**PARCEL COST ESTIMATE**

The endpoint /*getCostEstimateApi* returns a JSON which also includes the parcel delivery estimate cost.

METHOD: GET

x-www-form-urlencoded

{Params} **origins**

{Params 2} **destinations**

ENDPOINT: {{API\_DUKA\_LOCAL}}/getCostEstimateApi

**SERVER RESPONSE**

{

    "error": **false**,

    "message": "rate charge applicable found.",

    "base\_charge": 250

}

**BOOK REQUEST**

The endpoint invokes a live parcel service request. The expected feedback is in JSON format that contains a unique reference a TRIP i.e. Trip\_ID.

METHOD: POST

ENDPOINT: {{API\_DUKA\_LOCAL}}/bookApi

{Payload}

{

  "driver\_email": "2littledriver@gmail.com",

  "mobile\_number": "254700928123",

  "name": "AirDuka User2",

  "picture": "https://",

  "parcel\_size": "2",

  "recipient\_name": "Joe Doe",

  "recipient\_mobile": "254725239191",

  "recipient\_addresss": "westlands heights, westlands road",

  "contact\_name": "Joe Doe",

  "contact\_mobile\_number": "254725239191",

  "order\_id": 12345,

  "delivery\_notes": "order#12345 shipped to my doorstep",

  "pick\_up\_latlng": "-1.265955,36.805038",

  "pick\_up\_address": "My defined location",

  "drop\_off\_latlng": "-1.2648989,36.8372708",

  "drop\_off\_address": "Pangani shopping center, Nairobi, Kenya",

  "drop\_offs": [],

  "provider": "little"

}

DEPLOYMENT

The said micro service requires the following installed on the production server.

Node.JS

Express.JS

Libraries {axios,dotenv,mysql,}

Install PM2

This tool keeps the node micro-service up by restarting the application if it crashes. PM2 also restarts the node application every time the server is restarted and not forgetting the said tool keeps application logs of all unhandled exceptions.