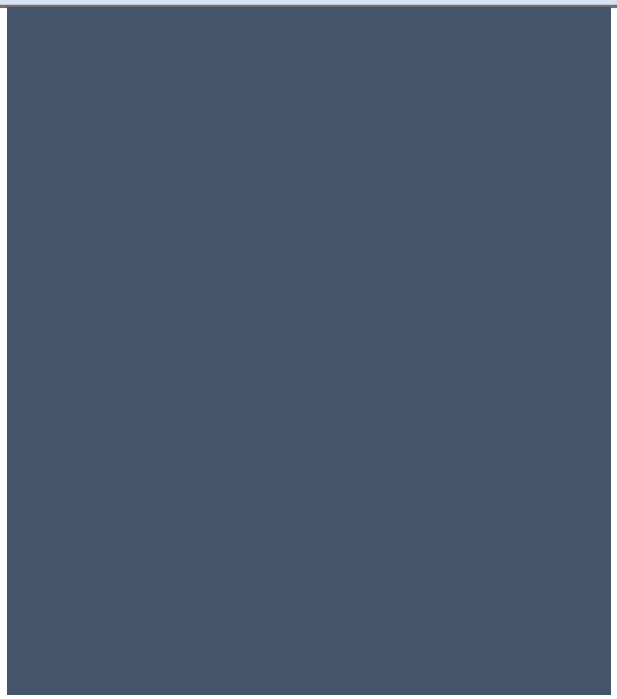


Adscope Installation Manual

Graymatics-SG Pte Ltd.

Content table:

Confidentiality

This document contains proprietary information that is confidential to Graymatics-SG Pte Ltd. Disclosure of this document in full or in part, may result in material damage to Graymatics-SG Pte Ltd. Written permission must be obtained from Graymatics-SG Pte Ltd prior to the disclosure of this document to a third party.

Installation

Dependencies

- Install Node: Node installation.
- Install MySQL for database: MySQL installation.
- Install Redis: Redis installation.
- To secure Redis: Redis security.
- Install Nginx: Nginx installation.

Extract files

- Unzip the attached zip file on the destination folder for Adscope.
- Install dependencies by doing the command “npm install” on the folder where the files were extracted. On the “ads“ folder.

Define details of the server

- On the file “app.properties”, change the values of “host”, “user”, and “password”, to the credentials of the MySQL server. If the server is in the same machine, “host” should be “localhost” as a value.
- If Redis has been set a password it needs to be specified on the “redis_password” field at “app.properties”.
- Define the value at “install” at “app.properties” to be “true” to install the database structure. This step needs to be changed after the first initialization of the server to “false”. This method prevents deleting the database after the first initiation.

Secure Server (optional)

- To use the HTTPS version of the system, requires the “cert.pem” file and “key.pem”. Both files need to be replaced in the main folder using the same names.
- Set Nginx to redirect port 443 to 3311. This will enable the default port of the server for the application. This is an optional step but required for different Tag managers.
  Additionally, it can be pointed from port 80 to 3311 if the certificates are needed to be only on the application.

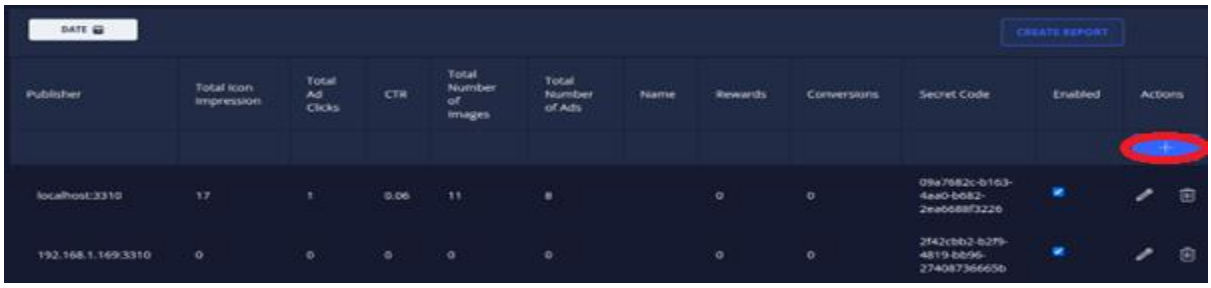
Initialize server

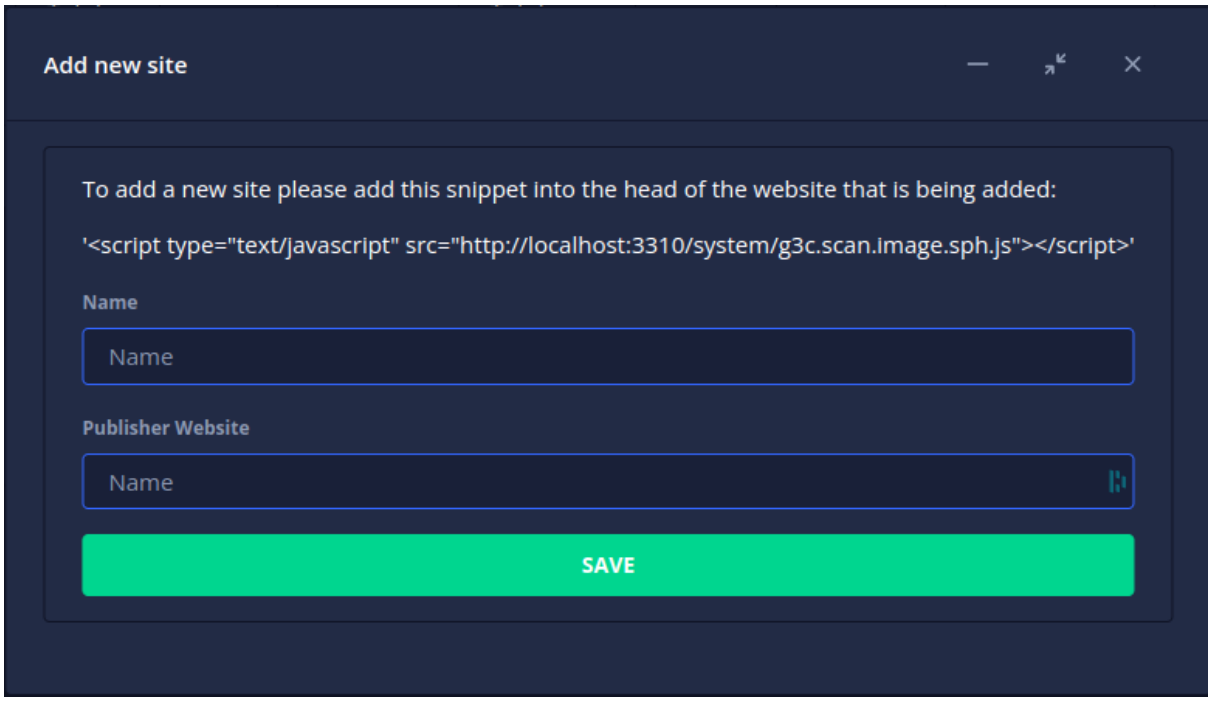
- Run the command “npm start” at the main folder. It will display the ports to be used by the server.

Website Management

Add website

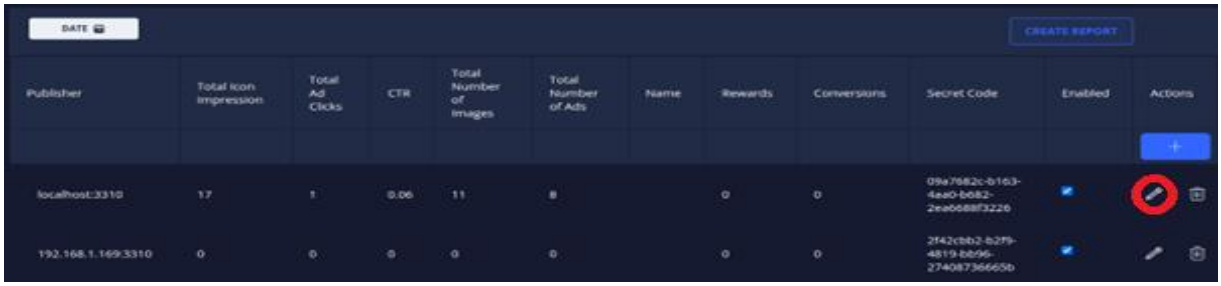
To add a new website, it needs to be pressed on the button highlighted. After this it needs to be filled the form and follow the instructions provided in the form.

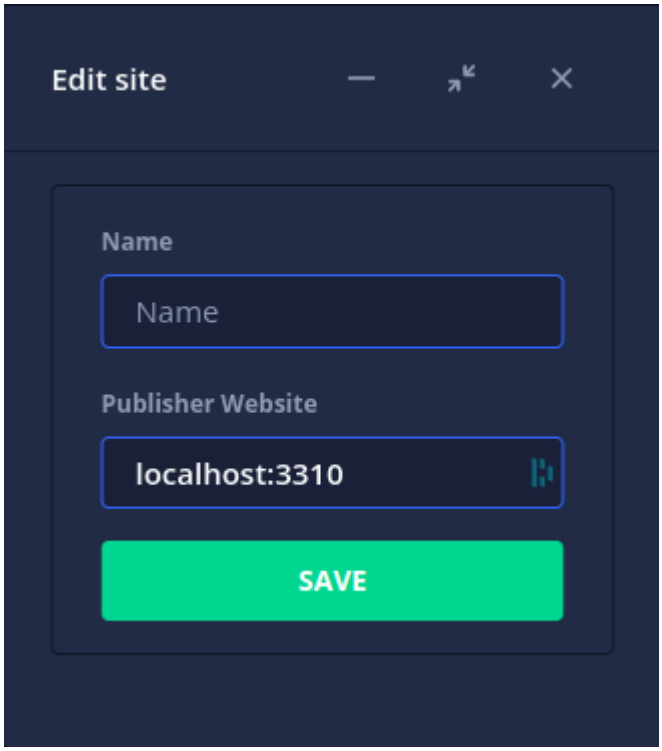




Edit website

To edit a website, it needs to be pressed on the edit button highlighted. After this it can be changed the name of the website to have a better understanding of the dashboard.

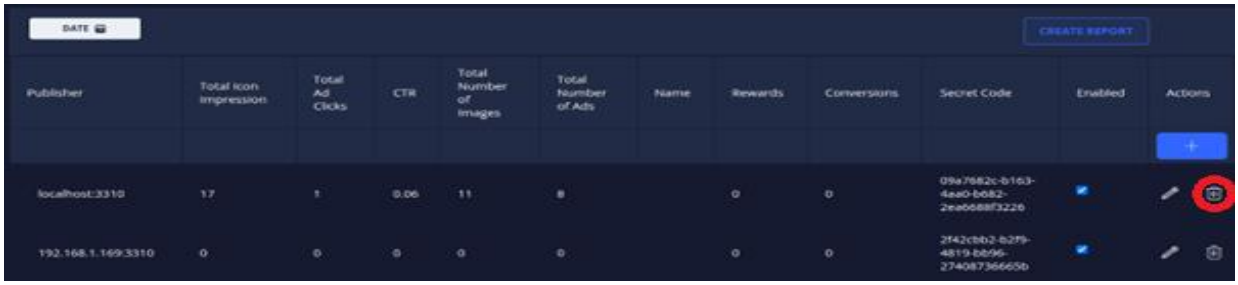




Delete website

To delete a website it needs to be pressed on the trash icon and then confirm this selection. After this it needs to be removed the snippet from the website to ensure it was completely removed.

Another alternative of this is to disable the site by clicking on the check icon under the column “enabled”.



Additional Information

Ports used

- HTTP port: 3310
- HTTPS port: 3311

Snippet

- HTTP snippet:
  <script type="text/javascript" src=http://{ip_server}/system/g3c.scan.image.sph.js></script>
- HTTPS snippet:
  <script type="text/javascript"
  src=https://{ip_server}/system/g3c.scan.image.sph.js></script>

Inside of the snippet, there is “{ip_server}” that needs to be replaced with the Ip of the server to be used. HTTPS snippet will work if the optional step has been completed.
*Notice that if Nginx is not being used, the snippet would have to use the port for each protocol.

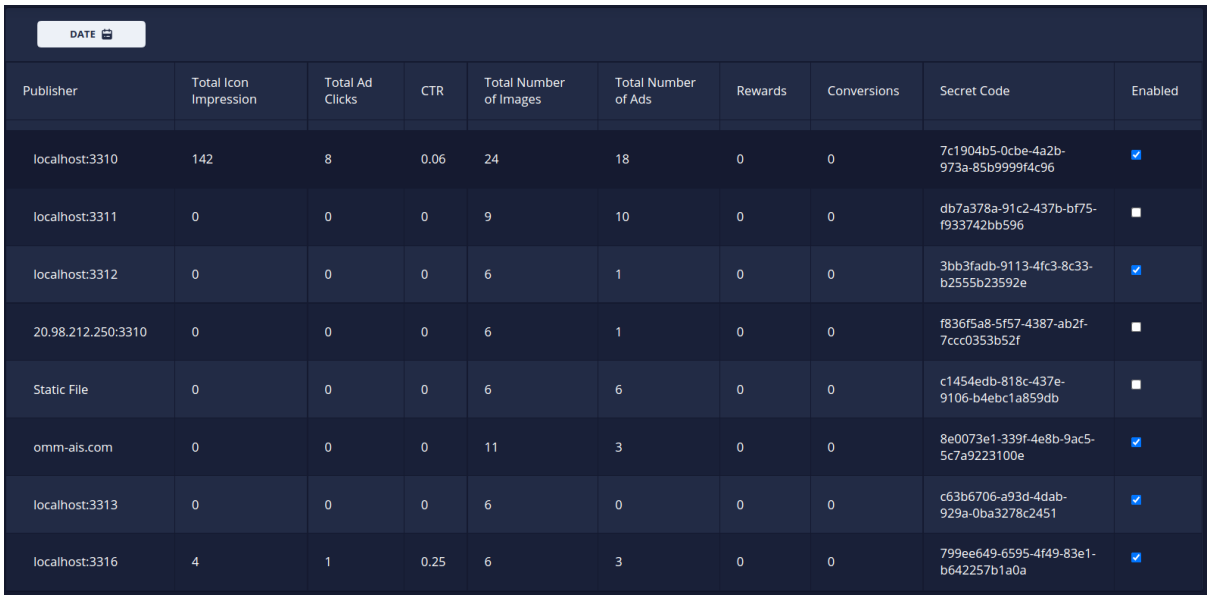
- HTTP snippet:
  <script type="text/javascript"
  src=http://{ip_server}:3310/system/g3c.scan.image.sph.js></script>
- HTTPS snippet:
  <script type="text/javascript"
  src=https://{ip_server}:3311/system/g3c.scan.image.sph.js></script>

Errors logs

To access errors, these would be present in the console after initialising the server with “npm start”. Additional, we stored all the requests made on the server and saved them on the “access.log” file. There is specified the date, URL of origin, status, and the user agent from where was originated the request.

Dashboard Access

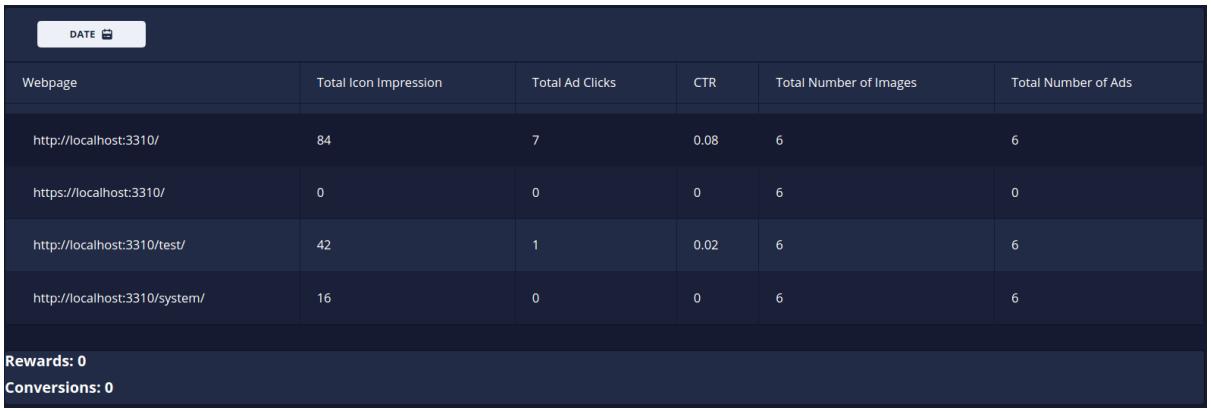
To access the dashboard, it needs to be accessed to "http://{ip_server}:3310/management".



DATE

Publisher	Total Icon Impression	Total Ad Clicks	CTR	Total Number of Images	Total Number of Ads	Rewards	Conversions	Secret Code	Enabled
localhost:3310	142	8	0.06	24	18	0	0	7c1904b5-0cbe-4a2b-973a-85b9999f4c96	☑
localhost:3311	0	0	0	9	10	0	0	db7a378a-91c2-437b-bf75-f933742bb596	☐
localhost:3312	0	0	0	6	1	0	0	3bb3fadb-9113-4fc3-8c33-b2555b23592e	☑
20.98.212.250:3310	0	0	0	6	1	0	0	f836f5a8-5f57-4387-ab2f-7ccc0353b52f	☐
Static File	0	0	0	6	6	0	0	c1454edb-818c-437e-9106-b4ebc1a859db	☐
omm-ais.com	0	0	0	11	3	0	0	8e0073e1-339f-4e8b-9ac5-5c7a9223100e	☑
localhost:3313	0	0	0	6	0	0	0	c63b6706-a93d-4dab-929a-0ba3278c2451	☑
localhost:3316	4	1	0.25	6	3	0	0	799ee649-6595-4f49-83e1-b642257b1a0a	☑

This will be receiving the information gathered into the database installed. From here once each publisher adds the snippet provided, it will auto-fill the dashboard with the information of each publisher. With each publisher, it will generate a secret code to provide the link of their dashboards, "http://{ip_server}:3310/client". Each publisher will be authorized to see their own data. The rewards system and the products will be enabled once ACCESSTRADE approves each campaign for each publisher. This might take up to 24 hours after the first access from the publisher site once added the snippet.



DATE

Webpage	Total Icon Impression	Total Ad Clicks	CTR	Total Number of Images	Total Number of Ads
http://localhost:3310/	84	7	0.08	6	6
https://localhost:3310/	0	0	0	6	0
http://localhost:3310/test/	42	1	0.02	6	6
http://localhost:3310/system/	16	0	0	6	6

Rewards: 0
Conversions: 0

Application Flow

1. First time.
    1.1. Get the images from the page
    1.2. Download the products from the API of Accesstrade
    1.3. Parse and upload the products to the MySQL database
    1.4. Send the image to Vista
    1.5. Receive and process Vista response
    1.6. Matching the image with a correspondent product
    1.7. Save that result in Redis DB
    1.8. Showing said to result on the page.
2. Second time without Redis.
    2.1. Get the image from the page
    2.2. Request products from MySQL DB
    2.3. Send the image to vista
    2.4. Receive and process Vista response
    2.5. Match the image with a correspondent product.
    2.6. Save the result in Redis DB
    2.7. Show the result on the page.
3. With Redis.
    3.1. Get the result of the previous iteration.
    3.2. Show the result on the page.

For visual explanation see the charts on the following pages.

ads

rodrigo garcia | December 14, 2021

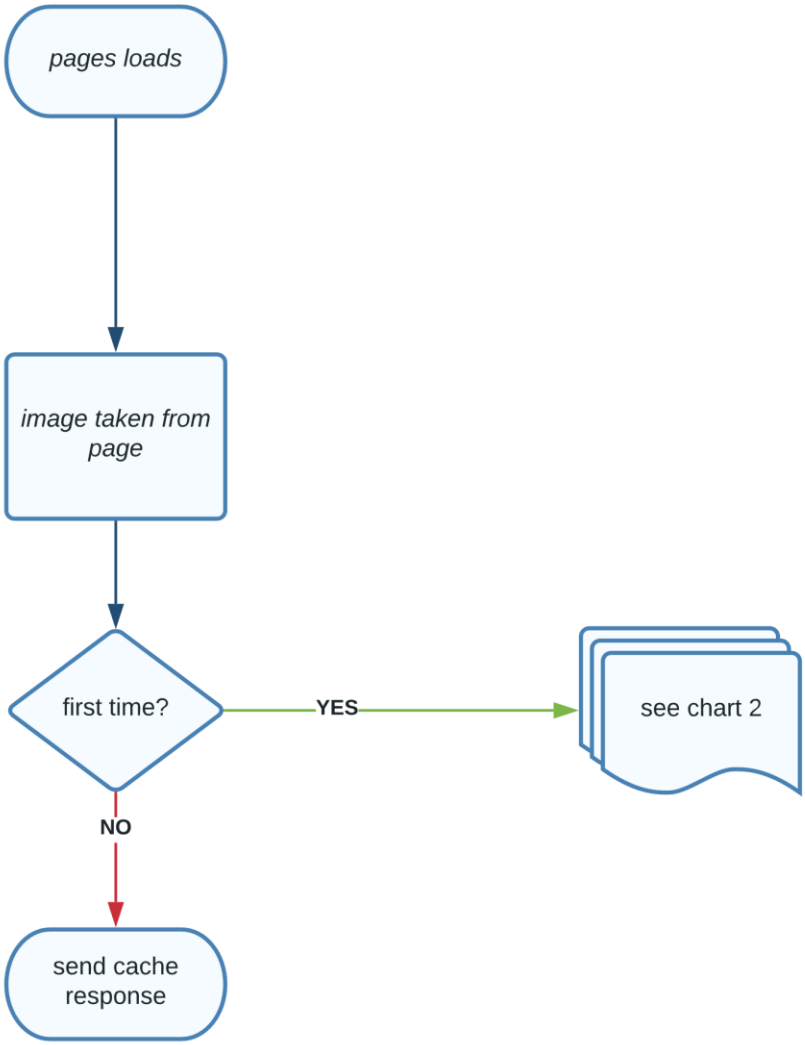


chart 2

rodrigo garcia | December 14, 2021