Python application wrapping in docker container

Python application Website must have:

1. Data model in backend
2. Frontend in Django/pyflask
3. Business model in between

Deliverables (Post on LinkedIn at the end):

1. Python application wrapped in a docker container
   1. App will be python information system
2. Latex documentation (1 in group) that will have UML, CID, OID, business domain, use cases
3. Video (Everyone) of the working of the project – host on YouTube

Overleaf.com – account – to work with latex documentation

Submit next week (teamname.txt):

1. Teammember
2. “Project concept”
3. URL of Overleaf account

Ideate your project Business domain with at least 3 business use cases

Sure, here’s a basic example of how you might structure the front-end of your Django application to fulfill the requirements mentioned above. This code includes views and templates for displaying a list of restaurants, viewing details about a single restaurant, and making a reservation.

Please note that this is a simplified example and doesn’t include all the necessary parts of a Django application. You’ll need to adjust this code to fit the specifics of your project, and you’ll also need to implement the back-end code and database models.

# views.py

from django.shortcuts import render

from .models import Restaurant, Reservation

def restaurant\_list(request):

restaurants = Restaurant.objects.all()

return render(request, 'restaurant\_list.html', {'restaurants': restaurants})

def restaurant\_detail(request, pk):

restaurant = Restaurant.objects.get(pk=pk)

return render(request, 'restaurant\_detail.html', {'restaurant': restaurant})

def make\_reservation(request, pk):

if request.method == 'POST':

restaurant = Restaurant.objects.get(pk=pk)

reservation = Reservation.objects.create(

customer=request.user,

restaurant=restaurant,

date=request.POST['date'],

time=request.POST['time'],

)

return render(request, 'reservation\_confirmation.html', {'reservation': reservation})

else:

return render(request, 'make\_reservation.html')

<!-- restaurant\_list.html -->

{% for restaurant in restaurants %}

<h2><a href="{% url 'restaurant\_detail' restaurant.pk %}">{{ restaurant.name }}</a></h2>

{% endfor %}

<!-- restaurant\_detail.html -->

<h1>{{ restaurant.name }}</h1>

<p>{{ restaurant.description }}</p>

<a href="{% url 'make\_reservation' restaurant.pk %}">Make a reservation</a>

<!-- make\_reservation.html -->

<form method="POST">

{% csrf\_token %}

<label for="date">Date:</label>

<input type="date" id="date" name="date">

<label for="time">Time:</label>

<input type="time" id="time" name="time">

<button type="submit">Make reservation</button>

</form>

<!-- reservation\_confirmation.html -->

<h1>Reservation confirmed!</h1>

<p>You've made a reservation at {{ reservation.restaurant.name }} for {{ reservation.date }} at {{ reservation.time }}.</p>

Remember to replace Restaurant and Reservation with your actual model names, and to add any additional fields or features as necessary. Good luck with your project! 😊