Kong Setup

!! For windows users only :
We are about to run some curl commands to configure Kong
To download it https://curl.haxx.se/windows/

SCHEMA OF KONG ARCHITECTURE

- Set environment variables

Linux and macOS
export kong_admin=IPADDRESS:8001
export aws_key=***KEY***
export aws_secret=***SECRET***
export aws_region=eu-west-2

Windows
set kong_admin=IPADDRESS:8001
set aws_key=***KEY***
set aws_secret=***SECRET***
set aws_region=eu-west-2

- Create a Kong service and routes

From a terminal or a console window run the following commands (you will find the file containing the commands here)

```
kong_admin=***IP/URL***:8001
aws_key=***KEY***
aws_secret=***SECRET***
aws_region=eu-west-2

#create generic products service
curl -X POST --url http://$kong_admin/services --data 'name=products' --data 'url=http://aws-lambda'

#create different routes to be able to have different aws-lambda call on each route
curl -X POST --url http://$kong_admin/services/products/routes --data 'methods[]=GET' --data
'paths[]=/workshop/products'
curl -X POST --url http://$kong_admin/services/products/routes --data 'methods[]=GET' --data
'paths[]=/workshop/products/\S+'


#create generic products service
curl -X POST --url http://$kong_admin/services --data 'name=deliveries' --data 'url=http://aws-lambda'

#create a route for POST deliveries
curl -X POST --url http://$kong_admin/services/deliveries/routes --data 'methods[]=POST' --data
'methods[]=OPTIONS' --data 'paths[]=/workshop/commands'
```

- Enable Kong AWS Lambda plugin

Create aws-lambda plugin to target the aws-lambda functions from the routes.
Get the route ids from products service (one id for /products and one id for /products/\S+)

curl $kong_admin/services/products/routes

```
#create aws-lambda plugin to target the aws-lambda functions from the routes
## get the route ids from products service (one id for /products and one id for /products/\S+)
curl $kong_admin/services/products/routes

###/products
curl -X POST $kong_admin/routes/{route_id}/plugins \
--data "name=aws-lambda" \
--data-urlencode "config.aws_key=$aws_key" \
--data-urlencode "config.aws_secret=$aws_secret" \
--data "config.aws_region=$aws_region" \
--data "config.unhandled_status=400" \
--data "config.function_name=getAllProducts"

###/products/\S+
curl -X POST $kong_admin/routes/{route_id}/plugins \
--data "name=aws-lambda" \
--data-urlencode "config.aws_key=$aws_key" \
--data-urlencode "config.aws_secret=$aws_secret" \
--data "config.aws_region=$aws_region" \
--data "config.forward_request_uri=true" \
--data "config.unhandled_status=400" \
--data "config.function_name=getProductDetails"

## get the route ids from products service (one id for /products and one id for /products/\S+)
curl $kong_admin/services/deliveries/routes
###/commands
curl -X POST $kong_admin/routes/{route_id}/plugins \
--data "name=aws-lambda" \
--data-urlencode "config.aws_key=$aws_key" \
--data-urlencode "config.aws_secret=$aws_secret" \
--data "config.aws_region=$aws_region" \
--data "config.forward_request_body=true" \
--data "config.unhandled_status=400" \
--data "config.function_name=createNewCommand"
```

- Create a reverse proxy to MapQuest

```
#create the service to MapQuest
curl -X POST --url http://$kong_admin/services --data 'name=mapQuest' --data
'url=http://www.mapquestapi.com/directions/v2/route'

#create the route to target the service
curl -X POST --url http://$kong_admin/services/mapQuest/routes --data 'methods[]=GET' --data
'paths[]=/distances'
```

- Create CORS plugins for local website

```
##Ajouter le plugin CORS sur les services suivant le mode déploiement du front end
curl -X POST http://kong:8001/services/products/plugins \
    --data "name=cors"  \
    --data "config.origins=*"

curl -X POST http://kong:8001/services/deliveries/plugins \
    --data "name=cors"  \
    --data "config.origins=*"
```