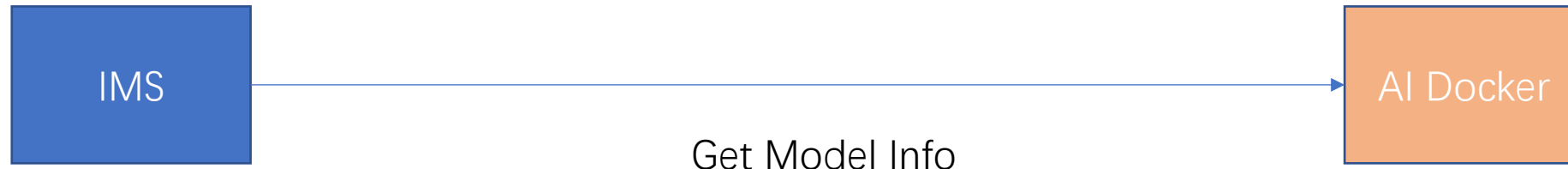


## 1. Get Capabilities and Info



Get: /get/info

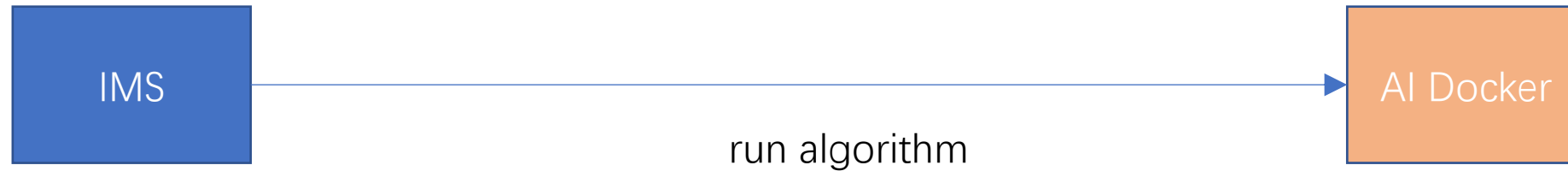
Response:

```
{
  name (should-unique):
  description:
  terms: [ "term-1" , "term-2" ,...]
  Supported_image_formats: [ "svs" , "ndpi" ,...]
  Supported_source_type:[ "forder" , "image" ]
  Supported_scope:[" whole-scan" , "ROI" ]
  Supported_algorithm_type: [ "key-value" , "geo" ,
    "content" ]
}
```

## 2. IMS run algorithm

Running status enum:

Created  
Runing  
Completed  
Failed



run algorithm

POST: /run/algorithm

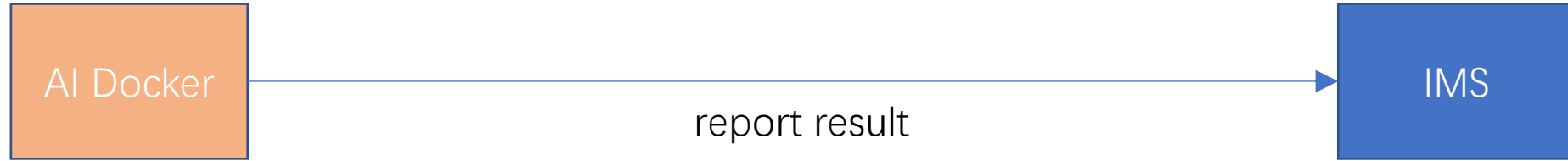
Request:

```
{
  session_id: GUID
  parameters: [ ]
  source: Image or Folder URI PATH
  Image_server:
}
```

Response:

```
{
  session_id: GUID
  code:
  status: Created
  message:
}
```

### 3. Actively report result to IMS



POST: /api/algorithm-result.json

Request:

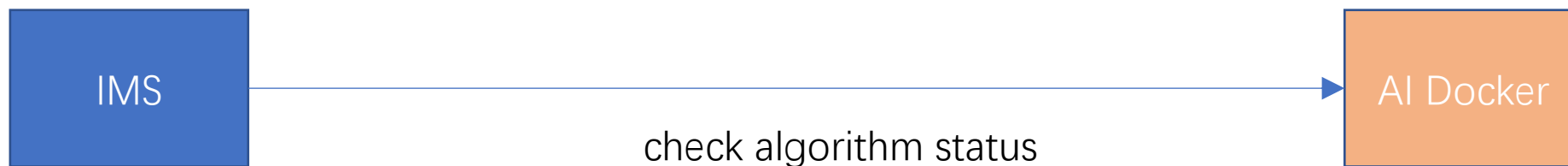
```
{
  session_id: GUID
  status: Completed
  data: [
    {
      image: image1
      [{
        result_type: "key-value"
        result:[
          {key-1,value-1}
          {key-2,value-2}...]
      }, {...}]
    },
    {
      image: image2,
      []
    }
  ]
}
```

Response:

```
{
  code: 200
}
```

#### 4. IMS tracking AI

Running status enum:  
Created  
Runing  
Completed  
Failed



check algorithm status

POST: /check/algorithm

Request:

```
{  
  session_id: GUID  
}
```

Response:

```
{  
  session_id: GUID  
  code:  
  status: Failed/Runing/...  
  message:  
  data:{  
  }  
}
```