GET/applications/services

|  |  |
| --- | --- |
| Description | If none of the parameters are passed |
| **Identifier** | MEC-MP1-SER-001 |
| **Test Purpose** | Check response received from server when all parameter is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/services |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying any parameters |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_INSTANCE\_ID – 200

|  |  |
| --- | --- |
| Description | If valid ser\_instance\_id is passed |
| **Identifier** | MEC-MP1-SER-002 |
| **Test Purpose** | Check response received from server when ser\_instance\_id is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={***Valid\_ser\_instance\_id***}&  ser\_name={ser\_name}&ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid ser\_instance\_id |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "ServiceInfo": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ]  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  }  },  "serializer": "JSON"  }  } |

PARAMETER – SER\_INSTANCE\_ID – 400

|  |  |
| --- | --- |
| Description | If ser\_instance\_id is not passed |
| **Identifier** | MEC-MP1-SER-003 |
| **Test Purpose** | Check response received from server when ser\_instance\_id is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={}&ser\_name={ser\_name}&  ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying ser\_instance\_id |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_INSTANCE\_ID – 404

|  |  |
| --- | --- |
| Description | If ser\_instance\_id is invalid |
| **Identifier** | MEC-MP1-SER-004 |
| **Test Purpose** | Check response received from server when ser\_instance\_id is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={***Invalid\_ser\_instance\_id***}&  ser\_name={ser\_name}&ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid ser\_instance\_id |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_NAME – 200

|  |  |
| --- | --- |
| Description | If valid ser\_name is passed |
| **Identifier** | MEC-MP1-SER-005 |
| **Test Purpose** | Check response received from server when ser\_name is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={***Valid\_ser\_name***}&ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid ser\_name |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "ServiceInfo": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ]  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  }  },  "serializer": "JSON"  }  } |

PARAMETER – SER\_NAME – 400

|  |  |
| --- | --- |
| Description | If ser\_name is not passed |
| **Identifier** | MEC-MP1-SER-006 |
| **Test Purpose** | Check response received from server when ser\_name is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={ }&ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying ser\_name |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_NAME – 404

|  |  |
| --- | --- |
| Description | If ser\_name is invalid |
| **Identifier** | MEC-MP1-SER-007 |
| **Test Purpose** | Check response received from server when ser\_name is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={Invalid\_ser\_name}&ser\_category\_id={ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid ser\_name |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_CATEGORY\_ID – 200

|  |  |
| --- | --- |
| Description | If valid ser\_category\_id is passed |
| **Identifier** | MEC-MP1-SER-008 |
| **Test Purpose** | Check response received from server when ser\_category\_id is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={ser\_name}&ser\_category\_id={Valid\_ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid ser\_category\_id |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "ServiceInfo": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ]  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  }  },  "serializer": "JSON"  }  } |

PARAMETER – SER\_CATEGORY\_ID – 400

|  |  |
| --- | --- |
| Description | If ser\_category\_id is not passed |
| **Identifier** | MEC-MP1-SER-009 |
| **Test Purpose** | Check response received from server when ser\_category\_id is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={ser\_name}&ser\_category\_id={Valid\_ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying ser\_category\_id |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SER\_CATEGORY\_ID – 404

|  |  |
| --- | --- |
| Description | If ser\_category\_id is invalid |
| **Identifier** | MEC-MP1-SER-010 |
| **Test Purpose** | Check response received from server when ser\_category\_id is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/services?ser\_instance\_id={ser\_instance\_id}&  ser\_name={ser\_name}&ser\_category\_id={Valid\_ser\_category\_id} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid ser\_category\_id |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |