**Design patterns:**

1. Single page application: Most of the interactions happen in a single HTML page.  
   Navigating to different pages happen dynamically on the client side.  
   This uses React router.  
   This does not require full page reloads.
2. Model View Controller (MVC): This is not a strict MVC architecture. But it works on separation of concerns. The core concept of managing data (Model), rendering (view) and handling logic (controller) are still relevant.  
   Examples:
   1. *Customer,* and *UserProfile* classes in **models.py** contain data.
   2. *CustomerViewSet* in **viewsets.py** process incoming requests. This is relevant to controller.
   3. *Login* and *Profile.tsx* handle the rendering of UI. This is relevant to View.
3. Frontend backend separation: React handles the frontend and Django APIs handle the backend.

**Flow Chart**

