Performance And Other considerations:

I decided to use NIO since I'm making http calls to an external server. I'm using Mono/Flux and WebClient since they provide NIO functionality. I'm using redis for caching and the CacheManager interface in spring. I decided to create a generic interface for the external weather service since this makes it easier to switch weather services if need be. The interface is called ExternalWeatherMethods which is what OpenWeatherMethodsImpl implements.

Naming conventions:
Controllers: have the name "controller" in them and reside in the "controllers" directory.
The convention is {name}Controller

Services: I generally name services in and {name}Methods and {name}MethodsImp.
Model: no db no models
DTO: Classes used to transfer data from layer A to B are name {name}DTO the common dto are in a common directory if they are shared.

Documentation:
I'm using swagger annotations on the controllers

Client Relationship with server:
The server provides restful end points that are consumed by a react SPA. This allow for the separation of presentation and functionality.