

# Create Shopify coupons for users going through an Iterable Journey

Last Modified on 07/01/2026 1:14 pm EDT



Using Digioh pipelines in an Iterable Journey depends on your current contract. Contact your account manager or Digioh Support for more information.

With Digioh you can call a Digioh Connect Pipeline from an Iterable Journey that will create new, unique Shopify coupons for each profile. The Pipeline will save the coupon code back to the Iterable profile, and send an event indicating a new coupon has been created.

## Step 1 - Create the necessary Digioh integrations

The first step is to ensure that you have the necessary Shopify and Iterable integrations in you Digioh account. You can follow these documents to create those:

- [How to install the Digioh Shopify app](#)
- [Connecting your Digioh account to Iterable](#)

## Step 2 - Create the Digioh Pipeline

Once the integrations are all set, you can then create the Digioh pipeline: Before moving forward with the integration with Iterable, You have to integrate with Shopify.

- Click on new integration
- Select Shopify Integration
- In the authentication button, Paste the coupon code of the Shopify access token that we created previously when creating automated coupons

1. Go to the Integrations tab

2. Click the 'Shopify Connections' link on the Shopify integration you added in Step 1:

Shopify US Production	Shopify	Shopify Connections   Delete	⌵
Shopify US Production	Shopify	Shopify Connections   Delete	⌵

Note: The next page may take some time to load

3. Select the Create Shopify Coupon, Save to Iterable, and Send Event To Iterable template:

## Available Shopify Pipeline Templates

### Create Shopify Coupon and Save To Iterable

This pipeline will retrieve an iterable user identified by the passed in email. If the user does not exist, then it is created. If the coupon code is not present it will create a new one using the configured Shopify Discount and save it to the Iterable contact

Create Pipeline

### Create Shopify Coupon, Save to Iterable, and Send Event To Iterable

This pipeline will retrieve an iterable user identified by the passed in email. If the user does not exist, then it is created. If the coupon code is not present it will create a new one using the configured Shopify Discount. The coupon code is saved to the iterable user, then a "digioh\_coupon\_created" event is sent to Iterable

Create Pipeline

4. Provide a name for the pipeline, select the appropriate Price Rule ID, and enter a Coupon Field name (Note: For the Coupon Field, no spaces can be used). If needed, you can also add a prefix and configure the Pipeline for a specific campaign:

### Create Pipeline From Template

This pipeline will retrieve an iterable user identified by the passed in email. If the user does not exist, then it is created. If the coupon code is not present it will create a new one using the configured Shopify Discount. The coupon code is saved to the iterable user, then a "digioh\_coupon\_created" event is sent to Iterable

**Template Name:**  
Create Shopify Coupon, Save to Iterable, and Send Event To Iterable

**New Pipeline Name:**

**Price Rule ID (Shopify Discount):**

**Coupon Field:**

**Coupon Prefix (optional):**

**Box (Optional):**

5. Hit 'Create.'
6. Once the Pipeline has been created, you can access it via the 'Edit' button under your current Shopify Pipelines:

Current Shopify Pipelines	
Test Digioh Pipeline	<a href="#">Edit</a>
Create New Coupon Code And Save to Iterable User	<a href="#">Edit</a>
Create Welcome Coupon	<a href="#">Edit</a>

7. Edit the Shopify step:

[Connectors Home](#) **(40488) Create Welcome Coupon**

[Show Details](#)

62426 Find Email in Payload Type: Map Data [+ Condition](#) [Edit](#) [Trash](#)  
Data Action: Merge

62427 Get Iterable User Type: Iterable API [Edit Conditions](#) [Edit](#) [Trash](#)  
Continue On Failure: Yes

62428 Create New Discount Code Type: Shopify [Edit Conditions](#) [Edit](#) [Trash](#)

62429 Create Iterable Update Payload Type: Map Data [+ Condition](#) [Edit](#) [Trash](#)

62430 Update Iterable User Type: Iterable API [+ Condition](#) [Edit](#) [Trash](#)

62431 Create Event Payload Type: Map Data [+ Condition](#) [Edit](#) [Trash](#)

In the configuration screen, select the Shopify connection and confirm that the Price Rule ID is correct. If needed, you can also add a coupon prefix and set the max length:

### Shopify (Create New Discount Code)

API Operation: Create Discount Code

Connection: Shopify Staging (Shopify)

Price Rule Id:

Discount Code

Prefix: (Optional)

Discount Code

Length: (Default 12, Max 100)

Discount Code Alpha Numeric

Type:

Path to Discount Item Data

Code: (Optional)

Save

Once you're all set, hit 'Save.'

- To connect the Pipeline to Iterable, select 'Edit' on each of the Iterable API steps: You can then select which Iterable integration the pipeline should use:

[Connectors Home](#)      **(40488) Create Welcome Coupon**

[Show Details](#)

▲ 62426 Find Email in Payload    Type: Map Data

+ Condition
Edit
🗑

Data Action: Merge

▲ 62427 Get Iterable User    Type: Iterable API

Edit Condition
Edit
🗑

Continue On Failure: Yes

▲ 62428 Create New Discount Code    Type: Shopify

Edit Conditions
Edit
🗑

▲ 62429 Create Iterable Update Payload    Type: Map Data

+ Condition
Edit
🗑

▲ 62430 Update Iterable User    Type: Iterable API

+ Condition
Edit
🗑

▲ 62431 Create Event Payload    Type: Map Data

+ Condition
Edit
🗑

▲ 62432 Send Coupon Created Event    Type: Iterable API

+ Condition
Edit
🗑

Iterable API (Get Iterable User)

API Operation: Get User

Connection: Iterable - Welcome Coupon Code Integration

Path to Email: Find Email in Payload

email

Merge Nested Objects

Save

9. Back in the Pipeline, expand the details (located just under the name at the top) and expand the Hook URLs. Copy down the Iterable URL, you'll need this for the next step:

(40488) Create Welcome Coupon

Hide Details

Pipeline ID: 40488

Name: Create Welcome Coupon

Notification: (comma separated)

Emails:

Check Data  (Avoids Duplicate Processing)

Hashes:

Hook URLs: Hide Details

JSON:

Salesforce:

Iterable:

10. Finally, you'll want to run a test of the pipeline to ensure that the necessary attribute and event are created in Iterable. At the bottom of the Pipeline screen, select 'Test':

62509 Send Coupon Created Event Type: Iterable API

Condition Edit

Publish Test Activity + Task

In the Test Data (JSON) enter: **`{ "data": { "email": "test@test.com" } }`** and then run the test. You should see a log showing each of the steps in the pipeline being run. As long as everything worked correctly, you should see a series of Successful steps:

Creating the Shopify coupon code

```

Task: 62505 Create New Discount Code
Status: Ran, Success
Duration0ms
Message: Successfully completed Shopify API Call pipelineTaskId: 62505
Output:
copy
{
  "success": true,
  "statusCode": "Created",
  "discount_code": {
    "id": "24627557269846",
    "price_rule_id": "3586445287766",
    "code": "PROVB3ER130FGVV",
    "usage_count": 0,
    "created_at": "2023-09-28T19:55:34+01:00",
    "updated_at": "2023-09-28T19:55:34+01:00"
  }
}

```

Updating the Iterable User

```

Task: 62506 Create Iterable Update Payload
Status: Ran, Success
Duration0ms
Message:
Output:
copy
{
  "email": "test@test.com",
  "dataFieldIds": {
    "digioh_coupon": "PROVB3ER130FGVV"
  }
}

```

```

Task: 62507 Update Iterable User
Status: Ran, Success
Duration0ms
Message: Successfully completed Iterable api pipelineTaskId: 62507
Output:
copy
{
  "msg": "User successfully created/updated.",
  "code": "Success",
  "params": null
}

```

Sending the Coupon Created Event

```

Task: 62508 Create Event Payload
Status: Ran, Success
Duration0ms
Message:
Output:
copy
{
  "eventName": "digioh_coupon_created",
  "email": "test@test.com",
  "dataFieldIds": {
    "digioh_coupon": "PROVB3ER130FGVV",
    "couponField": "digioh_coupon"
  }
}

```

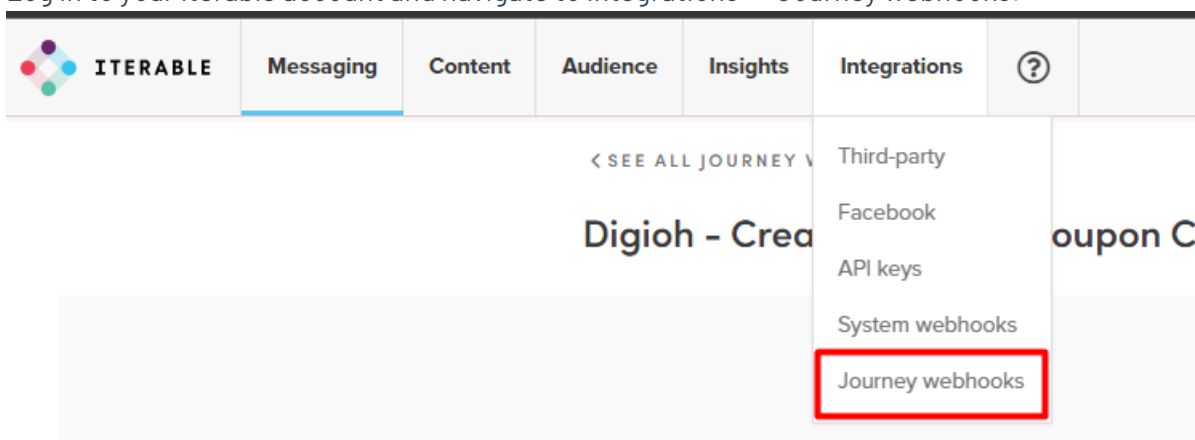
```

Task: 62509 Send Coupon Created Event
Status: Ran, Success
Duration0ms
Message: Successfully completed Iterable api pipelineTaskId: 62509
Output:
copy
{
  "msg": "Event with id: 71bc6b090e244e36b6f2d5a3a7eba057 tracked.",
  "code": "Success",
  "params": {
    "id": "71bc6b090e244e36b6f2d5a3a7eba057"
  }
}

```

### Step 3 - Create the Iterable Journey Webhook

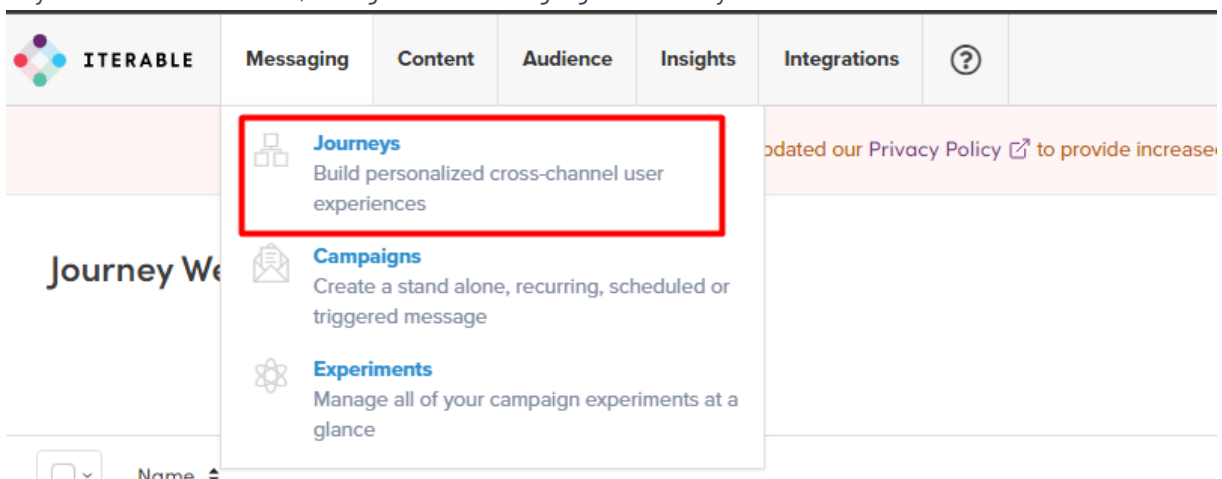
1. Log in to your Iterable account and navigate to Integrations -> Journey webhooks:



2. Add a new webhook.
3. Name your webhook as needed. The method should be POST. For the Endpoint, paste in the Hook URL from your Digioh Pipeline.
4. Hit 'Create webhook'

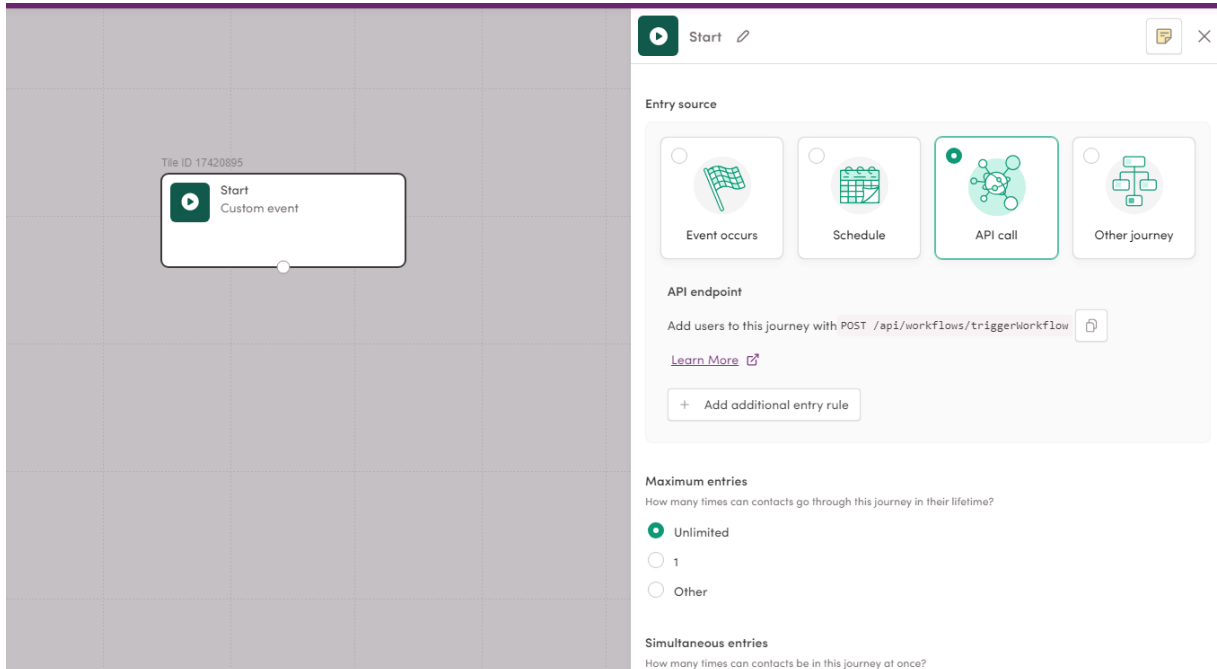
### Step 4 - Create the Iterable Journey

1. In your Iterable account, navigate to Messaging -> Journeys:



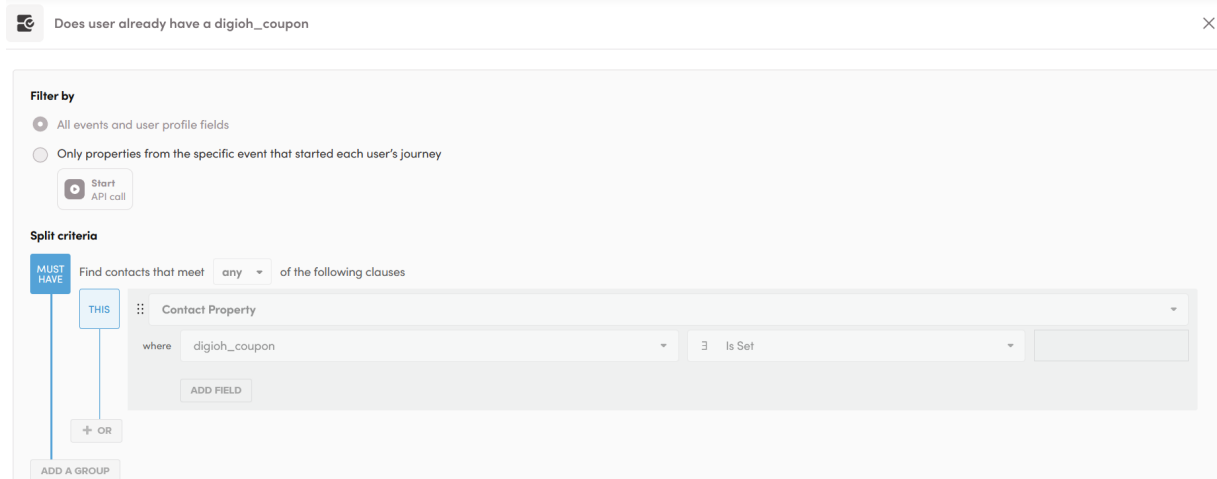
2. Create your new Journey as needed. For this example, our Journey will: Start with an API call. it will then check if the user already has a coupon on their profile or not: If they already do, it will send them an email; if they don't have a coupon, it will then call the Digioh Pipeline to generate one in Shopify.

3. Add the Iterable API call start tile:



The screenshot shows the Iterable Journey Builder interface. On the left, a grid contains a 'Start' tile with ID 17420895, labeled 'Start Custom event'. On the right, the configuration panel for the 'Start' tile is visible. It includes an 'Entry source' section with four options: 'Event occurs', 'Schedule', 'API call' (selected), and 'Other journey'. Below this is the 'API endpoint' section, which contains the text 'Add users to this journey with POST /api/workflows/triggerWorkflow' and a 'Learn More' link. There is also a button to 'Add additional entry rule'. The 'Maximum entries' section has three options: 'Unlimited' (selected), '1', and 'Other'. The 'Simultaneous entries' section has a question 'How many times can contacts be in this journey at once?'.

4. Add a Yes/no Split conditioned to check if the digioh\_coupon has already been set for the user:



The screenshot shows the 'Split criteria' configuration interface. At the top, it says 'Does user already have a digioh\_coupon'. Under 'Filter by', there are two radio buttons: 'All events and user profile fields' (selected) and 'Only properties from the specific event that started each user's journey'. Below this, a 'Start API call' tile is shown. The 'Split criteria' section has a 'MUST HAVE' label and a 'Find contacts that meet any of the following clauses' dropdown. A 'Contact Property' clause is added with 'where digioh\_coupon' and 'Is Set'. There are buttons for 'ADD FIELD', '+ OR', and 'ADD A GROUP'.

5. Add a Call Webhook tile and configure it to call the webhook you created in Step 3

6. Add a Hold Until tile and configure it to hold until the digioh\_coupon\_created event has occurred:



Hold until digioh\_coupon\_created event received

#### Hold criteria

Pause users at this tile until:

Event occurs

digioh\_coupon\_created

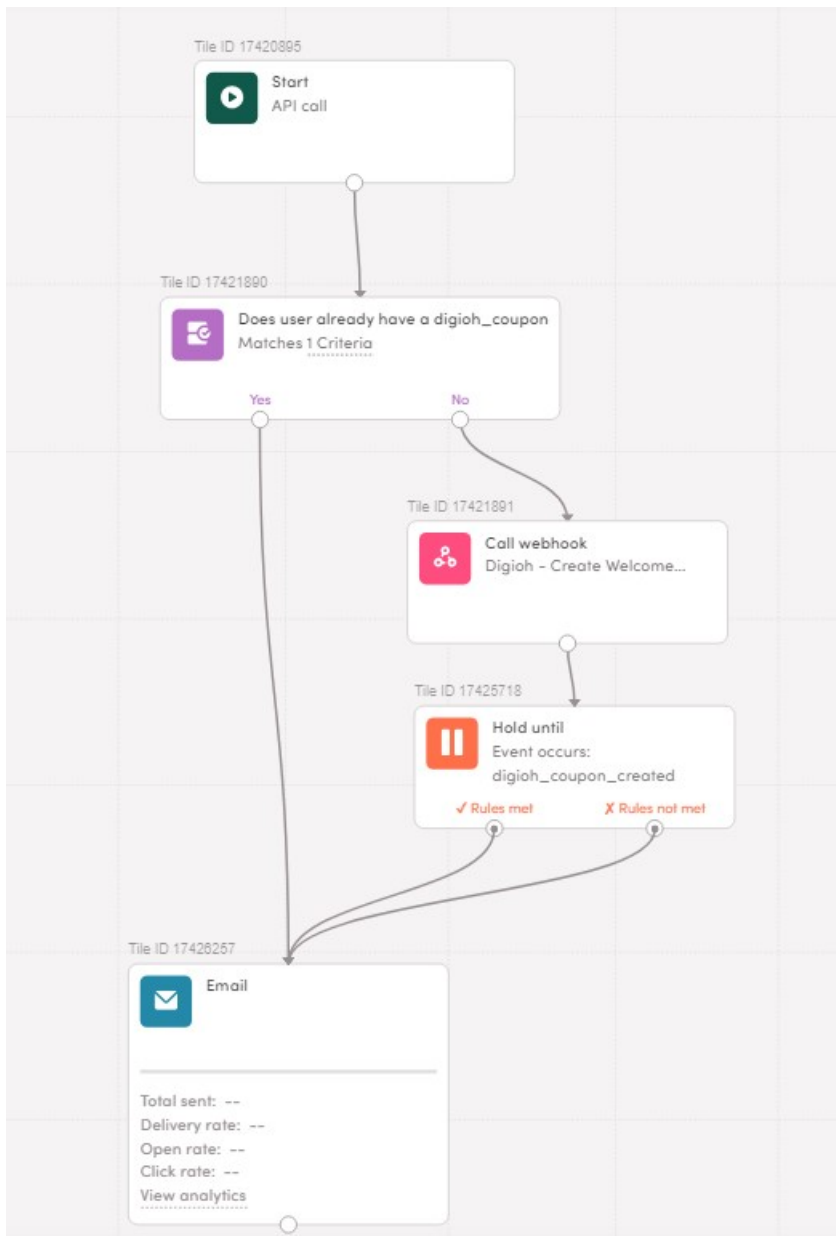
#### Hold duration

Users who don't meet your criteria by the end of this period will go down the "Rules not met" branch.

5 Minutes



7. Finally, add an Email tile, configuring its template as needed to send the user their coupon via the tag `{{digioh_coupon}}`. The Complete Journey:



Now that you have your configured Digioh Connect Pipeline, and have integrated it with an Iterable Journey, you can send profiles through the journey to create coupons, and send messages to those profiles that include their unique coupon code.

If you have any questions or run into any issues, feel free to reach out to our support team at [support@digioh.com](mailto:support@digioh.com)!

---