



JobRouter[®] Web Service Doc

Version 1.0.0 – 15.06.2016

Table of Contents

Jobrouter Web Services (v1)	3
Login Web Service:.....	3
Sample PHP code snippet:	3
Response (JSON):.....	4
Logout Web Service:	4
Sample PHP code snippet:	4
Response (JSON):.....	5
Processes Web Service:.....	5
Sample PHP code snippet:	5
Response (JSON):.....	6
Single Process Web Service:.....	6
Sample PHP code snippet:	6
Response (JSON):.....	7
Process Fields Web Service:	7
Sample PHP code snippet:	7
Response (JSON):.....	8
Process Incident Web Service:	8
Sample PHP code snippet:	9
Response (JSON):.....	10

Jobrouter Web Services (v1)

All web service calls can be made using class: **JobRouterWebService**. The class contains method: **sendCURLRequest** and it used to call JobRouter web services.

sendCURLRequest(url, params, request_method, token)

1. url (String) Complete URL to web service
2. params (Array) parameter used by web service
3. request method (String) [GET, POST, DELETE]
4. token (String) to access JobRouter web services token is required

Pass base URL when creating class: **JobRouterWebService** object i.e (Server/jobrouter/ws/v1) in constructor.

For example: \$obj = new JobRouterWebService(„http://myserver/jobrouter/ws/v1/“);

Login Web Service:

Feature	Description
Request Method	POST
URL	Server/jobrouter/ws/v1/login e.g (myserver /jobrouter/ws/v1/login)
Params	Username *, password *
Header	none
Response	JSON
Note	If false login count exceeded max login value then user will be blocked. Max login count can be adjust under System > configuration > Security Settings. Params with * post fix means required

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test JobRouter web services.

```
// Initialize web service call object
// Pass jobrouter web service URL as param in constructor
$jrwS = new JobRouterWebService(„http://myserver/jobrouter/ws/v1/“);
// Login web service params
$params = array(
    „username“ => „admin“,
    „password“ => „ABC123xyz“
);
// Create login web service URL
$url = $jrwS->getURL() . „login“;
// Return JSON response
$result = $jrwS->sendCurlRequest($url, $params, „POST“);
print_r($result);
```

Response (JSON):

Success	{“success”: {“userMessage”: “User logged in successfully”, “code”: 200, “token”: “i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt”}}
Failure	{“errors”: {“userMessage”: “Invalid Username or Password”, “more_info”: null, “code”: 400, “-”: [“Invalid Username or Password”]}}

Logout Web Service:

Feature	Description
Request Method	GET
URL	Server/jobrouter/ws/v1/logout e.g (myserver/jobrouter/ws/v1/logout)
Params	None
Header	Token (Return from login service)
Response	JSON
Note	Token should be passed under http header for security purpose.

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test JobRouter web services.

```
// Initialize web service call object
$jwrWS = new JobRouterWebService("http://myserver/jobrouter/ws/v1/");
// Logout web service params
$params = array();
// Token, fetched from login web service response
$token = „i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt“;
// Create logout web service URL
$url = $jwrWS->getURL() . 'logout';
// Return JSON response
$result = $jwrWS->sendCurlRequest($url, $params, 'GET', $token);
print_r($result);
```

Response (JSON):

Success	{"success":{"userMessage":"User logged out successfully","code":200}}
Failure	{"errors":{"userMessage":"Invalid token","more_info":null,"code":400,"-":["Invalid token"]}}

Processes Web Service:

This web service will list all processes details from JobRouter that a logged in user can manage.

Feature	Description
Request Method	GET
URL	Server/jobrouter/ws/v1/processes e.g (myserver/jobrouter/ws/v1/processes)
Params	None
Header	Token (Return from login service)
Response	JSON
Note	Token should be passed under http header for security purpose.

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test JobRouter web services.

```
// Initialize web service call object
$jwrWS = new JobRouterWebService("http://myserver/jobrouter/ws/v1/");
// processes web service params
$params = array();
// Token, fetched from login web service response
$token = „i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt“;
// Create processes web service URL
$url = $jwrWS->getURL() . 'processes';
// Return JSON response
$result = $jwrWS->sendCurlRequest($url, $params, 'GET', $token);
print_r($result);
```

Response (JSON):

Success	{"success":{"userMessage":"Request has been processed successfully","code":200,"processes":[{"processname":"XYZ"...}]}
Failure	{"errors":{"userMessage":"Invalid token","more_info":null,"code":400,"-":["Invalid token"]}}
Failure	{"errors":{"userMessage":"User is not authorized.","more_info":null,"code":405,"-":["User is not authorized."]}

Single Process Web Service:

This web service will list details of single process that a logged in user can manage.

Feature	Description
Request Method	GET
URL	Server/jobrouter/ws/v1/processes/:processname e.g (myserver/jobrouter/ws/v1/processes/Test)
Params	None
Header	Token (Return from login service)
Response	JSON
Note	Token should be passed under http header for security purpose.

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test JobRouter web services.

```
// Initialize web service call object
$jwrWS = new JobRouterWebService("http://myserver/jobrouter/ws/v1/");
// processes web service params
$params = array();
// Token, fetched from login web service response
$token = „i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt“;
// Create process web service URL
$processname = „Test“;
$url = $jwrWS->getURL() . 'processes/' . $processname;
// Return JSON response
$result = $jwrWS->sendCurlRequest($url, $params, 'GET', $token);
print_r($result);
```

Response (JSON):

Success	{"success":{"userMessage":"Request has been processed successfully","code":200,"processes":[{"processname":"Test","version":1,...}]}
Failure	{"errors":{"userMessage":"Invalid token","more_info":null,"code":400,"-":["Invalid token"]}}
Failure	{"errors":{"userMessage":"Invalid process name","more_info":null,"code":400,"-":["Invalid process name."]}}
Failure	{"errors":{"userMessage":"User is not authorized.","more_info":null,"code":405,"-":["User is not authorized."]}}

Process Fields Web Service:

This web service will list all input fields of process.

Feature	Description
Request Method	GET
URL	Server/jobrouter/ws/v1/process/:processname/fields e.g (myserver/jobrouter/ws/v1/process/Test/fields)
Params	None
Header	Token (Return from login service)
Response	JSON
Note	Token should be passed under http header for security purpose.

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test JobRouter web services.

```
// Initialize web service call object
$jwrWS = new JobRouterWebService("http://myserver/jobrouter/ws/v1/");
// processes web service params
$params = array();
// Token, fetched from login web service response
$token = „i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt“;
// Create process fields web service URL
$processname = „Test“;
$url = $jwrWS->getURL() . 'process/' . $processname . „/fields“;
// Return JSON response
$result = $jwrWS->sendCurlRequest($url, $params, 'GET', $token);
print_r($result);
```

Response (JSON):

Success	{ "success": { "userMessage": "", "code": 200, "processname": "Test", "fields": [{ "processname": "Test", "version": 1, ... }] }
Failure	{ "errors": { "userMessage": "Invalid token", "more_info": null, "code": 400, "-": ["Invalid token"] } }
Failure	{ "errors": { "userMessage": "Invalid process name", "more_info": null, "code": 400, "-": ["Invalid process name."] } }
Failure	{ "errors": { "userMessage": "User is not authorized.", "more_info": null, "code": 405, "-": ["User is not authorized."] } }

Process Incident Web Service:

This web service will start an incident of process that a logged in user can manage.

Feature	Description
Request Method	POST
URL	Server/jobrouter/ws/v1/process/:processname/incident e.g (myserver/jobrouter/ws/v1/process/Test/incident)
Params	Step *, summary, processtable, subtables
Header	Token (Return from login service)
Response	JSON
Note	Token should be passed under http header for security purpose. Please pass absolute file path to upload e.g (C:\doc\jobrouter\invoices\invoice1.pdf)

Sample PHP code snippet:

Note: Please make it sure that you are using class: **JobRouterWebService** to test jobrouter web services.

```
// Initialize web service call object
$jwrWS = new JobRouterWebService("http://myserver/jobrouter/ws/v1/");
// processes web service params
$params = array(
    'step' => 1,
    'processtable' => array(
        'Message' => 'Hello this is me!= Testing file uploading',
        'Test' => 'This is a Test',
        'Attachment1' => '@C:\doc\jobrouter\invoices\invoice1.pdf'
    ),
    'subtables' => array(
        'TESTSUBTABLE' => array(
            Array(
                'DocType' => 'Invoice In',
                'DocAttachment1' => '@C:\doc\jobrouter\invoices\Invoice2.pdf'
            ),
            Array(
                'DocType' => 'Invoice Out',
                'DocAttachment1' => '@C:\doc\jobrouter\invoices\invoice3.pdf'
            )
        )
    )
);
// Token, fetched from login web service response
$token = „i3hr5likkfjj2jiq9fd9jd6hsl5ukbbt“;
// Create process fields web service URL
$processname = „Test“;
$url = $jwrWS->getURL() . 'process/' . $processname . „/incident“;
// Return JSON response
$result = $jwrWS->sendCurlRequest($url, $params, 'POST', $token);
print_r($result);
```

Response (JSON):

Success	{"success":{"userMessage":"Request has been processed successfully","code":200,"task_id":"Test201606151465973770"}}
Failure	{"errors":{"userMessage":"Invalid token","more_info":null,"code":400,"-":["Invalid token"]}}
Failure	{"errors":{"userMessage":"Invalid process name","more_info":null,"code":400,"-":["Invalid process name."]}}
Failure	{"errors":{"userMessage":"Hey, you are not authorized to access this method","code":405}}
Failure	{"errors":{"userMessage":"File not found against field Attachment1","more_info":null,"code":400,"-":["File not found against field Attachment1"]}}