

# مکتب شریف

اولین بوتکمپ آموزشی - استخدامی ایران



پروژه نهایی

مکتب ۶۰





پروژه نهایی: فروشگاه در فروشگاه

با سلام خدمت همه دوستان عزیز

توضیحات پروژه نهایی به شرح ذیل تقدیم شما دوستان عزیز میگردد .

به صورت خلاصه پروژه تقریباً مثل اسنپ مارکت هست که شامل ۴ بخش می باشد

۱. بخش مدیریت فروشگاه
۲. بخش وب سرویس
۳. مدیر کل سایت
۴. وبلاگ

توضیحات کلی پروژه

بخش مدیریت فروشگاه:

هدف از این بخش ایجاد فروشگاه (های مختلف) می باشد .  
ابتدا کاربری که میخواهد فروشگاه ثبت کند، باید ثبت نام کند و پس از آن به پنل شخصی خود انتقال یابد.

ویژگی های پنل:

- ۱) در پنل خود باید امکان این را داشته باشد که فروشنده جدید را ثبت ، حذف و ویرایش کند .
- ۲) پس از ثبت یک فروشگاه ، فروشنده در وضعیت در حال بررسی قرار میگیرد و کاربر دیگر نمیتواند فروشگاه جدیدی را ثبت کند تا هنگامی که وضعیت فروشگاه ثبت شده به حالت تایید درآید.
- ۳) در پنل باید بتواند لیست فروشگاه های ثبت شده خود را مشاهده کند و قابلیت ویرایش فروشگاه خود را داشته باشد.
- ۴) هر کاربر میتواند چند فروشگاه یا تنها یک فروشگاه داشته باشد. داشتن یک فروشگاه یا چند فروشگاه بسته به استراتژی شماست. (هر کدام راحت تری 😊)
- ۵) توجه کنید در صورتی که وضعیت فروشگاه تایید شده باشد، اگر اطلاعات فروشگاه بر روز رسانی شود، وضعیت فروشگاه به حالت در حال بررسی قرار می گیرد.





- ۵) اگر کاربر بخواهد فروشگاه خود را حذف کند، تنها وضعیت فروشگاه به حالت حذف شده در می آید (اطلاعات آن موجود است در پایگاه داده) و این فرو شگاه و کلیه ملزومات آن دیگر نباید در سایر لیست ها نمایش داده شود (manger میتونه بهتون کمک کنه).
- ۶) کاربر باید بتواند برای هر فروشگاه خود، محصولات خود را ثبت نماید.
- ۷) هر محصول علاوه بر ویژگی هایی مانند نام، توضیحات، عکس و... دارای فیلدی به نام تعداد هست که نشان دهنده تعداد محصول می باشد.
- a) نکته: ممکن است محصول یک یا چند عکس داشته باشد (اختیاری). هر محصول حداقل یک عکس باید داشته باشد.
- ۸) در پنل قسمتی برای مشاهده خرید هایی که توسط مشتری انجام شده است، موجود است (لیست سبد خرید های مربوط به هر فروشگاه)
- a) توجه کنید: در لیست سبد خرید برای هر فروشگاه است و لیست سبد خرید بر اساس تاریخ مرتب شده است.
- ۹) در لیست سبد خرید یک باتنی (یا لینک یا ...) وجود دارد که با زدن بر روی آن جزییات آن سبد خرید را نشان می دهد.
- ۱۰) در این لیست کاربر (فروشگاه دار) میتواند وضعیت سبد خرید را تغییر دهد به عنوان مثال اگر وضعیت سبد خریدی در حال بررسی بوده به وضعیت تایید شده در آورد. در این لیست قابلیت فیلتر کردن بر اساس وضعیت و همچنین سرچ بر اساس بازه تاریخی وجود دارد. (پیشنهاد: برای تغییر وضعیت ها اگر از ajax استفاده کنید که عالی میشه)
- a) وضعیت سبد خرید شامل: تایید شده/ در حال بررسی/ کنسل شده /پرداخت شده
- ۱۱) **اختیاری:** در صورت کنسل شدن سبد خرید توسط فروشگاه دار، پیامی به کاربر داده شود که سبد خرید شما کنسل گردید.
- ۱۲) در پنل فروشگاه کاربر می تواند گزارش فروش هر فروشگاه خود را مشاهده کند (از نمودار استفاده کنید)
- ۱۳) در پنل فروشگاه، جدول (لیستی) وجود دارد که نشان دهنده ی مشتریانی است که از او خرید کرده اند. ستون های این لیست شامل: نام مشتری /تاریخ آخرین خرید/ تعداد خرید انجام شده /مبلغ خریداری شده توسط مشتری / تعداد کالای خریداری شده.
- ۱۴) برای لاگین در قسمت فروشگاه کاربر باید از ایمیل یا نام کاربری استفاده کند (یکی از این ها) **(فاز دوم)**
- a) کاربر باید بتواند با شماره موبایل خود نیز لاگین کند. **(فاز چهارم)**
- i) **اختیاری:** هنگامی که کاربر برای اولین بار لاگین شد در صورتی که شماره موبایل آن تاییده نشده، اقدام به تایید آن کند.



## بخش وب سرویس:

### ۱. وب سرویس ثبت نام (فاز چهارم)

a. ورودی: شماره موبایل

b. خروجی:

i. در صورت تکراری نبودن شماره تلفن ، شماره ثبت شود و کدی ارسال شود.

ii. در صورت تکراری بودن شماره خطای با کد ۴۰۰ همراه با یک پیام مناسب نمایش داده شود.

### ۲. وب سرویس تایید شماره موبایل (فاز چهارم)

a. ورودی: شماره موبایل و کد ارسالی به موبایل

b. خروجی: در صورت صحیح بودن کد ثبت نام انجام شود (توجه کنید که کد ارسالی تنها ۵ دقیقه اعتبار دارد).

در صورتی که کد منقضی یا اشتباه باشد خطا با کد ۴۰۰ و یک پیام جهت نمایش خطا.

### ۳. وب سرویس درخواست کد برای لاگین (فاز چهارم)

a. ورودی: شماره تلفن

b. خروجی: در صورت اکتیو بودن شماره موبایل ارسال کد

i. در غیر این صورت اعلام خطا

### ۴. وب سرویس لاگین: (فاز چهارم)

a. ورودی: شماره موبایل و کد .

b. خروجی: در صورت صحیح بودن شماره و موبایل ، ارسال JWT

i. در غیر این صورت اعلام خطا

### ۵. وب سرویس پروفایل

a. مشاهده

b. ایجاد

c. بروزرسانی ( update )

### ۶. وب سرویس لیست فروشگاه های تایید شده (با فیلتر روی نوع فروشگاه)

a. هر فروشگاه یک type دارد مثل سوپر مارکت ، اسپرت ، پوشاک و ...

### ۷. وب سرویس لیست تایپ فروشگاه ها

### ۸. وب سرویس لیست محصولات براساس فروشگاه

a. به عنوان مثال: ۱/product/?tag=۲mall/

b. قابلیت فیلتر روی

i. تگ

ii. قیمت (حداقل و حداکثر)





- iii. موجود یا ناموجود بودن
- iv. توجه کنید تنها باید کالاهایی را نشان دهید که موجودی آن ها بیشتر از صفر است .
۹. وب سرویس ایجاد سبد خرید
  - a. ورودی: id محصول
  - b. خروجی ایجاد سبد خرید و برگرداندن id سبد خرید
۱۰. وب سرویس اضافه کردن محصول به سبد خرید
۱۱. وب سرویس حذف محصول به سبد خرید
  - a. توجه کنید که هنگام ویرایش / حذف سبد خرید به تعداد موجودی یکی کم یا زیاد می شود
  - b. در صورتی بعد از حذف سبد خرید خالی باشد سبد خرید را حذف کنید.
۱۲. وب سرویس پرداخت سبد خرید
  - a. ورودی: id سبد خرید
  - b. خروجی: تغییر وضعیت سبد خرید به پرداخت شده
۱۳. وب سرویس مشاهده سبدهای خرید باز
۱۴. وب سرویس مشاهده خریدهای قبلی

## مدیر کل سایت

۱. در این قسمت باید ابتدا کل ادمین جانگو را شخصی سازی کنید.
  - a. شامل اضافه کردن فیلتر، سرچ و ... می باشد.
  - b. به جای تم دیفالت یک تم جدید جایگزین کنید (اختیاری)
۲. لیست فروشگاه ها بر اساس وضعیت باید فیلتر شود و وضعیت در این لیست قابلیت ادیت باشد تا ادمین بتواند وضعیت فروشگاه ها را در لیست بروز رسانی کند.
۳. یک اکشن تهیه کنید که با انتخاب آن اکشن تمام وضعیت فروشگاه های انتخابی به حالت تایید انجام گردد.
۴. در لیست محصولات علاوه بر شخصی سازی باید یک ستونی نیز اضافه شود که تصویر محصول نمایش داده شود.

## وبلاگ

وبلاگی که در پروژه قبلی پیاده سازی کردید در این پروژه بیاورید. با همان امکانات موجود.



## فازهای پروژه:

فاز	وظایف	مدت زمان لازم	تاریخ شروع - تاریخ تحویل
فاز اول	بلاگ + مدل + ERD + مدیریت کل سایت (ادمین جنگو)	یک هفته	شروع ۱ دی - تحویل ۹ دی
فاز دوم	مدیریت فروشگاه به جز موارد ۱۲ و ۱۳	یک هفته	شروع ۸ دی - تحویل ۱۷ دی
فاز سوم	وب سرویس (به جز موارد مربوط به موبایل) + موارد ۱۲ و ۱۳ در پنل فروشگاه	یک هفته	شروع ۱۵ دی - تحویل ۲۳ دی
فاز چهارم	لاگین بر اساس موبایل (در پنل فروشگاه و وب سرویس) + رفع باگ	یک هفته	شروع ۲۳ دی - تحویل ۳۰ دی
فاز نهایی و آزمون پایانی	تحویل نهایی و مصاحبه فنی	۱ تا ۵ بهمن	

با توجه به اینکه در مینی پروژه خود یک لاگین و رجیستر طراحی کرده اید (که ممکنه بر اساس نام کاربری یا ایمیل باشه). ابتدا فاز اول، دوم و سوم را بر همان اساس پیاده سازی کنید که کاربر بتواند لاگین، رجیستر کند.

به عنوان مثال اگر وبلاگ شما بر اساس ایمیل لاگین انجام می گردید فاز اول، دوم و سوم بر همان اساس لاگین کنید. اما توجه کنید که مدل مربوط به `user` باید کاستوم گردد و فیلد موبایل اضافه شود تا در فاز چهارم کل پروژه دچار مشکل نگردد.

توجه کنید برای لاگین در فاز سوم توکن `JWT` برگشت داده میشود.

نکته: توجه کنید در فاز سوم کاربر بر اساس نام کاربری یا ایمیل می تواند لاگین کند یا ثبت نام کند و در فاز چهارم بر اساس شماره موبایل.

## چند نکته کلی:

- برای شروع: اول مدل ها را داکيومنت کنید ( `ERD` کامل نمره خواهد داشت ) سپس داخل پروژه پیاده سازی کنید و به ادمین اضافه کنید تا دهننون ساختار درستی داشته باشه.
- دوم: برای پیاده سازی قسمت لاگین و یوزر برای راحتی کار میتونید از `abstracuser` استفاده کنید و آن را `override` کنید.





- سوم: دقت کنید که ما دیگه کار به **username** نداریم در عوض به شماره موبایل نیاز داریم .
- چهارم: برای لاگین شدن ما به دو صورت عمل میکنیم ۱. از طریق شماره موبایل که یک کد ارسال می شود (این قسمت برای مشتری). ۲. از طریق شماره موبایل و پسورد برای لاگین فرد فرو شگاه دارد. پس فیلد پسورد ممکنه برای بعضی یوزرها خالی باشه مهم نیست . خالی باشه
- پنجم: برای نکته چهارم جنگو یک ویژگی خیلی باحال داره. [Customizing authentication in Django](#)
- ششم: پایگاه داده مورد استفاده: **mysql or postgres**
- هفتم: هر جا گیر کردید **print** خیلی میتونه کمکتون کنه حس بگیرید الان کجایید.
- هشتم: از ارور زیاد نترسید. هرچه بیشتر به ارور بخورید باعث میشه ذهنیت شما درست بشه و در ادامه راه کمتر به مشکل برخورد کنید.
- نکته نهم: ممکن است جایی براتون پیش بیاد که **orm** به دردتون نخوره ، اونجا خیلی راحت از **query** استفاده کنید و کوئری خودتون رو بنویسید.
- برای استفاده از گیت دو برنچ لازم است یک برنچ **develop** و یک برنچ اصلی . در برنچ اصلی همیشه آخرین نسخه پایدار شما موجود است. به عنوان مثال در فاز سوم ، باید در برنچ مستر شما فاز یک + فاز دو موجود باشد.
- نکته آخر: تنها مجاز به استفاده از **class view** هستید.
- تحویل پروژه بخصوص سمت وب سرویس ها بر اساس **api** تست کیس ها و داکيومنتهای موجود در **swagger** شما خواهد بود در نتیجه سهم زیادی ( بیش از نصف ) از نمره به نوشتن تست ها و شخصی سازی **swagger** اختصاص خواهد داشت.
- پیشنهاد می کنیم وب سرویس های بخش سبد خرید را از **ViewSet** ها استفاده کنید.

در پناه حق