From no-help to self-help

Streamlining processes with Slack and Python

From no-help to self-help

Streamlining processes with Slack and Python

Background

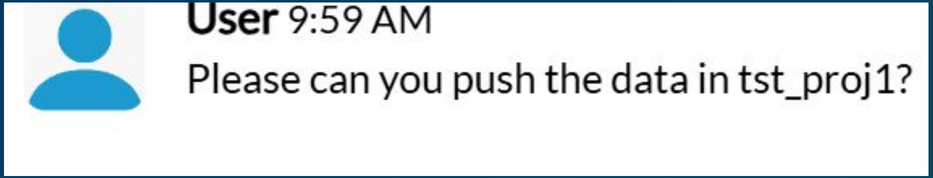
- Startup
- Product launched before finished
- Manual steps
- Bug reporting
- Feature requests

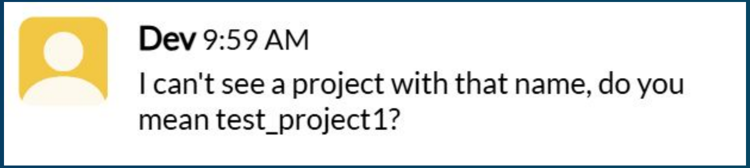
“We are trying to finish a rocket ship after it has already launched”

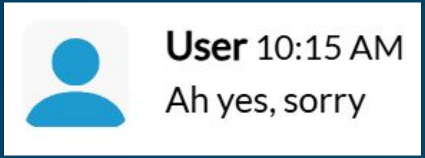
All about me

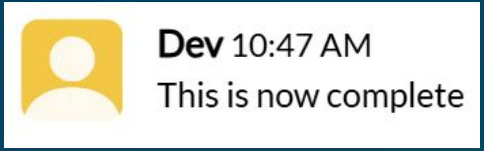
- Technical Lead @ Canopy Specialty Insurance
- Slack Certified Developer
- Hobbies:
    - Rugby
    - Running
    - Reading

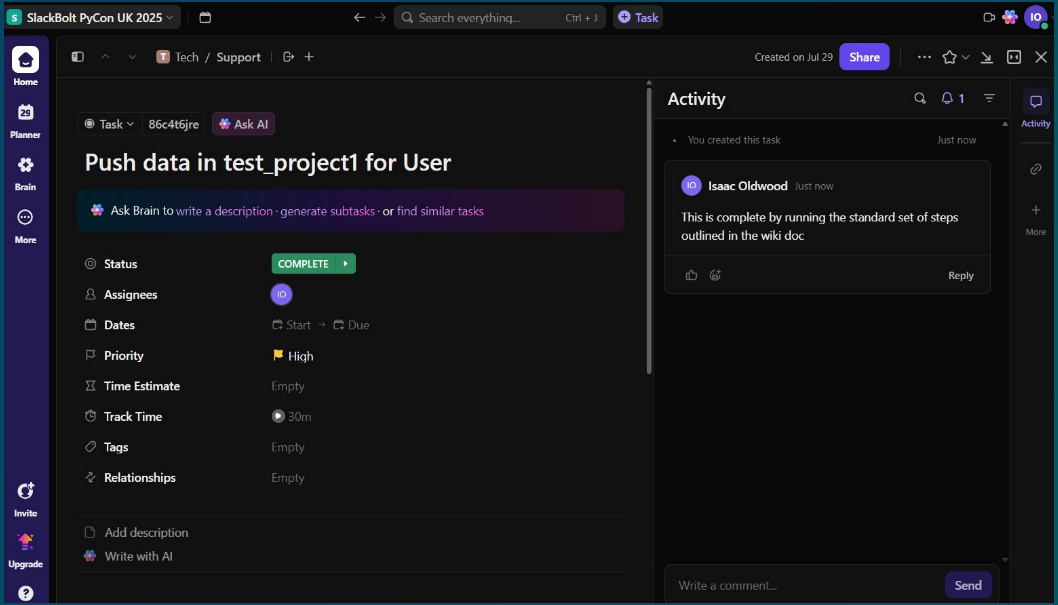
Stage 0 example


User 9:59 AM
Please can you push the data in tst_proj1?


Dev 9:59 AM
I can't see a project with that name, do you mean test_project1?


User 10:15 AM
Ah yes, sorry


Dev 10:47 AM
This is now complete



Push data in test_project1 for User

Ask Brain to write a description · generate subtasks · or find similar tasks

Status	COMPLETE
Assignees	IO
Dates	Start → Due
Priority	High
Time Estimate	Empty
Track Time	30m
Tags	Empty
Relationships	Empty

Add description
Write with AI

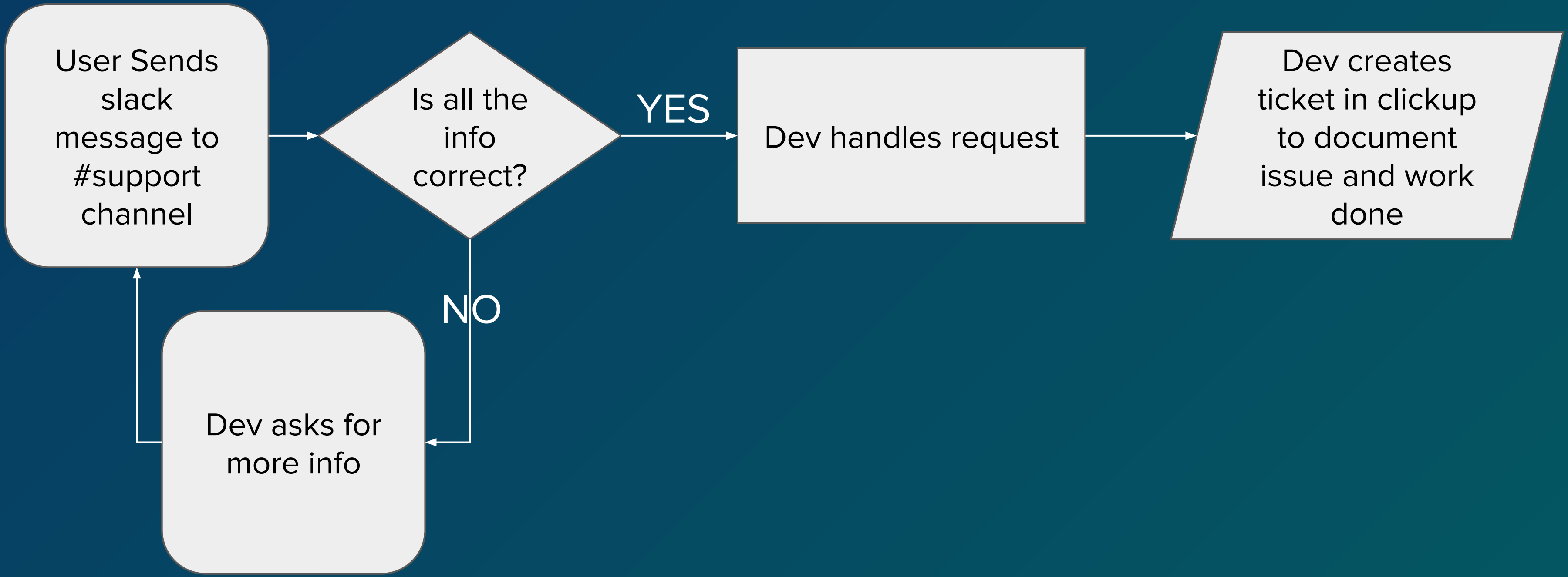
You created this task — Just now

Isaac Oldwood Just now
This is complete by running the standard set of steps outlined in the wiki doc

Reply

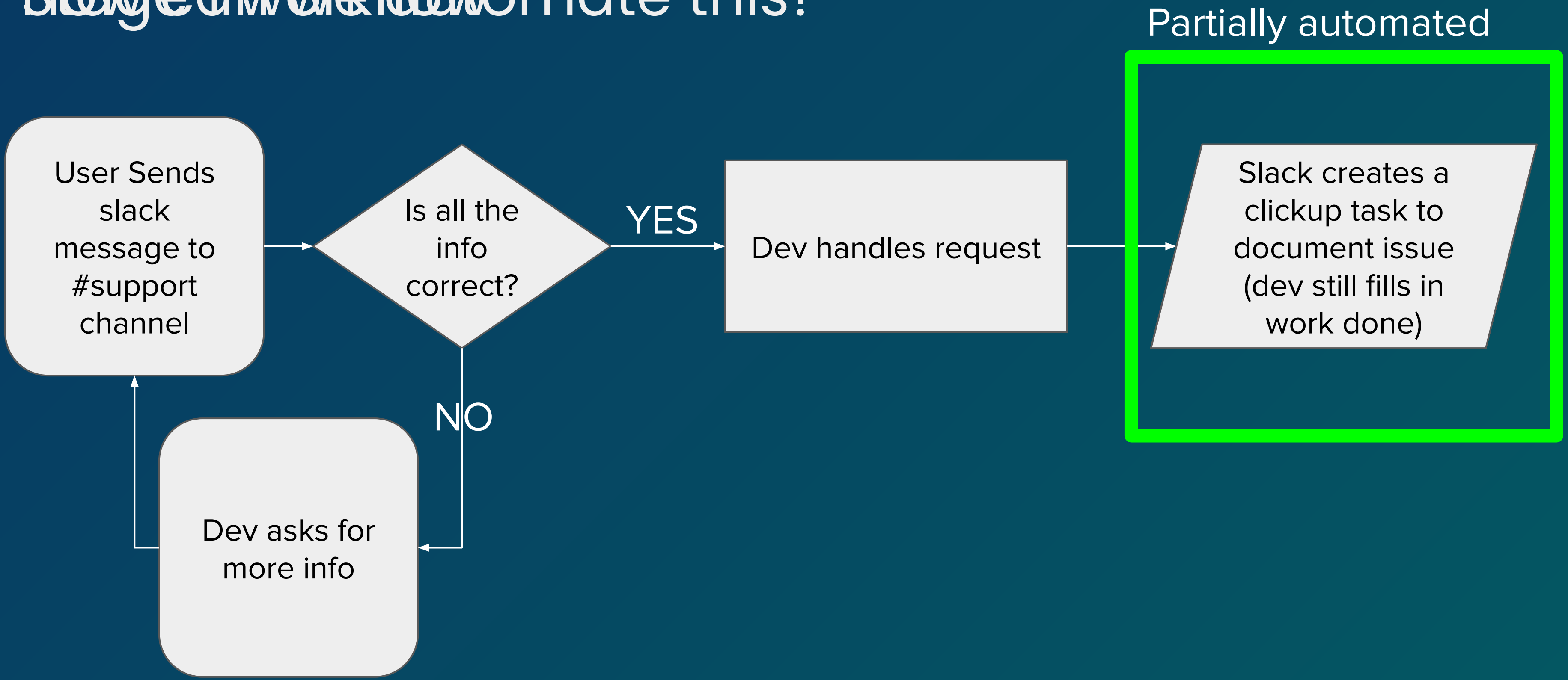
Write a comment... Send

Stage 0 workflow


User Sends slack message to #support channel
Is all the info correct?
YES
Dev handles request
Dev creates ticket in clickup to document issue and work done
NO
Dev asks for more info

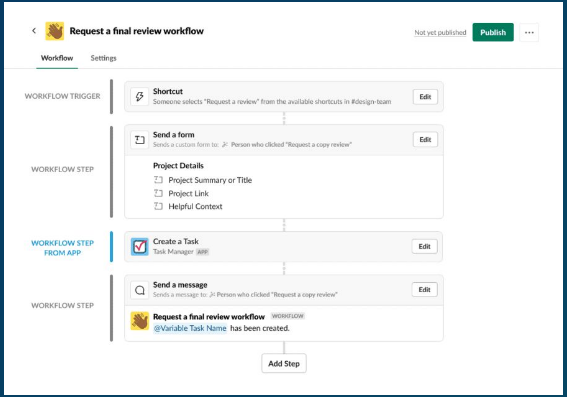
Stage 1 workflow

How can we automate this?


Partially automated
User Sends slack message to #support channel
Is all the info correct?
YES
NO
Dev handles request
Dev asks for more info
Slack creates a clickup task to document issue (dev still fills in work done)

Slack Automation Options

Workflows



Request a final review workflow

Not yet published · Publish · ···

Workflow · Settings

WORKFLOW TRIGGER — **Shortcut** — Someone selects "Request a review" from the available shortcuts in #design-team — Edit

WORKFLOW STEP — **Send a form** — Sends a custom form to: Person who clicked "Request a copy review" — Edit

Project Details
- Project Summary or Title
- Project Link
- Helpful Context

WORKFLOW STEP FROM APP — **Create a Task** — Task Manager APP — Edit

WORKFLOW STEP — **Send a message** — Sends a message to: Person who clicked "Request a copy review" — Edit

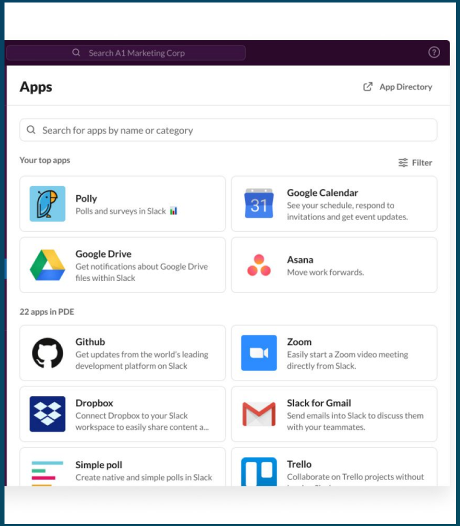
Request a final review workflow WORKFLOW
@Variable Task Name has been created.

Add Step

- No-code
- Paid

Image credit: Unito.io

Apps



Search A1 Marketing Corp

Apps · App Directory

Search for apps by name or category

Your top apps · Filter

Polly Polls and surveys in Slack	**Google Calendar** See your schedule, respond to invitations and get event updates.
Google Drive Get notifications about Google Drive files within Slack	**Asana** Move work forwards.

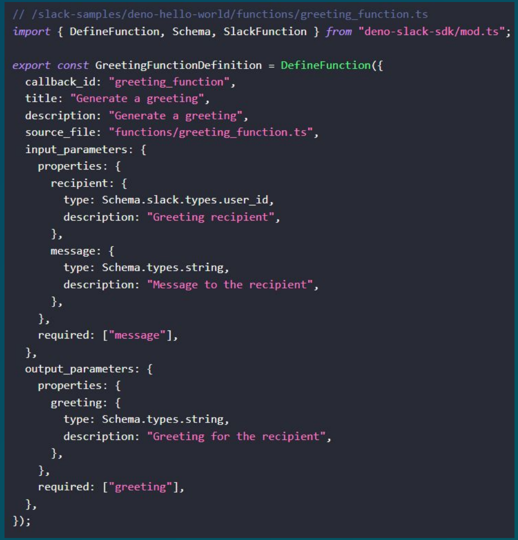
22 apps in PDE

Github Get updates from the world's leading development platform on Slack	**Zoom** Easily start a Zoom video meeting directly from Slack.
Dropbox Connect Dropbox to your Slack workspace to easily share content a...	**Slack for Gmail** Send emails into Slack to discuss them with your teammates.
Simple poll Create native and simple polls in Slack	**Trello** Collaborate on Trello projects without

- Code
- Self hosted
- Free

Image credit: Slack.com

Functions



```typescript
// /slack-samples/deno-hello-world/functions/greeting_function.ts
import { DefineFunction, Schema, SlackFunction } from "deno-slack-sdk/mod.ts";

export const GreetingFunctionDefinition = DefineFunction({
  callback_id: "greeting_function",
  title: "Generate a greeting",
  description: "Generate a greeting",
  source_file: "functions/greeting_function.ts",
  input_parameters: {
    properties: {
      recipient: {
        type: Schema.slack.types.user_id,
        description: "Greeting recipient",
      },
      message: {
        type: Schema.types.string,
        description: "Message to the recipient",
      },
    },
    required: ["message"],
  },
  output_parameters: {
    properties: {
      greeting: {
        type: Schema.types.string,
        description: "Greeting for the recipient",
      },
    },
    required: ["greeting"],
  },
});
```

- TypeScript
- Hosted on slack
- Paid

Image credit: slack.dev

Slack Apps

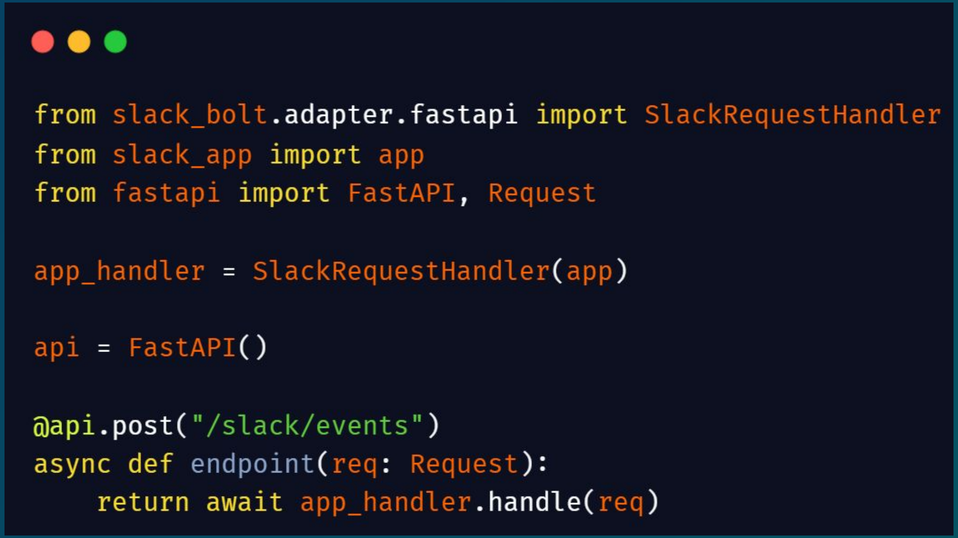
- Web APIs
- Split into two parts
  - App manifest on api.slack.com/apps
  - The self-hosted app
- Freedom to use any technology/stack
- Quickstart with Bolt
  - Javascript
  - Java
  - Python

Developing a Slack App

- install slack_bolt package
- Create a slack app https://api.slack.com/apps/new
- Write some code!

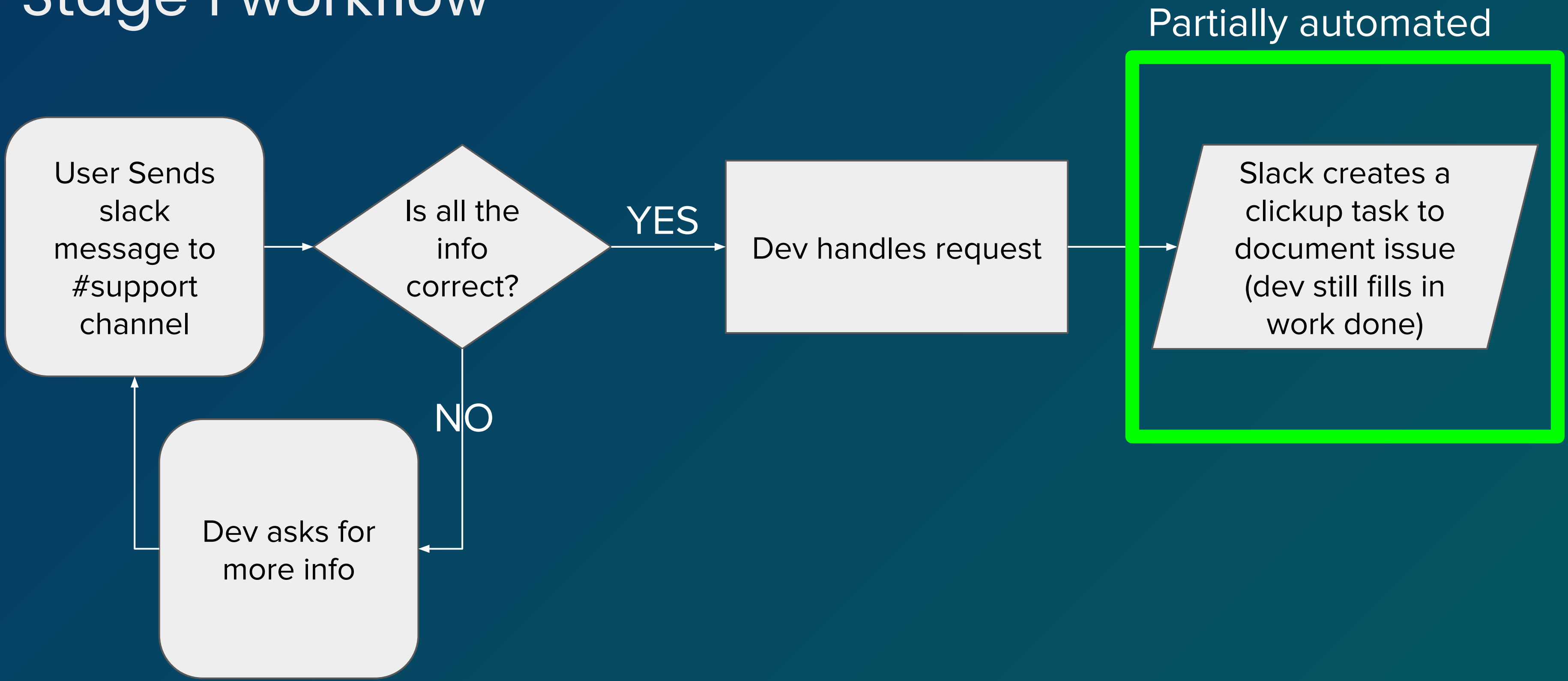
Deploying a Slack App

- Default adapter: `HTTPServer` - not suitable for production
- Wide range of adapters supported: Flask, Django, FastAPI etc
- Plug and play - deploy like any other app!

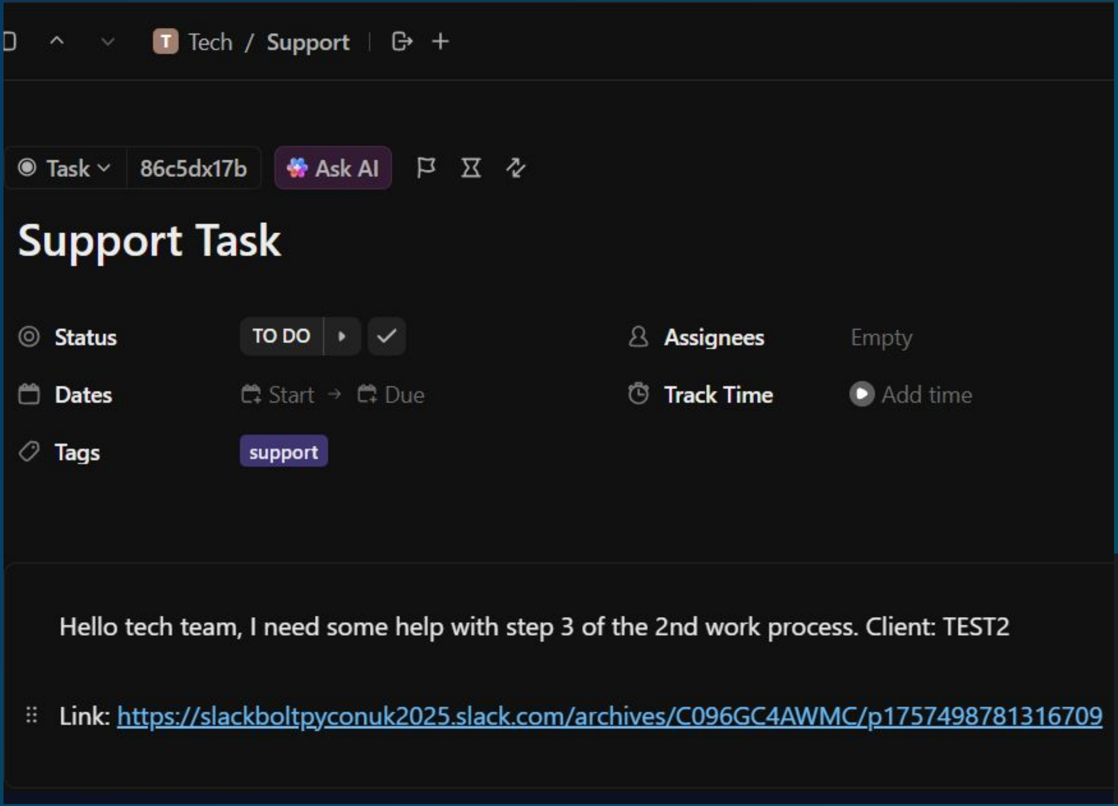


```python
from slack_bolt.adapter.fastapi import SlackRequestHandler
from slack_app import app
from fastapi import FastAPI, Request

app_handler = SlackRequestHandler(app)

api = FastAPI()

@api.post("/slack/events")
async def endpoint(req: Request):
    return await app_handler.handle(req)
```

Stage 1 workflow


Partially automated
User Sends slack message to #support channel
Is all the info correct?
YES
Dev handles request
Slack creates a clickup task to document issue (dev still fills in work done)
NO
Dev asks for more info

Stage 1 Automation

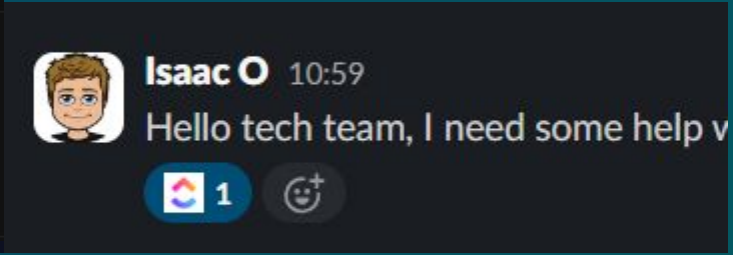


Support Task

Status	TO DO	Assignees	Empty
Dates	Start → Due	Track Time	Add time
Tags	support		

Hello tech team, I need some help with step 3 of the 2nd work process. Client: TEST2

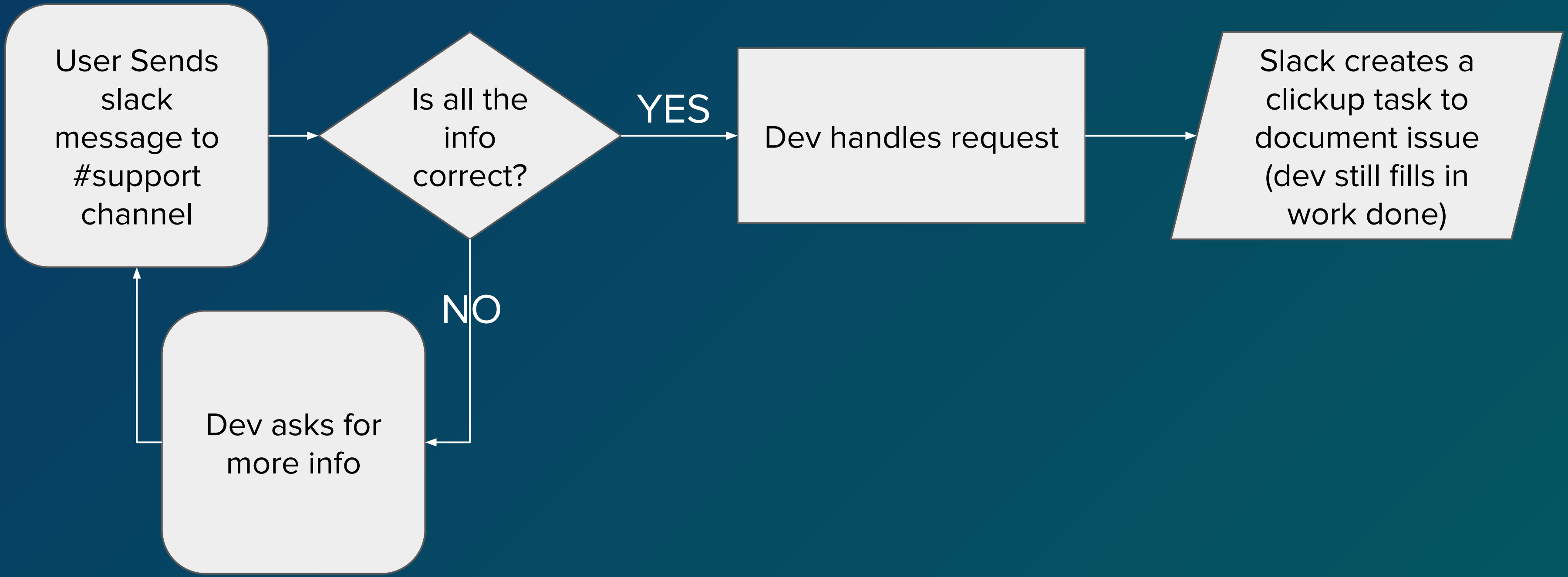
Link: https://slackboltpyconuk2025.slack.com/archives/C096GC4AWMC/p1757498781316709



Isaac O 10:59

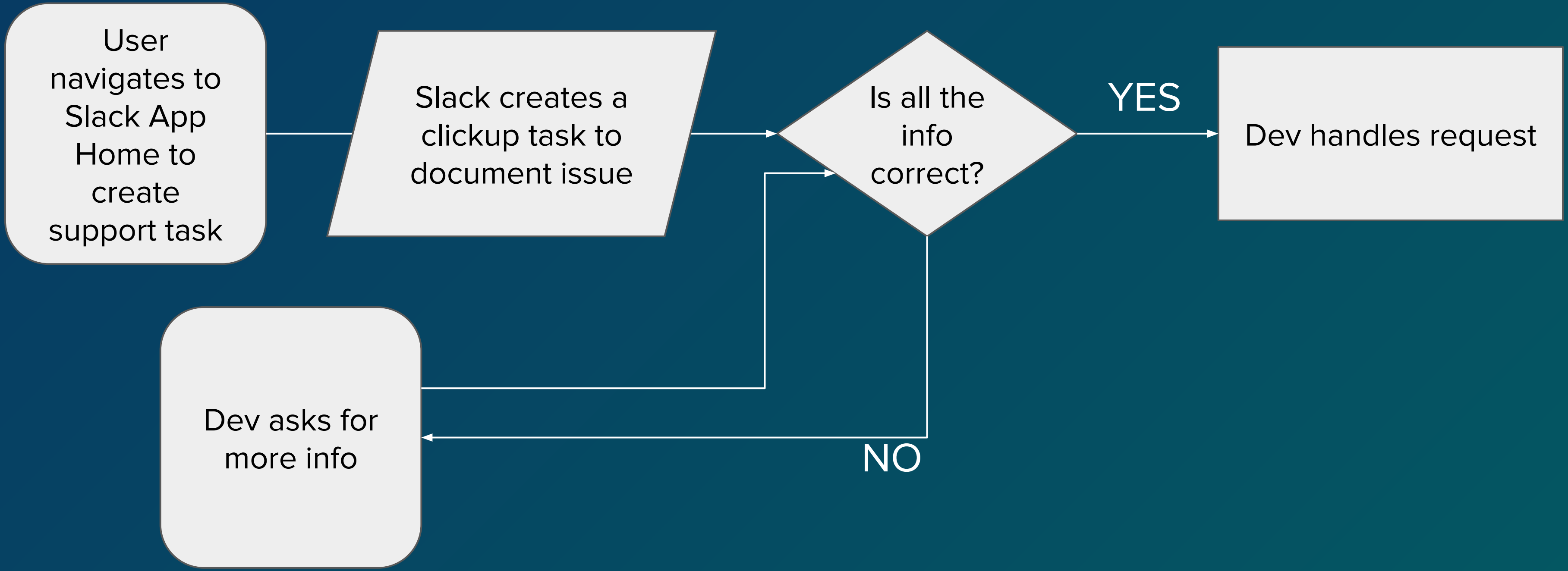
Hello tech team, I need some help v

Stage 1 workflow

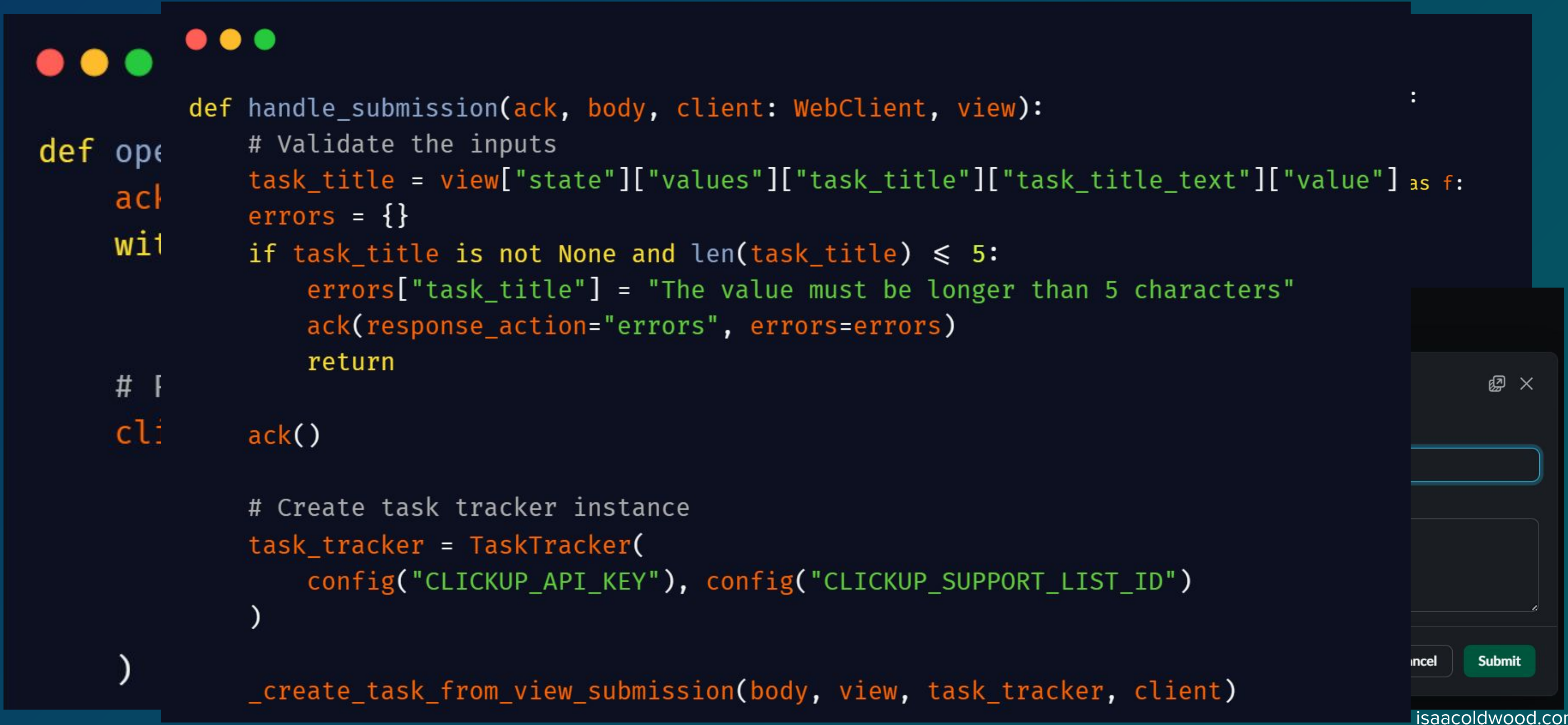

User Sends slack message to #support channel
Is all the info correct?
YES
Dev handles request
Slack creates a clickup task to document issue (dev still fills in work done)
NO
Dev asks for more info

isaacoldwood.com

Stage 2 workflow


User navigates to Slack App Home to create support task
Slack creates a clickup task to document issue
Is all the info correct?
YES
Dev handles request
NO
Dev asks for more info

Stage 2 Automation

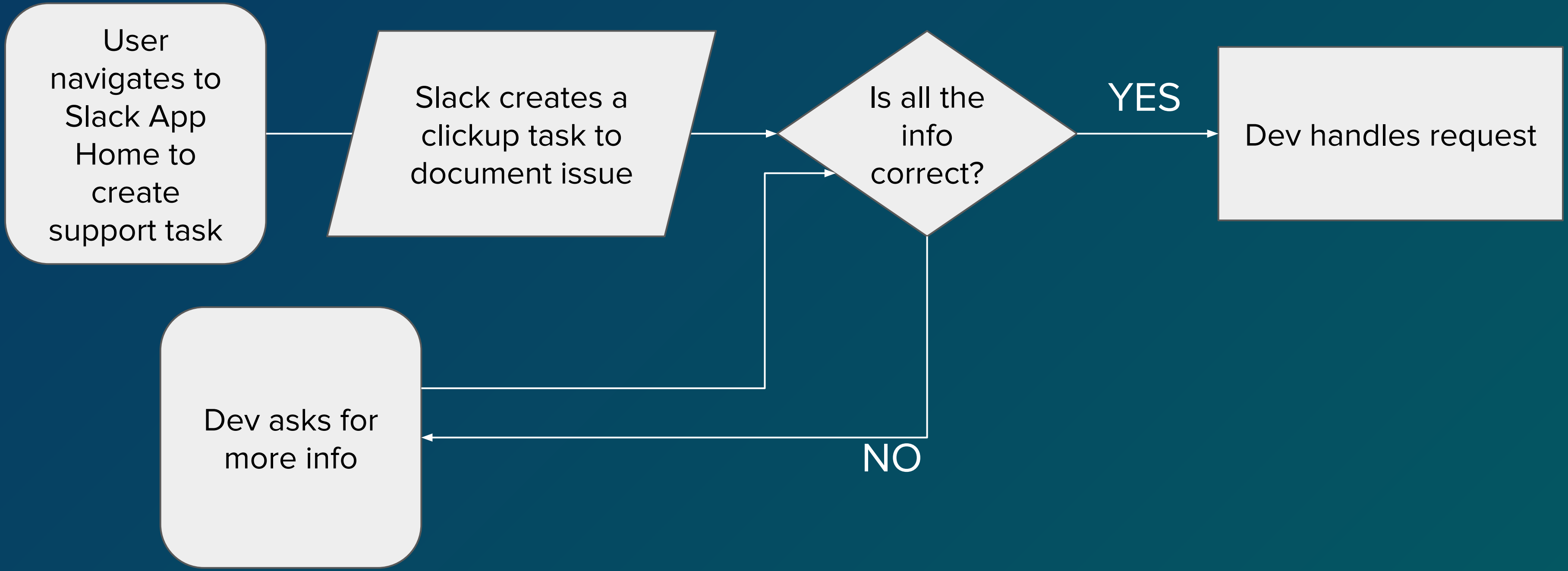


```
def ope
    ack
    wit

    # F
    cli

)
```

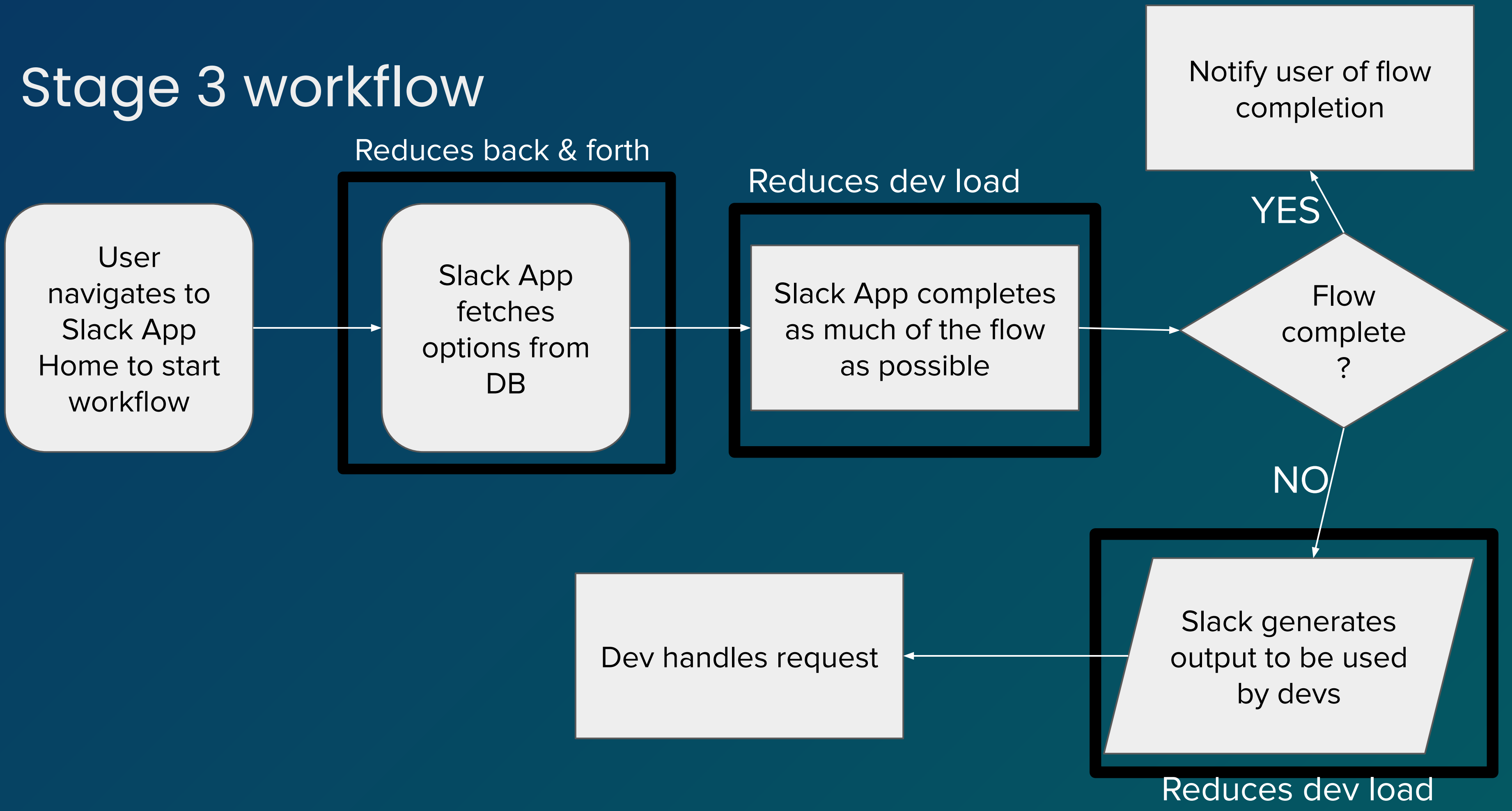
```
                                                                    :

                                                                    as f:
```

```
def handle_submission(ack, body, client: WebClient, view):
    # Validate the inputs
    task_title = view["state"]["values"]["task_title"]["task_title_text"]["value"]
    errors = {}
    if task_title is not None and len(task_title) ≤ 5:
        errors["task_title"] = "The value must be longer than 5 characters"
        ack(response_action="errors", errors=errors)
        return

    ack()

    # Create task tracker instance
    task_tracker = TaskTracker(
        config("CLICKUP_API_KEY"), config("CLICKUP_SUPPORT_LIST_ID")
    )

    _create_task_from_view_submission(body, view, task_tracker, client)
```

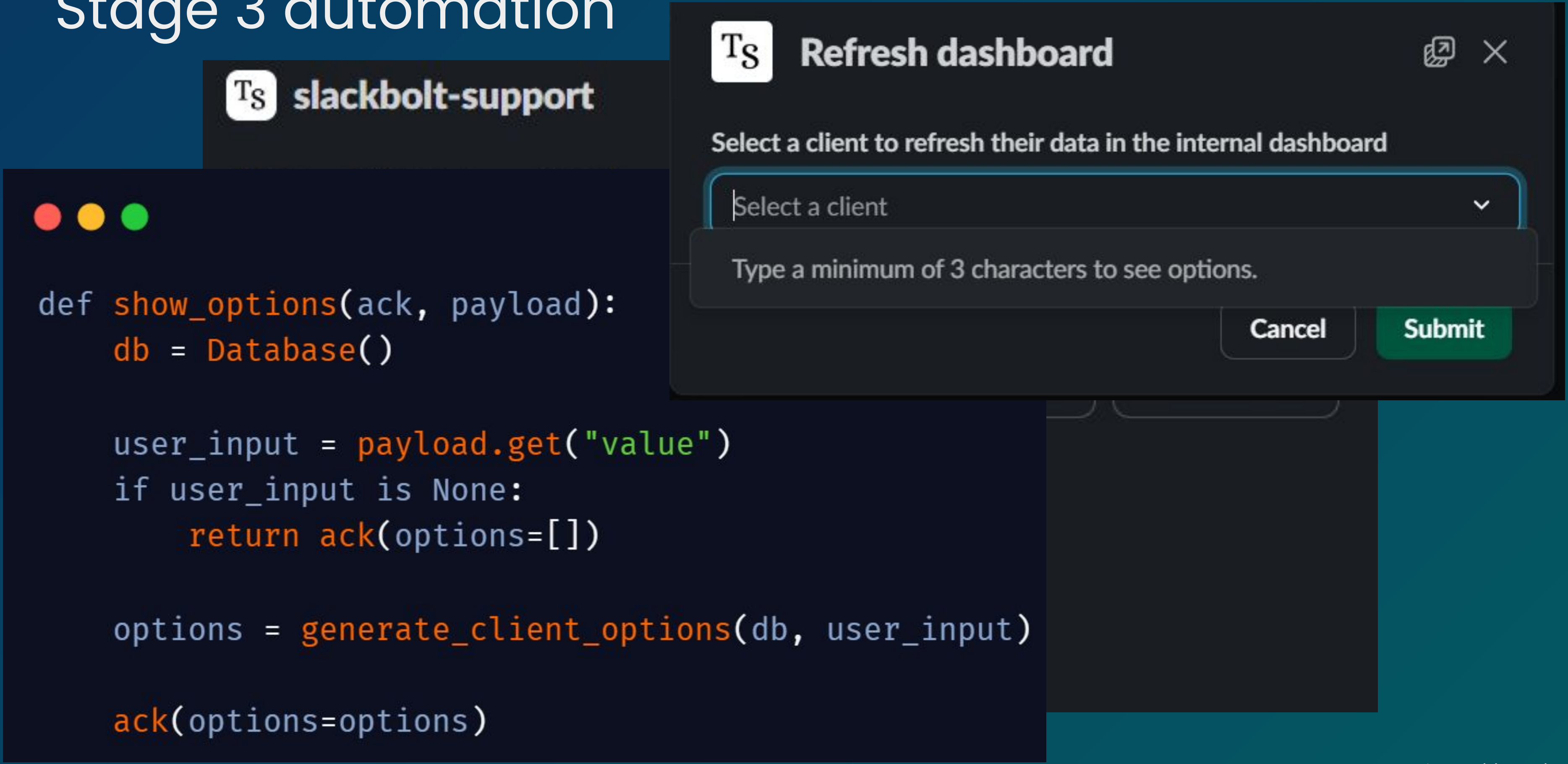
Stage 2 workflow


User navigates to Slack App Home to create support task
Slack creates a clickup task to document issue
Is all the info correct?
YES
Dev handles request
NO
Dev asks for more info

Stage 3 workflow


User navigates to Slack App Home to start workflow
Reduces back & forth
Slack App fetches options from DB
Reduces dev load
Slack App completes as much of the flow as possible
Flow complete ?
YES
Notify user of flow completion
NO
Slack generates output to be used by devs
Reduces dev load
Dev handles request

Stage 3 automation


slackbolt-support

Refresh dashboard

Select a client to refresh their data in the internal dashboard

Select a client

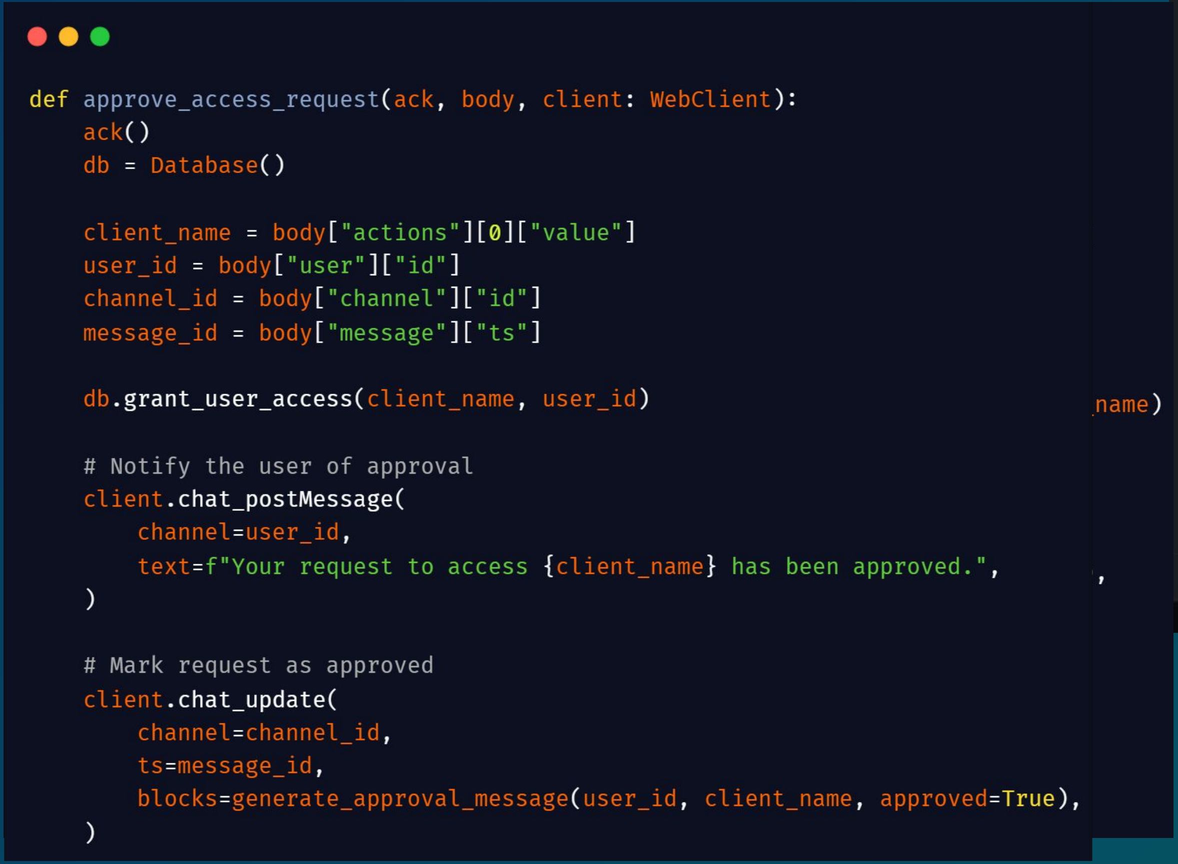
Type a minimum of 3 characters to see options.

Cancel

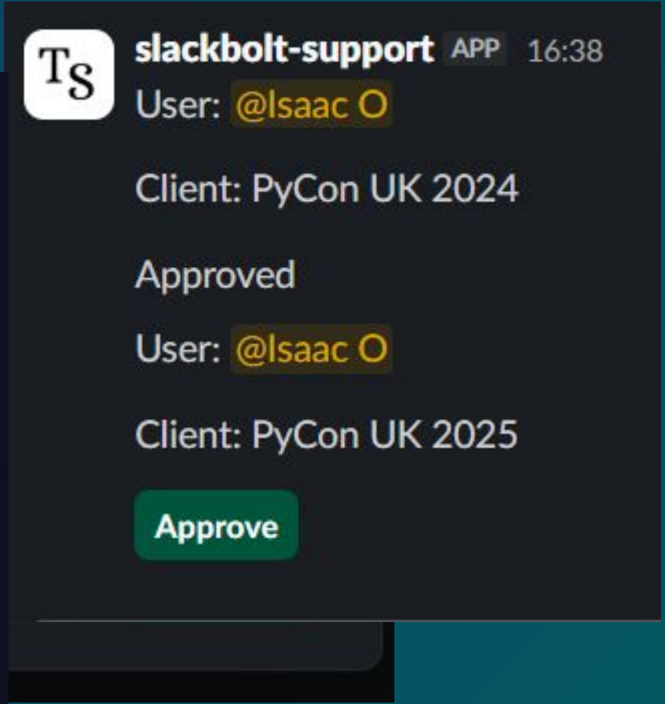
Submit

```
def show_options(ack, payload):
    db = Database()

    user_input = payload.get("value")
    if user_input is None:
        return ack(options=[])

    options = generate_client_options(db, user_input)

    ack(options=options)
```

Stage 3 automation



```python
def approve_access_request(ack, body, client: WebClient):
    ack()
    db = Database()

    client_name = body["actions"][0]["value"]
    user_id = body["user"]["id"]
    channel_id = body["channel"]["id"]
    message_id = body["message"]["ts"]

    db.grant_user_access(client_name, user_id)

    # Notify the user of approval
    client.chat_postMessage(
        channel=user_id,
        text=f"Your request to access {client_name} has been approved.",
    )

    # Mark request as approved
    client.chat_update(
        channel=channel_id,
        ts=message_id,
        blocks=generate_approval_message(user_id, client_name, approved=True),
    )
```



slackbolt-support APP 16:38

User: @Isaac O

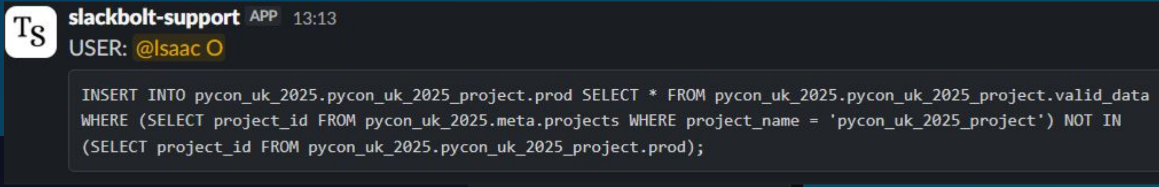
Client: PyCon UK 2024

Approved

User: @Isaac O

Client: PyCon UK 2025

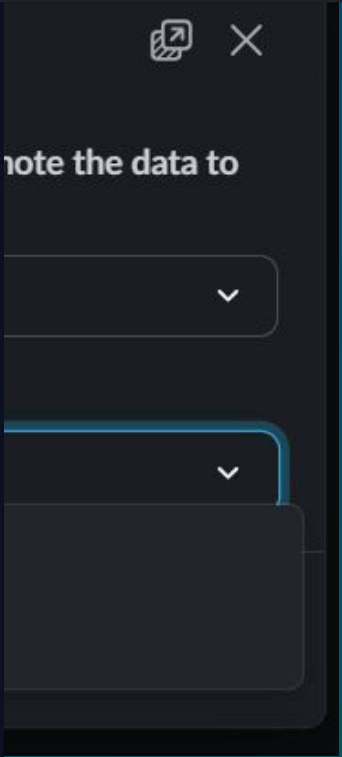
Stage 3 automation

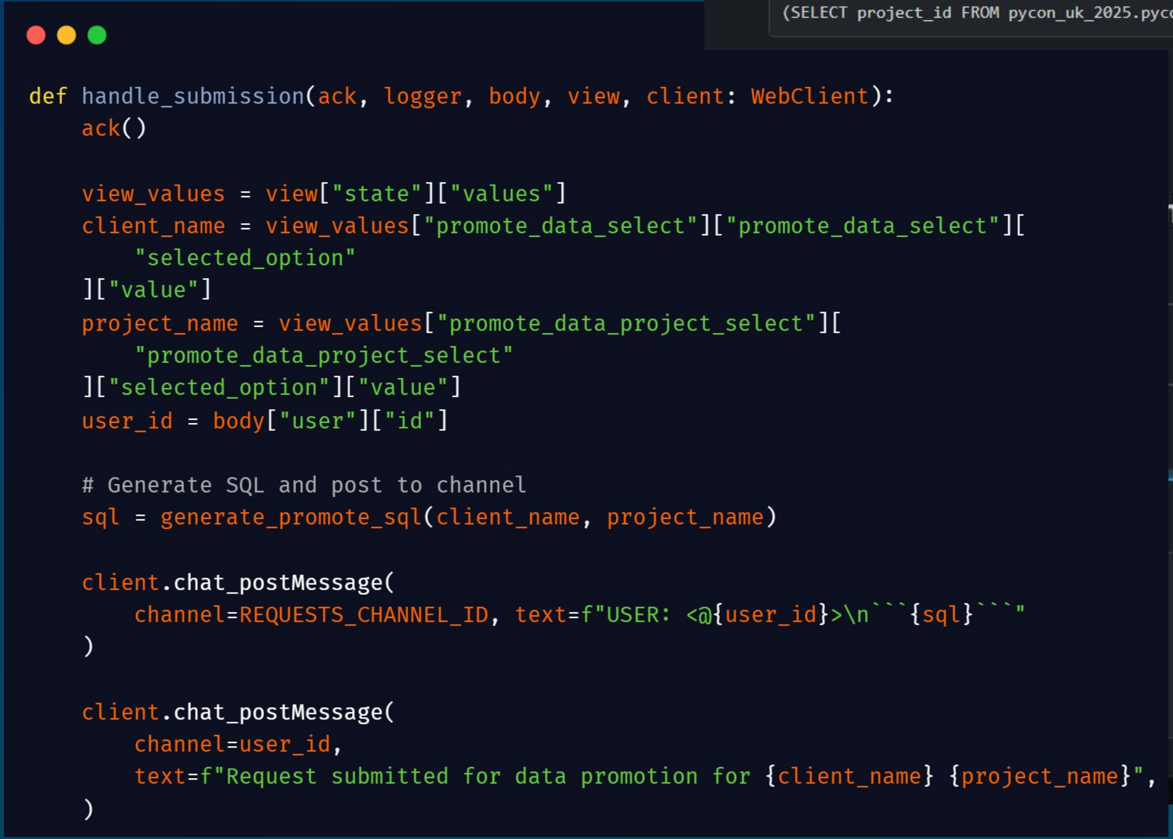


slackbolt-support APP 13:13

USER: @Isaac O

```
INSERT INTO pycon_uk_2025.pycon_uk_2025_project.prod SELECT * FROM pycon_uk_2025.pycon_uk_2025_project.valid_data
WHERE (SELECT project_id FROM pycon_uk_2025.meta.projects WHERE project_name = 'pycon_uk_2025_project') NOT IN
(SELECT project_id FROM pycon_uk_2025.pycon_uk_2025_project.prod);
```





```python
def handle_submission(ack, logger, body, view, client: WebClient):
    ack()

    view_values = view["state"]["values"]
    client_name = view_values["promote_data_select"]["promote_data_select"][
        "selected_option"
    ]["value"]
    project_name = view_values["promote_data_project_select"][
        "promote_data_project_select"
    ]["selected_option"]["value"]
    user_id = body["user"]["id"]

    # Generate SQL and post to channel
    sql = generate_promote_sql(client_name, project_name)

    client.chat_postMessage(
        channel=REQUESTS_CHANNEL_ID, text=f"USER: <@{user_id}>\n```{sql}```"
    )

    client.chat_postMessage(
        channel=user_id,
        text=f"Request submitted for data promotion for {client_name} {project_name}",
    )
```

Join my slack playground to **Try it out!**

Thanks for listening!

Check out the code

https://github.com/IsaacOldwood/slackbolt

Isaac Oldwood

Software Engineer | Event organiser

Add me on LinkedIn