

Hepsiburada Front-End Developer Çalışması

Tanım:

Hepsiburada için bir arama sayfası geliştiriniz. Sayfada olması gereken tasarım ve fonksiyonlar aşağıda belirtilmiştir.

Tasarım:

<https://www.figma.com/file/Wr7Usgaisuz4BPiWM8kuly/Hepsiburada-Storefront-React-Challenge?node-id=1%3A267>

Arama

- Arama kutusuna iki karakter girildikten sonra arama yapılmaya başlanmalıdır. Arama cümlesinin iki karakterden az olduğu durumlarda arama gerçekleşmemelidir.
- Arama ürün ismine göre yapılmalıdır ve listeleme ekranında ilgili arama terimini içeren ürünler göstermelidir.

Filtreleme

- Renk ve marka filtrelerinin içerikleri, oluşturduğunuz ürün verisinden beslenmeli ve içerisinde bulunan tüm renk ve markalar burada listelenmelidir. Aynı zamanda kaç adet olduğunu yazmalıdır.
- Kullanıcı bir marka veya renk filtrelediğinde listeleme kısmında ilgili ürünler gösterilmelidir ve filtreler getirilen ürünlere göre tekrar hesaplanmalıdır. Eğer daha önce bir sıralama yapılmışsa, bu tercih korunmalıdır. Seçilmiş bir filtre üzerine tıklandığında o filtrenin kaldırılması gerekmektedir.
- Sıralama filtresi ise mevcut listelenen ürünler üzerinde uygulanmalıdır. Tasarımda, sol tarafta ve yukarıda olmak üzere 2 tane sıralama componenti bulunmaktadır. Herhangi birinden seçim yapıldığında, diğerinin active state'i güncellenmelidir. (Örneğin sol taraftan bir sıralama filtresi seçildiğinde, yukarıda bulunan select box görünümündeki component'in active state'i değişmelidir)
 - En Düşük Fiyat
 - En Yüksek Fiyat
 - En Yeniler (A>Z)
 - En Yeniler (Z>A)

Sepet

- Sepet üzerine üründeki sepet sayısı gösterilmelidir. sepete ürün eklendiğinde veya silindiğinde, bu sayı güncellenmelidir.

- Aynı ürün birden fazla sepete eklenmemelidir. Eğer bir ürün daha önce sepete eklenmişse, o ürünün sepete ekle butonu pasif hale gelmelidir.
- Sepet componenti üzerine gelindiğinde sepete eklenen ürünler gösterilmelidir ve sepete eklenme tarihine göre sıralanmalıdır. Ürün sepetten silindiğine bir uyarı ve onay mesajı çıkmalıdır ve kullanıcı onayladıktan sonra ürün sepetten silinmelidir.

Sayfalama

- Liste sayfasında ürünlerin sayısı 12'den fazla ise sayfalama ile sayfalarabölünmeli ve bir sayfada 12 ürün olmalıdır.
- Uygulama ilk açıldığında, oluşturduğunuz veri özelinde, sayfalama çalışabilirolmalıdır. Bunun için en az 36 tane ürün oluşturmalsınız. Sayfalama ürün sayısına göre dinamik bir şekilde çalışmalıdır.

Ürün ve sepet verisi:

- Ürünlerin veri modeli, tasarım detayları ve doküman analiz edildikten sonra adaylar tarafında tarafından oluşturulmalıdır.
- Ürün verisi aday tarafından oluşturulmalıdır ve uygulama ilk çalıştığında hazır olmalıdır. Bonus çalışmayı yapmayacak adaylar ürün verisini local storage'da, bonus çalışmayı yapacak adaylar ürün verisini Backend API'de in-memory olarak tutmalıdırlar. Sepet verisi her zaman local storage'da tutulmalıdır.

Çalışma değerlendirilirken dikkat edilecek kriterler:

- Yukarıda belirtilen kriterlerin sağlanması zorunludur.
- Çalışma, **React, Redux** (veya **Context API**) kütüphaneleri kullanılarak geliştirilmelidir.
- Çalışma için yapılan geliştirmelerin unit test'leri yazılmalıdır. Unit test içermeyen projeler olumsuz değerlendirilecektir.
- Yazılan kodun kalitesi ve okunabilirliği önemlidir.
- Çalışmayı geliştirirken, CSS veya React (UI) Kütüphanesi kullanılmamalıdır.
- Uygulama ilk açıldığında, ürün listesi ve sepet ile ilgili sizin oluşturacağınız veriler, Local Storage katmanına kayıt edilmelidir. (**Bonus** çalışmasını yapmak isteyen adaylar, Bonus başlığı altındaki ürün listesi ile ilgili gereksinimleri inceleyebilirler.)
- Geliştirilen uygulama ya da uygulamaların container'da çalışabilir olması(containerization) ve buna dökümantasyonda yer verilmesi, değerlendirme sırasında ekstra katkı sağlayacaktır.
- Uygulamanın gereksinimleri, derlenmesi, çalıştırılması ve varsa varsayımlarınızı belirtmek için bir dökümantasyon hazırlanmalıdır.

Bonus:

- Client tarafta geliştirilen uygulamanın arama, ürün filtreleri, sıralama ve ürün listeleme işlemlerini gerçekleştirmek için bir backend API geliştirilip, client uygulama ile geliştirilen backend API ile konuşturulmalıdır.
- Geliştirilecek olan backend API, Node.js ya da tercih edilen başka bir programlama dilinde geliştirilebilir.
- Ürün verisi geliştirilen backend API'de **in-memory** olarak tutulmalıdır.
- Sepet verisi local storage'da tutulmaya devam etmelidir. Gerektiği durumda sepetteki ürünlerin detay bilgisi backend API tarafından sağlanabilir.