

Connecting an Iterable Pipeline

Last Modified on 07/16/2026 5:55 pm EDT

This article walks through connecting Digioh to Iterable and building your first Pipeline: one that sends campaign Submissions into Iterable as a list subscription, profile update, or tracked event. If you haven't read [Sending your Data: A Pipelines Overview](#) yet, start there for the concepts behind Pipelines.

If you've already connected Iterable in your Digioh account, you can jump directly to creating a pipeline.

Get a Server-Side API key from Iterable

1. In your Iterable project, go to **Integrations → API Keys**.
2. Click **New API Key**.
3. Choose **Server-Side** as the key type and give it a clear, recognizable name (for example, "Digioh").
4. Click **Create API Key**.



Copy the key immediately. Once you close the window in Iterable, you can't view the full key again. You'd need to generate a new one.

Connect Digioh to Iterable

1. In Digioh, hover over **Integrations** in the top navigation and select **Integrations**.
2. Scroll to **All Integrations**, search for Iterable, and select the Iterable card.
3. You'll be redirected to the Iterable connections page. Click **Add New Connection**.
4. Name the connection and paste in the Server-Side API key from Iterable. Treat this key like a password. Don't share or expose it.
5. Select the identifier your Iterable project uses: email, userId, or hybrid.
6. Click **Save**.



By default, Digioh restricts which Iterable profile fields it can retrieve to email, userId, and the IDs of subscribed lists, message types, and channels. If you need more fields available for targeting or personalization, you can add them in the connection's Include/Exclude Profile Fields settings.

Create a Pipeline: List, Profile, or Event

An Iterable Pipeline can do one or more of the following: subscribe a user to a static list, add or update their Iterable profile, update their subscriptions, or trigger a custom tracking event.

1. Go to your **Integrations** page and open the Iterable integration you just connected.
2. Scroll to the Iterable Pipeline Templates and choose the template that matches your goal: Subscribe to List, Profile Update, or Track Event.
3. Name your Pipeline.
4. Select the Iterable connection you set up above.
5. If you're using the Track Event template, enter the event name you want sent to Iterable (for example, newUserSignup or Quiz Taken).
6. Select the Campaign that should use this Pipeline.



The "event triggered" pattern is a popular way to kick off an Iterable Journey, for example by firing a Quiz Taken event instead of just adding someone to a list. All of the Submission's data is included in the event payload, so you can reference it in the Journey's email content, not just use it as a trigger condition.

Map your Fields and Check Every Mapping Before you Publish

1. Open your Pipeline and edit the **Map Profile Fields** step (and, for a Track Event pipeline, the **Map Event Data** step as well).
2. Map the **top-level email** (or userId) field first. This is separate from anything you put inside dataFields, and it's how Iterable identifies the contact. Use form.email, or the correct custom field if your form captures email differently.
3. For each additional field, set the **Input Field** using dot notation (form.first_name, form.custom_5 for a numbered custom field, or form.named_custom_fields.[Question Name] for quiz submissions) and the **Output Field** using Iterable's syntax: custom fields live under dataFields.your_field_name.
4. Apply a **Transform** wherever the Iterable field isn't a plain string. Iterable is strict about types. A date field needs a real date/timestamp, and a boolean field needs the BOOL transform, since form.opt_in otherwise arrives as the plain string "true" rather than a real boolean.
5. **Save** each step, then go back to the Pipeline and **Publish** your changes.

Tips to Keep in Mind:

- A blank top-level email is the single most common Iterable failure. If email only lives inside dataFields and the top-level identifier is empty, Iterable rejects every Submission with "successCount": 0. Confirm the top-level email mapping before you publish, every time.

- Fields that seem to be "missing" in Iterable are almost always a mapping problem, not a bug. Either the field isn't mapped, it's mapped to the wrong output name, or Remove Unmapped Fields stripped it. Double-check the mapping in Digioh before assuming something's broken on Iterable's side.
- If your Iterable project is userId based, you will need to provide a userId mapping, Iterable does not automatically generate userIds – You can use analytics.submission_guid for a unique value.

Once you're done, submit a live test and check **Pipeline › Activity** to confirm the request and response look right.

What's next

That's it. Submissions are flowing into Iterable. Ready to see how your Campaign is performing? Head to [Measuring your results](#). Need a hand? Reach out to support@digioh.com.
