

This Project is Developed using Django Rest API.

Overview You have to create a Django web application to receive data into the app server for an account and send it across different platforms (destinations) from that particular account using webhook URLs.

1. Account Module: Each account should have email id (Mandatory field & unique), account id (unique to each account), account name (Mandatory field), App secret token (automatically generated), Website (optional)

account id → auto generated

App secret token → 16-digit token will be auto generated

Adding New account → <http://127.0.0.1:8000/accounts/>

CRED operations for Accounts → <http://127.0.0.1:8000/accounts/<id>>

2. Destination Module: A destination belongs to an account. An account can have multiple destinations. A destination has URL (mandatory field), HTTP method (mandatory field) and headers (mandatory field & have multiple values).

URL → will be automatically generated & saved

CRED operations for Destinations.

Adding new Destination → <http://127.0.0.1:8000/destinations/>

CRED operations for Destinations → <http://127.0.0.1:8000/destinations/<id>>

3. An account can have multiple destinations. For example, if an account is deleted, the destinations for that account should also be deleted

<http://127.0.0.1:8000/accounts/<id>/destinations/>

if an account is deleted, all the destinations for under that account will be deleted

4. Data handler: Data handler receives only JSON data in the POST method. While receiving the data, it should have an app secret token sent through the header (header key is CL-X-TOKEN). Based on the secret token, the account should be identified. Using any Python http client, send the data across different available destinations (use the url, http method and headers) for that particular account.

Create an API for receiving the data.

- a. The API path is /server/incoming_data. The data should be received through the post method in the JSON format only.
- b. The app secret token should be received while receiving the data.
- c. If the HTTP method is GET and the data is not JSON while receiving the data, send a response as "Invalid Data" in JSON format d. If the secret key is not received then send a response as "Un Authenticate" in JSON format.
- d. After receiving the valid data, using the app secret token, identify the account and send the data to its destinations

Incoming Request Path of the request → http://127.0.0.1:8000/incoming_data

CL-X-TOKEN is used as key for passing the app secret token

[app_secret_token](#), [content_type](#), [post method](#), will be verified, before passing the data

[app_secret_token](#) is used to find the account and destinations accessioned with it,

URL, HTTP-Method, header data is fetched for each destination for sending the data to each of reserved through the post request.

[post.py](#) file is used for sending the Post request to the Data handler
