

```

<?php

// Iperion-ch Simple IIIF Manifest builder: Version 1.0

//
// This page has been put together to demonstrate a simple method
// for building very basic iiif (http://iiif.io) json manifests
// The images have been included for demonstration purposes only
//
// This file has been created as part of the IPERION-CH project
// http://www.iperionch.eu
//
// Further details of the two images will be available via the NG API:
// http://data.ng-london.org.uk/resource/009-00QL-0000
// http://data.ng-london.org.uk/resource/009-00BJ-0000
//
// Please note at the time of wrtiing the NG API is not yet fully working
// but it should be up and running in early 2018
//
// For further information please contact joseph.padfield@ng-london.org.uk
// 05/01/2018

// iiif server prefix, resolvable for full thumbnail paths
// for example: http://media.ng-london.org.uk/iiif/009-00QL-0000/full/full/0/default.jpg
$iiif_url = "http://media.ng-london.org.uk/iiif/";
//example manifest url - this one might not resolve - used for the id
$manifest_url = "http://media.ng-london.org.uk/manifest/";
//does not seem to need to resolve
$json_url = "http://media.ng-london.org.uk/public/manifests/sequence/iiif/";

ob_start();

// Basic example manifest information - this should be replaced with
// specific details or a database call for dynamic details.
$manifest = array (
    //manifest comment
    "comment" => "This is an example manifest demonstrating the php iiif ".
        "manifest process",
    //Unique id does not need to resolve
    "id" => "Example-Manifest-01",
    // "id" => "http://scientific.ng-london.org.uk/tools/md/manifest_simple.php",
    //Unique display label for the manifest
    "label" => "A few small images just to display a working manifest",
    //Resolvable url to required logo image
    "logo" => "http://research.ng-london.org.uk/ng/graphics/ng_logo_tr_125.png",
    //For simple manifests this does not need to change, does not need to resolve
    "sequence_id" => "http://example.example/public/manifests/sequence/normal.json",
    "licence" => "https://creativecommons.org/licenses/by-nc-nd/4.0/",
    "attribution" => "These images are © The National Gallery, London. These works are
        licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0
        International License (CC BY-NC-ND 4.0)
        https://creativecommons.org/licenses/by-nc-nd/4.0/"
);

// Just in case their are special characters that will break the json formatting
$manifest["comment"] = json_encode($manifest["comment"]);
$manifest["label"] = json_encode($manifest["label"]);
$manifest["attribution"] = json_encode($manifest["attribution"]);

// Basic simple image array - this should be replaced with
// specific details or a database call for dynamic details.
$ims = array(
    0 => array(
        "image_name" => "The Raising of Lazarus",

```

```

"image_height" => 800,
"image_width" => 604,
"image_ppmm" => false,
"image_caption" => "<p>The subject of this painting is taken from the New Testament
(John: 11). At the request of the sisters Martha and Mary, Jesus visits the grave of
their brother Lazarus and raises him from the dead.<br \/><br \/>This work was painted
for Cardinal Giulio de' Medici in Rome in competition with Raphael's
'Transfiguration', now in the Vatican Gallery. It was subsequently taken to the
Cathedral of Narbonne. Some of the main figures are based on drawings which
Michelangelo supplied for the guidance of Sebastiano.<br \/><br \/>The painting was
purchased from the Angerstein collection.<\p>",
// this is part of the iiif url you can use the full file path
// or in this case we have used an image PID that resolves locally
"image_path" => "009-00QL-0000",
),
1 => array(
"image_name" => "The Madonna of the Basket",
"image_height" => 800,
"image_width" => 599,
"image_ppmm" => false,
"image_caption" => "<p>The subject is an unusual one of the Virgin dressing the Christ
Child. The jacket into which he has put one arm has been made by the Virgin; wool of
the same colour is in the basket beside her. Saint Joseph is busy with his carpentry
in the background. The outstretching of Christ's arms may be taken to foreshadow his
Crucifixion.<br><br>\r\nThis painting has been dated to about 1524 on the strength of
stylistic comparisons with the 'Piet\u00e0' (Parma, Galleria Nazionale). A number of
copies are known. This picture has always been accepted as the original of the design.
It is in superlative condition.<\p>",
"image_path" => "009-00BJ-0000",
)
//, 2 => .....
);

echo <<<END
{
"@context":"http://iiif.io/api/presentation/2/context.json",
"@id":"$manifest[id]",
"@type":"sc:Manifest",
"label":$manifest[label],
"license" : [
"$manifest[licence]"
],
"attribution":$manifest[attribution],
"logo":"$manifest[logo]",
"metadata":[
{
"label":"Description",
"value":$manifest[comment]
}
],
"description":$manifest[comment],
"viewingDirection":"left-to-right",
"viewingHint":"individuals",
"sequences":[
{
"@id":"$manifest[sequence_id]",
"@type":"sc:Sequence",
"label":"Normal Order",
"canvases":[

```

```

{
$nm = json_encode($d["image_name"]);
$nm = trim($nm, "\\");
$nm = "\\$json_url".$nm."/normal.json\\";
$cp = json_encode($d["image_caption"]."\nFile:".$d["image_name"].
    " (".$d["image_height"]."x".$d["image_width"]."px)");

if ($d["image_ppmm"])
{
$scale = ($d["image_ppmm"]/$d["image_width"])/$d["image_width"];
// Can be added to the canvas below, but does not allow you so set
// scale for each image just for each set of images.
$addRuler = '
"service": {
"@context": "http://iiif.io/api/annex/services/physdim/1/context.json",
"profile": "http://iiif.io/api/annex/services/physdim",
"physicalScale": '.$scale.',
"physicalUnits": "mm"

},';
}
else
{$addRuler = '';}

echo <<<END
$c
{
    "@id":$nm,
    "@type":"sc:Canvas",
    "label":$cp,
    "height":$d[image_height],
    "width":$d[image_width],
    $addRuler
    "images":[
        {
            "@type":"oa:Annotation",
            "motivation":"sc:painting",
            "resource":{
                "@id":"$iiif_url$d[image_path]/full/full/0/default.jpg",
                "@type":"dctypes:Image",
                "format":"image/jpeg",
                "height":$d[image_height],
                "width":$d[image_width],
                "service":{
                    "@context":"http://iiif.io/api/image/2/context.json",
                    "@id":"$iiif_url$d[image_path]",
                    "profile":"http://iiif.io/api/image/2/level2.json"
                }
            },
            "on":$nm
        }
    ]
}

END;
$c = ",";
}
echo <<<END
]
}
]
}

END;

$json = ob_get_contents();

```

```

ob_end_clean(); // Don't send output to client

header('Content-Type: application/json');
header("Access-Control-Allow-Origin: *");
echo $json;

// Simple example function for generating an "im" array from a simple
// local file path
function findAllIms2BChecked ()
{
    $dir = '/pics/pyrCheck';
    $files = scandir($dir);
    $files = array_slice($files, 2);

    $ims = array();

    foreach ($files as $k => $file)
    {
        $im = array();

        $str = trim($file);
        if (preg_match ('/^(.+)(-PYR.tif)$/', $str, $m))
        {
            $im["image_name"] = $m[1];
            $im["image_caption"] = "Checking: ";
            $im["imagepath"] = $dir."/". $str;

            $inf = getImInfo ($im["imagepath"]);

            $im["image_height"] = $inf["Ysize"];
            $im["image_width"] = $inf["Xsize"];
            $im["image_ppmm"] = false;

            $ims[] = $im;
        }
    }

    return ($ims);
}

?>

```