# ETSI GS MEC 016 V2.1.1 (2019-04)


# Multi-access Edge Computing (MEC); UE application interface

***Disclaimer***

The present document has been produced and approved by the Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
RGS/MEC-0016v211UEAppInterface

Keywords
API, MEC

***ETSI***

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

# Contents

# Intellectual Property Rights

## Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

## Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

# Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Multi-access Edge Computing (MEC).

# Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

# 1 Scope

The present document contains the specification for the lifecycle management of user applications over the UE application interface. This interface is over the Mx2 reference point between the device application in the UE and the User Application LifeCycle Management Proxy (UALCMP) in the MEC system.

The present document covers the following lifecycle management operations: user application look-up, request for the user application instantiation, and the request for the user application termination. In addition, a mechanism is specified for the exchange of lifecycle management related information between the MEC system and the UE application.

The intended key audience of the present document are the application developers for the MEC system, since the UE application interface provides them with a method to instantiate their applications.

NOTE: User application mobility related lifecycle management operations are not covered by the present document.

# 2 References

## 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference/.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI GS MEC 010-2: "Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management".

[2] IETF RFC 2818: "HTTP Over TLS".

NOTE: Available at https://tools.ietf.org/html/rfc2818.

[3] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

NOTE: Available at https://tools.ietf.org/html/rfc8446.

[4] ETSI GS MEC 009: "Multi-access Edge Computing (MEC); General principles for MEC Service APIs".

[5] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

NOTE: Available at https://tools.ietf.org/html/rfc6749.

[6] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

NOTE: Available at https://tools.ietf.org/html/rfc6750.

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GS MEC 001: "Multi-access Edge Computing (MEC); Terminology".

[i.2] ETSI GS MEC 002: "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements".

[i.3] OpenAPI Specification.

NOTE: Available at https://github.com/OAI/OpenAPI-Specification.

# 3 Definition of terms, symbols and abbreviations

## 3.1 Terms

For the purposes of the present document, the terms given in ETSI GS MEC 001 [i.1] and the following apply:

**user application lifecycle management proxy:** system level functional element that allows specific and authorized requests from the device application for the user application lifecycle management

## 3.2 Symbols

Void.

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [i.1] and the following apply:

| | |
|---|---|
| AA | Authentication and Authorization |
| API | Application Programming Interface |
| OSS | Operations Support System |
| TLS | Transport Layer Security |
| UALCMP | User Application LifeCycle Management Proxy |
| UE | User Equipment |
| URI | Uniform Resource Identifier |

# 4 Overview

The present document specifies the API for the UE application interface to support the corresponding requirements defined for the Multi-access Edge Computing in ETSI GS MEC 002 [i.2].

Clause 5 describes how the UE application interface can be used by the device application and by the MEC system. It describes the information flows for the procedures over the UE application interface.

The information that is exchanged over the UE application interface is described in clause 6, providing detailed description of all information elements available on that interface.

Clause 7 describes the actual API of the UE application interface, providing detailed information how the information elements map into the RESTful API design of the interface.

Clause 8 describes the authentication, authorization and access control for the UE application interface.

# 5 Description of the service (informative)

## 5.1 Sequence diagrams

### 5.1.1 Introduction

The following clauses describe how the device application interacts with the UALCMP over the UE application interface. The sequence diagrams that are relevant for the UE application interface are presented.

The device application presents the access token to the UALCMP with every request in order to assert that it is allowed to access the resource with the particular method it invokes. The access token is included in the "Authorization" request header field as a bearer token according to IETF RFC 6750 [6].

### 5.1.2 User application look-up

The user application look-up is the procedure for requesting the list of available user applications in the MEC system to the requesting device application. The user application look-up procedure is illustrated in figure 5.1.2-1.

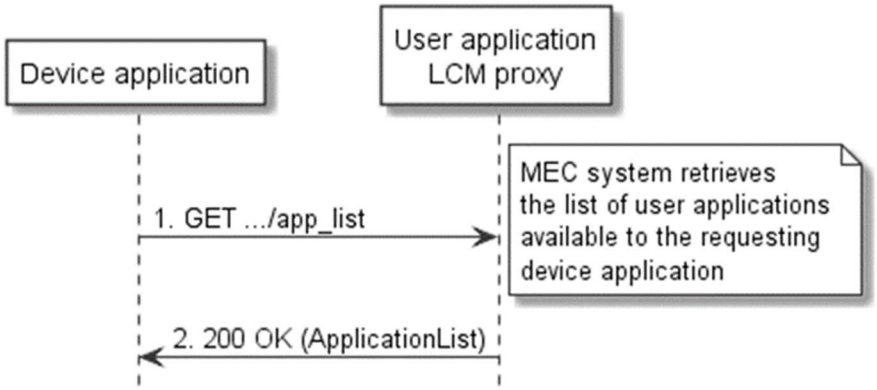


**Figure 5.1.2-1: User application look-up**

1) The device application submits the GET request to the UALCMP. The UALCMP authorizes the request from device application. The MEC system retrieves the list of user applications available to the requesting device application.

2) The UALCMP returns the 200 OK response to the device application, with the message body containing the data structure for the list of available user applications.

### 5.1.3 Application context create

The application context create is the procedure to request either to join with an available user application or to instantiate a new user application. The application context create procedure is illustrated in figure 5.1.3-1.

As part of the user application instantiation, the MEC system will create an associated application context that the MEC system maintains for the lifetime of the user application. The application context contains information specific to the application instance such as its unique identifier within the MEC system and the address (URI) provided for clients that are external to the MEC system to interact with the user application.

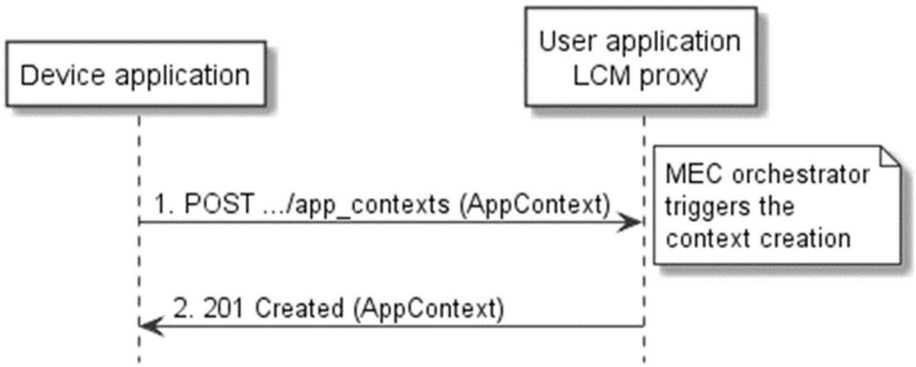


**Figure 5.1.3-1: Application context create**

1) The device application submits the POST request to the UALCMP. The message body contains the data structure for the application context to be created.

2) The UALCMP authorizes the request from the device application. The request is forwarded to the OSS. The OSS makes the decision on granting the context creation request. The MEC orchestrator triggers the creation of the application context in the MEC system.

3) The UALCMP returns the 201 Created response to the device application with the message body containing the data structure of the created application context, which includes the address (reference URI) provided for clients that are external to the MEC system to interact with the user application. The response message header contains the address of the resource relating to the application instance context created and maintained by the MEC system.

## 5.1.4 Application context delete

The application context delete is a procedure in which the device application requests the deletion of the application context. The application context delete procedure is illustrated in figure 5.1.4-1.

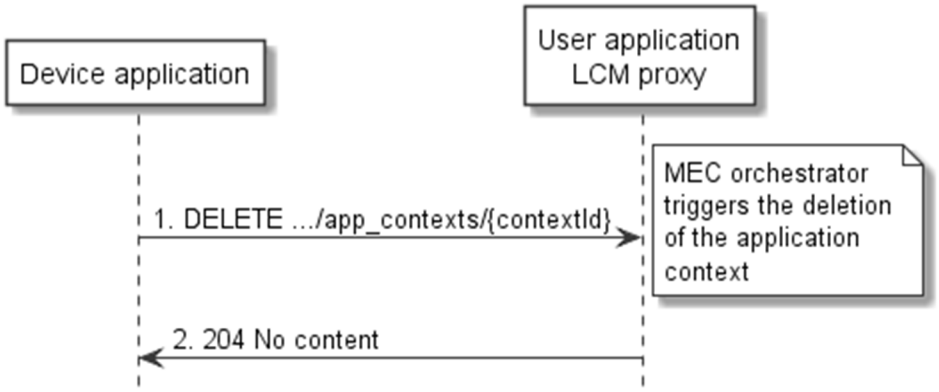


**Figure 5.1.4-1: Application context delete**

1) The device application submits the DELETE request to the UALCMP for the application context to be deleted.

2) The UALCMP authorizes the request from device application. The request is forwarded to the OSS. The OSS makes the decision on granting the deletion. The MEC orchestrator triggers the deletion of the application context, including deletion of the resource maintained by the MEC system that represents it.

3) The UALCMP returns "204 No content" response.

## 5.1.5 Application context update

The UALCMP is provided with an update of the application context. The procedure is illustrated in figure 5.1.5-1.

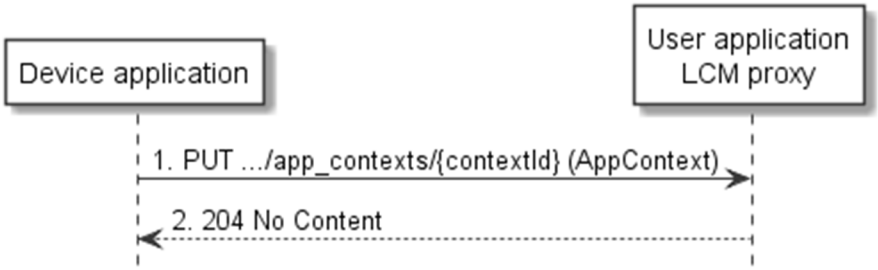


**Figure 5.1.5-1: Application context update**

1) The device application updates a specific application context by sending a PUT request to the resource within the MEC system that represents it, with the message body containing the modified data structure of AppContext in which only the callback reference may be updated.

2) The UALCMP returns a "204 No Content" response.

## 5.1.6 Receiving notification events

Figure 5.1.6-1 presents the scenario where the UALCMP sends notification events to the device application.

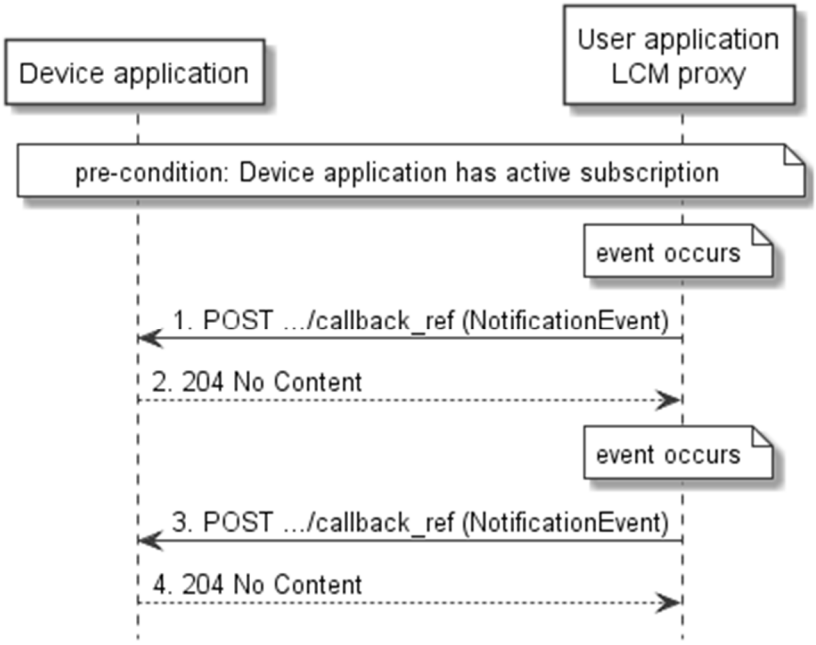


**Figure 5.1.6-1: Flow of receiving notification events**

Receiving notification events, as illustrated in figure 5.1.6-1, consists of the following steps:

1) The UALCMP sends a POST message to the callback reference address provided by the device application as part of application context creation, with the message body containing the notification event, indicating for instance a modification to the address of the user application.

2) The device application sends a "204 No Content" response to the UALCMP.

# 6 Data model

## 6.1 Introduction

The following clauses provide the description of the data model.

## 6.2 Resource data types

### 6.2.1 Introduction

This clause defines data structures to be used in resource representations.

### 6.2.2 Type: ApplicationList

This type represents the information on available applications. The device application can acquire this information by user application look-up procedure described in clause 5.1.2.

The elements of the ApplicationList shall follow the notations provided in table 6.2.2-1.

**Table 6.2.2-1: Definition of type ApplicationList**

| Attribute name | Data type | Cardinality | Description |
|---|---|---|---|
| appList | Array (Structure (inlined)) | 0..N | List of user applications available to the device application. As defined below. |
| >appInfo | Structure (inlined) | 1 | |
| >>appDId | String | 1 | Identifier of this MEC application descriptor. It is equivalent to the appDId defined in clause 6.2.1.2 of ETSI GS MEC 010-2 [1]. This attribute shall be globally unique. |
| >>appName | String | 1 | Name of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >>appProvider | String | 1 | Provider of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >>appSoftVersion | String | 1 | Software version of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >>appDVersion | String | 1 | Identifies the version of the application descriptor. It is equivalent to the appDVersion defined in clause 6.2.1.2 of ETSI GS MEC 010-2 [1]. |
| >>appDescription | String | 1 | Human readable description of the MEC application (see note 2). |
| >>appCharcs | Structure (inlined) | 0..1 | Characteristics of the application.<br>As defined below.<br>The application characteristics relate to the system resources consumed by the application. device application can use this information e.g. for estimating the cost of use of the application or for the expected user experience. |
| >>>memory | uint32 | 0..1 | The maximum size in Mbytes of the memory resource expected to be used by the MEC application instance in the MEC system. |
| >>>storage | uint32 | 0..1 | The maximum size in Mbytes of the storage resource expected to be used by the MEC application instance in the MEC system. |
| >>>latency | uint32 | 0..1 | The target round trip time in milliseconds supported by the MEC system for the MEC application instance. |
| >>>bandwidth | uint32 | 0..1 | The required connection bandwidth in kbit/s for the use of the MEC application instance. |
| >>>serviceCont | Enum | 0..1 | Required service continuity mode for this application.<br>Permitted values:<br>0 = SERVICE_CONTINUITY_NOT_REQUIRED.<br>1 = SERVICE_CONTINUITY_REQUIRED. |
| >vendorSpecificExt | Structure (inlined) | 0..1 | Extension for vendor specific information (see note 1). |

| Attribute name | Data type | Cardinality | Description |
|---|---|---|---|
| >>vendorId | String | 1 | Vendor identifier.<br>The length of the value shall not exceed 32 characters.<br>The rest of the structure of vendor specific extension is not defined. |
| NOTE 1: The vendor specific extension allows submitting information on the application lists that have been made available to the device application of the corresponding vendor.<br>NOTE 2: The language support may be limited The length of the value shall not exceed 128 characters. | | | |

## 6.2.3 Type: AppContext

This type represents the information on application context created by the MEC system.

The elements of the AppContext shall follow the notations provided in table 6.2.3-1.

**Table 6.2.3-1: Definition of type AppContext**

| Attribute name | Data type | Cardinality | Description |
|---|---|---|---|
| contextId | String | 0..1 | Uniquely identifies the application context in the MEC system.<br>Assigned by the MEC system and shall be present other than in a create request.<br><br>The length of the value shall not exceed 32 characters. |
| associateUeAppId | String | 1 | Uniquely identifies the device application.<br>The length of the value shall not exceed 32 characters. |
| callbackReference | URI | 0..1 | URI assigned by the device application to receive application lifecycle related notifications. Inclusion in the request implies the client supports the pub/sub mechanism and is capable of receiving notifications. This endpoint shall be maintained for the lifetime of the application context. |
| appInfo | Structure (inlined) | 1 | |
| >appDId | String | 0..1 | Identifier of this MEC application descriptor. This attribute shall be globally unique. It is equivalent to the appDId defined in clause 6.2.1.2 of ETSI GS MEC 010-2 [1].<br>It shall be present if the application is one in the ApplicationList. |
| >appName | String | 1 | Name of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >appProvider | String | 1 | Provider of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >appSoftVersion | String | 0..1 | Software version of the MEC application.<br>The length of the value shall not exceed 32 characters. |
| >appDVersion | String | 1 | Identifies the version of the application descriptor. It is equivalent to the appDVersion defined in clause 6.2.1.2 of ETSI GS MEC 010-2 [1]. |
| >appDescription | String | 0..1 | Human readable description of the MEC application.<br>The length of the value shall not exceed 128 characters. |
| >referenceURI | URI | 0..1 | Address of the user application.<br>It shall only be included in the response. |

| Attribute name | Data type | Cardinality | Description |
|---|---|---|---|
| >appPackageSource | URI | 0..1 | URI of the application package.<br>Included in the request if the application is not one in the ApplicationList.<br><br>appPackageSource enables on-boarding of the application package into the MEC system. The application package shall comply with the definitions in clause 6.2.1.2 of ETSI GS MEC 010-2 [1]. |
| NOTE 1: If a value of the attribute is included in the request, the same value shall be included in the response.<br>NOTE 2: The design of the current operation with callback reference assumes no web proxy between the entity that originates the notification and the entity that receives it.<br>NOTE 3: The language support for the application description may be limited. | | | |

## 6.3 Subscription data types

In the present document, no subscription data types are defined.

## 6.4 Notification data types

### 6.4.1 Introduction

This clause defines data structures for notifications.

### 6.4.2 Type: NotificationEvent

This type defines the parameters used in the method "Receiving notification events".

**Table 6.4.2-1: Definition of type NotificationEvent**

| Attribute name | Data type | Cardinality | Description |
|---|---|---|---|
| referenceURI | URI | 1 | Address of the user application. Used as the reference URI for the application.<br>Assigned by the MEC system. |

## 6.5 Referenced structured data types

In the present document, no referenced structured data types are defined.

# 7 API definition

## 7.1 Introduction

This clause defines the resources and operations of the UE application interface API on the Mx2 reference point.

## 7.2 Global definitions and resource structure

All resource URIs of this API shall have the following root:

**{apiRoot}/{apiName}/{apiVersion}/**

The "apiName" shall be set to "mx2"and the "apiVersion" shall be set to "v2" for the present document. It includes the scheme ("https"), host and optional port, and an optional prefix string. The API shall support HTTP over TLS (also known as HTTPS, see IETF RFC 2818 [2]). TLS version 1.2 as defined by IETF RFC 8446 [3] shall be supported. HTTP without TLS is not recommended. All resource URIs in the sub-clauses below are defined relative to the above root URI.

The content format of JSON shall be supported. The JSON format is signalled by the content type "application/json".

This API supports additional application-related error information to be provided in the HTTP response when an error occurs. See clause 7.15 of ETSI GS MEC 009 [4] for more information.

Figure 7.2-1 illustrates the resource URI structure of this API.

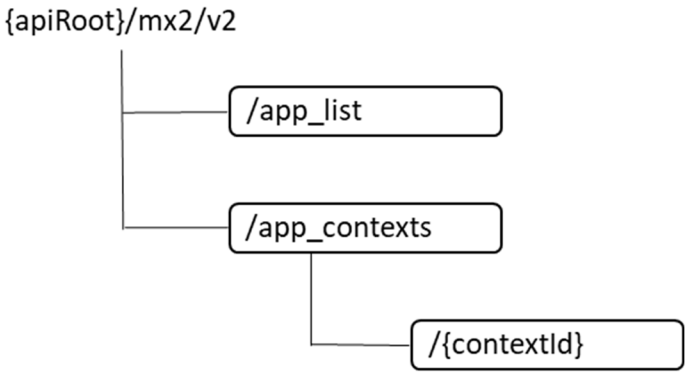


**Figure 7.2-1: Resource URI structure of the mx2 API**

Table 7.2-1 provides an overview of the resources defined by the present specification, and the applicable HTTP methods.

**Table 7.2-1: Resources and methods overview**

| Resource name | Resource URI | HTTP method | Meaning |
|---|---|---|---|
| meAppList | /app_list | GET | Retrieve available application information. |
| Parent resource of all ueAppContexts | /app_contexts | POST | For requesting the creation of a new application context. |
| Individual ueAppContext | /app_contexts/{contextId} | PUT | For updating the callbackReference of the existing application context. |
| | | DELETE | For requesting the deletion of an existing application context. |

## 7.3 Resource: meAppList

### 7.3.1 Description

This resource can be queried to retrieve information on available application information.

### 7.3.2 Resource definition

Resource URI: **{apiRoot}/mx2/v2/app_list**

Resource URI variables for this resource are defined in table 7.3.2-1.

**Table 7.3.2-1: Resource URI Variables for resource "meAppList"**

| Name | Definition |
|---|---|
| apiRoot | See clause 8.2 |

## 7.3.3 Resource Methods

### 7.3.3.1 GET

The GET method is used to query information about the available MEC applications.

This method shall comply with URI query parameters, request and response data structures, and response codes, as specified in the tables 7.3.3.1-1 and 7.3.3.1-2.

**Table 7.3.3.1-1: URI query parameters supported by the GET method on this resource**

| Name | Data type | Cardinality | Remarks |
|---|---|---|---|
| appName | String | 0..N | Name to identify the MEC application. |
| appProvider | String | 0..N | Provider of the MEC application. |
| appSoftVersion | String | 0..N | Software version of the MEC application. |
| serviceCont | Enum (inlined) | 0..1 | Required service continuity mode for this application.<br>Permitted values:<br>0 = SERVICE_CONTINUITY_NOT_REQUIRED.<br>1 = SERVICE_CONTINUITY_REQUIRED. |
| vendorId | String | 0..N | Vendor identifier |
| NOTE: The value of the attribute of the type String shall not exceed the length of 32 characters. | | | |

**Table 7.3.3.1-2: Data structures supported by the GET request/response on this resource**

| Request body | Data type | Cardinality | Remarks | |
|---|---|---|---|---|
| | n/a | | | |
| **Response body** | **Data type** | **Cardinality** | **Response codes** | **Remarks** |
| | ApplicationList | 1 | 200 OK | The response body contains an array of the user applications available to the querying device application. |
| | ProblemDetails | 0..1 | 400 Bad Request | Incorrect parameters were passed in the request.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 401 Unauthorized | An erroneous or missing bearer token.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 404 Not Found | The client provided a URI that cannot be mapped to a valid resource URI.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 1 | 403 Forbidden | The operation is not allowed given the current status of the resource.<br><br>More information shall be provided in the "detail" attribute of the "ProblemDetails" structure. |

#### 7.3.3.2 PUT

Not applicable.

#### 7.3.3.3 PATCH

Not applicable.

#### 7.3.3.4 POST

Not applicable.

#### 7.3.3.5 DELETE

Not applicable.

## 7.4 Resource: all ueAppContexts

### 7.4.1 Description

This resource represents the parent for all individual application contexts.

### 7.4.2 Resource definition

Resource URI: **{apiRoot}/mx2/v2/app_contexts**

Resource URI variables for this resource are defined in table 7.4.2-1.

**Table 7.4.2-1: Resource URI Variables for resource "all ueAppContexts"**

| Name | Definition |
|---|---|
| apiRoot | See clause 7.2 |

### 7.4.3 Resource Methods

#### 7.4.3.1 GET

Not applicable.

#### 7.4.3.2 PUT

Not applicable.

#### 7.4.3.3 PATCH

Not applicable.

#### 7.4.3.4 POST

The POST method is used to create a new application context. Upon success, the response contains entity body describing the created application context.

This method shall comply with the URI query parameters, the request and response data structures, and response codes, as specified in table 7.4.3.4-1.

**Table 7.4.3.4-1: Data structures supported by the POST request/response on this resource**

| | Data type | Cardinality | Remarks | |
|---|---|---|---|---|
| **Request body** | AppContext | 1 | Entity body in the request contains the Application Context as requested by the device application. | |
| **Response body** | **Data type** | **Cardinality** | **Response codes** | **Remarks** |
| | AppContext | 1 | 201 Created | The response body contains the Application Context as it was created by the MEC system, which includes the reference URI of the associated user application.<br>The URI of the resource created within the MEC system associated with the request, with its specific application context ID, shall be included in the “Location” HTTP header of the response. |
| | ProblemDetails | 0..1 | 400 Bad Request | Incorrect parameters were passed in the request.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 401 Unauthorized | An erroneous or missing bearer token.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 404 Not Found | The client provided a URI that cannot be mapped to a valid resource URI.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 1 | 403 Forbidden | The operation is not allowed given the current status of the resource.<br><br>More information shall be provided in the "detail" attribute of the "ProblemDetails" structure. |

#### 7.4.3.5 DELETE

Not applicable.

## 7.5 Resource: individual ueAppContext

### 7.5.1 Description

This resource represents one application context the MEC system has created.

### 7.5.2 Resource definition

Resource URI: **{apiRoot}/mx2/v2/app_contexts/{contextId}**

Resource URI variables for this resource are defined in table 7.5.2-1.

**Table 7.5.2-1: Resource URI Variables for resource " individual ueAppContext"**

| Name | Definition |
|---|---|
| apiRoot | See clause 7.2. |
| contextId | Uniquely identifies the application context in the MEC system. It is assigned by the MEC system. |

## 7.5.3 Resource Methods

### 7.5.3.1 GET

Not applicable.

### 7.5.3.2 PUT

The PUT method is used to update the callback reference of the existing application context. Upon successful operation, the target resource is updated with new callback reference.

This method shall comply with the URI query parameters, request and response data structures, and response codes, as specified in tables 7.5.3.2-1 and 7.5.3.2-2.

**Table 7.5.3.2-1: URI query parameters supported by the PUT method on this resource**

| Name | Data type | Cardinality | Remarks |
|---|---|---|---|
| n/a | | | |

**Table 7.5.3.2-2: Data structures supported by the PUT request/response on this resource**

| | Data type | Cardinality | Response codes | Remarks |
|---|---|---|---|---|
| **Request body** | AppContext | 1 | | Only the attribute callbackReference is allowed to be updated. Other attributes and their values shall remain untouched. |
| **Response body** | **Data type** | **Cardinality** | **Response codes** | **Remarks** |
| | n/a | | 204 No Content | Upon success, a response 204 No Content without any response body is returned. |
| | ProblemDetails | 0..1 | 400 Bad Request | Incorrect parameters were passed in the request.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 401 Unauthorized | An erroneous or missing bearer token.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 404 Not Found | The client provided a URI that cannot be mapped to a valid resource URI.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 1 | 403 Forbidden | The operation is not allowed given the current status of the resource.<br><br>More information shall be provided in the "detail" attribute of the "ProblemDetails" structure. |

#### 7.5.3.3 PATCH

Not applicable.

#### 7.5.3.4 POST

Not applicable.

#### 7.5.3.5 DELETE

The DELETE method is used to delete the resource that represents the existing application context.

This method shall comply with the URI query parameters, request and response data structures, and response codes, as specified in tables 7.5.3.5-1 and 7.5.3.5-2.

**Table 7.5.3.5-1: URI query parameters supported by the DELETE method on this resource**

| Name | Data type | Cardinality | Remarks |
|---|---|---|---|
| n/a | | | |

**Table 7.5.3.5-2: Data structures supported by the DELETE request/response on this resource**

| | | | | |
|---|---|---|---|---|
| **Request body** | **Data type** | **Cardinality** | **Remarks** | |
| | n/a | | | |
| **Response body** | **Data type** | **Cardinality** | **Response codes** | **Remarks** |
| | n/a | | 204 No Content | Upon success, a response 204 No Content without any response body is returned. |
| | ProblemDetails | 0..1 | 400 Bad Request | Incorrect parameters were passed in the request.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 401 Unauthorized | An erroneous or missing bearer token.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 0..1 | 404 Not Found | The client provided a URI that cannot be mapped to a valid resource URI.<br><br>More information should be provided in the "detail" attribute of the "ProblemDetails" structure. |
| | ProblemDetails | 1 | 403 Forbidden | The operation is not allowed given the current status of the resource.<br><br>More information shall be provided in the "detail" attribute of the "ProblemDetails" structure. |

# 8 Authentication, authorization and access control

## 8.1 Introduction

OAuth 2.0 is applied on UE application interface as defined in clause 8.2.

## 8.2 Description

An authentication and authorization entity is assumed to be available for both the REST client, i.e. the device application, and the REST server represented by the UALCMP. The AA entity exposes the token endpoint for the device application as defined by OAuth 2.0. In the present document, the client credentials flow of OAuth 2.0 (see IETF RFC 6749 [5]) shall be supported by the AA entity, and it may be used by the device application to obtain the access token. The AA entity performs the authentication for the credentials of the device application. Other means for the device application to obtain the access token are outside the scope of the present document. The AA entity, and the communication between it and the UALCMP are out of scope of the present document.

The device application shall present the access token to the UALCMP with every request in order to assert that it is allowed to access the resource with the particular method it invokes. The access token shall be included as a bearer token according to IETF RFC 6750 [6].

On UE application interface the access rights of the device application are bound to its access token. Additional policies can also be bound to access token, such as the maximum frequency of API calls. An access token has a lifetime after which it is invalid. An AA entity may revoke an access token before it expires.

# Annex A (informative): Complementary material for API utilization

To complement the definitions for each method and resource defined in the interface clauses of the present document, ETSI MEC ISG is providing for the UE application interface API a supplementary description file compliant to the OpenAPI Specification [i.3].

In case of discrepancies between the supplementary description file and the related data structure definitions in the present document, the data structure definitions take precedence.

The supplementary description file, relating to the present document, is located at https://forge.etsi.org/rep/gitweb.cgi/MEC.GS_016.git.

# History

| Document history | | |
|---|---|---|
| V1.1.1 | September 2017 | Publication |
| V2.1.1 | April 2019 | Publication |
| | | |
| | | |
| | | |