

مکتب شریف

اولین بوتکمپ آموزشی - استخدامی ایران



پروژه پایانی

مکتب 75



موضوع پروژه

سایت فروشگاه محصولات، شامل صفحات بازدید کنندگان (مشتریان) و صفحات مدیریت سایت مطابق با فایل ماک آپ ارسالی پیاده سازی شود و طی چند فاز ریلیز شود.

مشخصات فنی پروژه

- یک SPA با create react app برای تمام صفحات سایت و ادمین حاوی روت ها و لی اوت های لازم
- روش استایل دهی و انتخاب Library کاملاً آزاد. توصیه میشود از [reactstrap](#) یا [react-bootstrap](#) نسخه ۲ یا [material-ui](#) استفاده کنید.
- سرور بک اند در همان رپوی فرانت اند، با استفاده از json-server بر مبنای این رپو: <https://github.com/unicornist/maktab47-json-server-backend>
- استفاده از لایبرری axios جهت ارتباط با backend
- استفاده از لایبرری redux جهت مدیریت state
- رپوی github خود را بصورت private با فرمت maktab75-project-name-family بسازید و به اساتید و مربیان جهت بازدید دسترسی بدهید. **پروژه انفرادی است.**
- برای هر فاز، بر مبنای اصول git flow ریلیز صورت بگیرد، باگ های اعلامی توسط مربیان یا اساتید در کارتابل نیز بصورت hotfix اعمال، تست و ریلیز شود. (اطلاعات بیشتر در بخش فازبندی و تحویل)

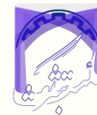
موجودیت های سیستم و فیلدهای آنها

- مشخصات کالاها: عکس، اسم، گروه، سرگروه، توضیحات بصورت html، قیمت، موجودی، تاریخ درج در سایت
- مشخصات گروه کالاها: آیکون، عنوان، عنوان سر گروه
- مشخصات سفارشها: فیلدهای مشخصات کاربر (نام، آدرس، تلفن)، مبلغ کل سبد، زمان درخواست تحویل، زمان ثبت سفارش، زمان پایان تحویل سفارش

صفحات پروژه و قابلیت های آنها

صفحات مدیریت

- لاگین پنل مدیریت: با نام کاربری و رمز کاربران ادمین، دکمه ی ورود
- صفحه ی سفارش ها:
 - صفحه ی لندینگ ورود به پنل این گزینه از منوی ادمین می باشد.
 - لیست سفارش های ثبت شده با pagination و filter سفارش های تحویل شده یا نشده
 - بررسی هر سفارش در مودال و نمایش اطلاعات کامل شامل جدول سبد خرید
 - بابت سفارش های تحویل نشده، دکمه ی ثبت اتمام تحویل
- صفحه مدیریت کالا:
 - لیست کالاها به همراه عکس، اسم و گروه بندی
 - قابلیت حذف هر کالا



○ قابلیت افزودن یا ویرایش در مودال فرم

○ قابلیت انتخاب یا تغییر عکس کالا در مودال فرم

○ قابلیت درج html در فیلد توضیحات

● مدیریت موجودی و قیمت ها:

○ لیست کامل کالاها به همراه موجودی و قیمت هر کالا با pagination

○ با کلیک روی هر عدد موجودی یا هر قیمت کالا آن فیلد به textbox تبدیل شده و عدد قبلی در آن قابل تغییر است (کد محصول تغییر یافته باید در آرایه‌ای نگهداری شود)

○ پس از اعمال تغییرات به روش فوق در لیست، بعد از زدن ذخیره، با استفاده از آرایه‌ای از ردیف‌های تغییر یافته، ajax call های متعددی بصورت سری جهت ویرایش هر محصول اتفاق می‌افتد و با پاسخ موفق آن ردیف از حالت textbox خارج میشود.

○ با زدن دکمه‌ی esc روی هر textbox، به منزله‌ی انصراف از تغییر textbox مخفی و مقدار قبلی نمایش داده میشود (کد محصول از آرایه‌ی تغییرات حذف شود)

○ در صورت عدم کلیک روی عددها و خالی بودن آرایه‌ی کد های محصولات تغییر یافته دکمه‌ی ذخیره غیر فعال باشد.

صفحات مشتریان

● صفحه اصلی: لیست گروه های منتخب ادمین به همراه ۶ محصول از هر گروه

● لیست محصولات (صفحه‌ی گروه محصولات):

○ به ازای هر گروه منحصر بفرد لیست کامل محصولات زیر مجموعه

○ دارای pagination (بصورت a href و قابل باز شدن در تب جدید)

○ طبق سورتینگ پیشفرض جدیدترین محصولات بالاتر هستند

○ ستون سمت راست لیست کامل گروه‌ها و زیر گروه‌ها به همراه آیکن که با کلیک به صفحه لیست آنها می‌رویم. (سرگروه صفحه ندارد)

● صفحه محصول:

○ مشخصات کامل کالا، بخش توضیحات باید html بدرستی نمایش داده شود

○ در صورت عدم موجودی: پیام «انجام موجودی»

○ در صورت موجودی: امکان افزودن کالا با تعداد مد نظر به سبد خرید (validation با در نظر گرفتن موجودی)

● صفحه سبد خرید:

○ لیست کالاهای موجود در سبد، شامل عکس، اسم، قیمت، تعداد، قیمت تعداد

○ قابلیت تغییر تعداد و حذف (validation با در نظر گرفتن موجودی)

○ نمایش جمع کامل

○ دکمه نهایی کردن خرید

● صفحه نهایی کردن خرید:

○ فرم اطلاعات تماس کاربر و تاریخ تحویل (validation)

○ دکمه پرداخت که با صدا زدن یک api در پاسخ لینک صفحه‌ی پرداخت که باید ریدایرکت کنیم

● صفحه‌ی ماگ درگاه: یک صفحه fake جهت ذخیره وضعیت سفارش و ریدایرکت به صفحه نتیجه

● صفحه نتیجه‌ی پرداخت: بر اساس query param کد سفارش، نتیجه را از سرور نمایش میدهم

نکته: صفحات ماک درگاه و نتیجه (برگشت از درگاه) بصورت ریدایرکتی و رفرش شدنی باید کار کنند.

فاز بندی، زمان و نحوه ی تحویل

تحویل بر مبنای git می باشد و هر فاز یک هفته بوده و deadline هر فاز پایان روز جمعه ی هر هفته می باشد. موارد قابل تحویل در هر فاز به همراه تاریخ های تحویل به شرح ذیل می باشد.

1. فاز اول^(۷): ایجاد رپو، ساختار پروژه، نصب لایبرری های مدنظر^۱، ساخت لی اوت ها^۲، روت ها^۳ و رسم نقشه ی ساختار کامپوننتی و فایل کامپوننت ها^۴
 2. فاز دوم^(۱۳): پیاده سازی هدر سایت^۱، پیاده سازی صفحه ی لاگین ادمین^۲، تکمیل db.json با جداول لازمه و داده های نمونه به میزانی که جهت pagination و filtering کافی باشد^۳، صفحات ادمین^۴ و نمایش لیست کامل جداول محصولات^۵ و قیمت/موجودی ها^۶ و سفارش ها^۷ به همراه عملکرد paging و filtering^۸، دکمه های عملیات (بدون عملکرد)^۹
 3. فاز سوم^(۱۳): مودال فرم افزودن/ویرایش کالا^۱، اتصال آن به لیست کالا^۲، حذف کالا^۳، تاگل ویرایش موجودی/قیمت^۴، ذخیره ی تغییرات موجودی/قیمت^۵، کامپوننت کارت محصول^۶، صفحه ی اول سایت^۷، صفحات گروه ها^۸، صفحه جزئیات محصول (بدون عملکرد دکمه ی افزودن به سبد)
 4. فاز چهارم^(۱۳): افزودن محصول به سبد خرید^۱، نمایش سبد خرید^۲، فرم ثبت سفارش^۳، فرایند رفت و برگشت پرداخت با صفحه ی ماک درگاه^۴، صفحه ی نتیجه ی پرداخت^۵، مودال نمایش جزئیات سفارش ادمین در هر دو حالت ارسال شده یا نشده و با عملکرد دکمه ی تغییر وضعیت سفارش^۶ نکات تحویل
- اصلاحیه ی باگ های گزارش شده توسط مربیان در هر هفته، تا انتهای همان فاز می بایست ریلیز شود.
 - هماهنگی جلسات ارائه ی انفرادی به مربیان و اساتید متعاقباً از طریق تلگرام انجام خواهد شد.
 - انجام زودتر فازهای بعدی بلامانع است، اما همچنان طبق همان اصول و نامگذاری ریلیز انجام شود.

پیشنهاد جهت تکمیل اختیاری و تمرینی

- برای هر ماژول از پروژه لایه ی model یا service در نظر بگیرید که مسئول ذخیره و بازیابی اطلاعات با فرمت مناسب است. مثلاً سبد خرید در لایه ی model بجای ajax از localStorage استفاده میکند.
- برای تمام لودینگ های ajax ای گرافیک مناسب loading در بخش مربوطه در ui درج کنید
- ماژول اصلی ajax داشته باشید که از axios استفاده میکند و بجای استفاده مستقیم از axios در model ها از ماژول خود استفاده کنید. همچنین خطاهای 4xx و 5xx را بطور یکجا در ماژول اصلی ajax خود هندل کنید.
- همانند لودینگ برای حالات خطای هر ajax call یک ui مناسب خطا در بخش مربوطه نمایش دهید.
- افزودن فیلترینگ و سورتینگ های بیشتر و کاربردی به صفحات سایت و ادمین.
- برای سمت کاربران قابلیت جستجوی محصولات در هدر سایت قرار دهید و صفحه ی نتایج جستجو داشته باشید.
- برای انتخاب تاریخ از یک لایبرری آماده ی date picker شمس استفاده کنید.
- برای فیلد توضیحات html ای محصولات از یک لایبرری WYSIWYG ادیتور استفاده کنید.
- برای بخش ادمین یک صفحه ی اصلی داشبورد اضافه کنید و نمودار بار چارت تعداد سفارش روزانه و بار چارت جمع مبلغ فروش روزانه و نمودار پای چارت تعداد کالای هر گروه اصلی و گیج درصد کالاهای بدون موجودی نمایش دهید.
- پروژه را بعد از تکمیل نهایی به عنوان نمونه کار خود با تغییر نام رپو بصورت Public منتشر کنید.

- جهت ارائه‌ی راحت این نمونه کار خود به کمک سرویس های رایگان اینترنتی نظیر netlify, heroku (و یا نهایتا یک هاست اشتراکی ارزان قیمت) سایت خود را بارگزاری کنید. (شاید نیاز به اعمال تغییرات در بک‌اند و لایه‌های model ماژول هایتان داشته باشید)
- در صورت بارگزاری سایت در جهت بهینه کردن ساینز باندل نهایی تلاش کنید.
- به پروژه‌ی خود SSR اضافه کنید. بهترین گزینه مهاجرت پروژه به nextjs است.