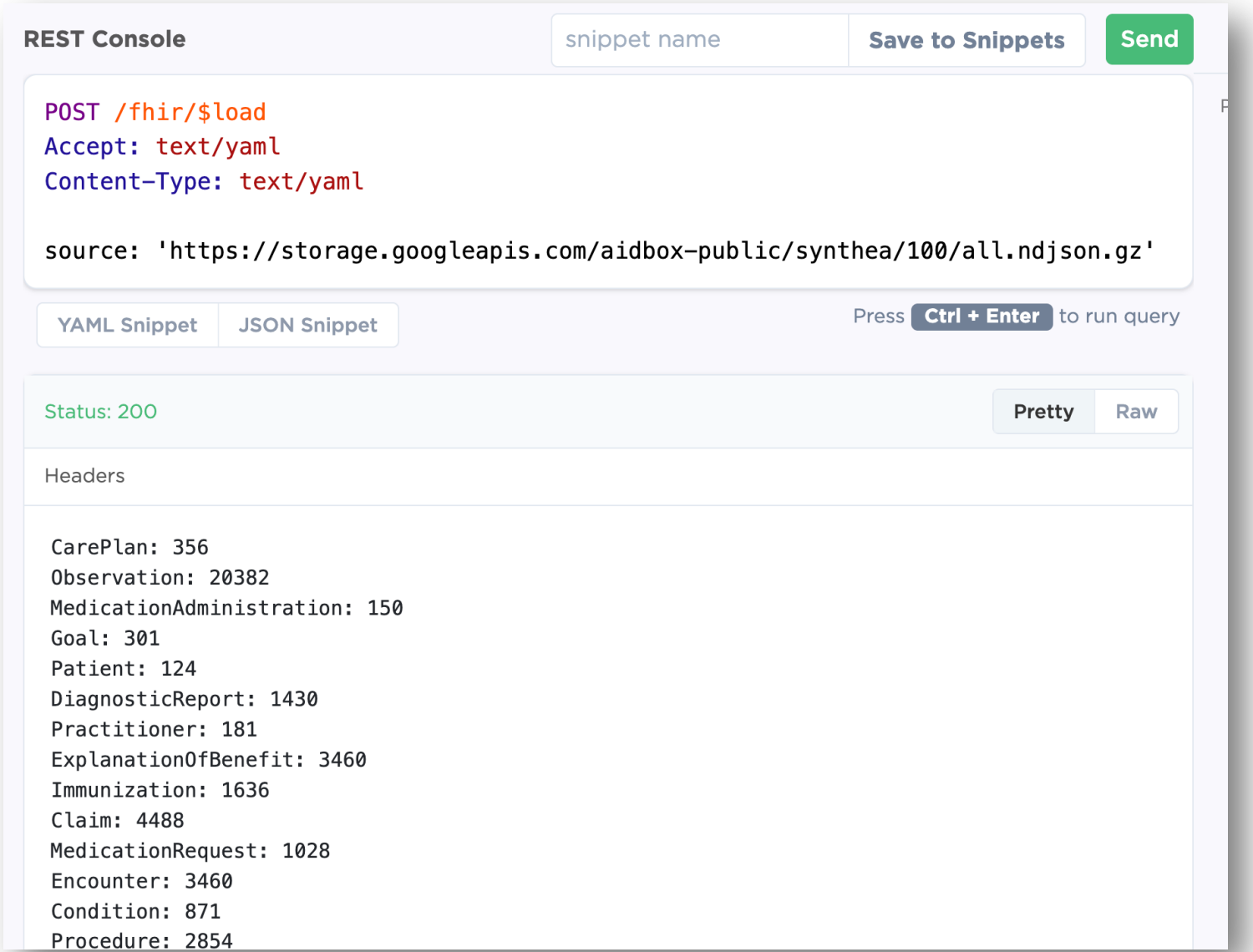
Default:
`admin / unsafeunsafe`



Confirm that able to login to Aidbox.
Go to REST console in menu, load Synthea data:

```
POST /fhir/$load
Accept: text/yaml
Content-Type: text/yaml

source: 'https://storage.googleapis.com/aidbox-public/synthea/100/all.ndjson.gz'
```

Expected response when Synthea data imported successfully:



9. Test federated analytics

Open the dashboard application URL

`https://dashboard-app.`**`your-hub-subdomain.your-domain.com`**

Login using **`test-user / unsafeunsafe`** if you haven't changed the defaults.

Querying age by site after creating 3 site nodes and loading them with variations of the same FHIR dataset.

CODA Home Stats Learning Active Connections API 04/09/2021 logout Fr

Select hospitals to include:

Site A Hospital × Site B Hospital × Site C Hospital ×

Select measures to retrieve:

Continuous variables (e.g. age)

count × mean × stdev × ci95 ×

Discrete (e.g. gender)

count × mode ×

Resources

Patient × Add (+)

Filters

AND OR Add filter Add group

Fields

age ×

Breakdown Disable Breakdown

Resource type

Patient

Resource attribute

age

Start (lower limit) 0

End (upper limit) 100

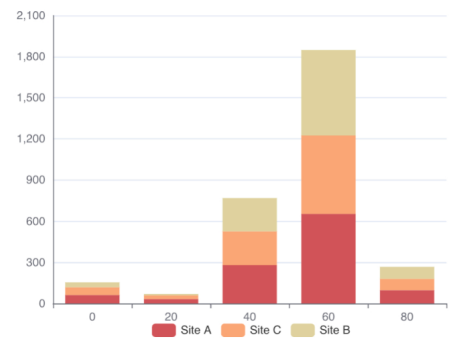
Stride (step size) 20

Results

Table 1. **Distribution of age by site**

0	20	40	60	80	Site	Total
57	27	246	572	84	Site C	988
36	11	243	623	87	Site B	1000
61	32	280	653	96	Site A	1124
154	70	769	1848	267	All	3112

Figure 1. **Distribution of age by site**

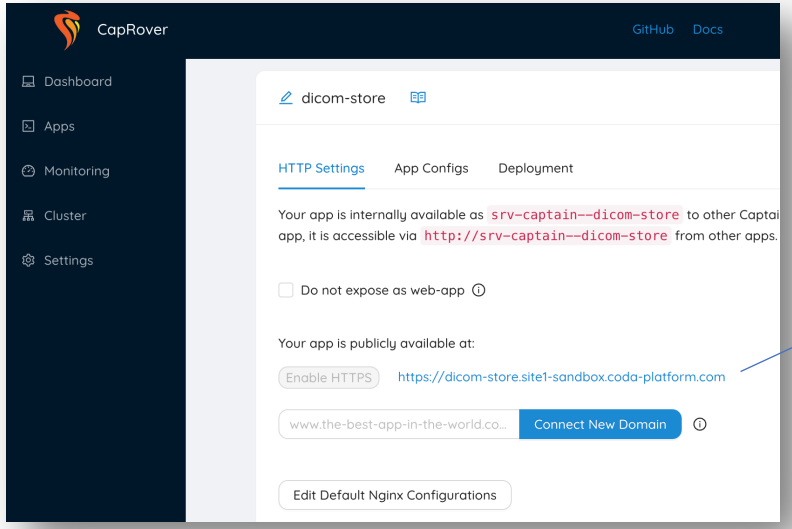


10. Load dummy DICOM data

Open the site CapRover control panel

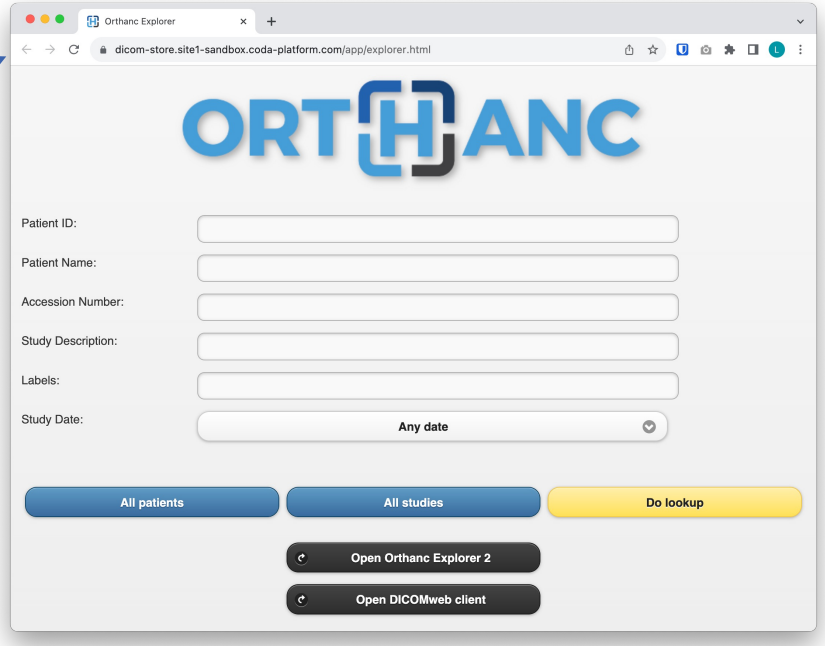
`https://captain.`**`your-site-subdomain.your-domain.com`**

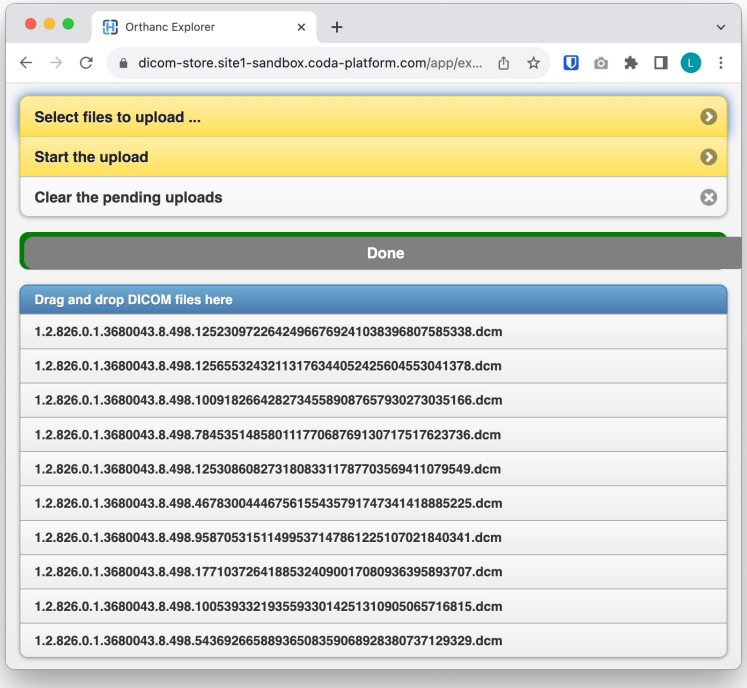
Open `dicom-store` app to get link to Orthanc



Login using Orthanc store credentials, i.e.

`CODA_DICOM_STORE_DB_USERNAME`
`CODA_DICOM_STORE_DB_PASSWORD`





Confirm that you are able to login with the default authentication credentials, and upload some non-sensitive DICOMs.

There are many free datasets available, e.g.

`https://cloud.google.com/healthcare-api/docs/resources/public-datasets/nih-chest`

11. Test multi-modal FL

Open the site CapRover control panel

`https://captain.`**`your-site-subdomain.your-domain.com`**

Predict death from age, gender and chest X-ray

CODA Home Stats Learning Active Connections API 10/26/2023 logout Fr

Learning

Launch a federated learning model.

Prepare Request

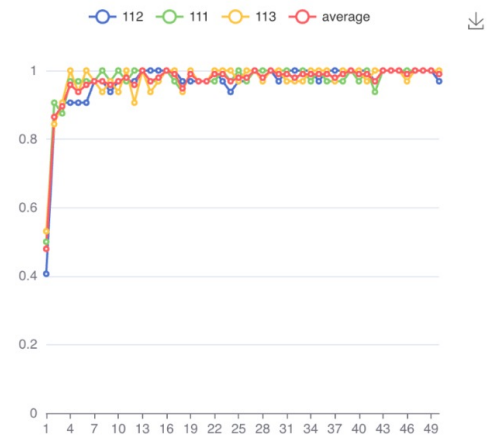
```
    ],
    "options": {
        "inputs": [
            "gender",
            "age",
            "chestXR"
        ],
        "outputs": [
            "isDeceased"
        ],
        "optimizer": {
            "name": "adam",
            "parameters": {
                "learning_rate": 0.00004,
                "validation_split": 0.33,
                "evaluation_split": 0.2,
                "epochs": 1,
                "batch_size": 16,
                "shuffle": 200
            }
```

Send Prepare Request

Data Count

Site Code	Count
113	100
112	100
111	100

Accuracy



Loss

Input configuration
(CODA API syntax)

```
{
  "selectors": [
    {
      "resource": "Patient",
      "label": "PA",
      "limit": 1000,
      "filters": [],
      "fields": [
        {
          "path": "age",
          "label": "age",
          "type": "integer"
        },
        {
          "path": "isDeceased",
          "label": "isDeceased",
          "type": "boolean"
        }
      ]
    },
    {
      "resource": "Observation",
      "label": "OB",
      "filters": [
        {
          "path": "code.coding.code",
          "operator": "is",
          "value": "20570-8",
          "type": "string"
        }
      ],
      "fields": [
        {
          "path": "value.Quantity.value",
          "label": "hematocrit",
          "type": "integer"
        }
      ]
    }
  ]
}
```

Model configuration
(Keras JSON syntax)

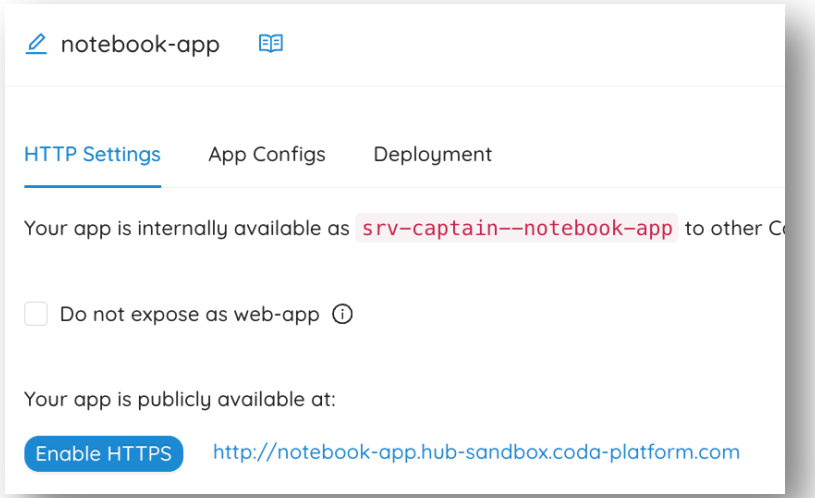
```
{
  "options": {
    "model": {
      "class_name": "Sequential",
      "config": {
        "name": "sequential_1",
        "layers": [
          "etc..."
        ]
      }
    },
    "inputs": [
      "gender",
      "age",
      "hematocrit"
    ],
    "outputs": [
      "isDeceased"
    ],
    "optimizer": {
      "name": "adam",
      "parameters": {
        "learning_rate": 0.00025,
        "validation_split": 0.33,
        "evaluation_split": 0.2,
        "epochs": 1,
        "batch_size": 20,
        "shuffle": 1000
      }
    },
    "compiler": {
      "parameters": {
        "loss": "binaryCrossentropy",
        "metrics": [
          "accuracy"
        ]
      }
    }
  }
}
```

12. Test interactive notebooks

Open the hub CapRover control panel

`https://captain.`**`your-hub-subdomain.your-domain.com`**

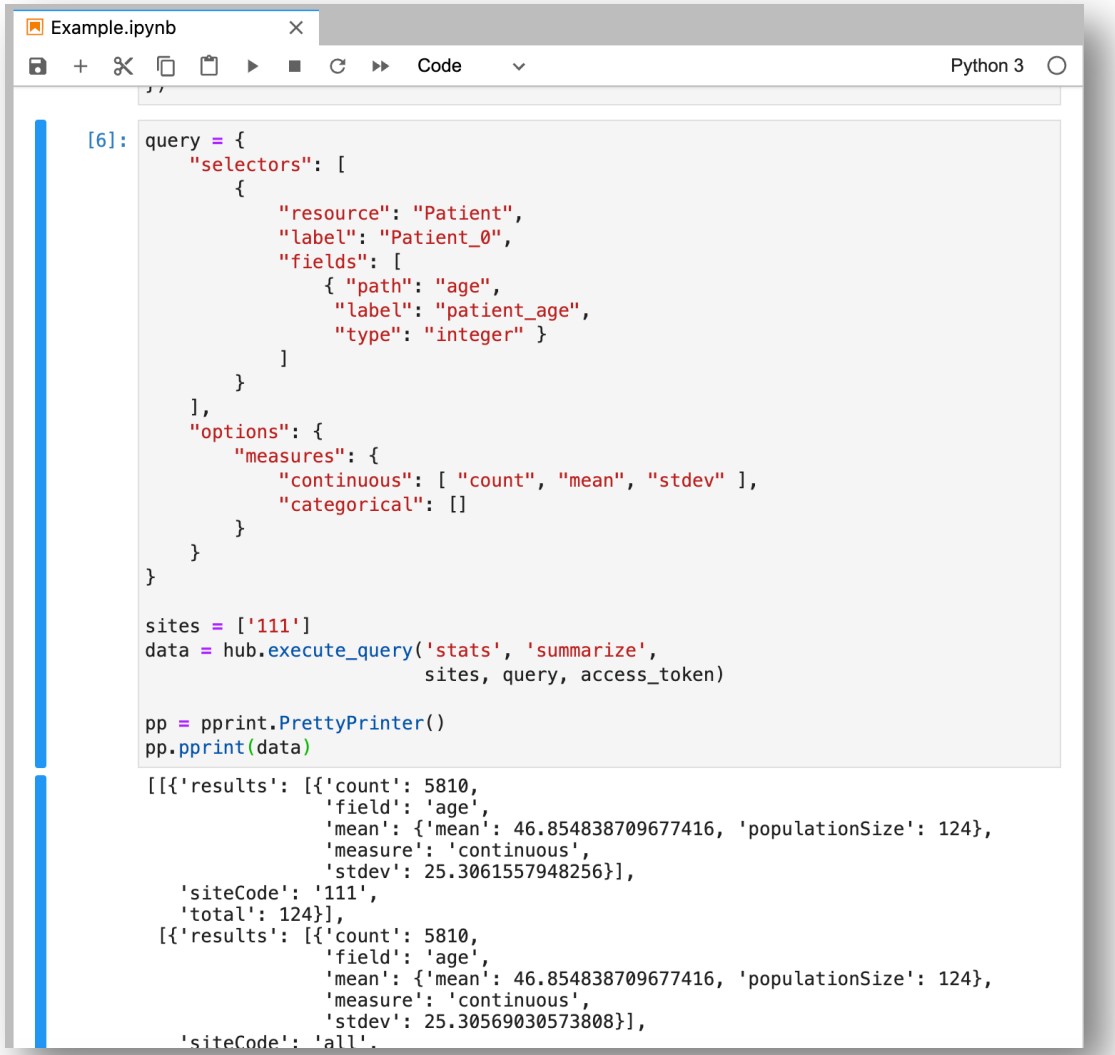
1. Open `notebook-app` in the hub CapRover and click on the link to open JupyterLab



2. Install the `coda_api` Python library inside the notebook and fetch an access token

```
[3]: %%capture
     !pip install coda_api

[5]: from coda_api import hub
     import pprint

     access_token = hub.get_access_token({
       'username': 'test-user',
       'password': 'unsafeunsafe',
       'client_id': 'notebook-app',
       'client_secret': '3bqaJltl87O12qH6ccDnKXhYlfQYqCgX',
       'grant_type': 'password'

     })
```

3. Use the `coda_api` to query the hub from the Jupyter notebook environment

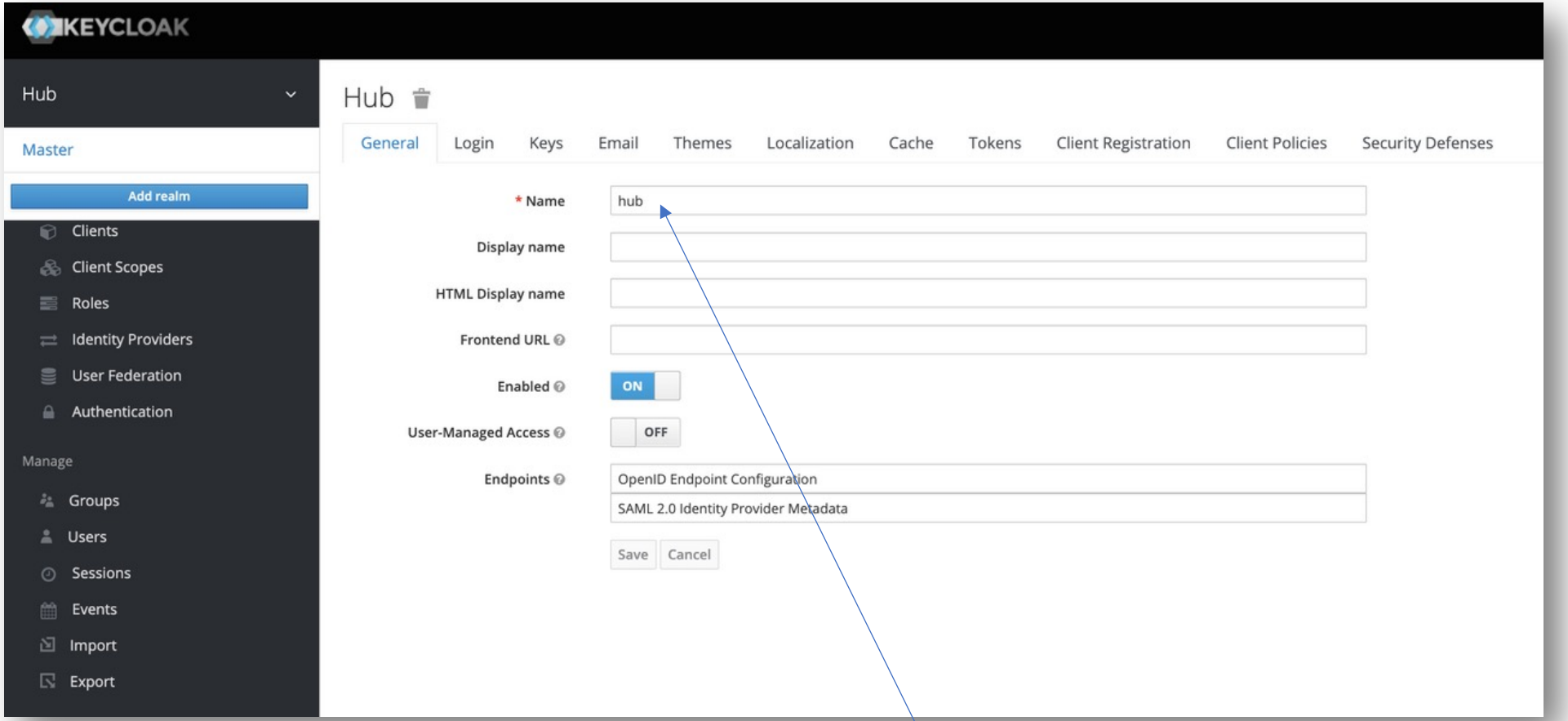


For more examples, visit the notebook-app GitHub:
https://github.com/coda-platform/notebook-app

Appendix A

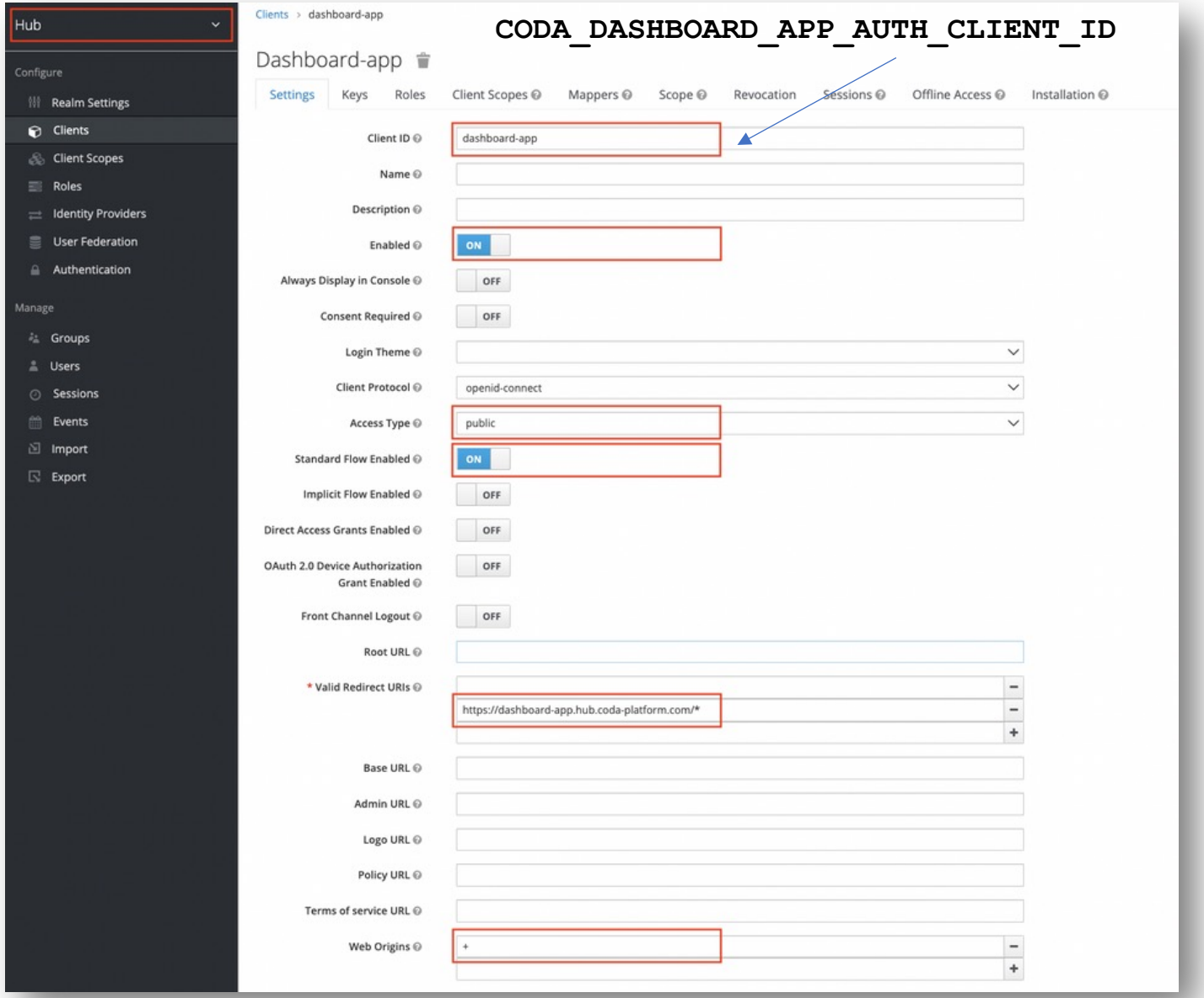
Detailed Keycloak setup

Create new realm named “hub”

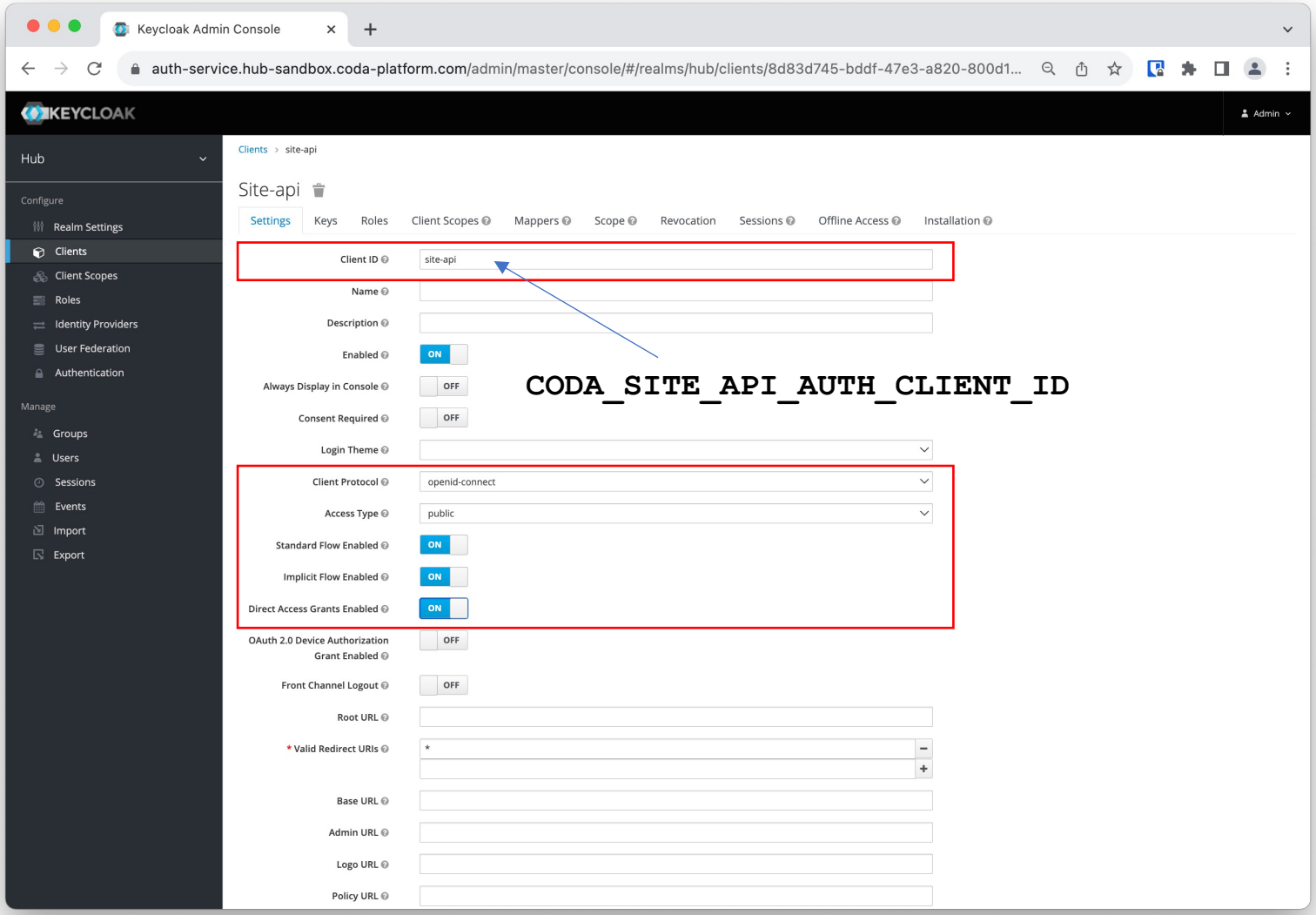


CODA_HUB_API_AUTH_REALM

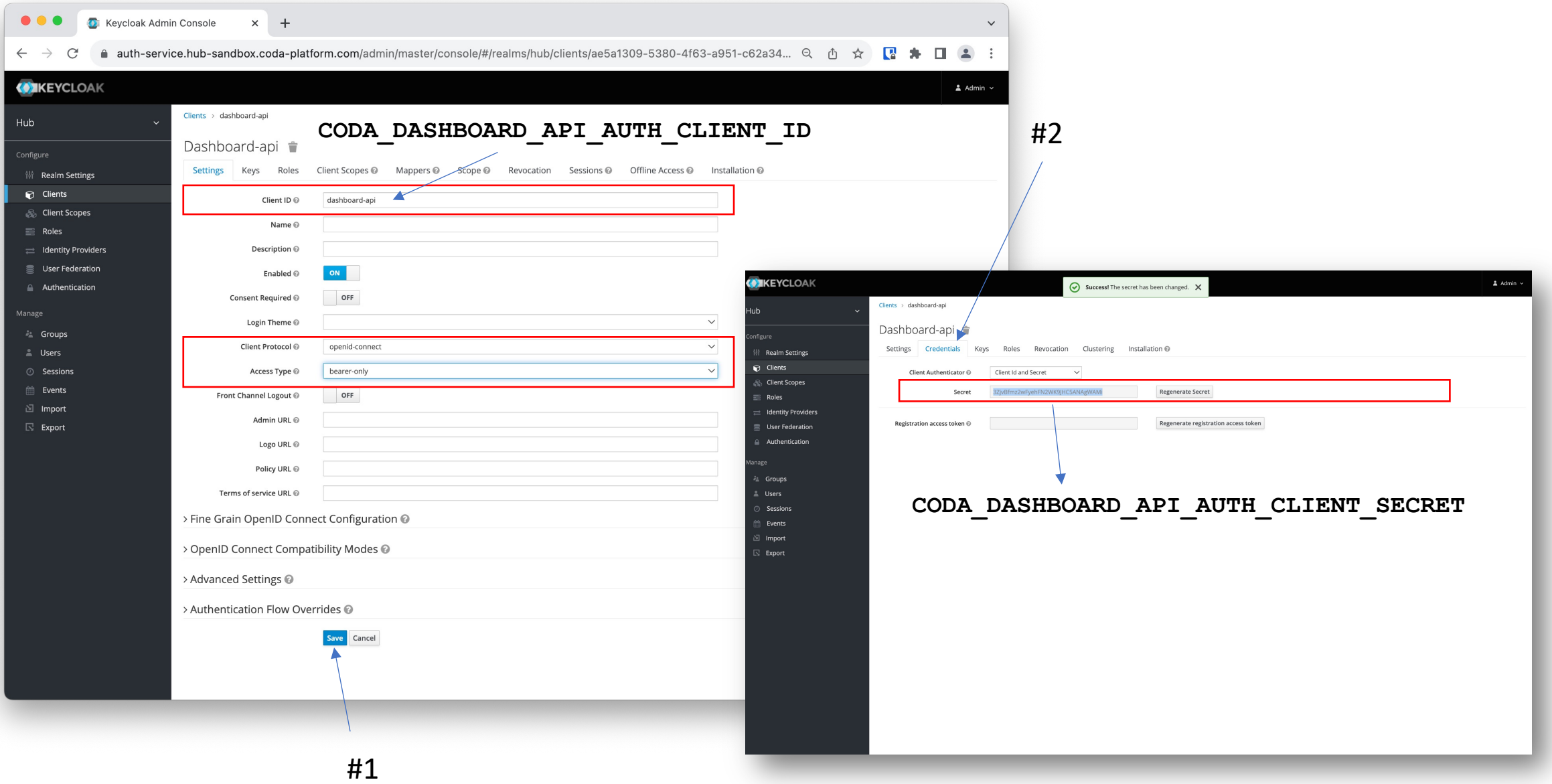
Create new client named “dashboard-app”



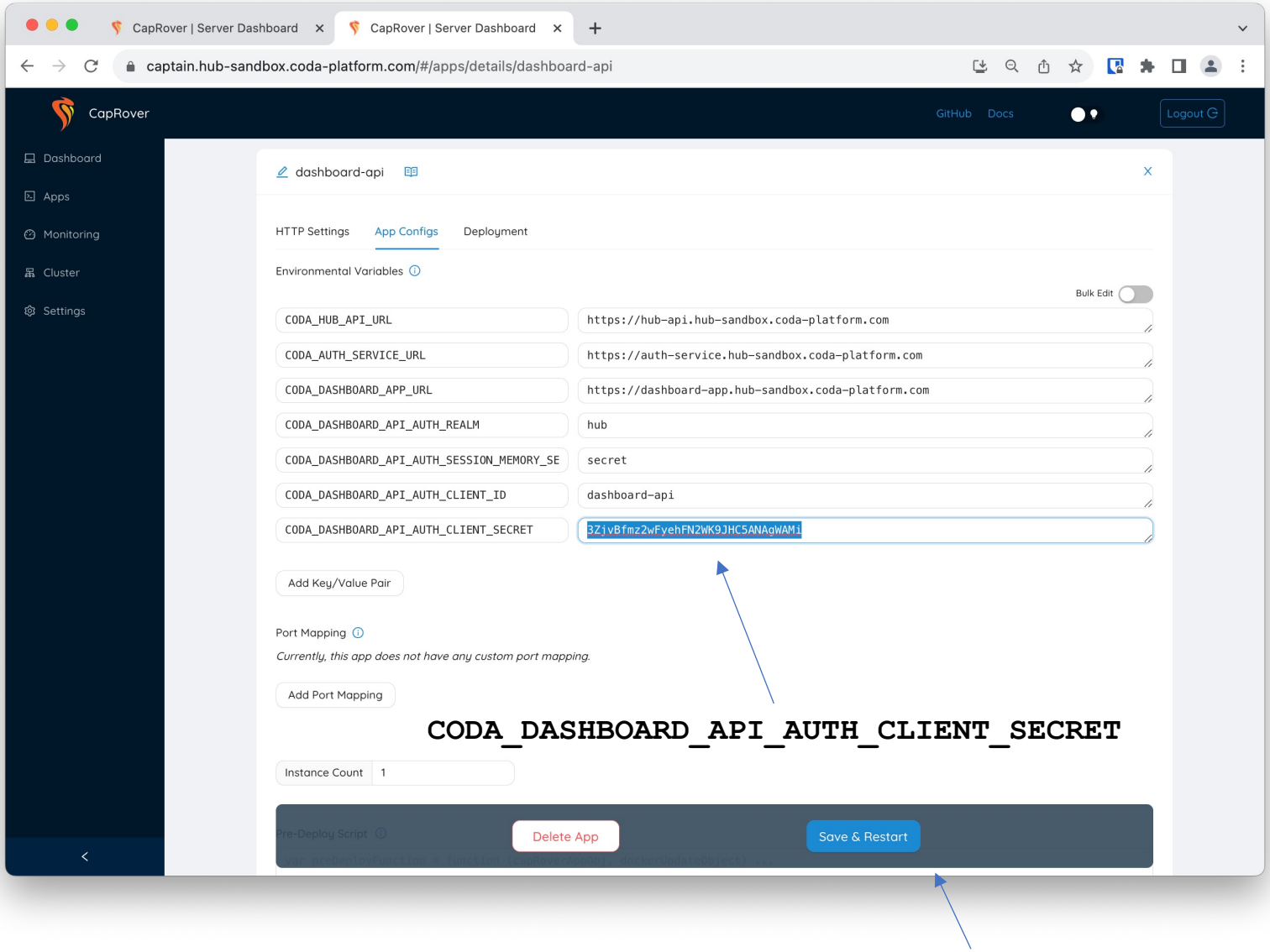
Create new client named “site-api”



Create new client named “dashboard-api”

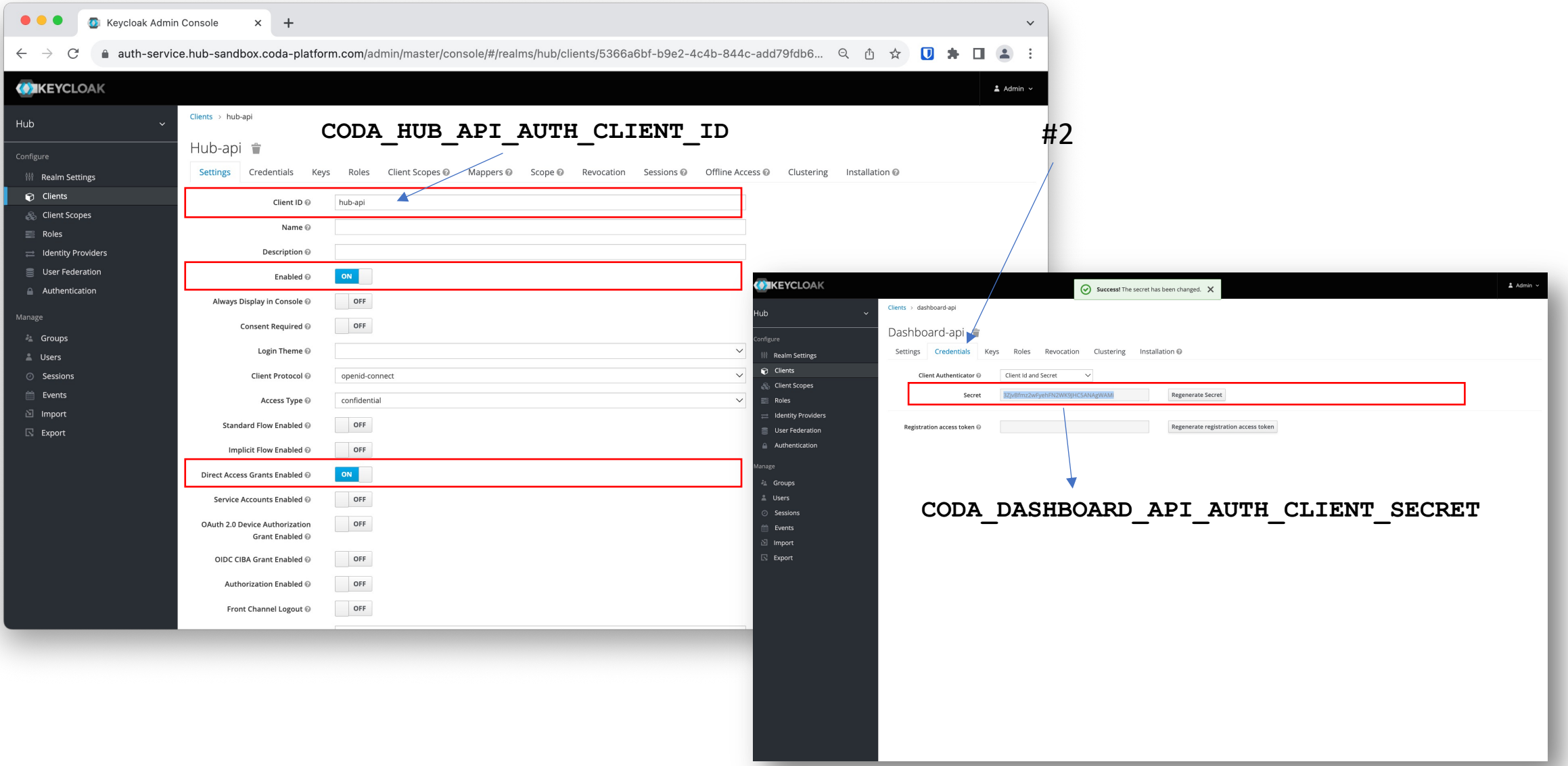


Update and restart secrets in dashboard-api app

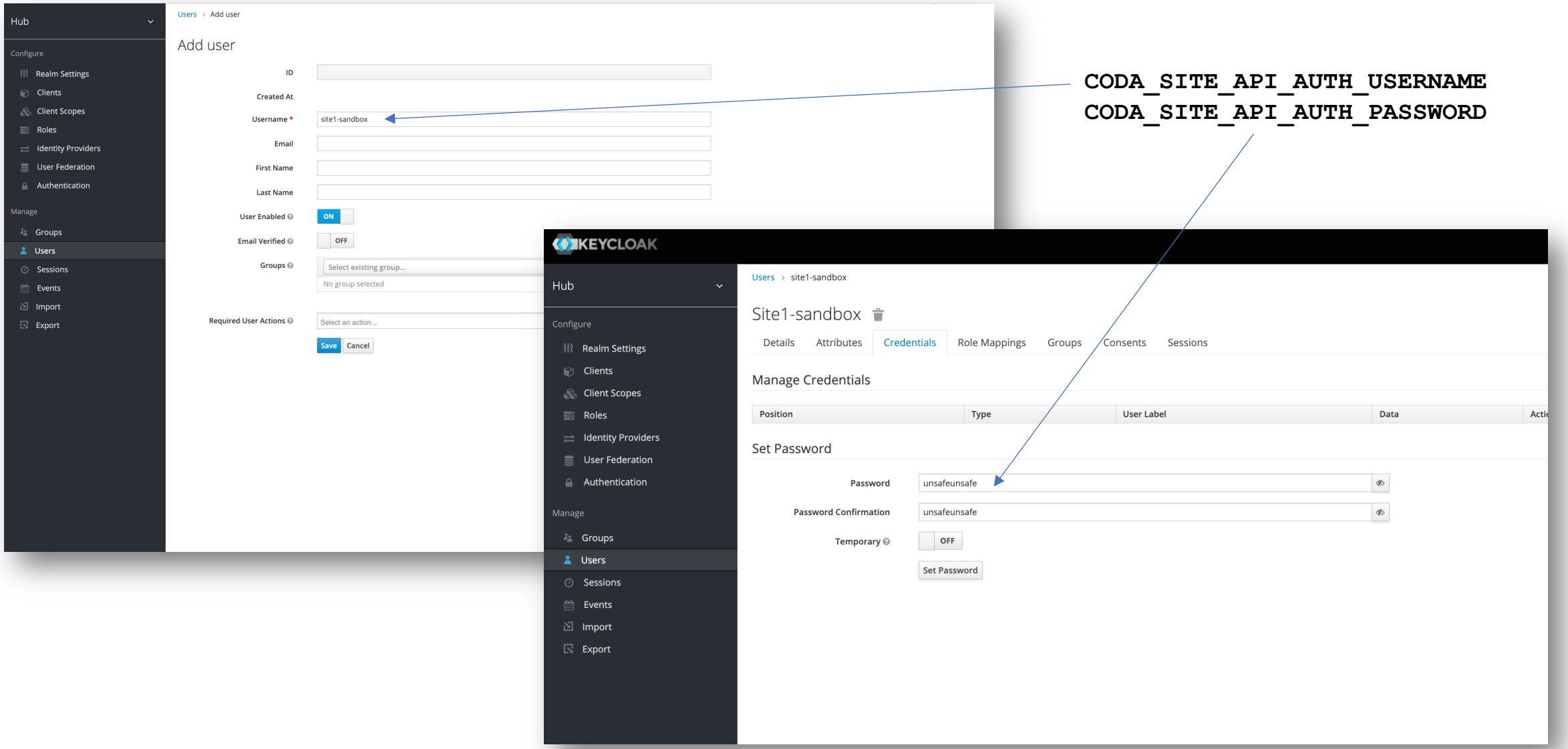


Update to value obtained from Keycloak, then click “Save & Restart”

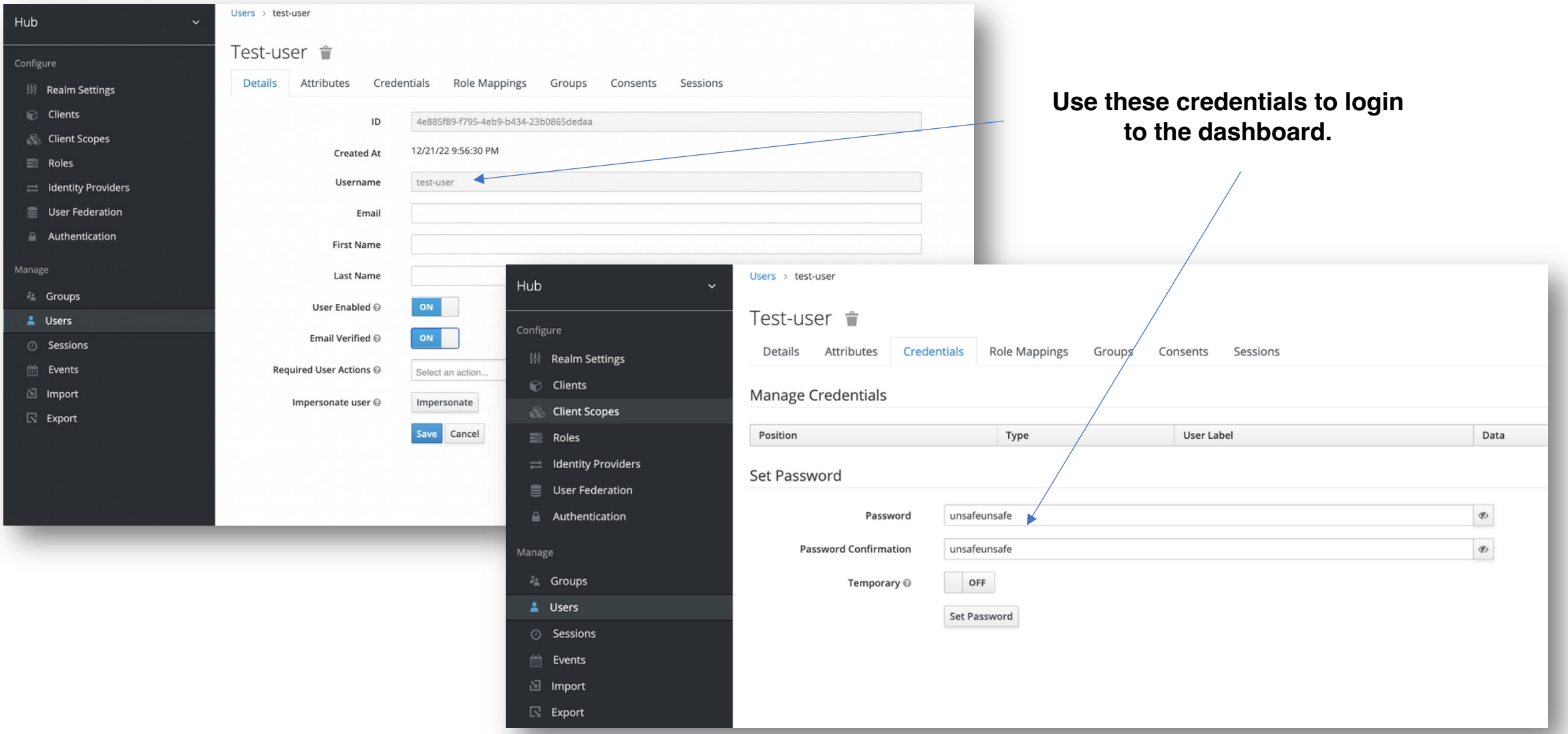
Create new client named “hub-api”



Create new user named “site1-sandbox”



Create new user named “test-user”



Use these credentials to login to the dashboard.

Appendix B

Interacting with CODA using the REST API

POST `{{CODA_AUTH_SERVICE_URL}}/realms/{{CODA_DASHBOARD_API_AUTH_REALM}}/protocol/openid-connect/token`

Example request

KEY	VALUE
username	site1-sandbox
password	unsafeunsafe
client_id	site-api
grant_type	password

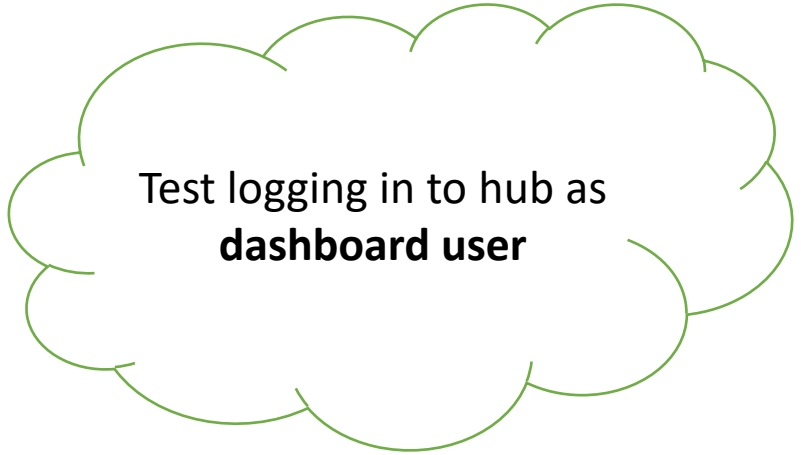
Example response


```
{
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUExgHhcBBKJe...",
    "expires_in": 300,
    "refresh_expires_in": 1800,
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2l...",
    "token_type": "Bearer",
    "not-before-policy": 0,
    "session_state": "378a9517-ac6a-414d-8056-b73c6f0c365a",
    "scope": "email profile"
}
```

POST `{{CODA_AUTH_SERVICE_URL}}/realms/{{CODA_DASHBOARD_API_AUTH_REALM}}/protocol/openid-connect/token`

Example request

KEY	VALUE
username	test-user
password	unsafeunsafe
client_id	dashboard-api
client_secret	vTruOr8IfSV2IjXfVvhi0iMddpEChN1G
grant_type	password



Example response


```
{
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2l...",
    "expires_in": 300,
    "refresh_expires_in": 1800,
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2...",
    "token_type": "Bearer",
    "not-before-policy": 0,
    "session_state": "661ee647-0bc6-4ddc-891a-cc6c675022cb",
    "scope": "email profile"
}
```

GET {{CODA_DASHBOARD_API_URL}}/sites

Example request

KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Token from response to `/openid-connect/token` when logging in as dashboard user (previous slide)

Test requesting a list of sites and catalog of available resources

Example response


```
{
    "connections": [
        {
            "uid": "111",
            "name": "Site A Hospital",
            "names": {
                "fr": "Hôpital Site A",
                "en": "Site A Hospital"
            },
            "api_version": "1.0.1",
            "resources": [
                {
                    "resource": "Patient",
                    "count": {
                        "estimate": 124
                    }
                },
                "more resources here..."
            ],
            "last_seen": "2023-10-26T10:29:32.363Z"
        },
        {
            "uid": "112",
            "name": "Site B Hospital",
            "names": {
                "fr": "Hôpital Site B",
                "en": "Site B Hospital"
```

GET {{CODA_HUB_API_URL}}/stats/summarize?sites=111,112

Example request

KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Example request body

```
{
    "selectors": [
        {
            "resource": "Patient",
            "label": "patient",
            "fields": [
                {
                    "path": "age",
                    "label": "patient_age",
                    "type": "integer"
                }
            ]
        }
    ],
    "options": {
        "measures": {
            "continuous": [
                "count",
                "mean",
                "stdev"
            ],
            "categorical": []
        }
    }
}
```

Example response

```
[
    [
        {
            "siteCode": "111",
            "total": 124,
            "results": [
                {
                    "field": "age",
                    "measure": "continuous",
                    "count": 5810,
                    "mean": {
                        "mean": 46.854838709677416,
                        "populationSize": 124
                    },
                    "stdev": 25.3061557948256
                }
            ]
        }
    ],
    [
        {
            "siteCode": "112",
            "total": 135,
            "results": [
                {
                    "field": "age",
                    "measure": "continuous",
```

Test federated analytics functionality

GET {{CODA_HUB_API_URL}}/learning/prepare?sites=111,112

Example request headers (same as previous)

KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Example request body

```
{
    "selectors": [
        {
            "resource": "Patient",
            "label": "patient",
            "limit": 1000,
            "filters": [],
            "fields": [
              { "path": "age", "label": "age", "type": "integer" },
              { "path": "isDeceased", "label": "isDeceased", "type": "boolean" }
            ]
        }
    ],
    "options": {
        "model": {},
        "inputs": ["age"],
        "outputs": ["isDeceased"],
        "optimizer": {
            "name": "adam",
            "parameters": { "batch_size": 2, "learning_rate": 0.0001 }
        },
        "compiler": {
            "parameters": {
                "loss": "binaryCrossentropy",
                "metrics": ["accuracy"]
            }
        }
    }
}
```

Example response

```
[
    {
        "job": "CcVNAu+LC89V/S3D",
        "siteCode": "111",
        "count": 124
    },
    {
        "job": "CcVNAu+LC89V/S3D",
        "siteCode": "112",
        "count": 124
    }
]
```

Test preparing a dataset for a training task

GET `{{CODA_HUB_API_URL}}/learning/train?sites=111,112`

Example request headers (same as previous)

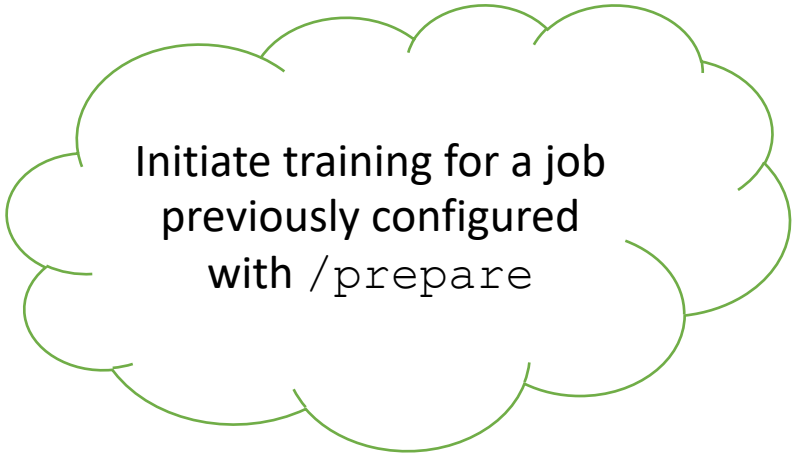
KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Example request body

```
{
    "job": "FAdmIfw5FxzhiDHJ",
    "rounds": 2
}
```

Example request response

```
{
    "job": "FAdmIfw5FxzhiDHJ",
    "rounds": 2
}

```



GET {{CODA_HUB_API_URL}}/learning/progress?sites=111,112

Example request headers (same as previous)

KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Example request body

```
{
    "job": "FAdmIfw5FxzhiDHJ"
}
```

Fetch the training progress for a job initiated with `/train`

Example request response


```
[
    {
        "acc": 0.75,
        "loss": 0.6873877048492432,
        "val_acc": 0.8064516186714172,
        "val_loss": 0.6868967413902283,
        "siteCode": "111",
        "currentRound": 1,
        "totalRounds": 2
    },
    {
        "acc": 0.7983870506286621,
        "loss": 0.6893560886383057,
        "val_acc": 0.8064516186714172,
        "val_loss": 0.6865110993385315,
        "siteCode": "112",
        "currentRound": 1,
        "totalRounds": 2
    },
    {
        "acc": 0.774193525314331,
        "loss": 0.6883718967437744,
        "val_acc": 0.8064516186714172,
        "val_loss": 0.6867039203643799,
        "siteCode": "average",
        "currentRound": 1,
        "totalRounds": 2
    }
]
```

GET {{CODA_HUB_API_URL}}/learning/evaluate?sites=111,112

Example request headers (same as previous)

KEY	VALUE
Content-Type	application/json
Authorization	Bearer {{access_token}}

Example request body

```
{
    "job": "FAdmIfw5FxzhiDHJ"
}
```

Compute evaluation metrics for a job completed by `/train`

Example response


```
[
    {
        "siteCode": "average",
        "acc": 0.8064516186714172,
        "loss": 0.6836749911308289
    },
    {
        "siteCode": "112",
        "acc": 0.8064516186714172,
        "loss": 0.6825025677680969
    },
    {
        "siteCode": "111",
        "acc": 0.8064516186714172,
        "loss": 0.6848474144935608
    }
]
```