Ulbora CMS Add-ons

Intro:

Ulbora CMS Add-ons are simple REST web services. Any existing REST web service meeting the Ulbora CMS Add-on specification can be registered and used from within Ulbora CMS. Access to Add-ons is provided by a single Ulbora CMS pass-through web service. Details about the specification and how to access Add-ons follows.

Specification:

Ulbora CMS Add-ons must meet the following specs:

1. Must be a REST service
2. Must use JSON as data-interchange format
3. Secure services must use basic authentication
4. Must implement CORS
5. Must return in 2 seconds or less for optimal performance.

As you can see, many existing web services already meet the Ulbora CMS Add-ons specification and can be registered and used from within Ulbora CMS with no issues.

How Add-ons Work:

Calls to Add-ons are invoked through a call to the following Ulbora CMS service:

A) Url:

POST: http://localhost:8080/UlboraCMSAPI/rs/addons/call

Your url here

B) Media Type:

application/json

In figure 1 below, jsonRequest is passed as a **UTF-8 encoded byte array**.

Request:

Figure 1


```
{
  "addOnId":230,
  "addOnName":"Payment Service",
  "method":"POST",
  "serviceName":"payment",
  "methodName":null,
  "jsonRequest":"PGRpdiBzdHlsZT0icGFnZS1icmVhay1iZWZvcmU6YWx3YXlzOyBwYWdlLWJyZWFrLWFmdGVyO
mFsd2F5cyI+PGRpdj48cD48Yj5UaGlzIGlzIHRoZSBUaXRsZQombHQ7L2ImZ3Q7PC9wPgo8cD5UaGlzIGlzIGEgdGVz
dCBwYXJhZ3JhcGguCjwvcD4KPGltZyBzcmM9Imh0dHA6Ly9sb2NhbGhvc3Q6ODA4MC9VbGJvcmFDTVNBUEkvcnMvaW1
hZ2UvZ2V0LzEyIiAgc3R5bGU9ImZsb2F0OiBsZWZ0OyIgIGFsdD0iaW1hZ2UiIC8+CjxwPlRoaXMgaXMgYW5vdGhlci
BwYXJhZ3JhcGguCjwvcD4KPHA+e3RhYmxlTGVmdFN0YXJ0fQpbe1Rlc3QgMX0ge1Rlc3QgdmFsfV0KW3tUZXN0IDJ9I
HtUZXN0IDIgdmFsfV0KPC9wPgo8cD57dGFibGVMZWZ0RW5kfQo8L3A+CjxwPnt0YWJsZUNlbnRlclN0YXJ0fQpbe1Rl
c3QgMX0ge1Rlc3QgdmFsfV0KW3tUZXN0IDJ9IHtUZXN0IDIgdmFsfV08L3A+Cgo8L2Rpdj48L2Rpdj4KPGRpdiBzdHls
ZT0icGFnZS1icmVhay1iZWZvcmU6YWx3YXlzOyBwYWdlLWJyZWFrLWFmdGVyOmFsd2F5cyI+PGRpdj48cD57dGFibG
VDZW50ZXJFbmR9CjwvcD4KPHRhYmxlIGFsaWduPSdyaWdodCcgc3R5bGU9J2JvcmRlcjogYmxhY2sgc29saWQgMXB4O
yAgd2lkdGg6IDEwMCU7Jz48dHI+PHRkPjxpbWcgc3R5bGU9J2JvcmRlcjogMTwvdGQ+PC90cj48L3RhYmxlPgo8cD48
dGFibGUgYWxpZ249J3JpZ2h0JyBzdHlsZT0nYm9yZGVyOiBibGFjayBzb2xpZCAxcHg7IHdpZHRoOiAxMDAlOyc+PHRy
PjxwPjwvcD48L3RkPjwvdHI+PC90YWJsZT4KPHA+PC9wPgo8cD48L3A+CjxwPjwvcD4KPHA+PC9wPgo8cD48L3A+Cg=="
}
```


Figure 2 shows a call to invoke an Add-on service that uses a GET. The Add-on is first registered with Ulbora CMS. The attribute “jsonResponse” in the response is also a **UTF-8 encoded byte array.** The call is as follows:

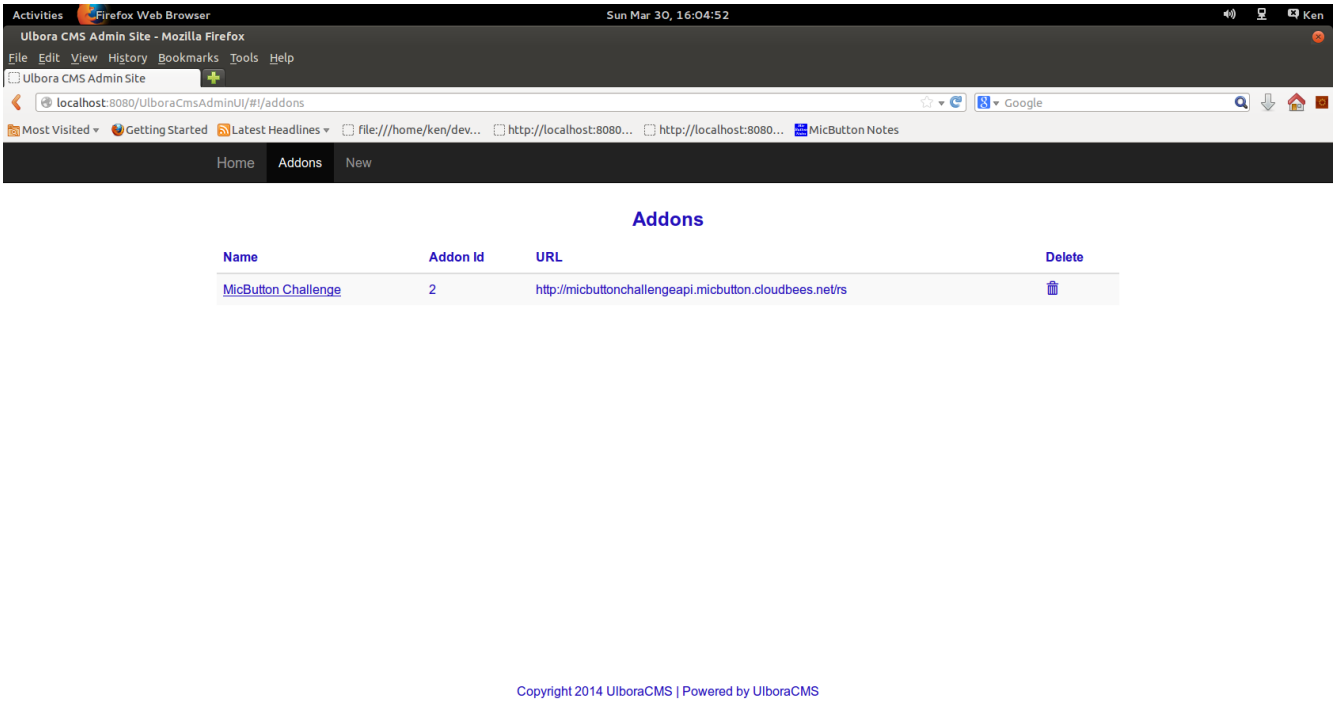

Figure 2 (request)


```
{
  "addOnId":2,
  "addOnName":"MicButton Challenge",
  "method":"GET",
  "serviceName":"challenge",
  "methodName":null,
  "queryParameters":{
    "languageCode":"en_US"
  }
}
```


Figure 3 (response)


```
{
   "success":true,
   "message":"Service Call Sucessful",
   "jsonResponse":"eyJxdWVzdGlvbiI6IldoaWNoIGlzIHdoaXRlLCBhIGR1Y2sgb3IgYSBibGFjayBjb3c/Iiwia2V5IjoiNzY2YjA3ZjAwMmFhODgwZjc3N2U5M2VjOWQyMjhhMDdlYWMxMzdhYyJ9"
}
```

The Attribute “jsonResponse” above translates to:

```
{
  "question":"Which is white, a duck or a black cow?",
  "key":"766b07f002aa880f777e93ec9d228a07eac137ac"
}
```

Sequence Diagram:

The sequence diagram below shows how an inventory Add-on could be used with a payment Add-on.

First, a call is made to the inventory Add-on to check the inventory for an item. Then a call is made to the payment Add-on to process a payment for that item. Once the payment is processed successfully, another call is made to the inventory Add-on to update payment information and inventory information.

The Ulbora CMS ***Add-on module service*** is merely a pass-through service call to the two registered Add-ons.

Figure 4

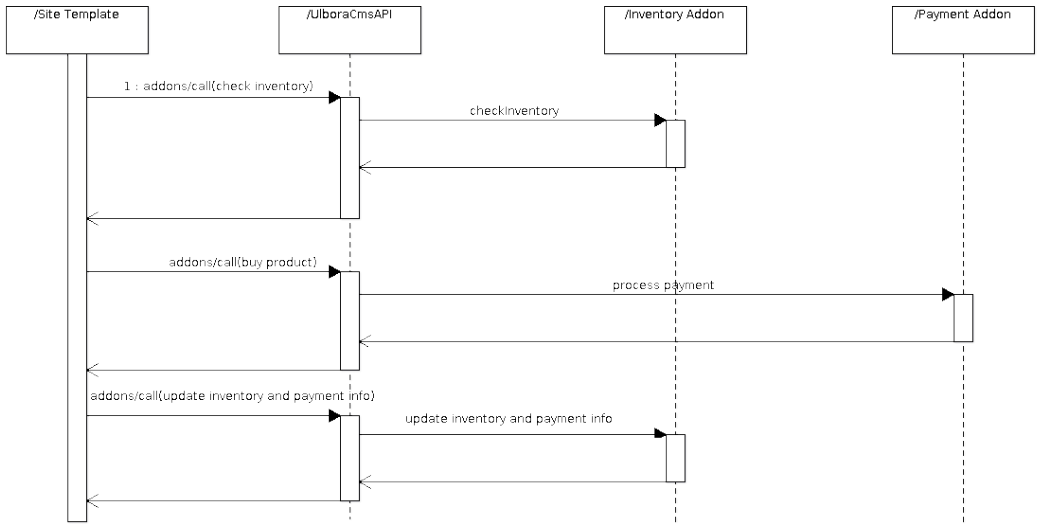

Figure 5 shows the complete Ulbora Add-on service request:

Figure 5 (Complete Request)

```
{
   "addOnId":230,
   "addOnName":"Addon Name",
   "method":"POST or PUT or DELETE or GET",
   "serviceName":"serviceName",
   "methodName":"methodName of null",
   "jsonRequest":"UTF-8 encoded byte array",
   "queryParameters":{
      "param1name":"param1value",
      "param2name":"param2value"
   },
   "pathParameters":[
      "param1value",
      "param2value"
   ]
}
```