REACT

1.

create-react-app və vite js dən istifadə edərək iki react projecti yaradın.İkisinidə eyni git reposuna add edib push edin.Folderlərdəki əlavə fileları silin.

2.

Bir List componentiniz olsun və bir item componentiniz olsun.List componentinizdə bir table yaradırsız.List componentinizdə 10 objectdən ibarət bir arrayiniz olur və bu arraydan istifadə edərək hər biri üçün bir item componenti generate edərək table üçün bir row yaradırsınız Hər bir rowda delete buttonu olur delete click edərkən o element arraydan silinir

3.

API - https://northwind.vercel.app/api/products

Header,Footer,Table və İtem componentləriniz olsun.Apidən dataları get edib Uİ-da göstərin hər birinin yanında delete buttonu olsun.Buttona click etdikdə axios üzərindən delete sorğusu göndərin və ui-dan həmin element silinsin.Table-ın üstündə formunuz olsun dataları doldurub göndərdikdə post sorğusu getsin və yeni product yaranıb listə add olunsun.

4.

Tapşırıq əlavədə verilən linkdəki book searchuygulaması olacaq. Design fərqli ola bilər. Responsivelik tələb **olunmur**. Datalar verilən API'dən alınacaq. Əlavədə API'nin documantation'ı əlavədə **verilir**. Projectdə kitabların preview'u üçün **yönləndirmə** və detailini göstərən **modal** olmalıdır. Modal hazır kitabxanalardan **götürülməməlidir**.Projecti deploy edib linki yerləşdirin. Project **private** github reposuna göndəriləcək.

Task Görünüşü - https://dadabooksearch.netlify.app/

APİ - https://developers.google.com/books/docs/v1/using  
  
  
ÖNƏMLİ NOT!  
1. Gec göndərilən tapşırıq nəzərə alınmayacaq.  
2. Köçürmə aşkarlansa **töhmətlə** sonlanacaq. (iki tərəf üçün də keçərlidir: köçürən, köçürməyə icazə verən)  
3. Linkdə repository yoxdursa və ya repository boşdursa, son commit vaxtı deadline'ı keçmiş isə tapşırıq **göndərilməmiş** kimi nəzərə alınacaq.

5.

Sinifdə yazdığmız taskı davam etdirirsiz.Datalar gələnə qədər ekranın ortasında spinner görünsün.Əgər inputdan daxil edilən dəyərə uyğun data tapılmazsa**, data not found** yazılsın.Hər kartın üzərində wishlistə əlavə edəcək icon olsun və həmin icona klik etdikdə wishlistə əlavə olunsun. Headerdə wislistdə olan filmlərin sayı görünsün.Üzərinə klik etdikdə wishlist açılsın və oradakı filmlər görünsün.Eyni zamanda filmi wishlistdən silmək də olsun. Əgər wishlistdə film yoxdursa **wishlist is empyt yazılsın.** Wishlistə əlavə edilmə localstoragelə olacaq.

6.

 Routing bu şəkildə olur:

 Main tərəfdə Products, ProductDetail, Home və Favorites səhifələriniz olur əlavə olaraq component kimi isə Navbar və Footer component-ləriniz.

 Admin Tərəfdə isə Products, AddProduct, EditProduct səhifələriniz olur və eyni şəkildə Admin tərəf üçün bir Navbar component-iniz olur.

 Admin Panel-dəki Navbar-da Products və Add Product Link-ləri olmalıdır (Main Site-ın linkini də yerləşdirə bilərsiniz)

 Main Site tərəfdə isə Home, Products, Admin və Favorites linkləri olmalıdır.

 Main tərəfdə Home Page-də sadəcə ortada bir yazırsınız ki, This is Home Page.  Products səhifəsində isə  MaterialUI-dan card component-ini götürürsünüz və card-larda Product-un datalarını yazırsınız. (Responsiv olmalıdır- Row,Col,Grid component-lerinden istifade ede bilersiniz). Card-lardan yuxarıda search Product input-u olmalıdır və search etdikcə change event-ında name-ə görə yalnız həmin product-lər görünməldir.  Və hər bir product-un adına click etdikdə isə həmin product-nin detail page-i açılmalıdır. Datalar yüklənənə qədər isə loading olmalıdır

Card-larda AddToFavorites button-u olmalıdır ona click etdikdə isə react-hot-toaster-dən alert gəlsin favorites-ə əlavə olunması ilə bağlı və Navbar-da ümumi favorites-dəki işçilərin sayı göstərilsin.Favorites səhifəsində favoriteslər görünməlidir.Adlarına görə search etməlisiz.Onları oradan tək-tək silə bilməliyəm. Həmçinin all clear buttonu olmalıdır və ona click etdikdə hamısı təmizlənməlidir.

Admin Tərəfdə də Products səhifəsində  table-da apidən data-ları qeyd edirsiniz və ada görə filter və pricea görə sort olmalıdır. Və hər birinin qabağında Edit product və Delete product button-ları olmaldır. Delete button-a click etdikdə həmin product silinməlidir həm API-dən həm də UI render olunub, UI-dan da siliməlidir.

Edit Button-a click etidkdə isə Edit səhifəsi açılmalıdır həmin product-nin data-ları ilə dolu şəkildə və edit edib submit etdikdə edit olunub Admin Panel-dəki Products səhifəsinə geri yönləndirilməlidir.

Add Product səhifəsinə keçdikdə isə bir form açılmalıdır və product-nin data-larını əlavə edib Submit etdikdə isə Sweet Alert-dən product added yazılıb Admin Panel-dəki products səhifəsinə geri yönləndirilməlidir.

Add və Edit formlarda formik istifadə edin.

İstənilən css kitabxanasını istifadə etməkdə sərbəstsiz.

Admin tərəfdə də data əlavə olunduqda, silindikdə, edit olunduqda sweetAlert və react-hot-toaster-dən istifadə edin.

7.

Task - Linkdə yerləşdirdiyim template-in home Page və Product Detail Page-ini yazırsınız.Productlar üçün Fake Store API-dən istifadə edin.  
Productu səbətə əlavə etmək mütləq olmalıdır.Hansı Producta Click etsəm onun detail pagei-i açılmalıdır.Basketi page olaraq yaza bilərsiniz.Routinigi yeni artitekturaa uyğun yazırsınız

Task Link - https://supro.arrowtheme.com/ru

8.

useHTTP  adında custom hook yaradırsınız.Parametr olaraq BaseUrl qəbul etsin.

İçərisində getData(),postData(),editData(),deleteData() adında 4 function olsun və hookunuz bu functionları return etsin.

getData heçbir parametr qəbul etməsin və url-dən datanı get edib return etsin.

postData parametr olaraq newData qəbul etsin və url-ə post request atsın.

editData parametr olaraq id və editedData qəbul etsin və url-ə put request atsın.

deleteData parametr olaraq id qəbul etsin və url-ə delete request atsın.

APİ - https://northwind.vercel.app/api/products

Admin və Site olaraq iki yerə bölürsünüz.

Admin tərəfdə APİ-dəki dataları get edib table-a doldurun.Table-da column olaraq edit delete və detail buttonları olsun.Detail Modal olsun.edit Modal olsun və Form Formik və yup ilə olsun.Headerinizdə add Button olsun və Add page açılsın.Add page-də istifadə edəcəyiniz form Formik və yup ilə olsun.

Site tərəfdə Home Page-inizdə dataları get edib hər biri üçün Card generate edin.Cardlar Home Page-ə 8 ədəd düşsün və Paginated olsun.Carda click etdikdə detail page açılsın.Headerinizdə Search inputu olsun.Dataları name üzrə search edin.Dataları price üzrə sort(selectbox olacaq) edin (qiyməti azdan çoxa,qiyməti çoxdan aza)

9.

https://dadanotepad.netlify.app/ - Task Link

Note-lar reduxda saxlanmalıdır.AddNote,deleteNote və search note funksionallıqları olmalıdır.Responsive yazırsınız.Deploy edilməyən task qəbul olunmayacaq.

10.

Template:https://preview.themeforest.net/item/smartbooks-responsive-book-store-ecommerce-bootstrap-4-template/full\_screen\_preview/23493015?\_ga=2.39369440.233775825.1701966198-258132269.1701966198&\_gac=1.44887952.1701967510.EAIaIQobChMI4P6hheP9ggMVGwuiAx2ZXwzREAAYASAAEgKEDvD\_BwE

Pagelər : Home Page,Detail Page,Shop Page(Left Side Bar),Card,CheckOut,Wishlist,Order Complate,Contact - Page-lərin tamamı responsive olmalıdır.(3rd party library istifadə edə bilərsiniz).

Admin:Books(table olacaq),Orders(table olacaq),Dashboard

Funksionallıqlar:

Admin

Books - Yeni kitab əlavə etmək(Kitab əlavə edərkən şəkillərini string olaraq qəbul edin.images=["image1.png","image2.png"]), Kitab edit etmək, kitab silmək , kitabın full detailini görmək(başqa page-də olacaq=>bookDetail)

Orders - Orderləri edit etmək(orderin statusunu dəyişirsiniz((pending,accepted,onCounrier,rejected)

Dashboard - Boş olsa da olar .

Site

    Home

1.Headerdəki search funksional olmalıdır.Headerdə link olaraq Home,Shop,Wishlist,Basket(template-də yoxdur table istifadə edib yazın dərsdə yazdığım kimi),Contact linklərini yerləşdirin.

2.Kitabları category üzrə filter etmək(featured products,new arrivals,most viewed products)

3.Kitabı wishlistə əlavə etmək

4.Kitabı Wishlistdən silmək

5.Kitabı Basket-ə əlavə etmək(həm dropdown həmdə basket page-ində görsənməlidir)

6.Kitabı basketdən silmək

7.Modal Detail

8.Basket və Wishlist əməliyyatları üçün react-hot-toast istifadə edin.

9.Dataları saxlamaq üçün redux istifadə edin.(basket üçün bir slice , wishlist üçün başqa slice yaradın)

    Shop Page

1.Category üzrə filter (qiymət üzrə axtarış üçün iki input qoyun,min price və max price).Category yanındakı ədədlər dinamik olmalıdır(kitabların sayı).

2.Shop page-də olan görüntü növlərindən sadəcə ilkini yazırsınız(3-3 olan)

3.Pagination olmalıdır.

    CheckOut

 1.Formdan userin datalarını və səbətindəki məhsulları götürüb Order create edirsiniz.Order create olunduqdan sonra Order Complate page-inə redirect edin və orderin bütün datalarını göstərin.(useNavigate istifadə edirsiniz)

APİ olaraq json-server istifadə edin.İki endpointiniz olacaq.Orders və Books

11.

Task - Linkdə yerləşdirdiyim template-i Material Uİ componentlərindən istifadə edərək yığın.  
Source Code - <https://github.com/RashadXalil/bf-202>

LINK: https://mui.com/material-ui/getting-started/templates/blog/?authuser=0