Airconet API

1. Login->

[POST] ../j_spring_security_check

Request :

{
“username”: “test@test.com”,
“password”: “1234567”,
“remember-me”: ”true”,
“isServer”: “false”,
“device-id”: “WiFi mac address”,
“firebase-token”: ”Firebase token”,
“os_type”: “device operation system type”,
“isDebug”: “tRue”
}

Response :


```
[
 {
  "deviceStatus": {
   "deviceStatus": {
    "id": 544,
    "mac": "CC50E33B0C67",
    "onoff": 85,
    "light": 0,
    "mode": 51,
    "fan": 1,
    "envTemp": 26,
    "envTempShow": 26,
    "tgtTemp": 19,
    "deviceType": "GR",
    "type": "AC"
   },
   "deviceName": "air",
   "model": "Air Conditioner"
  },
  "airConSetPoint": {
   "id": 124,
   "mac": "CC50E33B0C67",
   "groupId": null,
   "userId": 173,
   "maxPoint": 32,
   "minPoint": 16,
   "createTime": 1595752209000,
```

```
    "updateTime": null,
    "keepSetPointInRange": false
   },
   "online": false
  }
]
```


2. Sign up

[POST] ../user/signup

Request :


```
{
“username”: “user”,
“fullname”: “name”,
“address”: “China Shanghai”,
“password”: “1234567”,
“phone”: “+123456789”,
“email”: “test@test.com”,
“adminPassword”: “111”,
“countryState”: “Shanghai”,
“country”: “China”,
“isDebug”: “tRue”,
“token”: “”
}
```


Response :


```
{
  "result": 0,
  "reason": "",
  "attachment": {}
}
```


3. Forgot password

[POST] ../user/requestRecoverPassword

Request :


```
{“mail”, “test@test.com”}
```


Response :


```
{
  "result": 0,
  "reason": "",
  "attachment": {}
}
```

4. Current user

[GET] ../user/currentUserId

Request :

{}

Response :

{"returnCode":0,"values":[173]}

5. Get exist devices from server

[GET] ../dev/exist-devices?macs={Devices mac address}

Request :

{}

Response :

{

“isExist” : true,

“isBelongToCurrentUser” : true

}

6. Save device to server

[POST] ../dev/saveDevice?ssid={wifi ssid}

Request :

{

“mac” : “device mac address”,

“type” : “device type”,

“privateIp” : “device private ip”,

“publicIp” : “device public ip”,

“name” : “device name”,

“seriesNumber” : “device series number”,

“model” : “device model”,

“manufacturer” : “”,

“deviceType” : “AC or SW or PM”

}

Response :

{

"returnCode": 200

}

7. Get error messages and notify from server

[GET] ../alert/all?pageNumber={current page}&pageSize={totalPage}

Request :

{}

Response :

[


```
{
  "id": 9661,
  "userId": 173,
  "accountName": "",
  "name": "",
  "phone": "",
  "mac": null,
  "model": "",
  "serialNumber": "",
  "errorGroup": "",
  "severity": 1,
  "color": "red",
  "content": null,
  "description": "",
  "descriptionHebrew": "",
  "received": null,
  "errorCode": "data desc",
  "appCode": "",
  "comment": null,
  "technician": null,
  "fixTime": null,
  "holdTime": null,
  "startTime": 1599832690000,
  "startTimeTimezone": 1599847090000,
  "stopTime": null,
  "ackTime": null,
  "cleared": null,
```

```
   "clearTime": null,
   "deleted": false,
   "phaseNumber": 0,
   "deviceType": null,
   "alertType": null,
   "errorNumber": "",
   "localRegisterDate": null,
   "localHoldDate": null,
   "localFixedDate": null,
   "errorValue": 0,
   "errorType": "PARENT_USER_ERROR",
   "senderUserId": 177,
   "headerUserMessage": "data message",
   "ppm": null
  }
]
```


8. Delete all errors

[POST] ../alert/deleteAll

Request :

{}

Response :

"all alerts were deleted successfully"

9. Remove device

[GET] ../dev/removeDevice/{device mac address}

Request :

{}

Response :


```
{
  "returnCode": 0,
  "values": null
}
```

10. Device reorder

[POST] ../dev/order

Request :

{

“macs” : “device mac address”

}

Response :

0

11. Devices power meters

[GET] ../pm/app/master

Request :

{}

Response :

{

“id” : long;

“mac” : ”String”;

“name” : “String”;

“type” : “String”;

“masterEnabled” : boolean;

“ph1Watt” : int;

“ph2Watt” : int;

“ph3Watt” : int;

“totalWatt” : int;

“publicIp” : “String”;

“sensorWatt” : Integer;

“sensorAmpere” : Double;

“sensorInternal” : boolean;

“ph1Details” : [PowerMeterDetails];

“ph2Details” : [PowerMeterDetails];

“ph3Details” : [PowerMeterDetails];

“totalDetails” :[PowerMeterDetails];

“userList” : List<User>[{User}];

}

12. Delete device

[POST] ../dev/deletePM/{device mac address}

Request :

{}

Response :

“”

13. Get working hours

[GET] ../dev/working-hour?mac={device mac address}

Request :
{}
Response :
“783”

14. Get device temperature

[GET] ../setpoint/dev?mac={device mac address}

Request :
{}
Response :
200 ok

15. Reset working hour by device mac address

[POST] ../level3-info/restart-device-hours

Request :
{“mac”: “device mac address”}
Response :
Number: 1 if hours have been restarted
-1 if something went wrong)

16. Change current device temperature

[POST] ../dev/setHvac

Request :
{
“mac” : “device mac address”,
“onoff” : “device on/off state”,
“mode” : device mode state(Integer),
“fan” : device fan state(Integer),
“envTemp” : device min/max temp.,
“tgtTemp” : device current temp. ,
“envTempShow” : 0
}
Response :

{

"returnCode": 200,

```
 "values": [
  {
    "id": 0,

    "mac": "deviceMac"

    "onoff": 85,

    "light": 0,

    "mode": 33,

    "fan": 0,

    "envTemp": 0,

    "envTempShow": 0,

    "tgtTemp": 25,

    "deviceType": ""

    "type": "AC"
  }
 ]
}
```


17. Add link to power meter by device mac address

[POST] ../pm/udp/link

Request :
{“deviceMac” : “device mac address”}
Response :
200 ok

18. Update device by mac address

[POST] ../dev/updateDevice

Request :
{
“id” : “”
“mac” : “device mac address”,
“type” : “device type”,
“privateIp” : “device private ip”,

“publicIp” : “device public ip”,
“name” : “device name”,
“seriesNumber” : “”,
“model” : “device model”,
“manufacturer” : “”,
“deviceType” : “device type”,
“countryState”: “Shanghai”,
“country”: “China”
}

Response :
{
"returnCode": 200
}

19. Update device min/max temperature

[POST] ../setpoint/dev

Request:
{
“mac” : “device mac address”,
“max” : “device maximum temperature”,
“min” : “device minimum temperature”,
“enableSetPoint” : “keep set point within range” true or false
Response :

{

"id": 0,

"groupId": 0,

"userId": 0,

"maxPoint": 0,

"minPoint": 0.,

"createTime": "",

"updateTime": "",

"keepSetPointInRange": true
}

20. Get power meter link

[GET] ../pm/linked-pm?mac={device mac address}

Request :

{}

Response :


```
{

  "phaseNumber": 0,

  "powerMeter": {

  }

}
```


21. Get refreshed watt

[GET] ../pm/watt?deviceMac={device mac address}

Request :

{}

Response :

Number

22. Get group statuses

[GET] ../group/devices/app/{group id}

Request :

{}

Response :

23. group devices on/off

[POST] ../group/switchGroup

Request :

{

“groupId” : “group id”,

“on” : “group on/off state(true/false)”

}

Response :

24. delete group

[POST] ../group/remove

Request :
{ “groupId” : “group id” }

25. group reorder

[POST] ../group/order

Request :
{ “ids” : “groups id”}

Response:

26. save group

[POST] ../group/device

Request :
{
“groupName” : “group name”,
“id” : “group id”,
“macs” : “devices mac addresses”
}
Response :

27. get holidies

[GET] ../holiday/dates/

Request :
{}
Response :

28. save or remove holiday

[POST] ../holiday/addOrRemove/app

Request :
[

“yyyy-MM-dd”(1900-12-01)

]

Response :

29. check all devices power

[GET] ../dev/checkAllDevicesPower?isOn=true

Request :
{}

Response :

30. get holiday if it is today

[GET] ../holiday/istoday?timeZone={get current time zone}

Request :
{}
Response:

31. get all devices power on/off

[GET] ../dev/switchHvacs?on={All On/All Off}

Request :
{}
Response :

32. get messages count

[GET] ../alert/total-alerts/app

Request :
{}
Response :

33. enable or disable motion sensor

[POST] ../dev/updateMotionSensor

Request :
{
“mac” : “device mac address”,
“enabled : “true/false”

}
Response :

34. delete sensor device

[POST] ../dev/deletePM/{device mac address}

Request :
{}
Response :

35. get power meter link to device

[GET] ../dev/by/pm?mac={powermeter mac address}

Request :
{}
Response :

36. set master power meter

[POST] ../pm/app/{power meter mac address}/set-master

Request :
{“enabled” : “true/false”}
Response :

37. reset password

[POST] ../user/resetPassword

Request :
{
“email” : “user email”,
“verifyCode” : “verification code from email”,
“password” : “new password”
}

Response :
{
"result": 0,
"reason": "",
"attachment": {}
}

38. tech support

[POST] ../callme

Request :
{“content” : “text”(maximum 100 words)}

Response :

39. get timer data

[GET] ../timer/timers

Request :
{}
Response :

40. delete timer

[POST] ../timer/delete?timerId={timer id}&onOff ={timer power state}

Request :
{}
Response :

41. enable or disable timer

[POST] ../timer/enabledOrDisabled

Request :

```
[
  {
    “onOff”: “timer power state”,
    “timerId” : “timer id”,
    “enabled” : “true/false”
  },
  {
    “onOff”: “timer power state”,
    “timerId” : “timer id”,
    “enabled” : “true/false”
  }
]
```

Response :

42. timer exist

[POST] ../timer/isExist

Request:

```
{
   “deviceType” : “timer type”,
   “timerId” : timer id(long),
   “deviceId” : device id(long),
   “startTime” : “timer start time”,
   “daysList” : [
          {
   1,2,3….
}
        ],
   “onOff” : device power state (Integer),
   “timeZone” : “current time zone”
}
```

43. save or update timer

[POST] ../timer/app/save-update

Request :

```
{
  “Id” : timer id
  “onOff” : device power state 85/-86
  “days” : timer days
  “startTime” : “timer start time”
  “temperature” : device timer temperature(integer)
  “mode” : device timer mode type(integer)
  “fan” : device timer fan type(integer)
  “airConDevices”: [
                                {
                                    “id” : device id(long),
                                    “mac” : “device mac address”
                                },
                                {
                                    “id” : device id(long),
                                    “mac” : “device mac address”
                                }
                          ]
}
```

Response :

44. Get basic electric bill

[GET] ../electric-bill/baseBill/app

Request :
{}
Response :

45. Set basic electric bill day

[POST] ../electric-bill/base-bill/update-day

Request :
{
"baseBillId" : "basic electric bill id",
"day" : "basic electric bill day"
}
Response :

46. Set basic electric bill price

[POST] ../ electric-bill/base-bill/update-price

Request :
{
"baseBillId" : "basic electric bill id",
"price" : "basic electric bill price"
}
Response :

47. Get country list

[GET] ../country

Request :
{}
Response :

48. Update user country

[POST]../system/update-country

Request :
{"countryId" : "country id"}
Response:

49. Get hazard room temperature info

[GET] ../alertConfig/hazard-room-temperature

Request :
{}
Response :

50. Set new hazard room temperature info

[POST] ../alertConfig/hazard-room-temperature

Request :
{
“id” : hazard room temp id(integer),
“t3l” : (integer),
“t2l” : (integer),
“t1l” : room minimum temp (integer),
“t1h” : room maximum temp (integer),
“t2h” : (integer),
“t3h” : (integer),
“user” :[user info],
“settingsStatus” : true/false
}
Response :

51. Get maximum power data

[GET] ../user/peak-power/app

Request :
{}
Response :

52. Set new maximum power data

[POST] ../pm/change-peak-power/app

Request :
{“peakKW” : “set new max power”}
Response :

53. Refresh maximum power data

[POST] ../pm/udp/send-master-status-request

Request:
{}
Response:

54. Get working hour data

[GET] ../user/workingHour

Request :
{}
Response :

55. Set new working hour data

[POST] ../user/workingHour

Request :
{
“id” : working hour id(integer),
“mondayFromTime” : “”,
“mondayToTime” : “”,
“tuesdayFromTime” : “”,
“tuesdayToTime” : “”,
“wednesdayFromTime” : “”,
“wednesdayToTime” : “”,
“thursdayFromTime” : “”,
“thursdayToTime” : “”,
“fridayFromTime” : “”,
“fridayToTime” : “”,
“saturdayFromTime” : “”,
“saturdayToTime” : “”,
“sundayFromTime” : “”,
“sundayToTime” : “”,

“sqm” : “”
}
Response :

56. Get voltage data

[GET] ../ user/sensor-pm-consumption

Request :
{}
Response :

57. Set new voltage data

[POST] ../user/change-sensor-pm-consumption/app

Request :
{“volt” : “new voltage” (ex. 220)}
Response :

58. Set changed group fan

[POST] ../group/setGroupFan

Request :
{
“groupId” : “group id”,
“fan” : “group fan state”
}
Response :

59. Set changed group mode

[POST] ../group/setGroupMode

Request :
{
“groupId” : “group id”,
“mode” : “group mode state”
}
Response :

60. Set changed group temperature

[POST] ../group/setGroupTgtTemp

Request :
{
“groupId” : “group id”,

“tgtTemp” : “group temperature state”
}
Response :

61. get motion sensor arrive

[GET] ../motion-sensor/arrive-config

Request :
{}
Response :

62. set motion sensor arrive

[POST] ..motion-sensor/arrive-config

Request :
{
“id” : motion sensor arrive id (integer),
“motionSensorCommand” : “motion sensor arrive command”,
“tmp” : motion sensor arrive temperature(integer),
“mode” : motion sensor arrive mode state(integer),
“fan” : motion sensor arrive fan state(integer)
}

63. get motion sensor away

[GET] ../motion-sensor/away-config

Request :
{}
Response :

64. set motion sensor away

[POST] ../motion-sensor/away-config

Request :
{
“id” : motion sensor away id (integer),
“motionSensorCommand” : “motion sensor away command”,

“roomTmpFrom” : “motion sensor away from temperature”,
“roomTmpTo” : “motion sensor away to temperature”,
“lastingHour” : “motion sensor away last hour”,
}
Response:

65. get motion sensor setup

[GET] ../motion-sensor/setup-config

Request :
{}
Response :

66. set motion sensor setup

[POST] ../motion-sensor/setup-config

Request :
{
“id” : motion sensor setup id (integer),
“enabled” : “true/false”,
“delayAwayCountDown” : motion sensor setup delay away count down(integer),
“disabledSleepTime” : “motion sensor setup disabled sleep time”,
“sleepTimeFrom” : “motion sensor setup sleep from time”,
“roomTimeTo” : “motion sensor setup sleep to time”
}
Response:

67. get device air quality

[GET] ../air-quality?mac={device mac address}

Request :
{}
Response :

68. get alert detail by device

[GET] ../alert/last/byDevice?mac={device mac address}

Request :
{}
Response :

69. get timers by devices mac addresses

[GET] ../timer/isThereAnyTimer/all?macs={devices mac addresses}

Request :

{}

Response :

70. get devices

[GET] ../dev/devices

Request :

{}

Response :


```
{
  "returnCode": 0,
  "values": [
    {
      "id": 654,
      "name": "room18",
      "seriesNumber": "",
      "publicIp": "171.98.91.191",
      "privateIp": "192.168.2.42",
      "mac": "840D8E85AB2D",
      "type": "CH",
      "location": null,
      "state": 1,
      "realtimeStatus": null,
      "screenSize": null,
      "registDateTime": 1593097315000,
      "damageDateTime": null,
      "warrantyExpireDate": null,
      "lastOperation": null,
      "model": "Air Conditioner",
      "baseWatt": 0,
      "mfqWatt": 0,
      "inverter": false,
      "airSensor": false,
      "manufacturer": "",
      "maxPoint": 32,
      "minPoint": 16,
      "powerMeter": null,
      "powerMeterPhaseNumber": 0,
      "deviceType": "AC",
```

```
    "timeZone": "Asia/Bangkok",
    "disableAlert": false,
    "countryState": "Bangkok",
    "country": "Thailand",
    "warrantyYear": 0,
    "sumhours": 0,
    "index": 1,
    "pir": true
   }
 ]
}
```


71. get motion sensor by device mac addresses

[GET] ../motion-sensor/byDeviceMac?mac={device mac address}

Request :

{}

Response :

72. get all devices status

[GET] ../dev/deviceStatusSetPointCurrentStatus

Request :

{}

Response :

73. get groups

[GET] ../group/new

Request :

{}

Response :

74. get master power meter

[GET] ../pm/app/master-offlineOnline

Request :

{}

Response :

75. check current power meter status

[GET] ../control/checkDevicesCurrentStatus/{power meter mac adress}

Request :

{}
Response :

76. get power meter electric parameters

[GET] ../level3-info/power-meter-electric-parameters/app?pmMac={power meter mac address}

Request :
{}
Response :

77. get power meter proportion energy and efficiency

[GET] ../level3-info/proportion-energy/and/efficiency-rate-data?timeRange={power meter time range}&powerMeterMode={power meter mode}
Request :
{}
Response :

78. get power meter total energy

[GET] ../level3-info/total-energy/app?timeRange={power meter time range}
&powerMeterMode={power meter mode}

Request :
{}
Response :

79. get energy usage per device

[GET]../level3-info/energy-usage-per-device?timeRange={power meter time range}
&powerMeterMode={power meter mode}

Request :
{}
Response :

80. get power meter electric graph

[GET] ../electric-bill/baseBill-graph/app

Request :
{}
Response:

81. get power meter info

[GET] ../pm/info/

Request :
{}
Response :

82. get group power meter info

[GET] ../level3-info/device-mode/app?groupId={group id}&timeRange={group time range}&powerMeterMode={power Meter Mode}

Request :
{}
Response :

83. get device power meter info

[GET] ../level3-info/device-mode/app?deviceMac={device mac address} &timeRange={time range}&powerMeterMode={power Meter Mode}

Request :
{}
Response :

84. get device power meter

[GET] ../level3-info/power-meter-electric-parameters/app?acMac={device mac address}

Request :
{}
Response :

85. get single device graph

[GET] ../level3-info/total-energy-for-single-device/app?mac={device mac address}&timeRange= {time range}&powerMeterMode={power meter mode}

Request :
{}
Response :

86. get group graph

[GET] ../level3-info/total-energy-for-group-devices/app?groupId={group id}&timeRange= {time range}&powerMeterMode={power meter mode}

Request :
{}
Response :

87. get all devices power on/off state

[GET] ../dev/switchHvac/{device mac address}?on={true/false}

Request :

{}

Response :

88. update device watt

[POST] ../dev/update-pm-watt/app

Request :

```
{
  “mac” : “device mac address”,
  “watt” : “device base watt”
}
```

Response:

89. Socket

Ulr-> “https://www.socketurl.com:443/v2

Headers:

```
{
 “X-Timezone” : ”current time zone”,
 “isServer” , “false”,
 “isDebug” : “tRue”,
 “X-Client” : “Star.Touch”,
 “User-agent” : “BlueStar”,
 “Accept-Encoding” : “dzip, deflate”,
 “Connection” : “ keep-alive”,
 “Accept” : “*/*”
}
```