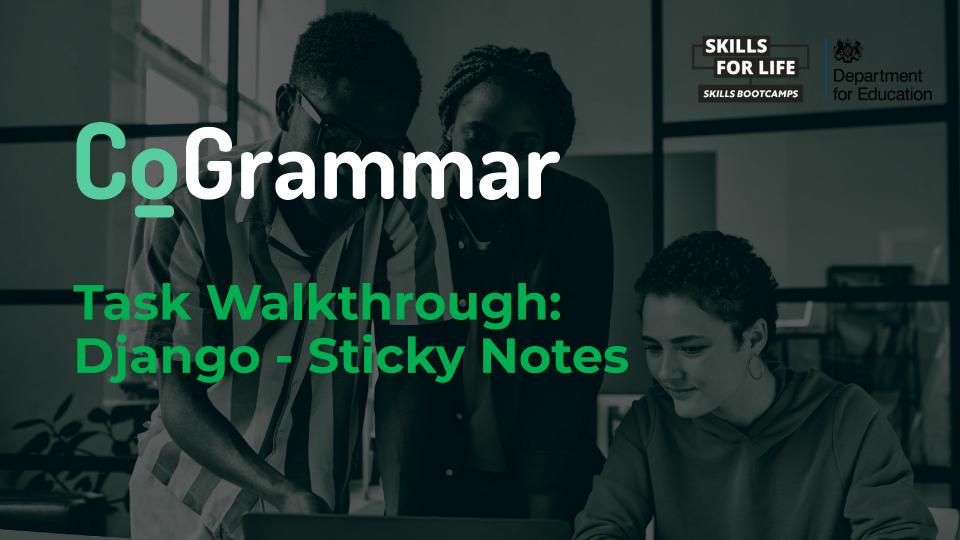
Welcome to this CoGrammar Task Walkthrough: Task 24/26

The session will start shortly...

Questions? Drop them in the chat.







Software Engineering Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
 (Fundamental British Values: Mutual Respect and Tolerance)
- No question is daft or silly ask them!
- There are **Q&A sessions** throughout this session, should you wish to ask any follow-up questions.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: <u>Questions</u>



Software Engineering Session Housekeeping cont.

- For all non-academic questions, please submit a query:
 www.hyperiondev.com/support
- Report a safeguarding incident:
 <u>www.hyperiondev.com/safeguardreporting</u>
- We would love your **feedback** on lectures: **Feedback on Lectures**

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member. or you feel like something isn't right, speak to our safeguarding team:



Ian Wyles Designated Safeguarding Lead



Simone Botes



Nurhaan Snyman



Ronald Munodawafa



Rafig Manan

Scan to report a safeguarding concern



or email the Designated Safeguarding Lead: Ian Wyles safeguarding@hyperiondev.com



Learning Outcomes

- Recall the core components of Django (models, views, templates, URLs).
- Describe the relationship between models, views, and templates in a CRUD app.
- Apply the CRUD functionality and MVT pattern to create a workout APP.
- Design and develop a complete Django application with full CRUD functionality.



Polls

- · Refer to the polls section to vote for you option.
- 1. How confident are you in setting up a Django project and app?
 - a) Not confident at all.
 - b) Somewhat confident.
 - c) Very confident.



Task Walkthrough: Practical Task

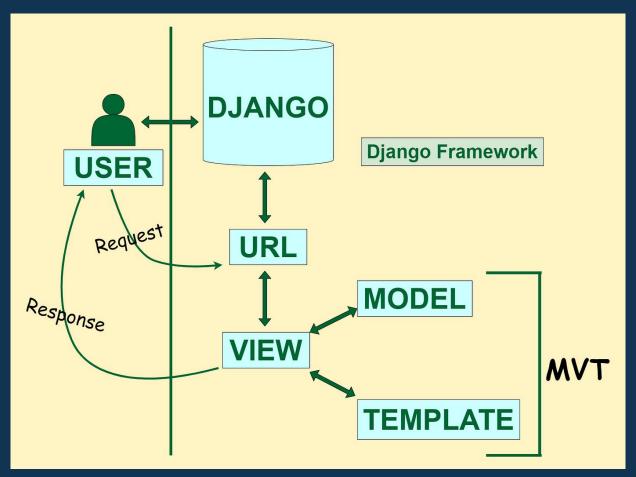


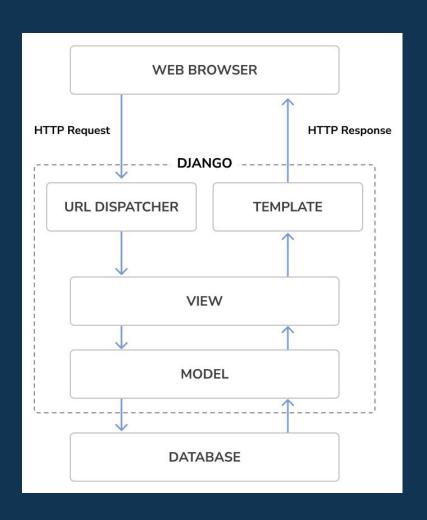
Instructions

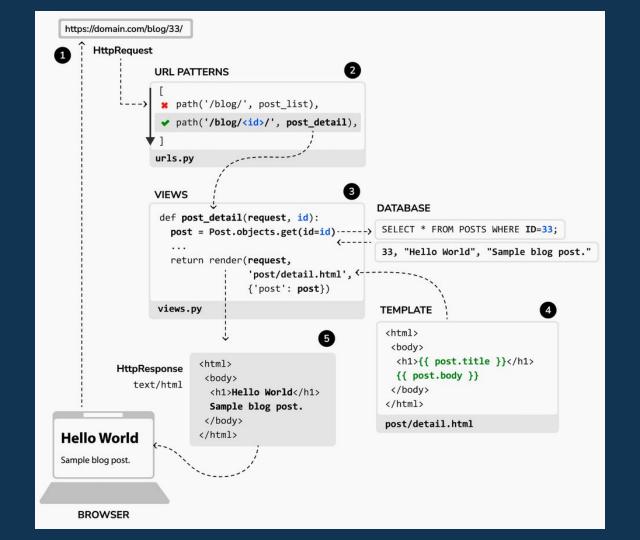
This practical task is designed to test your knowledge of Django by building a sticky notes application. You will be focusing on the creation of a Django project, defining the model, running migrations, and implementing views and forms. You will also focus on the configuration of URL patterns, creating templates, styling with CSS, collecting static files, running migrations, and creating posts for the sticky notes application.



MVT







Summary of the Task Steps

- 1. Set Up the Django Project and App:
 - Create a Django project: django-admin startproject projectname.
- 2. Define Models:
 - o Create a model in models.py (e.g., Post for a blog).
- 3. Create Views:
 - o Write view functions in views.py for CRUD operation.
- 4. Build Templates
 - Create HTML templates in a templates folder.



Polls

- Refer to the polls section to vote for you option.
- Can you explain the MVT (Model-View-Template) pattern in Django?
 - a) No, I'm still unclear.
 - b) I understand the basics but need more practice.
 - c) Yes, I can explain it clearly.



Questions and Answers





Thank you for attending







