

# User Upload Agent

Automating onboarding

Johan Codinha

# What is the User Upload Agent?

## The Problem

- Manual processing of user import tickets
- Repetitive data validation tasks
- Excel/CSV file format variations
- Multi-tenant authentication complexity

# What is the User Upload Agent?

## The Solution

- **Automated ticket monitoring** in Jira
- **AI-powered file parsing** (multi-sheet Excel support)
- **Smart column mapping** for varied formats
- **Tenant-aware authentication** via 1Password
- **Structured approval workflows** with SHA-256 validation

✨ **From hours of manual work to minutes of automation**

**100+**

Users processed in <2 minutes

**30MB**

Max attachment size

**5**

File formats supported





# Ticket Lifecycle

## Open Status

- ✓ New tickets detected
- ✓ AI analyzes intent
- ✓ Files downloaded & parsed
- ✓ Data validated

### Transitions to:

- Review (approval needed)
- Info Required (no credentials)

## Review Status

- ⌚ Waiting for approval
- 📎 Proposed CSV attached
- 🔄 Column mappings shown
- 💬 Reply "approved" to proceed

### Transitions to:

- Done (upload successful)
- Info Required (failures)

## Info Required

- 🔒 Missing 1Password entry
- ✗ Upload failures occurred
- 📝 Setup instructions posted

### Message examples:

"User upload completed with 3 failures"  
"No 1Password entry found"

**Automatic Transitions:** Open→Review (needs approval) • Review→Done (success) • Any→Info Required (errors/missing creds)

# Smart File Processing

## Simple Mode (Direct)

Files with exact column matches:

- `email`
- `first name`
- `last name`
- `job title`
- `mobile number`
- `teams`
- `user role`

✓ Deterministic system

## Complex Mode (AI-Assisted)

Handles challenging scenarios:

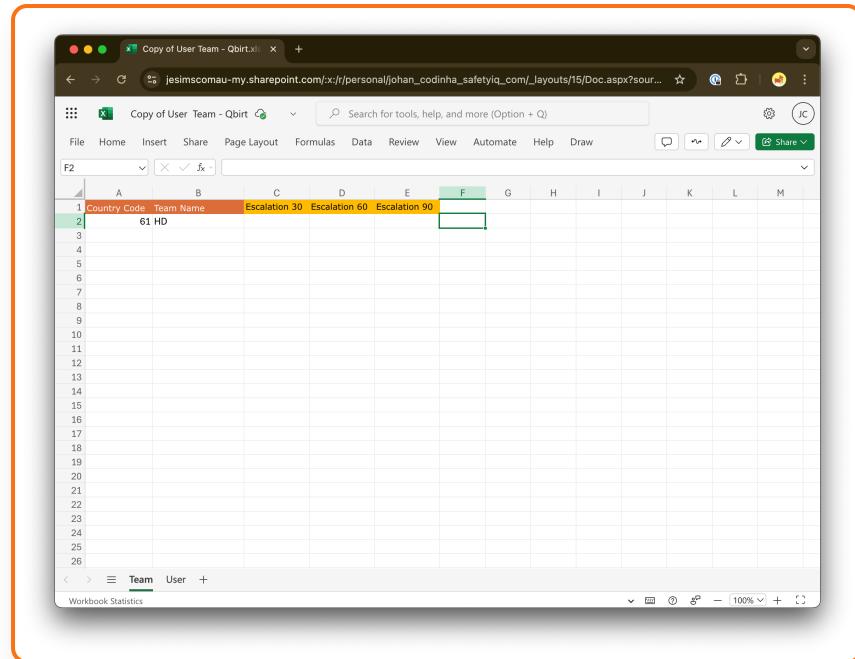
- Multi-sheet Excel files
  - Headers not in first row
  - Instruction rows before data
  - Non-standard column names
  - Sheet structure detection
-  [LLM driven data extraction](#)

# Real-World Excel Parsing Challenge

## Multi-Sheet Excel with Complex Structure

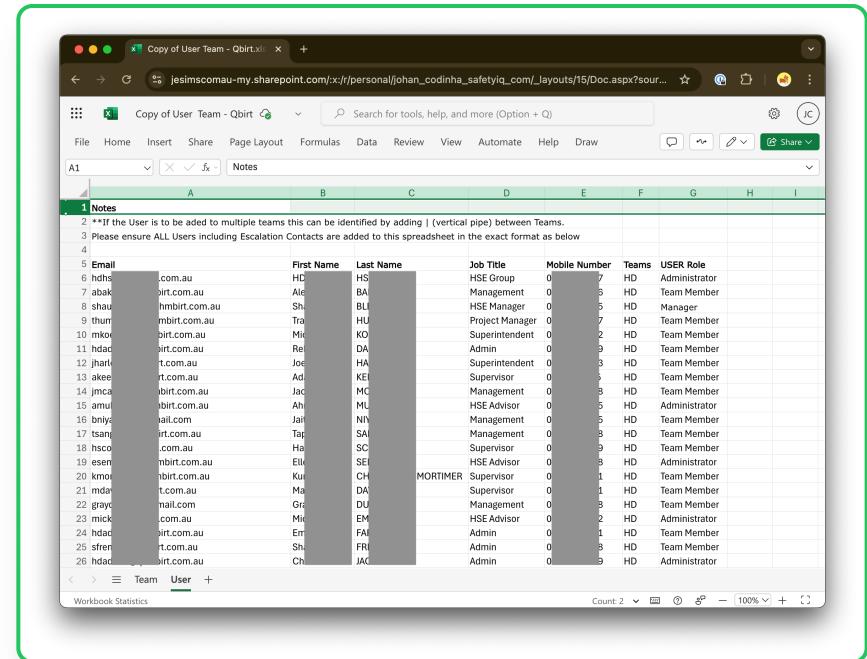
The AI agent navigates through multiple sheets to find and extract user data

### Sheet 1: Team Configuration



Country Code	Team Name	Escalation 30	Escalation 60	Escalation 90	
61	HD				
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					

### Sheet 2: User Data



Notes	A	B	C	D	E	F	G	H	I
1									
2									
3									
4									
5	Email	First Name	Last Name	Job Title	Mobile Number	Team	USER Role		
6	hdhs@qbit.com.au	HD	HS	HSE Group	0	7	HD	Administrator	
7	abak@qbit.com.au	Ale	BA	Management	0	6	HD	Team Member	
8	shau@qbit.com.au	Shau	BUL	HSE Manager	0	5	HD	Manager	
9	thum@qbit.com.au	Thum	HU	Project Manager	0	7	HD	Team Member	
10	mko@qbit.com.au	Mick	KO	Superintendent	0	2	HD	Team Member	
11	hdad@qbit.com.au	Rey	DA	Admin	0	9	HD	Team Member	
12	jhart@qbit.com.au	Joe	HA	Supervisor	0	3	HD	Team Member	
13	akee@qbit.com.au	Ad	KEI	Supervisor	0	3	HD	Team Member	
14	jmc@qbit.com.au	Jac	MC	Management	0	8	HD	Team Member	
15	amul@qbit.com.au	Ahi	MU	HSE Advisor	0	5	HD	Administrator	
16	bniy@qbit.com	Jai	NN	Management	0	5	HD	Team Member	
17	tsang@qbit.com.au	Tan	SAI	Management	0	8	HD	Team Member	
18	hsco@qbit.com.au	Han	SC	Supervisor	0	9	HD	Team Member	
19	esent@qbit.com.au	Elle	SEI	HSE Advisor	0	8	HD	Administrator	
20	kmo@qbit.com.au	Ku	CH	MORTIMER	Supervisor	0	1	HD	Team Member
21	mdar@qbit.com.au	Ma	DA	Supervisor	0	1	HD	Team Member	
22	grayd@qbit.com	Gray	DU	Management	0	8	HD	Team Member	
23	mick@qbit.com.au	Mick	EM	HSE Advisor	0	2	HD	Administrator	
24	hdad@qbit.com.au	Em	FAI	Admin	0	1	HD	Team Member	
25	stren@qbit.com.au	Shan	FRI	Admin	0	8	HD	Team Member	
26	hdad@qbit.com.au	Chi	JAC	Admin	0	9	HD	Administrator	

# Approval Request Comment

## **Structured Approval Request**

The agent posts a detailed comment with all processing information

# Approval Request Comment

## Structured Approval Request

The agent posts a detailed comment with all processing information



**Full Transparency**  
See exactly what was processed



**CSV Attachment**  
Review the clean data



**Simple Approval**  
Just reply "approved"

# Human-in-the-Loop Approval

## You Stay in Control

After AI processing, the agent creates a clean CSV proposal for your review

The screenshot shows a Jira work item titled '(Onboarding) QBirt: Upload User'. The work item is in the 'General' tab. The 'Description' section contains a note about HubSpot linked tickets and a placeholder for user uploads. The 'Attachments' section shows two files: 'users-for-approval.csv' (14 KB, added 14 Aug 2025 11:03am) and 'Copy of User Team - Qbirt.xlsx' (24 KB, added 08 Aug 2025 4:04pm). A red arrow points to the 'Review' button in the top right corner of the Jira interface. Another red arrow points to the 'users-for-approval.csv' attachment.

Projects / JESI / Add parent / JESI-7754

### (Onboarding) QBirt: Upload User

General User Story Details Release Categorize HubSpot

**Description**

HubSpot linked tickets:

- (Onboarding) QBirt: User Upload (Ticket ID: 27705868682)

(please, do not edit or duplicate in description)

Please upload the users on priority  
ETA: 11th Aug'25

User ID: customersolutions+qbirt@jesi.io

Customer Commitment

None

**Attachments** 2

Name	Size	Date added	Actions
users-for-approval.csv	14 KB	14 Aug 2025 11:03am	
Copy of User Team - Qbirt.xlsx	24 KB	08 Aug 2025 4:04pm	

Review Improve work item

**Details**

Assignee: Johan.Codinha

Reporter: Melissa Branscome

Story Points: None

Development

Open with VS Code

Create branch

Create commit

Fix versions: None

Pairing Partners: None

Priority Flag: None

## What Happened

-  Original Excel file processed (Complex multi-sheet)
-  AI mapped columns and extracted data
-  Clean CSV created: `users-for-approval.csv`
-  Attached to ticket for review
-  Status changed to "Review"

## Next Steps

-  Review the proposed CSV file
-  Verify data looks correct
-  Reply with "**approved**" to proceed
-  Agent will then create all users
-  Final report posted when complete



### Safety First

No users are created until you explicitly approve the proposed data

# Jira Comment Integration

## Automated Status Updates

The agent posts detailed comments at every stage

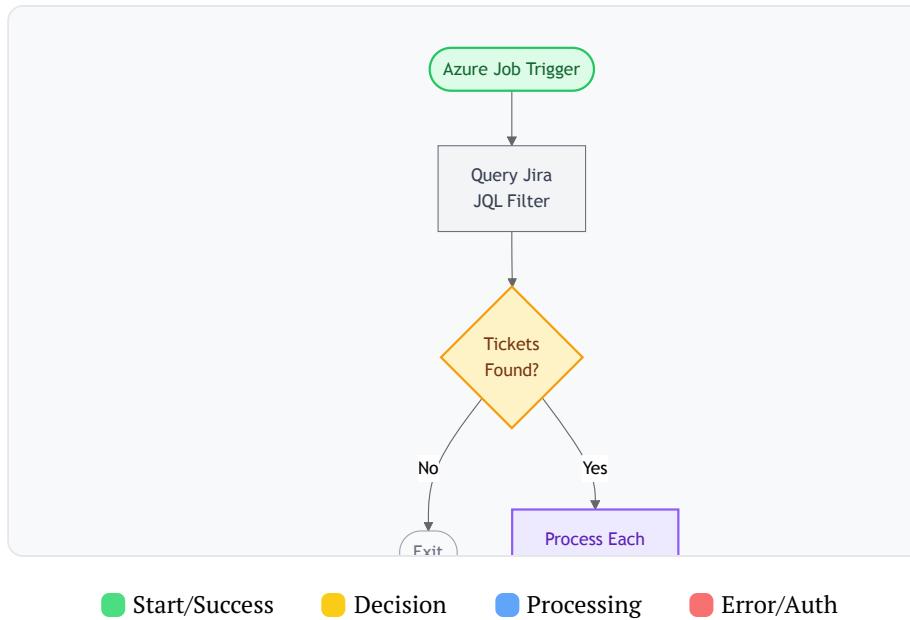
 **Successful Upload**

 **Missing Credentials**

# Technical Architecture & Implementation

Details for nerds

# Processing Workflow



# Intent Detection prompt

You are analyzing a Jira ticket to determine if it represents a user upload request.

A user upload request is a ticket asking to add, import, or upload user data (employees, team members, etc.) to a system.

Analyze the ticket information provided and determine if this is a user upload request.

Consider:

- Ticket summary and description content
- Attachment file names and types
- Keywords like "upload", "import", "add users", "new employees", "bulk users", etc.
- File extensions like .csv, .xlsx, .xls that commonly contain user data

Return ONLY JSON in this exact format with no additional text:

```
{"is_user_upload": <true/false>}
```

Examples:

Ticket: "Upload new team members" with attachment "employees.xlsx"

```
Response: {"is_user_upload": true}
```

Ticket: "Login page broken" with no attachments

```
Response: {"is_user_upload": false}
```

# Column Mapping Prompt

You are mapping file column headers to a standardized user schema.

You will be given:

1. A list of expected standard field names
2. A list of actual column headers from a file

Your task is to match each file header to the most appropriate expected field, or identify which expected fields have no match.

Common mappings:

- "Email Address", "E-mail", "Mail" → "email"
- "FirstName", "First Name", "Given Name" → "first name"
- "LastName", "Surname", "Family Name" → "last name"
- "Position", "Role", "Title" → "job title"
- "Phone", "Mobile", "Cell" → "mobile number"
- "Team", "Department", "Group" → "teams"
- "Access Role", "Permission Level" → "user role"

Be flexible with variations in:

- Case (upper/lower)
- Spacing vs underscores vs hyphens
- Abbreviations and synonyms
- Extra words ("User Email" → "email")

# Sheet Detection Prompt

You are analyzing an Excel file to find user data for upload to a system.

Your task is to identify:

1. Which sheet contains the user data
2. Which row contains the column headers
3. Which row the actual data starts

Expected user data columns include:

- Email addresses
- First and last names
- Job titles or roles
- Phone/mobile numbers
- Team assignments
- User permission roles (like "Administrator", "Team Member", etc.)

The file may have:

- Multiple sheets (some may be configuration or notes)
- Instruction rows before the actual headers
- Empty rows between sections
- Merged cells or formatting that creates empty columns

Analyze the provided Excel structure and return ONLY a JSON response in this exact format:

# Error Summary Prompt

You are summarizing user upload validation errors for a Jira ticket comment.

You will be given a list of validation errors that occurred while processing user data.

Create a concise summary suitable for posting as a Jira comment that:

- Gives an overall summary of what went wrong
- Lists the main error categories with bullet points
- Is professional and actionable
- Helps the requester understand what needs to be fixed

Return ONLY JSON in this exact format with no additional text:

```
{  
  "summary": "<brief overall summary>",  
  "bullet_points": [  
    "<error category 1>",  
    "<error category 2>",  
    "<specific details if needed>"  
  ]  
}
```

Example:

# Excel Sheet Detection

You are analyzing an Excel file to find user data for upload to a system.

Your task is to identify:

1. Which sheet contains the user data
2. Which row contains the column headers
3. Which row the actual data starts

Expected user data columns include:

- Email addresses
- First and last names
- Job titles or roles
- Phone/mobile numbers
- Team assignments
- User permission roles (like "Administrator", "Team Member", etc.)

The file may have:

- Multiple sheets (some may be configuration or notes)
- Instruction rows before the actual headers
- Empty rows between sections
- Merged cells or formatting that creates empty columns

Analyze the provided Excel structure and return ONLY a JSON response in this exact format:

# Agent Integration & Future Ideas

## Extending the Automation Platform

# Current Agent Capabilities

## What It Does Today

- Monitors Jira tickets continuously
- Processes CSV/Excel files
- Validates data comprehensively
- Manages approval workflows
- Creates users and teams
- Reports results automatically

## Key Benefits

- **Speed:** Hours → Minutes
- **Accuracy:** AI-powered validation
- **Security:** Tenant isolation
- **Visibility:** Complete audit trail
- **Reliability:** Idempotent operations
- **Flexibility:** Handles varied formats

**Processing 100+ users across multiple teams in under 2 minutes**

# Future Possibilities

- **Permission updates** - Bulk role changes
- **Team restructuring** - Move users between teams
- **Deactivation requests** - Offboarding automation
- **Access reviews** - Periodic audits
- **Report request** - Custom report generation
- **Risk assessment** - Setup - customized - visualizations - query.

Thank You