



New Hire Interconnection Testing Setup

The purpose of this document is to inform prospective agencies of the requirements and processes needed to begin testing the USAStaffing New Hire Interconnection. This document can be used in conjunction with the Authentication and Data Samples for New Hire Interconnection document.

Response Body

The response body should be in JSON (1) format and include the fields in table 1.

Property	Data Type	Description
success	Boolean	Indicates if the request succeeded.
error	string	Optional. In the event of an error a detailed message will be in this field

Figure 1 Response Body

Authentication

Authentication will be provided using JWT (2). The initial authentication will be by username/password credentials. This will result in a token that will be used for future requests. As of this version, there will only be one data request per successful authentication. Subsequent transmissions we will authenticate again.

The authentication request we will provide a username and password in JSON format as shown in figure 2. The user name should **not** be case-sensitive. The Content-Type will be application/json.

```
{  
  "username" : "testName",  
  "password" : "P@ssword!"  
}
```

Figure 2 Authentication Body

Upon successful authentication, the response should have an HTTP status code in of 200 and the body should include the JWT as shown in figure 3.

```
{
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bmlxdWVfbmFtZSI6InRlc3ROYW11IiwibmJmljoxNTA5MDM3MjU5LCJleHAiOiJE1MDkwNzMyNTksImIhdCI6MTUwOTAzNzI1OX0.zBv2nhMbhfEQEKcCDT54Uy7hhu3nxkTEba3Tle6Lu34"
}
```

Figure 3 JWT authentication token

If the authentication fails, the response should have an HTTP status code of 401.

Data Transmission

The data will be in JSON format and will have a field named **UniqueIdentifier** that uniquely identifies the New Hire in USAStaffing. This does not uniquely identify a person. Instead, it identifies the instance of the person for a given job selection.

Add New Record

New records may be added using either the POST or PUT method. A post is used to add a new record that must not exist. PUT is used when the record may exist, and will be replaced or added. See section 3.5.1 for more information.

Update Existing Record

Data will be updated using a PUT request. The PUT will include all the data fields. We will be using the PUT method if we believe the record has already been sent. If you do not have it, you may add the record (and return an HTTP status code of 201), or reject the record (and return an HTTP status code of 404). See section 3.5.2 for more information.

Delete Existing Record

Not supported with this version.

Get Single Record

Not supported with this version.

HTTP Methods (3)

The application will be of a restful (4) design and use the request verb, headers, body, and status code appropriately.

POST

The POST verb will be used for added a new record as described in section 3.1.

Return Codes

- 201 – The new record is created.
- 409 – The record already exists based on the unique identifier provided.
- 400 – The data failed validation.
- 401 – The authentication token is missing or invalid.
- 500 – Any server error.

PUT

The PUT verb will be used for added a new record as described in section 3.2.

- 200 – The record has been updated.
- 201 – The record did not exist, but it was added.
- 404 – The record does not exist and will not be added.
- 400 – The data provided failed validation.
- 401 – The authentication token is missing or invalid.
- 500 – Any server error.

Re-transmission

In the event of an HTTP response status of 500, we will attempt to re-send the record approximately once an hour for 120 hours (5 days). After this point it will require a manual operation to resend. For HTTP response status code of 400, 401, 404, or 409 we will not automatically resend the record and a manual operation will be required.

Works Cited

1. Introducing JSON. *json.org*. [Online] ecma, 2013. <http://www.json.org/>.
2. Jones, M, Bradley, J and Sakimura, N. JSON Web Token (JWT). *Internet Engineering Task Force (IETF)*. [Online] May 2015. <https://tools.ietf.org/html/rfc7519>.
3. w3. Method Definitions. *Hypertext Transfer Protocol*. [Online] 1999. <https://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html>.
4. Kearn, Martin. Introduction to REST and .net Web API. *MSDN*. [Online] Jan 5, 2015. <https://blogs.msdn.microsoft.com/martinkearn/2015/01/05/introduction-to-rest-and-net-web-api/>.