

استانداردها:

- در فولدر App-themes در فولدر CSS، به نامهای Page-layout-ltr و Page-layout-rtl وجود دارد.
- در فولدر asset در فولدر font، تمام فونتهای مورد احتیاج را، به صورتی که تمام ورژنها وجود داشته باشد، کپی می کنیم. فونتهای متفاوت در فولدرهای جداگانه قرار گیرند.
- در فولدر asset در فولدر plugin، تمام js ها برای تمام ورژنها باشد و مرتبطها در یک فولدر قرار بگیرند.

:Html

- در فایل های html، فایل های CSS ای که در صفحه لینک میشوند در بالای صفحه باشد. اولویت قرارگیری CSS ها به این صورت باشد که اول library ها باشد و سپس فایل های Customise قرار بگیرد.
- در فایل های HTML، فایل های JS ای که در صفحه لینک میشوند در پایین صفحه باشد. اولویت قرارگیری JSها هم مثل فایل های CSS است.
- برای ساخت منو از منوی bootstrap استفاده شود.
- Image ها دارای Width و Height در خود تگ باشد. برای تنظیمات در CSS ویرایش شود.
- Logo حتما در <a> قرار بگیرد و لازم نیست که در h1 قرار بگیرد.

:Css

- در اول صفحه Eric باشد. سپس font نوشته شود. مسیر CSS مشخص باشد. مسیر از Section به پایین نوشته شود. ترجیحا کامنت برای جداسازی Section ها نوشته شود.
- ترتیب CSSها:
 1. کلاسها و آی دی های مشترک و پرکاربرد
 2. کلاسها و آی دی های Section ها به ترتیب در صفحه اصلی
 3. کلاسها و آی دی های صفحه های داخلی
 4. کلاسها و آی دی های در حالت Responsive
- در حالت Responsive ترجیحا تا 1024 px منو داشته باشد ولی اگر امکانش نبود کوچکتر هم میتواند بدون منو باشد.
- در حالت Response ، منو و owlcarousel کامنت شود.
- استاندارد 1200، 992، 768، 480، 320 لزوما رعایت نمیشود. هر زمان که صفحه به هم ریخته شد، درست بشود. تا حد امکان از 992 حالت موبایل پیدا کند.

:Config

- جاوااسکریپت هایی که برای صفحه نوشته میشود، کامنت شود که مربوط به چه قسمتی است.
- در Bundle.config همه CSS ها را لینک می کنیم. این فایل merge میکند.
- در App-code در قسمت config، فایل bundle.config، JSها را لینک می کنیم و merge میشوند.
- اگر جاوا اسکریپت ای بود که حالت Customize داشت، در پایین صفحه اصلی HTML و در تگ Script می نویسیم و اگر در صفحات داخلی جاوا اسکریپتی بود، در Head و در پایین link ها مینویسیم.
- در App-Global Resource در Modules، ماژولهایی که کاربر نمیخواهد را 0 میکنیم.
- سایت را با سایتهای w3c.org و GTmetrix.com چک می کنیم.

- Robot را در زمانی که کار تمام نشده، همه را disallow و در زمان اتمام سایت، فقط اولی را Allow می کنیم. ادمین و پرینت همیشه disallow.
- اگر بخواهیم jsها با هم merge نباشند که بتوانیم مشکل را پیدا کنیم، فایل web.config را باز میکنیم و Debug را true میکنیم. در حالت عادی همیشه False است.

روند کار:

- اگر سایت دارای دو نوع مثلا فارسی و انگلیسی بود، اول یکی را با جزییات درست میکنیم و بعد دومی را انجام می دهیم.
- ورژن انگلیسی را از روی ورژن فارسی کپی کرده، فایلها را از ادمین اضافه می کنیم. و سپس left و right.
- تقریبا فایل فتوشاپ سایت باید با صفحه ای که درست کرده ایم یکی باشد. فقط اگر فونتی یا چیزی از این قبیل فرق داشت مشکلی ندارد.
- صفحه باید در Firefox و Internet Explorer چک شود که حتما درست باشد. در Firefox کاملا باید درست باشد و در IE مرتب باشد طوری که به هم ریختگی نداشته باشد اگر فقط انیمین یا قسمت کوچکی مشکل داشت قابل اغماض است.
- معمولا صفحات داخلی اگر صفحات مهمی باشند باید طرح آنها داده شود. و گرنه به سلیقه خودمان مرتب می کنیم. ولی Header و Footer صفحات داخلی هم طبق طرح است.
- اگر صفحات چند زبانه بود باید برای نوشته ها از () GetViewString استفاده شود.
- ترتیب درست کردن صفحات:
 1. صفحه اصلی
 2. Responsive صفحه اصلی
 3. صفحه داخلی فارسی
 4. صفحه داخلی انگلیسی
 5. Responsive صفحه داخلی
 6. بررسی سایت در Browser های مختلف
- برای Icon معمولا از سایت fontawesome استفاده میشود.
- اگر مجبور بودیم که به جای icon از عکس استفاده کنیم، آنها را باید با هم Merge کنیم و بعد به صورت Background از آن استفاده میکنیم و با استفاده از background-position آن را ادیت میکنیم. به دلیل اینکه gtmatrix به عکسهای خیلی کوچک اشکال میگیرد.
- Image ها باید optimize بشوند. هیچ عکسی نباید دوبار آپلود شود. و اگر دو جا احتیاج است، از لینک آن استفاده شود.
- در جایی که نیاز نیست از کدهای HTML استفاده شود از Before و After استفاده شود.

:Publish

- یک فولدر Publish در drive پروژه ها داریم که تمام سایتهای Publish شده در آن قرار دارد و هر سایت هر زمان publish میشود، در فولدری با تاریخ و ساعت همان زمان قرار میگیرد.
- اگر قرار باشد که تغییری در css یا js رخ بدهد، فقط آن فایل را آپلود میکنیم و نیازی به publish نیست.
- اگر تغییری در HTML باشد، HTML را آپلود نمیکنیم. Bin چون تغییر میکند، bin را آپلود می کنیم و برای آپلود bin ، فایلها را Core و Telerik مثل core را آپلود نمیکنیم.
- فقط در صورتی دوباره Publish میکنیم که HTML تغییر کند.

:Map

- تمام map ها را با استفاده از یک ایمیل برای تمام سایتهای درست میکنیم.

- آدرس را پیدا میکنیم و اسم شرکت موردنظر را مینویسیم و در توضیحات معمولا آدرس را مینویسند.
- حتما map باید به google.map لینک شود.
- دسترسی را از حالت private به public تغییر میدهیم.
- کد HTML را از طریق embedded بر میداریم.