# OPENi Mobile Client for Web Application

# OPENi JavaScript SDK (v1.0)

## # OPENi WP4 Release 1.0 (March 2014)

## Introduction

This document serves as user guide for OPENi application developers, detailing the use and features of **OPENi JavaScript SDK v1.0**.

OPENi JavaScript SDK is a lightweight client library (SDK) that enables OPENi web and mobile web applications to communicate and interact with OPENi Cloudlet Platform and the corresponding Cloudlet API.

**OPENi Mobile Client (OPENi js SDK) Features**

1. **A lightweight SDK.** OPENi jsSDKis an easy to integrate, use and exploit SDK imposing minimum overhead to application developers and thus, in the functionality of the application.
2. **A Dynamically Adapted SDK.** Due to the multiplicity and dynamicity of OPENi API (that is new API calls, objects, object types, etc, can be included/updated/deleted by OPENi application developers), OPENi js SDK follows the same dynamic nature. To achieve the latter, open source [Swagger](https://github.com/wordnik/swagger-core/wiki) Tool (Apache License, Version 2.0) has been exploited for both Cloudlet API documentation/run-time demo purposes and thus, for *dynamically configuring OPENi js SDK*.

**Assumptions**

In the rest of this document we make the following assumptions.

1. The reader of this document is familiar to web/mobile web applications development.
2. The reader (OPENi Application Developer) has access to a stable Cloudlet environment, either provided by OPENi or deployed in his own facilities (by downloading and installing OPENi Cloudlet code).
3. For completeness purposes, in the following an example Cloudlet installation has been assumed where, Cloudlet API UI Swagger-enabled is located under the following base URL: <http://openi-qa.velti.com/api-docs> (an OPENi Application developer should replace the latter URL with the one of her/his own Cloudlet installation Swagger URL).

## OPENi Cloudlet js SDK (v1.0)

In this section the use of OPENi Cloudlet js SDK is detailed, enabling OPENi applications to call Cloudlet API. The OPENi SDK for JavaScript provides a rich set of client-side functionality that enables OPENi application developers to:

* Create and manage end mobile users Cloudlets.
* Manage the Cloudlet objects used, created and update by their application.
* Search on object, object types, e.t.c,
* Makes it easy to call into Cloudlet API

This document details you how to setup the SDK and get it to make some basic Cloudlet API calls.

## I. Initiation and Basic Setup

OPENi SDK for JavaScript doesn't have any standalone files that need to be downloaded or installed. Moreover, OPENi application developer for web/mobile web applications need to include a short piece of **JavaScript** in their HTML/HTML5 that will asynchronously loads the SDK into their pages. The async load means that it does not block loading other elements.

The following lines should be inserted to <head> section of HTML.

<script src='shred.bundle.js' type='text/javascript'></script>

<script src='swagger.js' type='text/javascript'></script>

Basic version of the SDK can be initiated by placing the following snippet at <head> section on each page:

<script type="text/javascript">

function initSwagger() {

window.swagger = new SwaggerApi({

// url: "http://dev.openi-ict.eu/api-spec/v1/",

url: "http://openi-qa.velti.com/api-spec/v1/",

success: function () {

if (swagger.ready === true) {

console.log("swagger is ready");

} else {

console.log("swagger is not ready");

}

}

});

}

</script>

**Note 1:** swagger.js and shred.bundle.js need to be added using script tag.

**Note 2:** “url” is the url of the Cloudlet API swagger-enabled server. url contains the api resource declaration and generates JavaScript methods on the fly. This should be replaced with the corresponding one according to OPENi application developer OPENi Cloudlet swagger-enabled server.

## II. Using the SDK to call OPENi Cloudlet API

To read or write, create or delete data via OPENi Cloudlet API, upon Cloudlet swagger UI is ready, developers can call methods for every API. **Note:** Existing APIs in Cloudlet Release 1 (Deliverable D4.2) are:

swagger.apis.cloudlets

swagger.apis.objects

swagger.apis.search

swagger.apis.subscription

swagger.apis.attachments

To get available methods for each API use help() method. For example, available methods for cloudlets API are:

swagger.apis.cloudlets.help()

[" createCloudlet", " deleteCloudletById", " exportCloudletById"]

These are the method names. A description on how these methods can be called can also be retrieved with help().

**Example:** Let’s try getting help on how to use createCloudlet() method:

swagger.apis.cloudlets.createCloudlet.help();

"\* data (required) - The Cloudlet data"

To create a cloudlet, requested data is a JSON object that should be passed at createCloudlet() method. Data structure are described in detail at swagger UI, so it is recommended to visit it (e.g., in our example <http://openi-qa.velti.com/api-docs/> ) and see how data needs to be constructed.

**Note:** OPENi application developers are highly advised to use the same approach on all other methods you will use at your application.

The actual JavaScript code needed to create a cloudlet is the following:

function createCloudlet(username, password) {

console.log("Creating cloudlet");

var json = JSON.stringify({

alias: username,

username: password

});

console.log(json);

swagger.apis.cloudlets.createCloudlet({

body: json

}, function (response) {

console.log(response);

if (response.status == 200) {

console.log("Cloudlet created successfully.");

var data = JSON.parse(response.data);

var cloudletId = data.data.cloudletId;

console.log("cloudletId: " + cloudletId);

}

}, function (error) {

console.log(error);

});

}