



Creating  
DuoFactory  
*An Orchestration  
Ecosystem with  
Airflow*

Summit 2025

3.0



Belle Romea,  
Software Engineer  
@ Duolingo



**Duolingo** ✓

@duolingo

Remember to not miss any lessons today! It wouldn't be very nice if we had to lock up your family.

12:07 PM · 30 Mar 19 · [Twitter Web Client](#)

2,578 Retweets 5,869 Likes



**FBI** ✓ @FBI · 47m

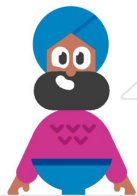
Replying to [@duolingo](#)

We are working on tracking you down. It is not okay to kidnap families because someone didn't complete a lesson.



Duolingo is a popular,  
free  
language-learning  
app that uses a  
gamified, bite-sized  
lesson approach to  
teach reading, writing,  
listening, and  
speaking skills

Write this in English



¿Cuáles son tus libros favoritos?

park

are

good

your

What

favorite

books

to learn

bedroom

our mission is to  
develop the best  
education in the  
world and make it  
universally  
available

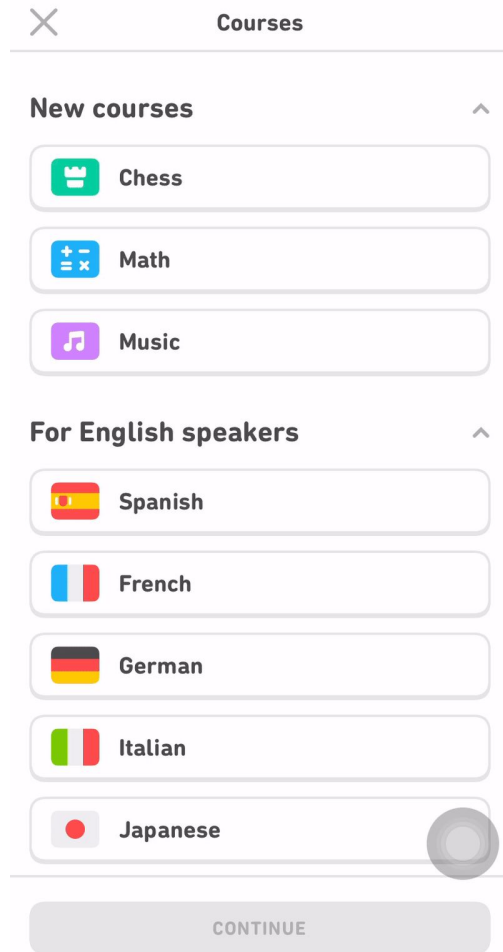


... but how do we do that?

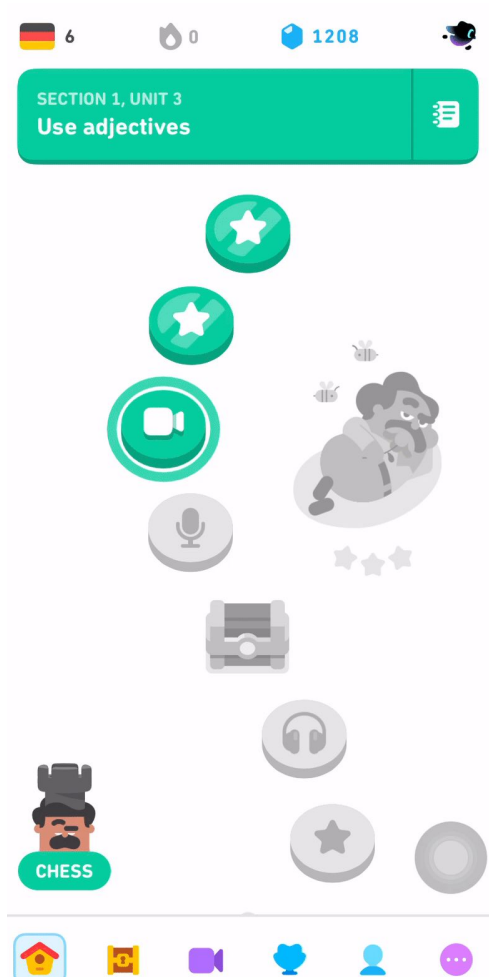




meet the  
duolingo  
curriculums



each course  
needs a lot  
of content





just to create the new sentences  
it take around 600 hours per  
course!

Write this in English



¿Cuáles son tus libros favoritos?



park

are

good

your

What

favorite

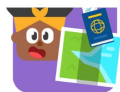
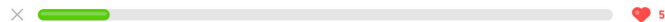
books

to learn

bedroom

and we want rich content





### Voyage en Espagne

Vikram parle avec son amie, Béa.



Salut, Béa ! Comment ça va ?



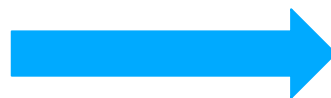
Super ! Je suis très très contente d'aller en Espagne !

stories



podcasts

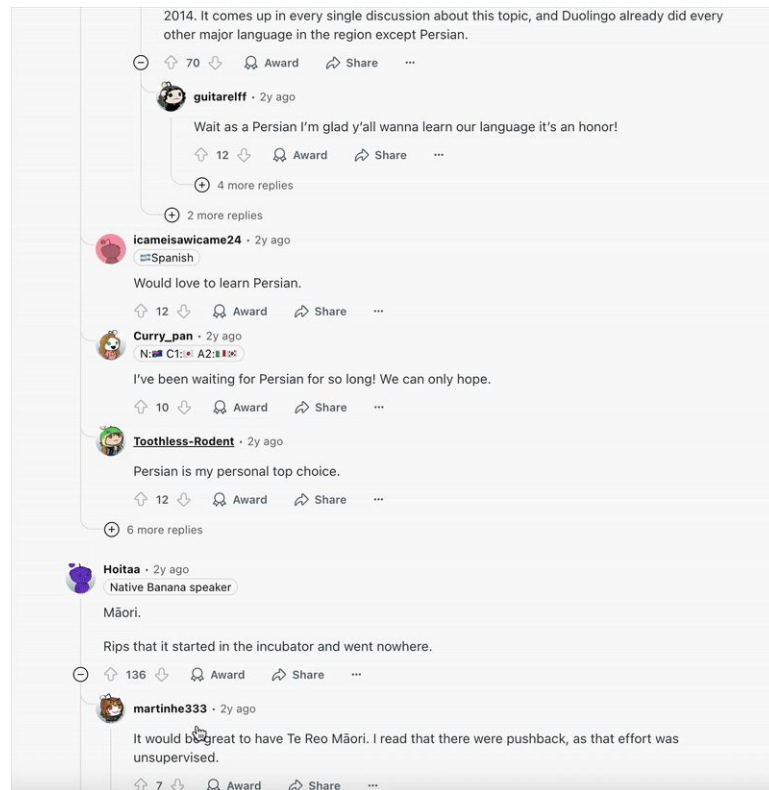
but this is timely + expensive



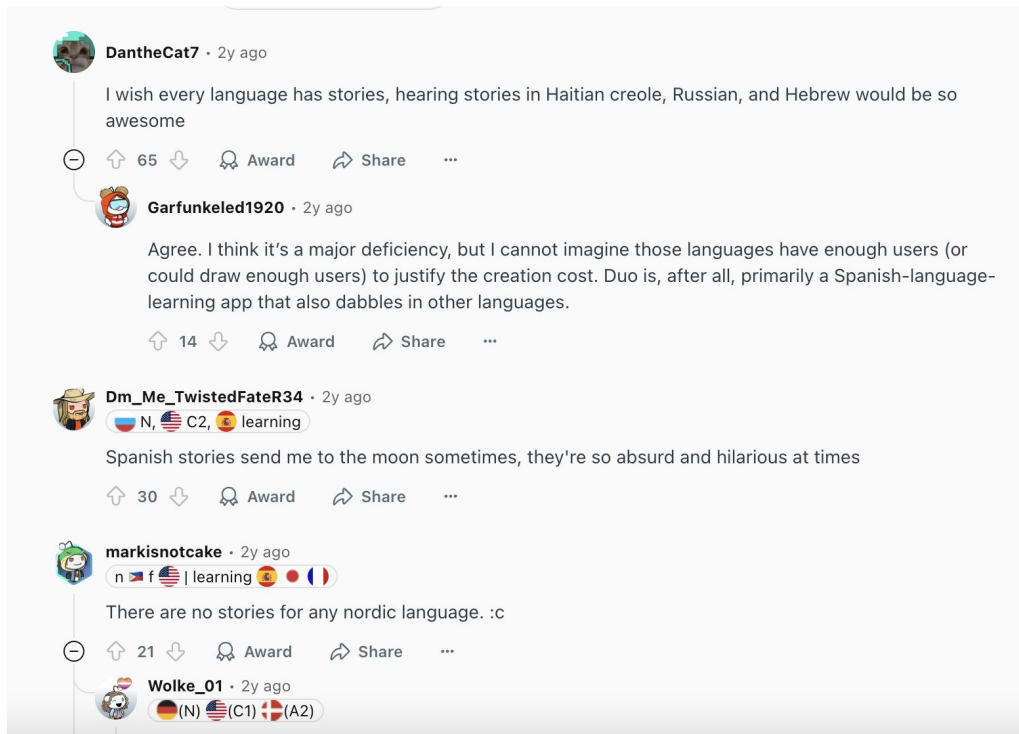


the result

# limited courses



# limited content per course



but then the quality of  
LLM generation  
improved...

so much that LLMs  
can generate our  
content to speed us  
up

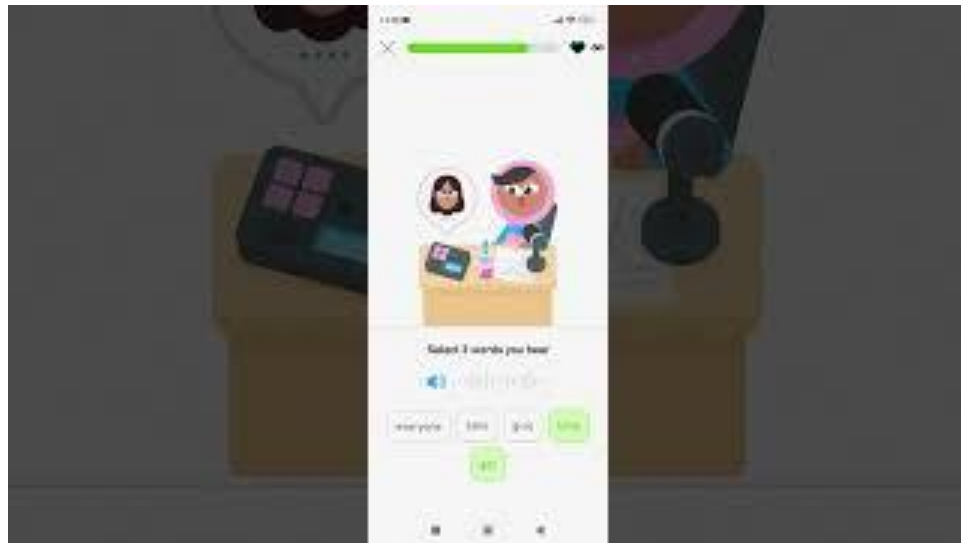




# example



a Duoradio  
takes 6 hours  
with LLMs+  
human review



okay ... problem solved





hold your horses



limitations to genai scalability

# engineering costs increasing

as we implemented new ai approaches,  
we had fragmented tooling landscape

- Bespoke one-off endpoints on existing services → technical debt
- No easy way to share and reuse logic across workflows → reinventing the wheel
- Homegrown orchestration was non-trivial to scale





enter Airflow



# DuoFactory

wrapper around  
Airflow for genai  
orchestration

The screenshot shows the Airflow web interface. At the top, there's a navigation bar with links like DAGs, Cluster Activity, Datasets, Security, Browse, Admin, Docs, and Other. Below this is a header bar with the Airflow logo and a message: "Looking for DuoFactory help? Start here!". The main section is titled "DAGs" and contains a table of DAGs. The table has columns for DAG name, Owner, Runs, Schedule, Last Run, Next Run, and Recent Tasks. The DAGs listed include "add\_cb\_exercises\_tags", "add\_topic\_can\_do\_grammar\_to\_scope\_and\_sequence", "ai\_art\_pipeline\_for\_vocab\_object", "airflow-utils.log-analyzer", "airflow-utils.truncate-database", "apply\_manual\_review\_updates", "auto\_label\_with\_prompt\_ensemble", "backfill\_flashcard\_alternate\_answers", "batch\_run\_test\_cases", and "benchmark\_bt\_evaluators". Each DAG row shows its status (e.g., Running, Failed), a list of runs with their dates and times, and a set of icons representing recent task statuses.

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks
add_cb_exercises_tags <small>extension_policy.medium</small>	generated_exercises artifice	● ● ● ● ●	None			● ● ● ● ●
add_topic_can_do_grammar_to_scope_and_sequence <small>extension_policy.medium</small>	artifice curriculum_and_infrastructure	● ● ● ● ●	None			● ● ● ● ●
ai_art_pipeline_for_vocab_object <small>extension_policy.medium</small>	artifice ai_art_team	● ● ● ● ●	None			● ● ● ● ●
airflow-utils.log-analyzer <small>extension_policy.medium</small>	artifice automation_platform	● ● ● ● ●	None			● ● ● ● ●
airflow-utils.truncate-database <small>extension_policy.medium</small>	artifice automation_platform	● ● ● ● ●	8:51***	2025-10-03, 17:00:00	2025-10-03, 18:00:00	● ● ● ● ●
apply_manual_review_updates <small>extension_policy.medium</small>	artifice curriculum_and_infrastructure	● ● ● ● ●	None			● ● ● ● ●
auto_label_with_prompt_ensemble <small>extension_policy.medium</small>	generated_exercises artifice	● ● ● ● ●	None			● ● ● ● ●
backfill_flashcard_alternate_answers <small>extension_policy.medium</small>	artifice short_term_experience	● ● ● ● ●	None	2025-10-03, 14:27:45		● ● ● ● ●
batch_run_test_cases <small>extension_policy.medium</small>	artifice automation_platform	● ● ● ● ●	None			● ● ● ● ●
benchmark_bt_evaluators	artifice	● ● ● ● ●	None	2025-10-03, 15:46:00		● ● ● ● ●



why airflow was a good  
candidate





# in-house orchestration is hard

Airflow provides rich  
orchestration tooling, including  
scheduling, dependency  
management, retries, and  
observability



# modularity

- tasks are reusable logic
- high ROI of eng work
- easily swap out parts of the pipeline as tech improves



# examples of tasks

- read/write google sheets
- read course data
- run prompts
- write to s3 to be viewed in the app



# idempotency

each step in a DuoFactory DAG  
can be retried safely without  
corrupting data or producing  
inconsistent result

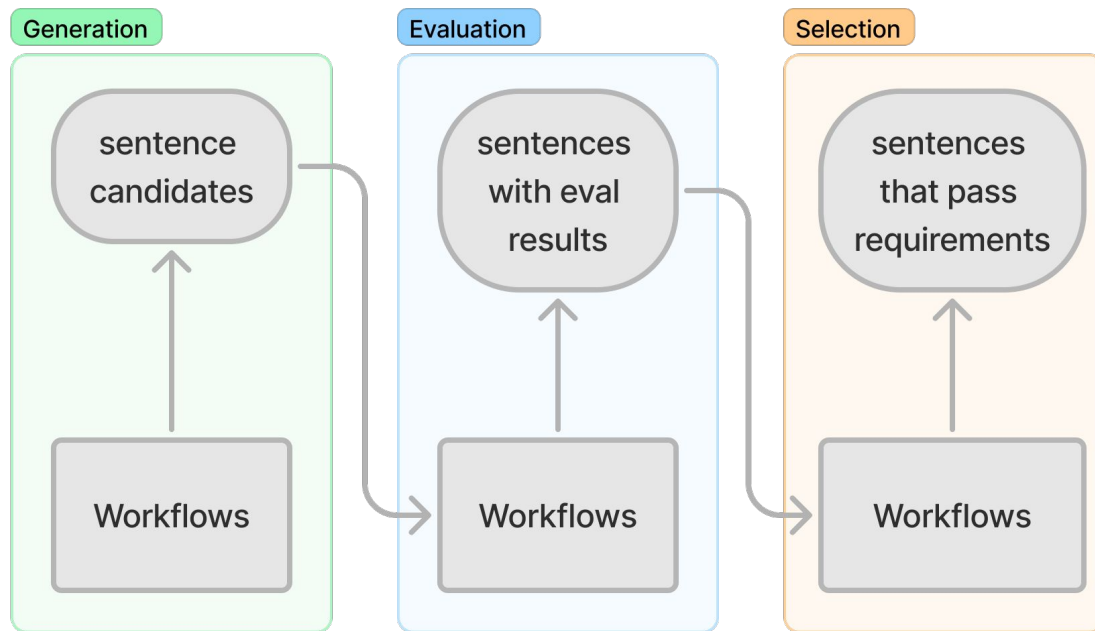


# ecosystem compatibility and extensibility

Airflow integrates easily with  
Duolingo's existing AWS,  
PostgreSQL, and  
Terraform-based infrastructure



Duofactory in action



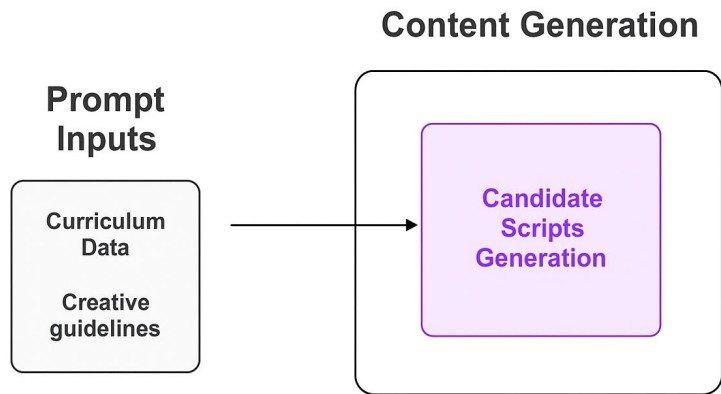


# example

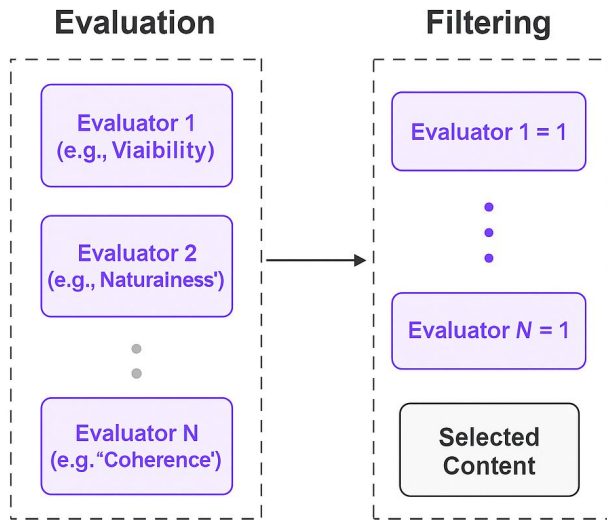
## Duoradio generation



# script generation DAGs



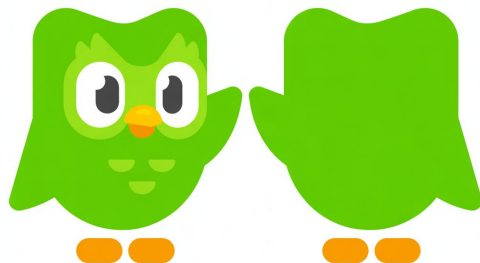
# script evaluation DAG





then repeat for exercises  
generation -> evals -> selection

extending to end to end





formatting DAG

- API calls to TTS/Viseme generation

publishing DAG



wins

created a pipeline to  
generate duoradio  
episodes with 0  
minutes of human  
intervention





cut costs by 99%





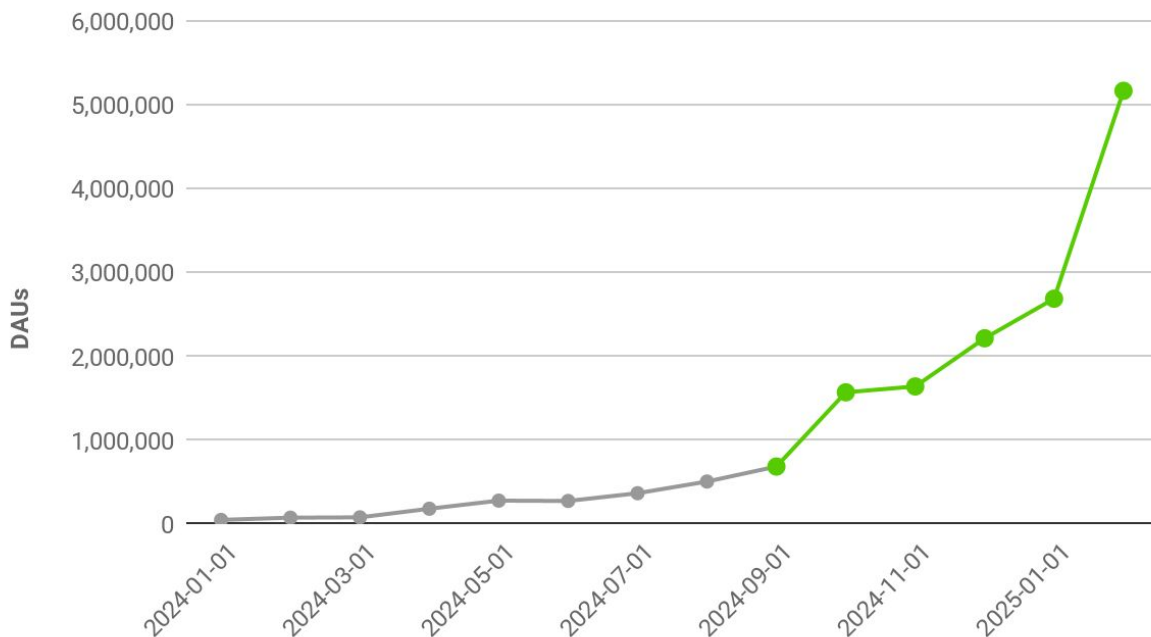


impact

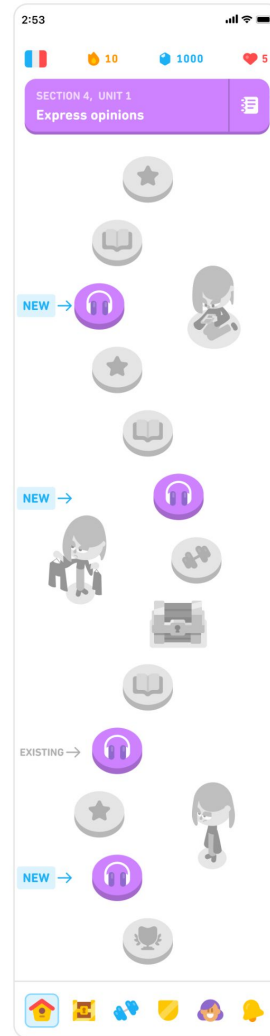
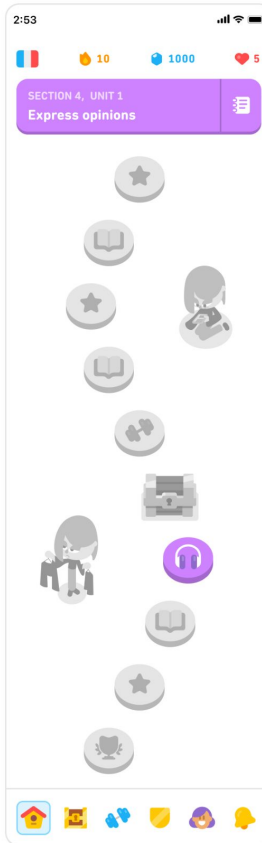
launched duoradio in 243  
courses



# reaching more learners



more  
episodes  
per unit



# expanding the ecosystem





prompt editor



HOME

belle test

DRAFT

Version 1

PUBLISH VERSION

RUN PROMPT

Chat Template

1 System Message

SWITCH TYPE

The prompt sent by the system...

2 User Message

SWITCH TYPE

The prompt sent by the user...

+ Create New Step

3 Assistant Message

SWITCH TYPE


☒ Auto Generate this Message

VIEW JSON

Input Variables

No variables were found in your prompt.  
In your prompt, you can define a custom variable using the `{{myVariableName}}` syntax.  
For image variables, use `{{image:myImageName}}` syntax.  
Image variables need to be put at the end of the message. For example, *describe the image* `{{image:image1}}`.

Chat



RUN YOUR PROMPT

You'll see messages after running the prompt.  
Use Command + Enter, or click Run Prompt.

Model Configuration

Output Variables



google sheets



	A	B	C	D	E	F	G	H	I	J	K	L	M	
1	wrapper_name	section_type	variability_type	en	es	fr	de	pt	it	ja	hi	ko	ru	vi
2	Lily and Strangers	intro	fixed	Welcome to "Lily and Strangers," where I'm forced to talk to p	Te doy la bienve	Bienvenue dans	Willkommen bei	Olá. Esse é o "L	Vi diamo il benve	こんにちは。「リ	"लिली और अजनबी	'릴리와 낯선 사람들	Привет. Это шо	Xin ch
3					1a) Soy Lily, y es 1a) Je m'appelle 1b) Soy Lily, y es 1b) Je m'appelle				1a) Eu sou a Lily		1) मैं लिली हूँ और य			
				1) I'm Lily and here's our new guest... Let's get this over with.. 2a) Soy Lily, y he 2a) Je m'appelle 1) Ich bin Lilli un					1) Io sono Lily. P 1) リリーです。: चलिए, कइके खत व 1) 저는 릴리예요. 1) Я Лили, а это 1) Tòi					
				2) I'm Lily, and today we have another guest who wants to talk 2b) Soy Lily, y he 2b) Je m'appelle 2) Ich bin Lilli un					2) Eu sou a Lily 2) Io sono Lily e 2) 리リーです。: 2) मैं लिली हूँ और 2) 저는 릴리예요. 2) Я Лили, и сег 2) Tòi					
				3) I'm Lily, and today we're listening to another stranger... Let 3) Soy Lily, y he 3) Je m'appelle 3) Ich bin Lilli un					3) Io sono Lily e 3) 리リーです。: 3) मैं लिली हूँ और 3) 저는 릴리예요. 3) Я Лили, и сег 3) Tòi					
	Lily and Strangers	intro	semi_fixed	4) I'm Lily, and today we have another person here to waste o 4) Soy Lily, y aq 4) Je m'appelle 4) Ich bin Lilli un					4) Eu sou a Lily 4) Io sono Lily e 4) 리リーです。: 4) मैं लिली हूँ और 4) 저는 릴리예요. 4) Я Лили, и сег 4) Tòi					
4	Lily and Strangers	outro	fixed	See you next time... or not...	Hasta la próxima	À la prochaine...	Bis zum nächste	Até a próxima...	Alla prossima...	では、また次回: फिर मिलेंगे...या शायद 다음 편에서 만나도	Увидимся в сле	Hen g		
5	Lucy Is Listening	intro	fixed	Welcome back to "Lucy Is Listening," where I uncover hidden	Te doy la bienve	Bienvenue dans	Willkommen zur		Vi do il benvenuto	こんにちは。「ル	"लूसी सुन रही है" में 'रुसी'가 듣고 있어	Приветствую ва	Xin ch	
6	Lucy Is Listening	intro	semi_fixed	I'm your host, Lucy, and today we're listening to two people [+	Soy tu anfitrióna	Je m'appelle Luc	Ich bin Lucy, dei	Aquí é a Lucy, a	Io mi chiamo Luc	どうも。ホスト(に) मैं आपकी होस्ट लूसी	저는 진행을 맡은	меня зовут Люс	Tòi là	
7	Lucy Is Listening	outro	fixed	I'll get to the bottom of this. Remember, I'm always listening...	Llegaré al fondo	Je découvre touj	Ich werde der St	Vamos desvend	Andrò a fondo di	必ず真相を突き。 मैं इसकी तह तक जा	제가 속속들이 파헤	Все тайное ста	Tòi sẽ	
8	Vikram's Community Hotline	intro	fixed	Welcome to "Vikram's Community Hotline," where members c	Les doy la bienv	Bienvenue dans	Willkommen bei	Queridos ouvint	Vi diamo il benve	こんにちは。「ウ	"विक्रम की कम्यूनिटी	'비크람의 지역 전화	Добро пожалов	Chào
9	Vikram's Community Hotline	intro	semi_fixed	I'm your host, Vikram, and today's message is all about [subj	Soy tu anfitrión,	Je m'appelle Vik	Ich bin dein Mod	Aquí é o Vikram,	Sono Vikram, e i	ホストのヴィク	मैं आपका होस्ट विक्र	저는 MC 비크람입	меня зовут Вик	Tòi là
10	Vikram's Community Hotline	intro	fixed	Let's hear from our first caller.	Escuchemos nu	Écoutons mainte	Hören wir uns de	Vamos ouvir nos	Ascoltiamo il pri	さて、最初に電	चलिए, अपने पहले व	그런 첫 번째 분	의 U нас уже есть	Và sai
11	Vikram's Community Hotline	outro	fixed	Thank you for joining us on "Vikram's Community Hotline." Re	Gracias por esc	Merci d'écouter	Das war „Vikram	Obrigado por ou	Grazie per aver	「ヴィクラムの	"विक्रम की कम्यूनिटी	'비크람의 지역 전화	Спасибо, что б	Câm c
12	Oscar's Antique Roadshow	intro	fixed	I'm your host, Oscar, and this is "Oscar's Antique Roadshow,"	Soy tu anfitrión,	Je m'appelle Os	Guten Tag und h	Eu sou seu anfit	Sono Oscar, e si	どうも。「オス	फा मैं आपका होस्ट ऑ	저는 MC 오스	ваш ведущий	Tòi têt



content management system



DuoRadio / Prod  
EN < ES




FILTER

<input type="checkbox"/>	EPISODE NAME	VIABILITY	WRAPPER	STATUS	SLOT	SCHEMA VERSION	PIPELINE STATUS	GLOBAL ATTEMPTS	LAST UPDATED		
▼ Empleeos (A1.0 – 63f532236a038ee957b7154aadddbcd)											
<input type="checkbox"/>	Work in Chicago	0.85	Lily and Strangers	PUBLISHED	0	3	Published	2	12/15/2024 3:46 AM EST		
<input type="checkbox"/>	Interesting Job	0.86	Falstaff's Guide to Humans	PUBLISHED	0	duoradio_a1_flagships	Published	1	6/20/2025 12:13 PM EDT		
<input type="checkbox"/>	Are You a Doctor?	0.86	Lucy Is Listening	PUBLISHED	1	duoradio_a1_flagships	Published	1	6/20/2025 12:13 PM EDT		
<input type="checkbox"/>	Artists and Offices	0.85	Lucy Is Listening	PUBLISHED	1	3	Published	1	11/20/2024 6:56 PM EST		
<input type="checkbox"/>	Work at the Hospital	0.87	Late Nights with Lin	PUBLISHED	2	duoradio_a1_flagships	Published	1	6/20/2025 12:13 PM EDT		
<input type="checkbox"/>	Meeting Humans	0.87	Falstaff's Guide to Humans	PUBLISHED	2	3	Published	1	11/20/2024 8:38 PM EST		
▼ Encuentros (A1.0 – 46f57661460e7b1621fed84e29bc0720)											
<input type="checkbox"/>	A Bilingual Encounter	0.94	Lucy Is Listening	PUBLISHED	1	duoradio_a1_flagships	Published	1	6/20/2025 12:13 PM EDT		
<input type="checkbox"/>	Nice to Meet You	0.92	Falstaff's Guide to Humans	PUBLISHED	1	3	Published	1	11/20/2024 11:14 PM EST		
<input type="checkbox"/>	Do You Speak English?	0.94	Falstaff's Guide to Humans	PUBLISHED	2	duoradio_a1_flagships	Published	1	6/20/2025 12:13 PM EDT		
<input type="checkbox"/>	Suspicious Nice to Meet You	0.96	Lucy Is Listening	PUBLISHED	2	3	Published	1	11/20/2024 10:15 PM EST		
<input type="checkbox"/>	Conversation in Two Languages	0.85	Falstaff's Guide to Humans	PUBLISHED	3	duoradio_a1_4_nodes_flagship	Published	2	7/14/2025 2:14 PM EST		

prod





workflow builder

codeless DAG  
using predefined  
building blocks

P

← HOME

A1 DuoRadio Title, Exercises & Formatting for Lucy IT<EN

SAVE

SAVE AND RUN

VIEW RESULTS

✎

⚙️

📁

IMPORT

PROMPT

EVALUATE

EVALUATE (PARALLEL)

SORT

SELECT

CUSTOM

⋮ Import 1

Import using Google Sheets

🗑️

⋮ Prompt 1

A1 DR episode title (UI)

Run prompt 1 time(s).

🗑️

⋮ Prompt 2

A1 DuoRadio 3-in1 exercises: Listen+Comp+CAP

Run prompt 2 time(s).

🗑️

⋮ Evaluate 1

Evaluate using Loading...

🗑️

⋮ Prompt 3

🗑️



each workflow block maps to a  
corresponding DAG based on its type and  
metadata

## execution flow

- load workflow definition from the workflow table
- trigger the corresponding block-type DAG, passing outputs from the previous block
- use `should_execute()` to stop once workflow completion is detected



# static architecture constraint

DAGs require a fixed structure known at scheduling time... but we get the definition at run time

always iterate MAX\_BLOCKS times, even if some iterations are skipped





```
workflow_id = cast(WorkflowId, "{{ params.workflow_id }}")
use_local_db = cast(bool, "{{ params.use_local_db }}")
workflow = get_workflow(workflow_id=workflow_id, use_local_db=use_local_db)
blocks = get_workflow_blocks(wf=workflow)
```



```
for i in range(MAX_BLOCKS):
    # For downstream blocks, pass the output of the previous block as the `payloads` parameter
    # If the previous outputs are empty, seed the first block with an empty payload
    payloads = previous_block_output or [Payload(block_results={})]

    # This returns both the first task and the actual output of the @task_group so we can set
    # the dependencies correctly
    first_task, current_block_output = run_current_block.override(
        default_args={
            "trigger_rule": TriggerRule.ALL_SUCCESS,
        },
        group_id=f"run_current_block_{i}",
    )(
        global_vars=cast(dict[str, Any], "{{ params.global_variables }}"),
        index=i,
        input_payloads=payloads,
        wf=workflow,
        wf_id=workflow_id,
        use_local_db=use_local_db,
    )

    if previous_block_output:
        # Short circuit the current block if we are at the end of the workflow
        # This will cause the downstream blocks to be skipped (until the final task group,
        # which has its own trigger rules, since we need to store the results at the end.)
        short_circuit = should_execute.override(
            ignore_downstream_trigger_rules=False,
            task_id=f"short_circuit_{i}",
            trigger_rule=TriggerRule.ONE_SUCCESS,
        )(
            curr_block=i,
            n_blocks=num_blocks,
        )
        chain(previous_block_output, short_circuit, first_task)

    previous_block_output = set_latest_output_payloads.override(
        task_id=f"set_latest_output_payloads_{i}",
    )(current_block_output, previous_block_output)
```

thank you



questions?



[belle@duolingo.com](mailto:belle@duolingo.com)

LinkedIn:  
[@isabel-romea](#)