

# JobMount Job Board v1.4.

## Data Upload Guide

### Introduction:

This document provides a step-by-step guidance on configuring fields matching of Job Board and your CSV data file. System can save presets so you can re-use same configuration multiple times.

It is possible to upload following types of data:

1. Vacancies
2. Employers
3. Resumes
4. Candidates

### JobMount Data Structure

Please keep in mind following structure of data on JobMount Job Board:

1. Vacancies are linked to Employers and cannot exist separately. For example:  
Bank of America  
– Vacancy 1: Team lead  
– Vacancy 2: HR manager  
...  
IBM  
- Vacancy 1: Senior developer  
...
2. Resumes are linked to Candidates and cannot exist separately. For example:  
John Smith  
- Resume 1 for Team Lead positions  
- Resume 2 for Senior Developer positions  
...

So in order to upload Vacancies you need to upload Employers first. Also Candidates should be uploaded together with Resumes.

## Data upload preparation – Important:

1. Finalize configuration of data fields for (Vacancy Settings, etc).
2. Finalize dictionary listings (Dictionary Management).
3. Prepare your CSV data file in a format that matches job board data configuration as close as possible.
4. First line is considered to be header and ignored on CSV upload.
5. Last line should not be empty.
6. If checkboxes are to be uploaded then values in CSV should be *false* or empty for unticked and *true* for ticked checkbox.

## Creating upload preset

Before data upload the tool should be configured to match CSV columns with fields in the system and such configuration is stored in Upload presets. You can have multiple presets for vacancies, employers, etc.

To create and access upload templates navigate to *Tools - CSV upload management*:

The list of saved presets is shown and you can either edit them or create a new preset by clicking Add button:

#	Preset	Actions
<b>Employer</b>		
1.	<a href="#">Recruiter</a>	 
<b>Candidate</b>		
1.	<a href="#">Candidate</a>	 

Once you click Add button you will see general preset settings:

## Create Upload Preset

<b>Preset Name*</b>	Vacancy
<b>Entity*</b>	Vacancy <span>▼</span>
<b>Separated by*</b>	Comma <span>▼</span>
<b>Text Delimiter*</b>	" <span>▼</span>

- **“Preset name”** should be something you will recognize, for example “Vacancies”.
- **“Entity”** is the type of data you need to upload. Choose one of: Vacancy, Candidate, Employer, Resume.
- **“Separated by”** is the character that delimits columns of the CSV data file: Tab, Semicolon, Comma, Space or other.
- **“Text delimiter”** is needed for large amounts of text, for example when you have vacancy description stored in one column cell and it is delimited by apostrophes or by quotes.

Buttons **“Save & Return”** and **“Save & Proceed”** – first one will save data and return you to the presets list page while second one will save and take you to the matching page.

Once you save the general preset settings you will be able to add fields matching. There are two dropdowns: 1) select entity which will be filled by column #1 and 2) select the field to fill:

<b>Columns</b>	Candidate <span>▼</span>	First Name <span>▼</span>	<input type="button" value="Add"/>
----------------	--------------------------	---------------------------	------------------------------------

By clicking Add button you will be asked for column data, where the most important setting is the “Column Number”. You should check which column in CSV file is relevant for selected field:

## Create Column

<b>Entity</b>	Candidate
<b>Field</b>	First Name
<b>Field Name*</b>	First Name
<b>Column Number</b>	1
<b>Regex</b>	
<b>Default Value</b>	What to out into field if no data in CSV file for this field
<b>Validate Duplicate by This Field</b>	<input type="checkbox"/>

“Validate duplicate by this field” will check if there is already a candidate with such name present and will not import a candidate with duplicate info. If 2 or more of the fields are ticked for Duplicate validation, a record will be considered a duplicate only if ALL of the fields are duplicate. If at least one of the fields is different a new unique record will be added.

Tip: set this checkbox for Email field as it is to be unique and used as login.

Match all of the fields. Your resulting matching page would look like example below:

#	Field Name	CSV Col #	Field	Validate Duplicate	Actions
<b>Candidate</b>					
1.	<u>Email</u>	4	Email	Yes	
2.	<u>Password</u>	5	Password	No	
3.	<u>First Name</u>	6	First Name	No	
4.	<u>Middle Name</u>	7	Middle Name	No	
5.	<u>Last Name</u>	8	Last Name	No	
6.	<u>Home Address1</u>	9	Home Address1	No	
7.	<u>Home Address2</u>	10	Home Address2	No	
8.	<u>Country</u>	11	Country	No	
9.	<u>Location</u>	12	Location	No	
10.	<u>Or type other</u>	13	Or type other	No	
11.	<u>City/Town</u>	13	City/Town	No	
12.	<u>Post Code</u>	14	Post Code	No	
13.	<u>Phone Number</u>	15	Phone Number	No	
14.	<u>Mobile/Cell</u>	16	Mobile/Cell	No	
15.	<u>Subscribe for news</u>	17	Subscribe for news	No	
16.	<u>Status</u>		Status	No	
17.	<u>Date Created</u>		Date Created	No	

Once candidate fields are matched you can move on to create Resume matching. First of all create connection between a resume and a candidate by adding following field:

Columns

And leave all fields to be default:

<b>Entity</b>	Resume
<b>Field</b>	Candidate
<b>Field Name*</b>	Candidate
<b>Column Number</b>	
<b>Column Type</b>	Select Please... <input type="button" value="v"/>
<b>Default Value</b>	Select Please... <input type="button" value="v"/>
<b>Validate Duplicate by This Field</b>	<input type="checkbox"/>

All other resume fields should be matched to CSV columns as usual.

**Column type should be**

- string if they are listed as names in your CSV file, e.g. London
- ID is used if you have already prepared your CSV column to match dictionary IDs instead of names, e.g. 1568 (Id's aren't available for employer reference)
- Multiselect separator character should not be the same as column or text delimiters.

## Matching lists

Quite often there is a need to match items you have in JobMount Dictionary Management with the text data in the CSV file. I.e. Sectors or Locations. The problem is that usually there are various spellings of some entries like "New York" in Dictionary management and NYC in CSV file. Here is the solution:

Create new matching by selecting needed field:

Columns

Hit *Add* and fill out info as usual and set Column type = String:

Column Type

System will retrieve data from Dictionary Management. Next step: compile all possible City values from CSV file, click "Add multiple" button and paste cities into the new window for auto-matching:

### Dictionary Multiple Matching

<b>Items*</b>	NYC Auckland Manukau City
---------------	---------------------------------

System automatically matches all CSV text values to Dictionary ones. For those not matched: manually assign value from dropdown. Screenshot: choose matching value to replace all North Shore City records from CSV:

<u>North Shore City</u>	<input type="text"/>	<input type="checkbox"/>	<input type="text" value="Select Please..."/> <ul style="list-style-type: none"> <li>Select Please...</li> <li style="background-color: #0070c0; color: white;">North Shore</li> <li>Palmerston North</li> <li>Wellington City</li> </ul>
<u>Porirua</u>	<input type="text"/>	<input type="checkbox"/>	
<u>Pratapgarh</u>	<input type="text"/>	<input type="checkbox"/>	
<u>Raumati</u>	<input type="text"/>	<input type="checkbox"/>	

If CSV file has cities that aren't present in Dictionary you can either add that city to the Dictionary Management and return to configure match or leave it empty. Empty matching would mean no value assigned to the field.

## System fields to be filled

In order to make correct upload some additional system fields are to be filled out during CSV upload.

They should have **empty Column number** specified (unless it's present in the CSV).

### Candidates:

1. Status field with following settings:

Entity	Candidate
Field	Status
Field Name*	Status
Column Number	
Column Type	String <span>▼</span>
Default Value	Active <span>▼</span>
Validate Duplicate by This Field	<input type="checkbox"/>

2. Date created:

Entity	Candidate
Field	Date Created
Field Name*	Date Created
Column Number	
Date Format	
Default Value (CURRENT_DATE for current date)	CURRENT_DATE
Validate Duplicate by This Field	<input type="checkbox"/>

### Resumes:

1. Candidate reference:

Entity	Resume
Field	Candidate
Field Name*	Candidate
Column Number	
Column Type	Select Please... <span>▼</span>
Default Value	Select Please... <span>▼</span>
Validate Duplicate by This Field	<input type="checkbox"/>

2. Post date created:

Entity	Resume
Field	Post Date
Field Name*	Post Date
Column Number	
Date Format	
Default Value (CURRENT_DATE for current date)	CURRENT_DATE
Validate Duplicate by This Field	<input type="checkbox"/>

**Employers:**

1. Date created:

Entity	Employer
Field	Date Created
Field Name*	Date Created
Column Number	
Date Format	
Default Value (CURRENT_DATE for current date)	CURRENT_DATE
Validate Duplicate by This Field	<input type="checkbox"/>

**Vacancies:**

1. Employer reference:

Entity	Vacancy
Field	Employer
Field Name*	Employer
Column Number	
Column Type	Select Please...
Default Value	Select Please...
Validate Duplicate by This Field	<input type="checkbox"/>

2. Post date:

Entity	Vacancy
Field	Post Date
Field Name*	Post Date
Column Number	
Date Format	
Default Value (CURRENT_DATE for current date)	CURRENT_DATE
Validate Duplicate by This Field	<input type="checkbox"/>

## Matching done, let's upload

Once the upload preset is configured and checked you are ready to do upload. To do this go to "CSV Upload Management" in admin, select a preset and browse for CSV file:

### CSV Upload



Upload Presets\* Candidate

Upload File\*  Browse...

Upload

After clicking **Upload** you will see the progress and the final results:

Row 21 successfully processed

Row 22 successfully processed

**CSV file successfully uploaded!**

Undo

## Undo function

Verify data once it uploaded via Candidates or Employer Management to check if all fields are properly filled out.

If something went wrong: use **Undo function** to revert data but please note you will need to logout and login again for it to become available.