Coding Assignment

Create a simple inventory management system that does the following:

a. Create an app called "inventory" that will handle showing item listings and viewing each individual item. The model should contain at least the following data:

Inventory
id
name (char)
description (char)
note (text)
stock (int)
availability (bool)
supplier (FK)

Supplier
id
name (char)

The database of choice can be anything you want, can use SQLite for simplicity's sake. Use constraint where makes sense.

b. Create a view at "/inventory" to display the list of inventories and their suppliers. Data should be retrieved from a separate Django Rest Framework (DRF) API endpoint available at "/api/inventory". No pagination is needed for either of the views. The API endpoint should have filter support to search by at least name using a URL query parameter, e.g. "?name=<query>". On the list view, you should only display the name, supplier name and availability. Optimise database queries where necessary.

c. Create a view at "/inventory/<id>" to display a selected item from the list page using only views and template rendering to show the item. Show all information of the item including description, note, stock, supplier, etc. Use a static image as a thumbnail placeholder.

d. Add the inventory model to the Django's built-in admin panel to enable CRUD management of inventory (Add, update, remove stock). Create a superuser account to access the admin panel.

e. Write simple unit test cases to test the functionality you created. The following criteria should be checked:
    - ☐ The "/inventory"page returns 200 OK status
    - ☐ The "/api/inventory" page returns 200 OK status
    - ☐ The "/inventory/<id>" page returns 200 OK status