

Master 1 RSD - Technologie des Applications Web <i>Examen de Rattrapage</i> 21 Juin 2025 - Durée : 60 Minutes	Nom & Prénom	Gr.	Note

Questions de cours :

1. Étiquetez le schéma ci-contre représentant l'architecture logicielle du Framework Django en utilisant les libellés suivants :

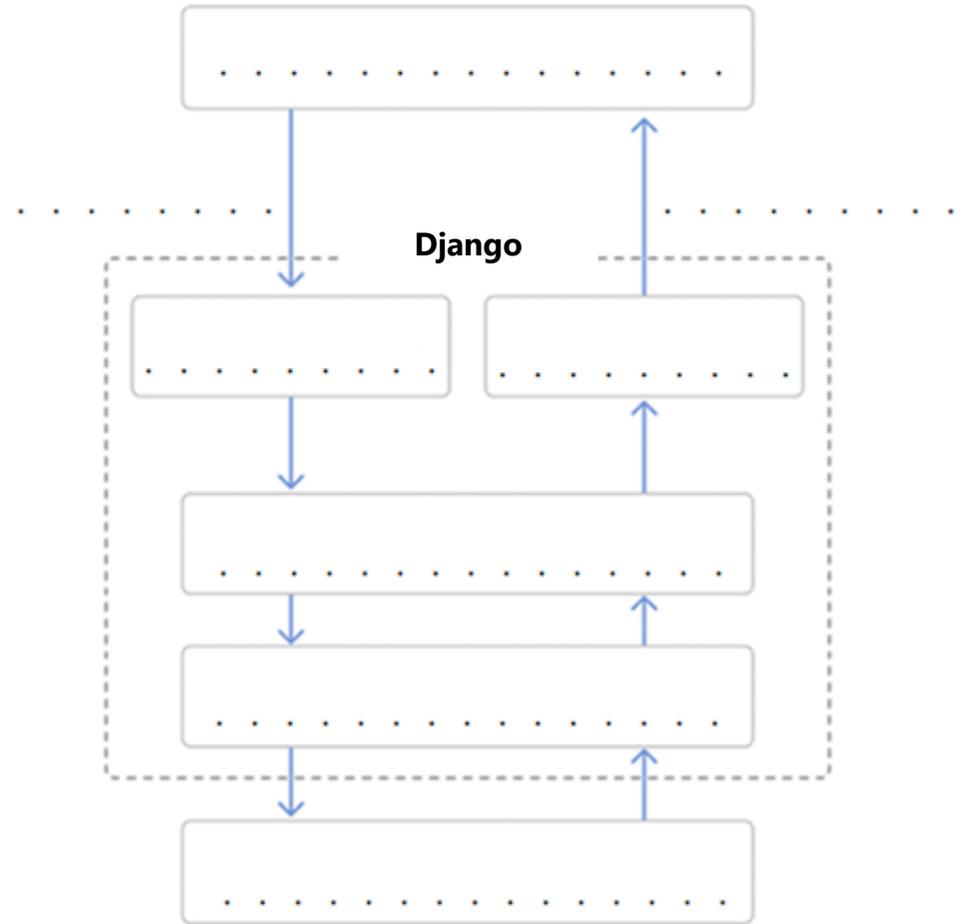
VIEW, TEMPLATE, HTTP-Request, URL-DISPATCHER, MODEL, DATABASE, WEB-BROWSER et HTTP-Response.

(8 x 0.5pt = 4 points)

2. Django suit le modèle MTV (Model-Template-View). Ce schéma est légèrement similaire au célèbre modèle MVC (Model-View-Controller), dans lequel le Template agit comme la vue et le Framework lui-même agit comme le contrôleur.

En se basant sur la question précédente, explique comment Django traite les requêtes et gère le cycle requête/réponse HTTP à travers ses principaux composants. Pour plus de précisions, le traitement des requêtes se résume en six étapes :

(6 x 1pt = 6 points)



Architecture de Django

Etape	Description
1
2
3
4
5
6

Nous allons développer une application Django 5, **sportify**, permettant de gérer des activités sportives (tournois, séances d'entraînement, compétitions) sur une plateforme en ligne. Les utilisateurs pourront consulter les activités, s'inscrire, et obtenir des détails sur chaque événement sportif

1. Les modèles de l'application Sportify sont les suivants :

Modèle	Champ	Détails
Athlete	first_name	Prénom l'athlète (max. 100 caractères).
	last_name	Nom de l'athlète (max. 100 caractères).
	email	Adresse email unique.
	phone	Numéro de téléphone (max. 15 caractères).
Activity	tag	Code unique de l'activité (max. 50 caractères).
	title	Titre de l'activité (max. 200 caractères).
	date	Date de l'activité.
	location	Lieu de l'activité (max. 200 caractères).
	description	Description détaillée de l'activité.
	capacity	Nombre maximal de participants.
Registration	athlete	Clé étrangère vers le modèle Athlete .
	activity	Clé étrangère vers le modèle Activity .
	reg_date	Date d'inscription.

Activity

tag	title	date	location	description	capacity
f5c23	Foot 5 contre 5	2025-02-15	Stade Communal	Match amical pour débutants.	20
mtb10	Tournoi de badminton 2025	2025-03-05	Gymnase Université	Tournoi ouvert à tous les niveaux.	32
rt50	Randonnée dans les montagnes	2025-04-20	Parc National	Excursion guidée pour amateurs.	50

2. Exemples de données

Athlete

first_name	last_name	email	phone
Lamis	Foughali	lamis.f@mail.com	0771234567
Louey	Foughali	louey.f@mail.com	0669876543
Ouail	Foughali	ouail.f@mail.com	0557891234

Registration

email	tag	reg_date
lamis.f@mail.com	f5c23	2025-02-01
lamis.f@mail.com	mtb10	2025-03-01
ouail.f@mail.com	rt50	2025-04-10

3. Complète le fichier `cfg/urls.py` du projet principal `cfg` pour inclure l'application `sportify` (1 pt)

```
# cfg/urls.py

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    .....

]
```

4. Complète le fichier `sportify/urls.py` pour inclure la vue `register` permettant d'enregistrer un athlète. (2 pts)

```
# sportify/urls.py

# Format de l'url : http://127.0.0.1/sportify/register/tag/email

# Exemple : http://127.0.0.1/sportify/register/f5c23/lamis.f@mail.com

from django.urls import path

from .views import .....

urlpatterns = [

    .....

]
```

