INSTALLMATE REST API

MICHAEL MEDING* , MMEDING@OUTSMARTINC.COM

CONTENTS

ABSTRACT

This article will include details regarding the usage and implementation of all available methods supported by the Outsmart app server. For the purposes of this article “BASE-URL” should be substituted with "http://ia.outsmartinc.com/" as it will be used often as a prefix for the following URL’s. In addition this documentation only includes one class with several services which have not been fully implemented at the time of this writing. They include, getOpStatus, pingMac and savePanelData.

1 AUTHSERVICE

1.1 fetchModes

```
#Curl script
curl -H "Origin:*" -v -X POST
      BASE-URL/installmate/fetchModes
```

OUTPUT (JSON)

```
1 {"unavaiableCalls":["fetchPanelReferences","fetchPanelData"],"
    haveELR":true}
```

URL PATTERN

BASE-URL/installmate/fetchModes

EXPLANATION Simple call that returns the modes "elr/ops". Including the -H "Origin:*" is nessary for cross origin resource sharing.

1.2 fetchCustomers

```
#Curl script
cu_{}rl -H "Origin:*" -v -X POST
      BASE-URL/installmate/fetchCustomers
```

OUTPUT (HEADER TEXT)

```
1 {"status":"OK","version":1.0,"list":["OutSMart Ppwer SUstems","
    OutSmart Power Systems","laber"]}
```

URL PATTERN
BASE-URL/installmate/fetchCustomers

EXPLANATION This method should be called fetchAllCustomers. It returns all the customers it has stored as a list of strings.

1.3 fetchLocations

```
#Curl script
curl -v -H "Accept: application/json"
      -H "Content-Type: application/json"
      -X POST -d "{\"customerName\":\"laber\"}"
      BASE-URL/installmate/fetchLocations
```

OUTPUT (HEADER TEXT)

```
1 {"status":"OK","version":1.0,"list":["glork","lall"]}
```

URL PATTERN
BASE-URL/installmate/fetchLocations

EXPLANATION I figure by this point you must understand what is going on. The only differance between this method and the ones prior is that this one has a single argument that must be passed as JSON.

1.4 fetchPanelData

```
#Curl script
curl -v -H "Accept: application/json"
      -H "Content-Type: application/json"
      -X POST -d "{\"customerName\":\"laber\"
      ,\"locationName\":\"lall\"}"
      BASE-URL/installmate/fetchPanelData
```

OUTPUT (HEADER TEXT)

```
{"anObject":"aString"}
```

URL PATTERN
BASE-URL/installmate/fetchPanelData

EXPLANATION Again, if you understand everything else up to this point then you should have no problem figuring out what is going on here.

1.5 fetchPanelData

```
#Curl script
curl -v -H "Accept: application/json"
      -H "Content-Type: application/json"
      -X POST -d "{\"customerName\":\"laber\"
      ,\"locationName\":\"lall\"}"
      BASE-URL/installmate/fetchPanelData
```

OUTPUT (HEADER TEXT)

```
{"anObject":"aString"}
```

URL PATTERN
BASE-URL/installmate/fetchPanelData

EXPLANATION Again, if you understand everything else up to this point then you should have no problem figuring out what is going on here.

1.6 savePanelData

```
#Curl script
curl -v -H "Accept: application/json"
      -H "Content-Type: application/json"
      -X POST -d "{\"customerName\":\"laber\"
      ,\"locationName\":\"lall\",
      ,\"panelData\":\"\"{\"variable\":\"some_data\"}\"\",}"
      BASE-URL/installmate/savePanelData
```

OUTPUT (HEADER TEXT)

```
{"status":"OK","version":1.0}
```

URL PATTERN
BASE-URL/installmate/savePanelData

EXPLANATION You put the JSON in, some magic happens and BLAM! Your stuff gets saved. Amazing!