

Interface Design Description (IDD) DeviceDiscovery

September 25, 2019

1	Interface Design Description Overview	2
2	Interfaces	2
3	Information Model	3
3.1	DeviceRegistryEntry	3
3.2	DeviceQueryForm	4
3.3	DeviceQueryResponse	4
4	Revision history	5
4.1	Amendments	5

Document title	Document type
DeviceDiscovery	IDD
Date	Version
September 25, 2019	1.2
Author	Status
Ani Bicaku	Draft
Contact	Page
ani.bicaku@fh-burgenland.at	2(5)

1 Interface Design Description Overview

This document describes the DeviceDiscovery service with the REST interface, which uses a basic three-tier architecture:

- Presentation tier
 - implements the RegistryResource interface named DeviceRegistryResource.java
 - transforms the view into domain specific objects and vice versa
- Application tier
 - implements the RegistryService interface named DeviceRegistryService.java
 - is responsible for any business logic and extensive validation
- Data tier
 - taken from the arrowhead common project (DatabaseManager.java), which is a generic and hibernate specific implementation to deal with any entity in the arrowhead code

2 Interfaces

According to SD document(...), DeviceDiscovery service has three functions implemented as shown in the table below.

Table 1: DeviceDiscovery Functions

Function	URL Path	Method	Input	Output
Publish	"/publish"	POST	DeviceRegistryEntry	DeviceRegistryEntry
Unpublish	"/unpublish"	POST	DeviceRegistryEntry	DeviceRegistryEntry
Lookup	"/lookup"	GET	Integer	DeviceRegistryEntry

Document title	Document type
DeviceDiscovery	IDD
Date	Version
September 25, 2019	1.2
Author	Status
Ani Bicaku	Draft
Contact	Page
ani.bicaku@fh-burgenland.at	3(5)

3 Information Model

3.1 DeviceRegistryEntry

POST /deviceregistry/publish

Parameters

Cancel

No parameters

Request body

application/json

Edit Value | Model

```
{  "providedDevice": {    "deviceName": "an IoT device"  },  "macAddress": "01:23:45:67:89:AB",  "endOfValidity": "2029-09-24T11:38:38.167Z"}
```

Cancel

Reset

Execute

Clear

Figure 1: DeviceRegistryEntry



Document title	Document type
DeviceDiscovery	IDD
Date	Version
September 25, 2019	1.2
Author	Status
Ani Bicaku	Draft
Contact	Page
ani.bicaku@fh-burgenland.at	4(5)

3.2 DeviceQueryForm

GET /deviceregistry/lookup/{id} Searches a DeviceRegistryEntry by Id

Parameters

Name	Description
id * required integer (path)	559

Execute Clear

Figure 2: DeviceQueryForm

3.3 DeviceQueryResponse

Responses

Curl

```
curl -X GET "http://0.0.0.0:8438/deviceregistry/lookup/559" -H "accept: application/json"
```

Request URL

```
http://0.0.0.0:8438/deviceregistry/lookup/559
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": 559, "providedDevice": { "id": 558, "deviceName": "an IoT device" }, "macAddress": "01:23:45:67:89:AB", "endOfValidity": "2029-09-24T11:38:38" }</pre> <p>Download</p> <p>Response headers</p> <pre>access-control-allow-credentials: true access-control-allow-headers: origin, content-type, accept, authorization access-control-allow-methods: GET, POST, PUT, DELETE, OPTIONS, HEAD access-control-allow-origin: * access-control-max-age: 600 content-length: 176 content-type: application/json</pre>

Figure 3: DeviceQueryResponse





Document title	Document type
DeviceDiscovery	IDD
Date	Version
September 25, 2019	1.2
Author	Status
Ani Bicaku	Draft
Contact	Page
ani.bicaku@fh-burgenland.at	5(5)

4 Revision history

4.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2017-09-10	1.0	Initial Version	Ani Bicaku
2	2017-11-15	1.1	Initial Version	Ani Bicaku
3	2019-09-23	1.2	Updated Version	Ani Bicaku, Mario Zsilak

