

Intelligent Claims System

Installation Guide

Prerequisites

Before proceeding, please check that your system fulfills the following requirements:

Docker

Please ensure that Docker is installed in your system. To install, follow the guide here:

<https://www.docker.com/get-started>

Docker Compose

Please ensure that Docker Compose is installed in your system. To install, follow the guide here:

<https://docs.docker.com/compose/install/>

Ngrok

Please ensure that Ngrok is installed in your system. To install, follow the guide here:

<https://ngrok.com/download>

Network Ports

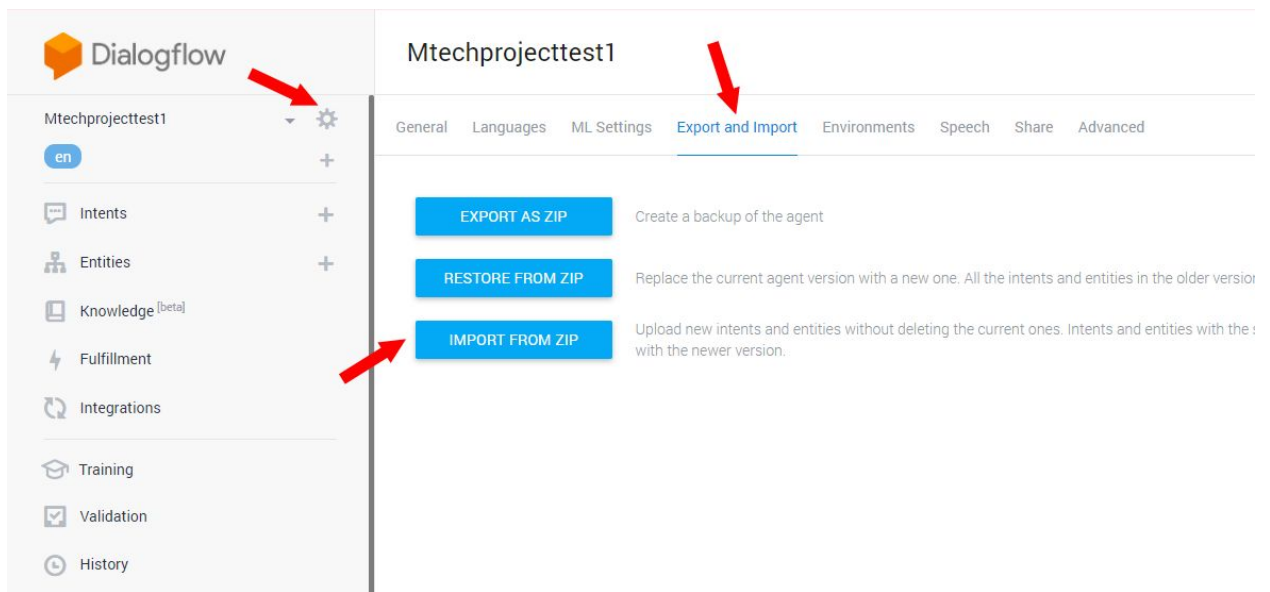
Please ensure that the following ports on your machine are free:

- 3307
- 5000
- 5001
- 5003
- 5672
- 8081
- 9001
- 9999
- 15672

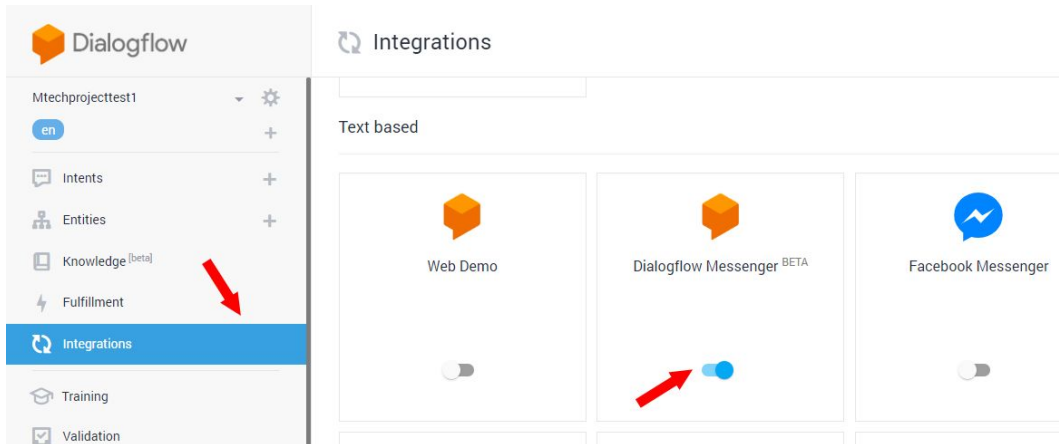
In the event that you need these ports, you can configure the system to use other ports instead. To change the port number being used, open `/docker-compose.yml` in the project, and replace all instances of the ports you wish to replace, with a free port.

Installation

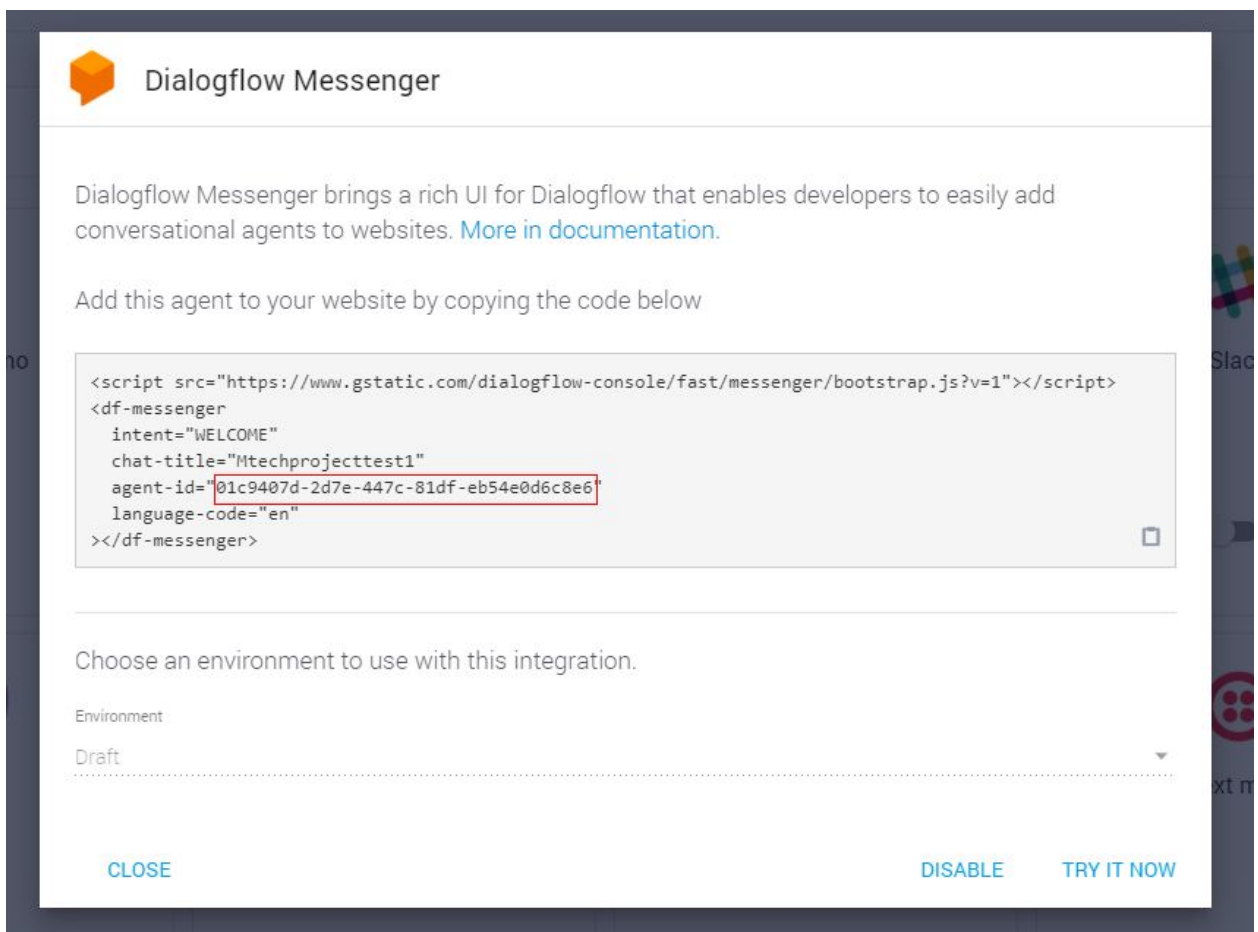
1. Ensure the above prerequisites have been fulfilled.
2. Clone the git repository from https://github.com/francislow82/mtech_irs_team15 to your local file system.
3. Create new Dialogflow agent at <http://dialogflow.com>
4. Import agent by going to settings > “Export / Import” > “Import from ZIP”. Use the zip file provided at SystemCode/claimBot/ClaimBot.zip.



5. Go to Integrations and turn on Dialogflow Messenger



6. A window should popup. Copy the agent-id from the provided script



7. Paste into /SystemCode/docker-compose.yml > claimsubmissionfrontend > DIALOGFLOW_AGENT_ID

```
# CLAIM SUBMISSION FRONTEND
claimsubmissionfrontend:
  build: ./claimSubmission/frontend
  ports:
    - "5001:80"
  restart: always
  links:
    - "claimsubmissionserver"
  environment:
    API_HOST: http://127.0.0.1:5000
    DIALOGFLOW_AGENT_ID: 01c9407d-2d7e-447c-81df-eb54e0d6c8e6
```

8. Go to settings > General and copy the projectID

The screenshot shows the Google Dialogflow console interface. On the left is a sidebar with navigation options: Mtechprojecttest1, Intents, Entities, Knowledge (beta), Fulfillment, Integrations, Training, Validation, History, Analytics, Prebuilt Agents, and Small Talk. The main panel displays the 'General' settings for the project 'Mtechprojecttest1'. The 'DEFAULT TIME ZONE' is set to '(GMT+8:00) Asia/Hong_Kong'. Under 'GOOGLE PROJECT', the 'Project ID' is highlighted with a red box and contains the value '01c9407d-2d7e-447c-81df-eb54e0d6c8e6'. The 'Service Account' is listed as 'iam.gserviceaccount.com'. The 'API VERSION' is set to 'V2 API'. Under 'BETA FEATURES', the 'Enable beta features and APIs' toggle is turned on. The 'API KEYS (V1)' section is visible at the bottom.

9. Replace the original value in /SystemCode/docker-compose.yml: services > claimsubmissionserver > environment > DIALOG_FLOW_PROJECT_ID

```

45 # CLAIM SUBMISSION SERVER
46 claimsubmissionserver:
47   build: ./claimSubmission/server
48   ports:
49     - "5000:5000"
50   restart: always
51   links: |
52     - "claimrepository"
53   environment:
54     HOST: 0.0.0.0
55     PORT: 5000
56     DIALOG_FLOW_PROJECT_ID: claimbot-vxtlgn
57     GOOGLE_APPLICATION_CREDENTIALS: private_key.json
58     CLAIM_REPOSITORY_HOST: http://claimrepository:8081
59     OCR_FEATURE: 1
60     GOOGLE_API_KEY: AIzaSyBuZXEcDehWONb3Np0axuAjoJg8l9K00pw
61 #   PYTHONUNBUFFERED: 1
62

```

10. On dialogflow.com, click on Service Account

The screenshot shows the Dialogflow console interface for a project named 'Mtechprojecttest1'. The left sidebar contains a navigation menu with options: Intents, Entities, Knowledge (beta), Fulfillment, Integrations, Training, Validation, History, Analytics, Prebuilt Agents, and Small Talk. The main panel displays the 'General' tab with the following settings:

- DEFAULT TIME ZONE:** (GMT+8:00) Asia/Hong_Kong
- GOOGLE PROJECT:**
 - Project ID: [redacted]
 - Service Account: [redacted] iam.gserviceaccount.com
- API VERSION:** V2 API (selected)
 - Use Cloud API as default for the agent. Your webhook will receive and return V2 format messages.
- BETA FEATURES:**
 - Enable beta features and APIs (checked)
 - Be the first to get access to the newest features and latest APIs. (Full V2-beta API reference)
- API KEYS (V1):** [empty field]

A 'SAVE' button is visible in the top right corner of the main panel.

11. If you don't already have a Google Cloud Platform account, you may be prompted to create one

12. After that, it should bring you to the Google Cloud Platform console. Click on the matching account

The screenshot shows the Google Cloud Platform IAM & Admin console for project 'Mtechprojecttest1'. The left sidebar lists navigation options: IAM, Identity & Organization, Policy Troubleshooter, Organization Policies, Quotas, Service Accounts (highlighted), Labels, and Settings. The main content area is titled 'Service accounts' and includes a '+ CREATE SERVICE ACCOUNT' button and a 'DELETE' button. Below this, there is a section 'Service accounts for project "Mtechprojecttest1"' with a brief description and a link to 'Learn n'. A table lists the service accounts with columns for checkboxes, Email, Status, Name, and Description. One service account is highlighted with a red box.

	Email	Status	Name	Description
<input type="checkbox"/>	[redacted].gserviceaccount.com	✓	Dialogflow Integrations	
<input type="checkbox"/>	[redacted].gserviceaccount.com	✓	Dialogflow Integrations	

13. Click on Edit, then create Key

The screenshot shows the 'Dialogflow Integrations' service account details page in the Google Cloud Platform IAM & Admin console. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Dialogflow Integrations' and includes an 'EDIT' button (highlighted with a red arrow) and a 'DELETE' button. Below this, there is a section 'Service account details' with fields for Name, Description, Email, and Unique ID. The 'Service account status' section shows 'Account currently active' and a 'DISABLE SERVICE ACCOUNT' button. The 'Keys' section shows a list of keys with a 'Key ID' and a 'Key ID' field. A red arrow points to the '+ CREATE KEY' button at the bottom of the page.

Service account details

Name: Dialogflow Integrations

Description:

Email: dialogflow-kvaxmt@mtechprojecttest1-xyawd.iam.gserviceaccount.com

Unique ID: 109095603882999736371

Service account status

Disabling your account allows you to preserve your policies without having to delete it.

✓ Account currently active

DISABLE SERVICE ACCOUNT

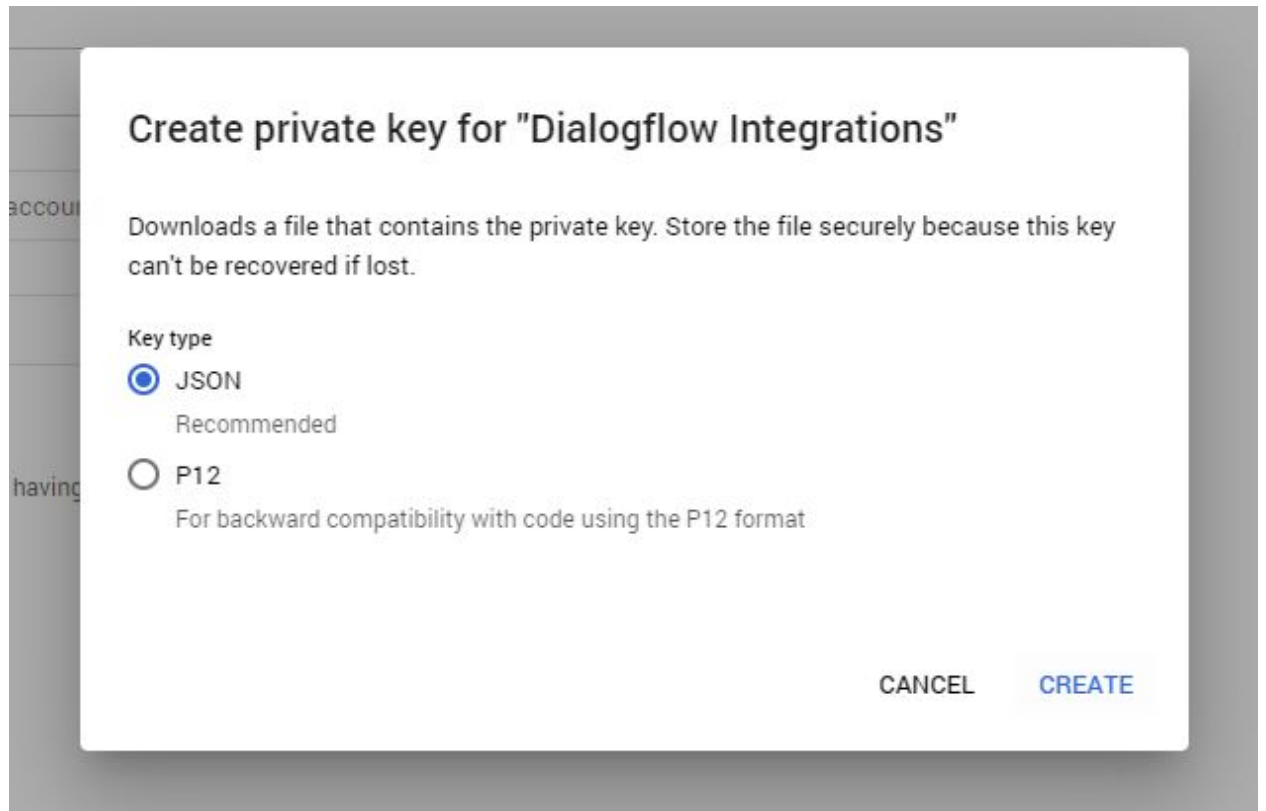
SHOW DOMAIN-WIDE DELEGATION

Keys

Key ID: 1c8b7b1223b50f840d50bc630b7c3a572d7e9da7

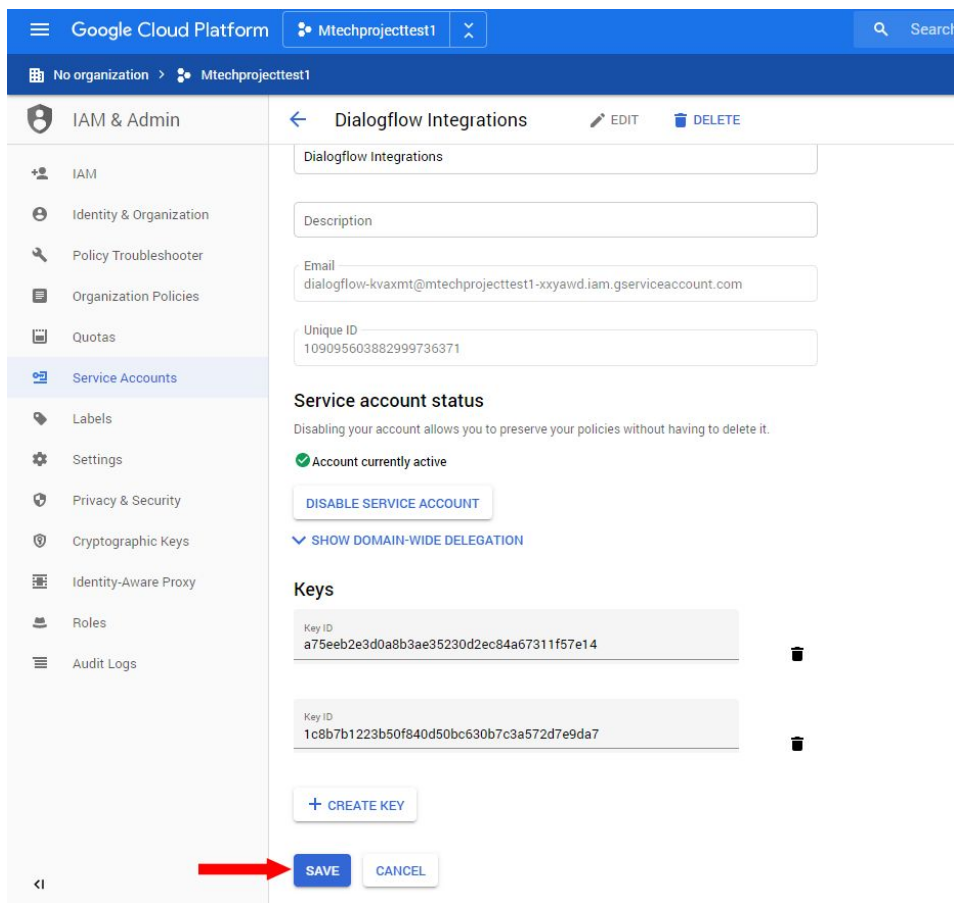
+ CREATE KEY

14. Choose JSON and click "Create"



15. A file will be downloaded to your machine

16. Click on “Save”



17. Copy rename that file as “private_key.json” and use that to overwrite the file at /SystemCode/claimSubmission/server/private_key.json

Running

1. Navigate to your ngrok installation directory, and run the command: `ngrok http 5000`. that the Docker host is set to be localhost / 127.0.0.1. It may differ based on your configuration. If it is not, run this command instead: `ngrok http <ENTER YOUR DOCKER HOST>:5000`
2. Copy the https forwarding url (the one in the red box)

Session Status	online					
Account	[REDACTED]					
Update	update available (version 2.3.35, Ctrl-U to update)					
Version	2.3.23					
Region	United States (us)					
Web Interface	http://127.0.0.1:4040					
Forwarding	http://8d8ccc59.ngrok.io -> http://localhost:5000					
Forwarding	https://8d8ccc59.ngrok.io -> http://localhost:5000					
Connections	ttl	opn	rt1	rt5	p50	p90
	0	0	0.00	0.00	0.00	0.00

3. Login to Dialogflow at: <https://dialogflow.com/>
4. Navigate to the Fulfillment tab, and update the Webhook URL with the forwarding url from step 2. Note to append /claim/intenthandler at the end of it. For example, if forwarding url "https://12345.ngrok.io", then enter
["https://12345.ngrok.io/claim/intenthandler"](https://12345.ngrok.io/claim/intenthandler)

The screenshot shows the Dialogflow Fulfillment configuration page. The 'Webhook' section is active, with a toggle switch set to 'ENABLED'. The 'URL*' field is highlighted with a red box and contains the text 'https://53d3ff8.ngrok.io/claim/intenthandler'. Below this, there are fields for 'BASIC AUTH' (username and password), 'HEADERS' (key and value), and 'SMALL TALK' (a dropdown menu). At the bottom, there is an 'Inline Editor' section for managing fulfillment code via Cloud Functions.

5. Open command prompt (Windows) / terminal (Mac / Linux) and navigate to the /SystemCode of the cloned repository
6. Run the following command: `docker-compose -p icms up --build`
7. You should see the docker services building / firing up.
8. Wait for it to complete. You should see something like that once all the services are up and running:

```

eventdispatcher_1 | 2020-05-07 12:50:24.760 [info] <0.306.0> Running boot step notify_cluster defined by app rabbit
eventdispatcher_1 | 2020-05-07 12:50:24.760 [info] <0.306.0> Running boot step networking defined by app rabbit
eventdispatcher_1 | 2020-05-07 12:50:24.765 [info] <0.446.0> started TCP listener on [::]:5672
eventdispatcher_1 | 2020-05-07 12:50:24.766 [info] <0.306.0> Running boot step cluster_name defined by app rabbit
eventdispatcher_1 | 2020-05-07 12:50:24.766 [info] <0.306.0> Running boot step direct_client defined by app rabbit
eventdispatcher_1 | 2020-05-07 12:50:24.767 [info] <0.306.0> Running boot step os_signal_handler defined by app rabbit
eventdispatcher_1 | 2020-05-07 12:50:24.767 [info] <0.448.0> Swapping OS signal event handler (erl_signal_server) for our own
eventdispatcher_1 | completed with 0 plugins.
eventdispatcher_1 | 2020-05-07 12:50:25.233 [info] <0.9.0> Server startup complete; 0 plugins started.
eventdispatcher_1 | 2020-05-07 12:50:28.193 [info] <0.455.0> accepting AMQP connection <0.455.0> (172.18.0.5:53960 -> 172.18.0.3:
5672)
eventdispatcher_1 | 2020-05-07 12:50:28.201 [info] <0.455.0> connection <0.455.0> (172.18.0.5:53960 -> 172.18.0.3:5672): user 'gu
est' authenticated and granted access to vhost '/'
eventdispatcher_1 | 2020-05-07 12:50:28.658 [info] <0.467.0> accepting AMQP connection <0.467.0> (172.18.0.6:53542 -> 172.18.0.3:
5672)
eventdispatcher_1 | 2020-05-07 12:50:28.717 [info] <0.467.0> connection <0.467.0> (172.18.0.6:53542 -> 172.18.0.3:5672): user 'gu
est' authenticated and granted access to vhost '/'
claimrepository_1 | Claims Repository service started on port: 8081

```

9. To visit the dashboard, open this url in a browser: <http://127.0.0.1:9001>. Note this assumes that the Docker host is 127.0.0.1. It may differ based on your configuration.
10. The dashboard should now load. For details on how to use the dashboard, please read the User Guide.

Stopping

1. On the same command prompt / terminal, press “Ctrl + C” (Windows) or “Command + C” (Mac)
2. Alternatively, open a separate command prompt / terminal and run the command:
`docker-compose -p icms down`