MAKE WORKFLOW INSTRUCTIONS

Three Pre-Built Automation Workflows for Event Professionals

What is Make?
Make (formerly Integromat) is a no-code automation platform that connects apps and services. Think Zapier, but more powerful for complex workflows. Free tier includes 1,000 operations/month.

Get Started: https://make.com

WORKFLOW #1: SPEAKER CONTENT AUTOMATION

What It Does

Takes the clean output from your Speaker Content Processor agent and automatically distributes it to 6 systems:

1. WordPress website (speaker bio page)
2. Event app (via API)
3. Social card generator (Bannerbear or Canva)
4. Email marketing (Mailchimp/HubSpot)
5. Google Sheet (for printed program)
6. Slack notification (to your team)

Prerequisites

- Make account (free tier works)
- Google account (for Sheets trigger)
- API keys for your platforms (WordPress, event app, email system)
- Bannerbear or Canva API key (for social cards)
- Slack webhook URL

Workflow Structure

TRIGGER: Google Sheets (Watch New Rows)

- Watches a specific Google Sheet for new rows
- When you paste agent output into the sheet, workflow triggers automatically

STEP 1: Parse Row Data

- Extracts: Speaker name, 50-word bio, 100-word bio, 1-sentence intro, session title, session abstract, key takeaways, alt text, tech requirements

STEP 2: Router (Splits into 6 parallel paths)

Path A: WordPress → HTTP Module

- Makes POST request to WordPress REST API
- Endpoint: `yoursite.com/wp-json/wp/v2/posts`
- Body:

json

```
{
  "title": "{{speaker_name}}",
  "content": "{{100_word_bio}}<h2>Session</h2>{{session_abstract}}<ul><li>{{takeaway_1}}</li><li>{{takeaway_2
  "status": "publish",
  "categories": [12]  // Your "Speakers" category ID
}
```

- Authentication: Basic Auth or Application Password

Path B: Event App API → HTTP Module

- Example for Whova API:
- Endpoint: `https://whova.com/api/v2/event/{{event_id}}/speakers`
- Method: POST
- Body:

json

```
{
  "name": "{{speaker_name}}",
  "bio": "{{100_word_bio}}",
  "session_title": "{{session_title}}",
  "session_description": "{{session_abstract}}"
}
```

- Headers: `Authorization: Bearer {{your_api_key}}`

Path C: Bannerbear (Social Card) → Bannerbear Module

- Template ID: (Create a template in Bannerbear first)
- Modifications:
  - `speaker_name`: {{speaker_name}}
  - `session_title`: {{session_title}}
  - `headshot_url`: (Upload headshot to cloud storage first, pass URL)
- Output: Returns image URL
- Optional: Post to Buffer/Hootsuite for scheduling

Path D: Mailchimp (Email Drip) → Mailchimp Module

- Action: Add/Update Subscriber
- List ID: Your speaker email list
- Email: {{speaker_email}}

- Merge Fields:
  - `FNAME`: {{first_name}}
  - `LNAME`: {{last_name}}
  - `BIO`: {{50_word_bio}}
  - `SESSION`: {{session_title}}
- Tags: ["speaker", "{{event_name}}"]

Path E: Google Sheets (Program Tracker) → Google Sheets Module

- Action: Add a Row
- Spreadsheet: "Event Program Master Sheet"
- Values:
  - Column A: {{speaker_name}}
  - Column B: {{50_word_bio}}
  - Column C: {{session_title}}
  - Column D: {{session_abstract}}
  - Column E: {{tech_requirements}}

Path F: Slack Notification → Slack Module

- Action: Send a Message
- Channel: #event-production
- Message:

✅ Speaker packet processed: *{{speaker_name}}*

Session: {{session_title}}

✅ Bio posted to website
✅ Added to event app
✅ Social card generated: {{social_card_url}}
✅ Added to email list
✅ Logged in program tracker

Tech requirements: {{tech_requirements}}

Setup Instructions

Step 1: Create Trigger Google Sheet

1. Create a new Google Sheet named "Speaker Content Pipeline"
2. Headers in Row 1:
   - A: Speaker Name
   - B: 50-word Bio
   - C: 100-word Bio
   - D: 1-sentence Intro
   - E: Session Title

- F: Session Abstract
- G: Takeaway 1
- H: Takeaway 2
- I: Takeaway 3
- J: Alt Text
- K: Tech Requirements

Step 2: Import Workflow to Make

1. Log in to Make
2. Click "Create a new scenario"
3. Import the JSON file (provided separately)
4. OR build manually following the structure above

Step 3: Connect Your Apps

1. Google Sheets: Authorize Make to access your Google account
2. WordPress: Add your site URL + Application Password
3. Event App: Add API key from your event app dashboard
4. Bannerbear: Add API key from Bannerbear dashboard
5. Mailchimp: Authorize Make to access Mailchimp
6. Slack: Add webhook URL from Slack workspace settings

Step 4: Test the Workflow

1. Manually add a test row to your Google Sheet
2. Watch the workflow run in Make
3. Verify each output (check WordPress, event app, email list, etc.)
4. Debug any errors in Make's execution log

Step 5: Deploy

1. Turn on the scenario in Make
2. Schedule: Real-time (watches for new rows continuously)
3. Paste agent output into Google Sheet → workflow triggers automatically

---

WORKFLOW #2: SPONSOR DELIVERABLES TRACKING

What It Does

Parses sponsor contracts, creates deliverables tracking records, schedules reminder emails, and notifies your team when deliverables are due or completed.

Prerequisites

- Make account
- Airtable account (free tier works)
- Gmail or SendGrid account
- Slack webhook URL

Workflow Structure

TRIGGER: Airtable (Watch New Records)

- Base: "Event Management"
- Table: "Sponsors"
- Watches for new sponsor records

STEP 1: Parse Deliverables

- Extract deliverables from "Contract Terms" field
- Split by newline or comma
- Create array of deliverables with due dates

STEP 2: Iterator → Create Deliverable Records

- For each deliverable in array:
  - Create new record in "Deliverables" table:
    - Sponsor Name: {{sponsor_name}}
    - Deliverable: {{deliverable_name}}
    - Due Date: {{due_date}}
    - Status: "Pending"
    - Reminder Sent: No

STEP 3: Schedule Reminder Emails

- Tool: Make's built-in scheduler
- Runs daily at 9am
- Checks "Deliverables" table for:
  - Status = "Pending"
  - Due Date = 14 days from today
  - Reminder Sent = No

STEP 4: Send Reminder Email

- To: {{sponsor_contact_email}}
- Subject: "Reminder: {{deliverable_name}} due in 2 weeks"
- Body:

Hi {{sponsor_contact_name}},

This is a friendly reminder that your {{deliverable_name}} deliverable is due on {{due_date}}.

Details:
- Event: {{event_name}}
- Deliverable: {{deliverable_name}}
- Due Date: {{due_date}}

Please let us know if you have any questions!

Best,
{{your_name}}
{{your_title}}

STEP 5: Update Airtable Record

- Set "Reminder Sent" = Yes
- Add note: "Reminder sent on {{today}}"

STEP 6: Slack Notification

- Channel: #sponsors
- Message:

📧 Reminder sent to {{sponsor_name}}
Deliverable: {{deliverable_name}}
Due: {{due_date}}

STEP 7: Completion Workflow

- Trigger: Airtable (Watch Updated Records)
- Condition: Status changed to "Complete"
- Actions:
  - Send thank you email to sponsor
  - Slack notification: "✅ {{sponsor_name}} delivered {{deliverable_name}}"
  - Update "Sponsor Fulfillment Report" Google Sheet

Setup Instructions

Step 1: Create Airtable Base

1. Create "Event Management" base
2. Create "Sponsors" table with fields:
    - Sponsor Name (Single line text)
    - Contact Name (Single line text)
    - Contact Email (Email)
    - Contract Terms (Long text) - paste deliverables list here
    - Sponsorship Level (Single select)
3. Create "Deliverables" table with fields:
    - Sponsor Name (Linked to Sponsors table)
    - Deliverable (Single line text)
    - Due Date (Date)
    - Status (Single select: Pending, In Progress, Complete, Overdue)
    - Reminder Sent (Checkbox)
    - Notes (Long text)

Step 2: Import Workflow to Make

1. Import JSON file or build manually
2. Connect Airtable, Gmail/SendGrid, Slack

Step 3: Test

1. Add a test sponsor with deliverables
2. Watch workflow create deliverable records
3. Manually set due date to 14 days from today
4. Trigger reminder workflow
5. Verify email sent and Airtable updated

WORKFLOW #3: POST-EVENT SURVEY ANALYSIS

What It Does

Takes raw survey responses, sends to Claude API for analysis, generates executive summary with insights and recommendations, creates visualizations, and distributes report to stakeholders.

Prerequisites

- Make account
- Claude API key (from Anthropic)
- Google account (for Sheets and Docs)
- Survey tool account (SurveyMonkey, Typeform, Google Forms)

Workflow Structure

TRIGGER: Google Sheets (Watch New Rows) OR Webhook

- Option A: Export survey responses to Google Sheet, workflow watches for new rows
- Option B: Survey tool sends webhook when survey closes

STEP 1: Aggregate Responses

- Collect all responses from sheet
- Format as JSON array

STEP 2: Claude API Analysis → HTTP Module

- Endpoint: `https://api.anthropic.com/v1/messages`
- Method: POST
- Headers:
  - `x-api-key`: {{your_claude_api_key}}
  - `anthropic-version`: 2023-06-01
  - `content-type`: application/json
- Body:

json

```
{
  "model": "claude-sonnet-4-20250514",
  "max_tokens": 4000,
  "messages": [
    {
      "role": "user",
      "content": "Analyze these post-event survey responses and provide:\n\n1. Executive Summary (3-4 sentences)\n2. Ser
    }
  ]
}
```

STEP 3: Parse Claude Response

- Extract JSON from Claude's response
- Map to variables:
  - exec_summary
  - sentiment_positive
  - sentiment_neutral
  - sentiment_negative
  - top_wins (array)
  - top_improvements (array)
  - key_themes (array)
  - quotes (array)
  - action_items (array)

STEP 4: Create Google Doc Report

- Template: "Post-Event Survey Report"
- Populate with:
  - Event Name: {{event_name}}
  - Event Date: {{event_date}}
  - Total Responses: {{response_count}}
  - Executive Summary: {{exec_summary}}
  - Sentiment Breakdown:
    - ✅ Positive: {{sentiment_positive}}%

        - 😐 Neutral: {{sentiment_neutral}}%
        - ❌ Negative: {{sentiment_negative}}%
    - Top 3 Wins:
        - {{top_wins[0]}}
        - {{top_wins[1]}}
        - {{top_wins[2]}}
    - Top 3 Improvements:
        - {{top_improvements[0]}}
        - {{top_improvements[1]}}
        - {{top_improvements[2]}}
    - Key Themes: {{key_themes}}
    - Notable Quotes: {{quotes}}
    - Recommended Actions: {{action_items}}

STEP 5: Create Visualization (Optional)

- Google Sheets chart module
- Create pie chart for sentiment breakdown
- Create bar chart for rating questions
- Export as image, insert into Google Doc

STEP 6: Email Report to Stakeholders

- To: {{stakeholder_emails}}
- Subject: "Post-Event Survey Analysis: {{event_name}}"
- Body: Link to Google Doc + executive summary
- Attachment: PDF export of report

STEP 7: Slack Notification

- Channel: #event-reports
- Message:

📊 Survey Analysis Complete: {{event_name}}

Responses: {{response_count}}
Sentiment: {{sentiment_positive}}% positive

Top Win: {{top_wins[0]}}
Top Improvement: {{top_improvements[0]}}

Full report: {{google_doc_url}}

Setup Instructions

Step 1: Get Claude API Key

1. Go to console.anthropic.com
2. Create API key
3. Copy key (starts with `sk-ant-`)
4. Add $5-10 credit (analysis costs ~$0.10-0.50 per survey)

Step 2: Export Survey Data

1. Export survey responses to Google Sheets
2. Format columns: Question | Response | Respondent ID | Timestamp
3. Share sheet with Make

Step 3: Create Report Template

1. Create Google Doc: "Post-Event Survey Report Template"
2. Add placeholders: {{event_name}}, {{exec_summary}}, etc.
3. Share with Make

Step 4: Import Workflow to Make

1. Import JSON or build manually
2. Connect Google Sheets, Google Docs, Gmail, Slack
3. Add Claude API key

Step 5: Test

1. Run with sample survey data (5-10 responses)
2. Verify Claude analysis is accurate
3. Check Google Doc is populated correctly
4. Confirm email and Slack notifications sent

Step 6: Deploy

1. Schedule to run after survey closes
2. OR trigger manually when you're ready for analysis

TROUBLESHOOTING

Common Issues

"Invalid API key" error

- Double-check API key is copied correctly (no extra spaces)
- Verify API key has not expired
- Check API key has sufficient credits

Workflow doesn't trigger

- For Google Sheets: Make sure "Watch New Rows" is set to correct sheet and range
- For webhooks: Test webhook URL in your trigger app
- Check Make scenario is turned ON (not paused)

Missing data in outputs

- Check field mapping in Router modules
- Verify variable names match exactly

- Test with simple data first, then add complexity

Rate limit errors

- Add delays between API calls (1-2 seconds)
- Reduce frequency of scheduled workflows
- Upgrade to paid tier if hitting operation limits

COST BREAKDOWN

Make:

- Free tier: 1,000 operations/month (enough for testing)
- Core plan: $9/month for 10,000 operations (recommended for production)

Claude API:

- ~$0.10-0.50 per survey analysis (depending on response count)
- $5 minimum credit purchase

Bannerbear (Social Cards):

- Free tier: 50 images/month
- Starter plan: $19/month for 500 images

Total estimated monthly cost for all 3 workflows:

- Free tier: $0 (limited usage)
- Production tier: $30-50/month (unlimited usage within reason)

NEXT STEPS

1. **Start with Workflow #1** (Speaker Content Automation)
    - Easiest to test
    - Immediate time savings
    - Clear ROI
2. **Add Workflow #2** once comfortable (Sponsor Tracking)
    - More complex but high value
    - Prevents missed deadlines
3. **Deploy Workflow #3** post-event (Survey Analysis)
    - Runs once per event
    - Impressive stakeholder deliverable

Need help? Email me: [your email]

Want to see these workflows in action? Watch the demo videos in the Google Drive folder.