

Contact Import Modal: Quick Guide (CSV, TSV, Excel)

Contact Import Modal: Quick Guide (CSV, TSV, Excel)

What this does

- Quickly add many contacts from a file into BuilderPal.
 - Uses headers-only AI mapping: only your column names are analyzed to auto-map fields. Your actual contact data stays local by default.
 - This modal does not support QuickBooks IIF. Use CSV/TSV/TAB or Excel.
-

Before you start

- Make sure your file's first non-empty row is a header row (e.g., `First Name`, `Last Name`, `Email`).
- Include at least `First Name` and `Last Name`. `Email` or `Phone` is recommended.

Supported file types

- CSV (`.csv`)
- TSV/TAB (`.tsv`, `.tab`)
- Delimited text (`.txt`) using comma, tab, semicolon, or pipe
- Excel (`.xlsx`, `.xls`)

Note: IIF is not supported for contacts. Use the Team Member Import for IIF (Employee) files.

How to import (Step?by?Step)

1. Open Contacts → Import
 - Go to the Contacts area and click Import.
2. Upload your file
 - Choose a `.csv`, `.tsv`, `.txt`, `.xlsx`, or `.xls` file.
3. Review auto-mapping (if shown)

- If a mapping screen appears, confirm the suggested column matches (e.g., `First Name` → `first_name`, `Email` → `emails`). Adjust if needed.

4. Review the preview grid

- Each row shows a contact with fields such as First, Last, Emails, Phones, Company, Title, and Notes.
- Rows are pre-selected to import. Uncheck any you don't want to save.

5. Edit or add contacts

- Use Edit to fix a row inline (names, emails, phones, etc.).
- You can also add a new contact manually if needed.

6. Save

- Click Save Contacts. If conflicts occur (for example, `email` or `phone` already in use), you'll see inline warnings. Fix and save again.

What gets imported

The grid uses these fields per contact:

- `first_name` (string)
- `last_name` (string)
- `nickname` (string, optional)
- `emails` (array of `{ type: string, email: string }`)
- `phones` (array of `{ type: string, phone: string, country_code: string | null }`)
- `company_name` (string, optional)
- `company_position` (string, optional)
- `notes` (string, optional)

Notes:

- Multiple emails/phones are supported via the `emails` and `phones` arrays. If your file has only one, it becomes the first element (type is inferred from the header when possible, e.g., `work`, `mobile`).
- Phone digits may be normalized during processing.

Common header names that auto?map

- Names: `First Name`, `Firstname`, `Given Name` → `first_name`
- Names: `Last Name`, `Surname` → `last_name`
- Nickname: `Nickname`, `Preferred Name` → `nickname`

- Emails: Email, Email Address, Work Email, Personal Email → emails
 - Phones: Phone, Mobile, Cell, Work Phone, Home Phone → phones
 - Company: Company, Organization → company_name
 - Title: Title, Position, Role → company_position
 - Notes: Notes, Comments → notes
-

Examples

Example CSV

```
First Name,Last Name,Email,Phone,Company,Title,Notes
Gabe,Dahl,gabe@example.com,(604) 613-2792,Gabe Renovations,Owner,Prefers text
Jane,Doe,jane@xyz.com,604-555-0199,XYZ Inc,Marketing Director,
```

How those rows appear to the importer (simplified)

```
[
  {
    "first_name": "Gabe",
    "last_name": "Dahl",
    "emails": [ { "type": "work", "email": "gabe@example.com" } ],
    "phones": [ { "type": "work", "phone": "(604) 613-2792", "country_code": null } ],
    "company_name": "Gabe Renovations",
    "company_position": "Owner",
    "nickname": null,
    "notes": "Prefers text"
  },
  {
    "first_name": "Jane",
    "last_name": "Doe",
    "emails": [ { "type": "work", "email": "jane@xyz.com" } ],
    "phones": [ { "type": "work", "phone": "604-555-0199", "country_code": null } ],
    "company_name": "XYZ Inc",
    "company_position": "Marketing Director",
    "nickname": null,
    "notes": ""
  }
]
```

Using the chat in the modal (optional)

- You can type instructions (e.g., “Please load this file and map headers”) and attach your CSV/TSV/Excel.
- The assistant only analyzes your header row to infer mapping by default; the actual contact values are processed locally.
- You can also paste small lists of contacts in plain text and let the assistant structure them for you.

Tip: The modal includes quick actions like “Fill Example Text” or “Load Example” to demo the format.

Conflicts and error messages

On Save, you may see conflicts per row. The modal highlights the field and shows a message. Typical keys:

- `email` — email already in use or invalid
- `phone` — phone already in use or invalid
- `contact` — a general contact-level conflict (e.g., duplicate contact)

Unselect or edit conflicted rows and save again.

Troubleshooting

- “Could not detect a consistent delimiter.”
 - Re-save as CSV (comma) or TSV (tab). Ensure the first non-empty row contains headers.
 - “Header row appears empty or malformed.”
 - Confirm the first non-empty row is the header row and has at least two columns.
 - Email or phone didn’t map.
 - Make sure headers are recognizable (e.g., `Email`, `Phone`, `Mobile`). Adjust mapping if prompted.
 - Multiple email or phone columns.
 - The importer will populate the `emails`/`phones` arrays. The first recognizable header becomes the first entry; additional matched headers may be added with inferred `type`.
-

Privacy at a glance

- Headers-only AI mapping by default: we send only your column labels to AI to figure out mapping.
 - Your actual contact values (names, emails, phone numbers) are handled locally in BuilderPal by default.
-

Quick checklist

- File type is supported (`.csv`, `.tsv`, `.txt`, `.xlsx`, `.xls`).
 - First non-empty row has clear headers (`First Name`, `Last Name`, `Email`, etc.).
 - Columns include at least `First Name` and `Last Name`; ideally `Email` and/or `Phone`.
 - Review the preview grid, make edits if needed, then click Save Contacts.
 - Remember: IIF files are not supported in this modal.
-

Revision #1

Created 27 October 2025 21:49:49 by Admin

Updated 17 November 2025 16:52:29 by Admin