

Fuse Technical
Documentation

HRIS INTEGRATION WITH FUSE



fuse

TABLE OF CONTENTS

Fuse Technical Documentation	1
Table of contents	2
Overview	3
How to Provision Users on Fuse Platform	4
Automated CSV Process	6
Fuse REST API	8
Customer Support for HRIS Integttation	9
Important Information	10

OVERVIEW

Most Fuse customers will have an HRIS integration for automatic user provisioning. Typically this will form part of their Fuse Implementation project.

Benefits of HRIS Integration

Integrating your Fuse instance with your HRIS can remove much of the administration overhead associated with managing users at scale and provide all the typical user flows - Create, Update, Deactivate, Reactivate.

In addition, our Audience Management functionality provides powerful logic-based rules to dynamically assign users to their learning, content and roles on Fuse. The Fuse Audience Management uses If/Then/Else logic to reference user profile field data so can be driven dynamically by a HRIS integration.

For example, if a user changes role or location within the business and this is reflected within their Fuse user profile via the automated HRIS integration, Audience Management can automatically reassign them to new Fuse communities and learning plans. No manual community or learning plan assignment is required.

HOW TO PROVISION USERS ON FUSE PLATFORM

Users can be provisioned and managed by:

- The Manage Users utility (available to Administrators via the Admin Tools)
- Manual CSV upload via the Imports utility (available to Administrators via the Admin Tools)
- Custom Batch CSV process via the Fuse FTP server
- Fuse REST API (/manage/users endpoint)

Batch CSV processing and the Fuse REST API will be covered later in this document.

User Profile Fields

In addition to the typical profile fields such as Username, Email, First Name, Last Name, Fuse supports customer-defined profile fields. HRIS integration projects should always start by defining what HR data is available and what data needs to be pushed to Fuse to populate the user profile fields. Only authorised Fuse staff can create custom user profile fields. The defining of user profile fields is part of the standard Fuse implementation process. Updates to customer user profile fields can be requested via the Fuse Support ticketing system.

- LinkedIn Learning

User Profile Field Considerations

Customers are reminded that they are the Data Controller for their Fuse instance and share responsibility for their user data within Fuse. Please get approval from your Data Officer or CTO for the HR data being referenced by the integration.

Whilst Fuse can support numerous customer-defined profile fields, in the interest of simplicity, data protection and API overhead these should be kept to a minimum. Only data that is required for Fuse user management, reporting and analytics should be referenced.

Based on numerous customer HRIS integrations we have listed the common user profile fields below, based on standard business data and organization hierarchy:

Parameter	Description
Email	Must be unique on the Fuse instance. Must include a valid email format.
Username	Mandatory field. Must be unique on the Fuse instance (up to 32 characters). In addition to standard usernames this can also be an employee ID or HR code used as a key field to cross reference users across systems. This is often referenced as the NameID for SSO authentication also.
First Name	As described
Last Name	As described
Employee ID	Often specified to serve as a key field for lookups across customer systems.
Job Title	As described
Is Manager	Yes/No - useful identifying if a user is a manager so they can be directed to managerial learning/content/landing page.
Department	As described
Function	As described
Region	As described
City	As described
Start Date	Useful for defining onboarding audience management rules. Must be a valid date format such as ISO8601, dd/mm/yyyy, yyyy/mm/dd.
Start Date UA	Static date field for use with Fuse Universal Analytics.
Additional Custom Field 1	Typically mapped to customer organisational structure.
Additional Custom Field 2	Typically mapped to customer organisational structure.
Additional Custom Field 3	Typically mapped to customer organisational structure.
Additional Custom Field 4	Typically mapped to customer organisational structure.
Additional Custom Field 5	Typically mapped to customer organisational structure.
Additional Custom Field 6	Typically mapped to customer organisational structure.
Landing Page	Often used to predefine a custom landing page for a user based on their user persona.

Note: Special characters such as Æ, Ø, Å, Ü, Ö and Ä are not currently supported in user-names.

It should be stressed that there is little value in replicating complex organisational structures within Fuse as it just adds unnecessary complexity. Most customers can sort and manage their users within Fuse using a field structure similar to the one shown above.

AUTOMATED CSV PROCESS

Overview

Customers with legacy HR systems or without the resource to leverage our REST API's can provision users to Fuse via a daily CSV feed to our FTP server. Due to the customisable nature of Fuse this is a custom process for each customer. The automated CSV process is typically scoped and developed as part of the initial Fuse implementation process.

Note: Special characters such as Æ, Ø, Å, Ü, Ö and Ä are not currently supported in usernames.

Functionality

The automated CSV process is based on an agreed CSV template mapped to the user profile fields on their Fuse. Customers will need to define the necessary HRIS report to populate these CSV fields and automate the upload of the daily CSV's to the Fuse FTP. FTP credentials will be provided.

Due to the constraints of CSV's we typically perform the initial bulk upload of users for the customer rather than use the automated CSV process. Customers can then check their user data, community and learning plan assignments before going live.

The CSV process supports all the standard user provisioning flows -

- Create new users
- Update existing users
- Deactivate existing users
- Reactivate deactivated users
- Assign managers to users

The Fuse FTP server is part of our secured AWS environment. It supports FTPS, SFTP and PGP encryption of the CSV's.

The CSV should be UTF-8 format. The customer is responsible for data integrity of the CSV's. Practical CSV field character limit for the process is 60 characters.

Named customer contacts or mailboxes will receive daily process notifications listing successful creations/updates as well unsuccessful creations/updates and errors.

Automated CSV Process Implementation

The scope of the process will be defined in a Statement of Work beforehand. Once approved, it will typically take 8 to 10 working days to develop, test and deploy. Timescales are dependent on the customer providing the necessary test data and data validation. The process will be developed and tested on the customer's Staging instance first, before being deployed to their Production instance once approved. Once live, change requests and issues should be raised via the Fuse support ticketing system.

FUSE REST API

API Documentation		
manage : Operations about manages		
Show/Hide	List Operations	Expand Operations
DELETE	/manage/users	Delete users
GET	/manage/users	Users list
POST	/manage/users	Create users
PUT	/manage/users	Update users
DELETE	/manage/users/{id}	Delete user
GET	/manage/users/{id}	User info
PUT	/manage/users/{id}	Update user
GET	/manage/users/{id}/learning-plans	User's learning plans list
GET	/manage/users/{id}/learning-plans/{learning_plan_id}	User's detailed learning plan
PUT	/manage/users/{id}/restore	Restore a disabled user
GET	/manage/users/{user_id}/scorm-course-data/{course_id}	Scorm course data for user
PUT	/manage/users/restore	Restore disabled users

Overview

Fuse has a suite of REST APIs (JSON) that enable customers to manage their users, content and learning. For user provisioning and management customers can leverage the /manage/users endpoint:

https://<instance_name>.fuseuniversal.com/api/<version>/manage/users

The Fuse API Manage Users Endpoint is free to use for existing customers. Please see our SLA for more information and terms.

Our API uses the standard HTTP Request Methods/Operations for communication:

- **GET** to lookup/retrieve user/s details.
- **POST** to create new user(s).
- **PUT** to edit/update/reactivate existing user(s).
- **DELETE** for deactivating user(s).

Fuse API supports JSON (JavaScript Object Notation).

Fuse APIs are versioned. We urge customers to always use the latest version of our API.

CUSTOMER SUPPORT FOR HRIS INTEGRATIONS

Throughout a customer implementation or development project, Fuse will work closely with customer development teams. Once an integration project has been defined we will typically have regular project calls to ensure customer developers are up to speed with our APIs and work through any issues. We provide this support free of charge for new and existing customers. Existing customers can request support through the Fuse support ticketing system.

A customer Staging instance will be made available for developments. Once tested and approved the integration can be deployed to the customer Production instance.

API Wiki

API Documentation

Show documentation for version: v4.3 - latest | [v4.2](#) | [v4.1](#) | [v4.0](#) | [v3.1](#) | [v3.0](#) | [v2.0](#) | [v1.8](#) | [v1.7](#) | [v1.6](#) | [v1.5](#) | [v1.4](#) | [v1.3](#) | [v1.2](#) | [v1.1](#) | [v1](#)

Fuse API documentation

This is the Fuse API Documentation and contains all endpoints for use by the client.

Base URL: <https://acme.fuseuniversal.com/api/v4.3>

admin : Operations about admins

Show/Hide | List Operations | Expand Operations

GET	/admin/events	This endpoint allows community admins and site admins to retrieve details for events in all communities, including name, description, price, occurrence count, settings, and visibility state.
GET	/admin/events/{id}	This endpoint allows community admins and site admins to retrieve details for a specific event, including name, description, price, occurrence count, settings, and visibility state.
GET	/admin/events/{id}/occurrences	This endpoint allows community admins and site admins to retrieve details for all occurrences of a specific event.
GET	/admin/events/{id}/occurrences/{occurrence_id}	This endpoint allows community admins and site admins to retrieve details for specific occurrence in specific event.
GET	/admin/events/{id}/occurrences/{occurrence_id}/users	This endpoint allows community admins and site admins to retrieve users for event occurrence.
GET	/admin/events/{id}/occurrences/{occurrence_id}/users/{user_id}	This endpoint allows community admins and site admins to retrieve specific user for event occurrence.

Each Fuse instance contains a Swagger-based interactive API Wiki that lists all our API's and profile fields associated with that instance. This is a great resource for information on all the available Fuse APIs. This is available to Administrator level users in Admin Tools:

https://<instance_name>.fuseuniversal.com/site-admin/api_documentations

IMPORTANT INFORMATION

UserID

Every user account on Fuse will have a corresponding Fuse userID that is generated automatically by the system at the time of user creation. The userID will be unique and cannot be changed. The userID forms part of many of the /manage/user API requests. Customers can lookup a user's userID via any attribute (username, email etc).

Username

The Fuse username is a key field that can be used for authentication. It is a mandatory field that must be unique for each user (up to 32 characters). In addition to standard usernames this is often linked to an employee ID or HR code so it can be a key reference field across systems. This is often referenced as the NameID for SSO authentication.

Email Address

Email address is optional but if used must be unique for each user on an instance. Email address is a key field and is often used for authentication. Only valid email address syntax is accepted.

Password Controls

Customers can align their instance password controls to their corporate password policy. If setting passwords as part of a user creation the password must comply with the password control settings of the instance.

Case-Sensitive

Fuse field names including custom fields are case-sensitive. If a field does not update or you are getting an error response, check that the field name being sent within the request matches the field name on fuse exactly.

Settings

Most of the user configuration settings can be configured at the instance level so are generally not required to be set at the user level via API but they are available. For example:

```
"accessibility_enabled": false,  
"accessibility_theme": "theme_v1",  
"plain_text_emails": false,  
"notify_when_shared_with_me": true,  
"notify_when_shared_with_my_community": true,  
"notify_when_commented_my_content": true,  
"notify_when_commented_followed_content": true,  
"notify_when_liked_my_content": true,  
"send_email_for_all_enabled_notifications": true,  
"daily_digest_enabled": false,  
"weekly_digest_enabled": false,  
"chat_digest_enabled": false,  
"instant_emails": false,  
"send_instant_for_comment": false,  
"send_instant_for_share": false,  
"send_instant_for_like": false,  
"send_instant_for_revision": false,  
"digest_emails": "Off",  
"notification_block_in_digest": false
```

Language Support

Fuse supports over 45 languages (more being added continually). Whilst language can be set via API at user level most customers will want users to select their own language on first login. If set via API the user's language settings will be overwritten each time the API fires:

```
"default_locale": "en-GB",
```

/Manage/User API Examples

The following are generic common examples of /manage/users API. Please see your API Wiki for more detailed information. API authorisation via Auth0 JWT based token (Oauth).

Lookup user details via user ID

Method: GET

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users/1<userid>>

Lookup user details via Fuse user via username

Method: GET

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users&username=test-user>

Lookup user details via Fuse user via email

Method: GET

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users&username=test.user@acme.com>

Lookup user via any profile field name

Note profile names are case sensitive.

Method: GET

Resource URL:

https://<instance>.fuseuniversal.com/api/v4.3/manage/users?given_name=John

Create a user

Create a single user.

Method: POST

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users>

Request Body:

```
{
  "users": [
    {
      "username": "John.Smith",
      "password_confirmation": "hU42!z9-Hkslk",
      "password": "hU42!z9-Hkslk",
      "display_name": "John Smith",
      "public_name": null,
      "given_name": null,
      "family_name": null,
      "email": "john.smith@acme.com",
      "manager_ids": [453232],
      "profile": {
        "First Name": "John",
        "Last Name": "Smith",
        "Department": "Sales",
        "Job Role": "Sales Rep",
        "Office": "London"
      },
      "settings": {
        "default_locale": "en-UK"
      }
    }
  ]
}
```

Update user parameter

Method: PUT

Resource URL:

https://<instance_name>.fuseuniversal.com/api/v4.3/manage/users/<valid_user_id>

Request Body:

```
{  
  "username": "John.Smith",  
  "family_name": "Smith"  
}
```

Update user's Manager

Specify the userID of the manager.

Method: PUT

Resource URL:

https://<instance_name>.fuseuniversal.com/api/v4.3/manage/users/<valid_user_id>

Request Body:

```
{  
  "manager_ids": [120194],  
}
```

Deactivate a user

Method: DEL

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users/<userid>>

Deactivate a user and wipe personal information

Method: DEL

Resource URL:

https://<instance>.fuseuniversal.com/api/v4.3/manage/users/<userid>?delete_personal_data=true

Restore a user

Note you cannot restore a user if their personal information has been deleted.

Method: PUT

Resource URL:

<https://<instance>.fuseuniversal.com/api/v4.3/manage/users/<userid>/restore>