

به نام خدا

طراحی واسط کاربری UI Design

حمید رضا نیرومند

<http://niroomand.ir>

سرفصل‌های پیشنهادی برای این درس

مقدمات و مفاهیم رابط و تجربه کاربری

آشنایی با دوره و اصطلاحات آن

تاریخچه و تعاریف رابط کاربری و تجربه کاربری

اهمیت رابط کاربری و تجربه کاربری

آشنایی با اسکچ Sketch

آشنایی با وایرفریم Wireframe

آشنایی با پروتوتایپ Prototype

طراحی گرافیک Graphic Designer –

طراحی رابط کاربری UI Designer –

طراحی بصری Visual Designer –

محقق تجربه کاربری UX Designer –

طراح تعامل Interaction Designer –

آشنایی با نرم افزارهای UI/UX

تحقیق کاربر یا User research

استراتژی UX

مصاحبه، نظرسنجی و Card sorting

کاربرد پذیری Usability –

قابلیت دسترسی Accessibility -

معماری اطلاعات IA -

روانشناسی کاربر

تست A/B در کاربرد پذیری

تست Guerrilla Usability در کاربرد پذیری

تست First Click در کاربرد پذیری

تست Eye Tracking در کاربرد پذیری

اصول طراحی المان ها در رابط کاربری

روانشناسی رنگ

حواس پنجگانه و تاثیر آنها بر طراحی

اصول طراحی بصری

طراحی منو

اصول طراحی فرم

فواصل و سایه ها در طراحی رابط کاربری

فونت، تایپوگرافی و هم ترازی متون

آیکن ها

طراحی واکنشگرا و گریدبندی

اصول طراحی اپلیکیشن موبایل

نرم افزار Photoshop و جایگاه آن در طراحی رابط کاربری

نرم افزار Illustrator و جایگاه آن در طراحی رابط کاربری

تعامل نرم افزارهای Adobe با یکدیگر

تفاوت نرم افزار Photoshop و Xd

شروع کار با نرم افزار Xd

نصب نرم افزار و معرفی بخش های مختلف آن

معرفی toolbar و اجزای آن

متون، رنگ ها، سایه ها و اعمال تغییرات گرافیکی

move, select, layers, zoom

Interface

Artboard

zoom and panning

Assets

Objects

Symbols

Pen tool

Mask

Prototype

آشنایی با سبک های طراحی

آشنایی با ui kit ها در طراحی

آشنایی با سبک طراحی متریال دیزاین گوگل

آشنایی با سبک های طراحی اپل

آشنایی با سبک های طراحی ماکروسافت

نحوه خروجی گرفتن و اشتراک گذاری

طراحی پروژه کامل – وب سایت

طراحی با نرم افزار Adobe Xd

استفاده از نرم افزار Photoshop برای ویرایش های گرافیکی

استفاده از نرم افزار Illustrator برای طراحی و ویرایش وکتورها و آیکن ها

طراحی پروژه کامل – اپلیکیشن موبایل

طراحی با نرم افزار Adobe Xd

استفاده از نرم افزار Photoshop برای ویرایش های گرافیکی

استفاده از نرم افزار Illustrator برای طراحی و ویرایش وکتورها و آیکن ها

آشنایی با تکنولوژی وب و کاربرد آن در طراحی رابط کاربری

آشنایی و معرفی HTML و CSS ۵ و ۳

مفاهیم ریسپانسیو سازی در وب

آشنایی با فریمورک ها و grid بندی آنها

مفاهیم transition و animation

تعامل طراح رابط کاربری و برنامه نویس وب

خصوصیات یک تیم حرفه ای UI و UX

جمع بندی

ابزارهای آنلاین و غیر آنلاین UI/UX

معرفی نرم افزارهای Figma و Sketch

معرفی نرم افزارهای Balsamiq و Invasion

معرفی کتاب

معرفی نخبگان داخلی و خارجی UI/UX

معرفی وب سایت

مسیر پیش رو

تعریف طراحی واسط کاربری:

User interface (UI) design or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience.

طراحی رابط کاربری (UI) یا مهندسی رابط کاربری، طراحی رابط های کاربری برای ماشین ها و نرم افزارها، مانند کامپیوتر، لوازم خانگی، دستگاه های تلفن همراه، و سایر دستگاه های الکترونیکی با تمرکز بر به حداکثر رساندن کاربردپذیری و تجربه کاربر است.

در طراحی نرم افزار و در کامپیوتر، طراحی واسط کاربری فرایند ساخت واسطه هایی است که از نظر هنری خوشایند باشند. طراحان به دنبال طراحی واسطه هایی هستند که کار با آن ها آسان و خوشایند باشد.

هدف طراحی واسط کاربری این است که تعامل کاربر را برای رسیدن به اهدافش، تا حد ممکن ساده و کارا کند. در حقیقت با یک نوع طراحی کاربرگرا مواجه هستیم.

انواع واسط کاربری:

1 - واسط کاربری گرافیکی یا Graphical User Interface = GUI

a. در این واسط کاربران با نمادهای بصری که بر روی صفحه نمایش کامپیوتر ظاهر می شود تعامل می کنند؛ دسکتاپ یک نمونه از GUI است.

2 - واسطه های کنترل شده با صدا (Voice)

a. کاربران از طریق صدای خود با این واسطه ها تعامل می کنند؛ دستیارهای صوتی مانند Siri و Alexa نمونه هایی از این نوع واسطه ها هستند.

3 - واسطه های تعاملی که با حرکات کاربر (Gestures) کار می کند

a. کاربران از طریق بدن خود با محیط های سه بعدی تعامل می کنند؛ مانند بازی های واقعیت مجازی

(Virtual Reality Games)

طراحی واسط کاربری با طیف گسترده ای از پروژه ها درگیر است؛ از سیستم های کامپیوتری گرفته تا هواپیماهای تجاری و نکته مهم این است که: در عین حال که همه آن ها با تعاملات ساده انسان درگیر هستند اما طراحی هر یک از آن ها دانش و مهارت خاص خود را می طلبد. بنابراین می توان گفت هر طراح واسط کاربری متخصص یک نوع طراحی خاص است. (فرض کنید یک طراح واسط نرم افزار کامپیوتری بخواهد یک واسط برای یک دستگاه صنعتی در یک شرکت طراحی کند. اگر او یک واسط لمسی که احتمالاً از نظر طراح، بسیار

پیشرفته و به روز است برای این دستگاه در نظر بگیرد، چه مشکلی پیش خواهد آمد؟ اپراتور این دستگاه مجبور خواهد بود برای فشردن هر دکمه، دستکش ایمنی خود را در بیاورد تا بتواند دکمه را لمس کند!!)

یک طراح خوب ممکن است از توجه به برخی موارد غیر ضروری خودداری کند تا انجام کار را برای کاربر آسان کند. ممکن است طراح از طراحی گرافیکی و تایپوگرافی برای تأمین کاربردپذیری و همینطور برای زیباسازی ظاهر هنری کار استفاده کند؛ اما باید توجه داشت که همین زیبایی‌های هنری در طراحی ممکن است توانایی کاربر برای انجام کار از طریق واسط کاربری را بهبود دهد و یا **کاهش دهد**. (گاهی استفاده نابه‌جا از یک زیبایی هنری باعث کاهش کارایی کاربر می‌شود؛ نمونه این مشکلات را می‌توان در محصولات شرکت اپل دید. به عنوان نمونه درباره مشکلات طراحی کیبورد و موس اپل از نگاه کاربران تحقیق کنید)

نکته بسیار مهم: فرایند طراحی باید بین دو موضوع کارایی فنی و عناصر بصری توازن ایجاد کند.

The design process must balance technical functionality and visual elements.

و سیستمی بسازد که نه تنها عملیاتی است، بلکه قابل استفاده نیز باشد و خود را با نیازهای کاربر که دائماً در حال تغییر است سازگار کند.

تفاوت طراحی UI و UX

طراحی واسط کاربری (UI Design) که اغلب با طراحی تجربه کاربری (UX Design) اشتباه گرفته می‌شود، بیشتر درباره سطح و نمای کلی یک طراحی است. طراحی رابط کاربری نقش مهمی در ایجاد یک تجربه کاربری خوب دارد. در حالی که واژه طراحی تجربه کاربری به کل فرایند ایجاد تجربه کاربری اشاره دارد. (UI Design زیرمجموعه UX Design و یکی از ابزارهای آن است)

مثال: فرض کنید یک سایت حاوی نقدهای فیلم‌هاست. ممکن است طراحی واسط کاربری آن بسیار زیبا و قوی باشد اما اگر فقط فیلم‌های استودیوهای خاصی را در خود داشته باشد، برای کاربری که به دنبال اطلاعات درباره یک فیلم خاص است ممکن است تجربه کاربری ضعیفی ایجاد کند.

فرایندهای طراحی رابط کاربری

یکی از لازمه‌های طراحی واسط کاربری این است که طراح درک درستی از «نیازهای کاربر» داشته باشد. منظور از نیازها می‌تواند نیازهای بستری باشد که رابط برای آن طراحی می‌شود و یا انتظارات کاربر.

فرایندها و گام‌های مختلفی در طراحی واسط کاربری وجود دارد که برخی از آن‌ها نسبت به دیگری بیشتر مورد تقاضای پروژه‌های مختلف است.

(توجه: از این پس هر گاه واژه «سیستم» به کار برده می‌شود، منظور هر نوع پروژه‌ای است؛ مانند وب‌سایت، برنامه کاربردی و یا یک دستگاه)

1 - جمع‌آوری نیازهای کاربردی: در این گام باید یک لیست از عملکردهای مورد نیاز در

سیستم و نیازمندی‌های بالقوه کاربران جهت رسیدن به اهداف پروژه جمع‌آوری می‌شود.

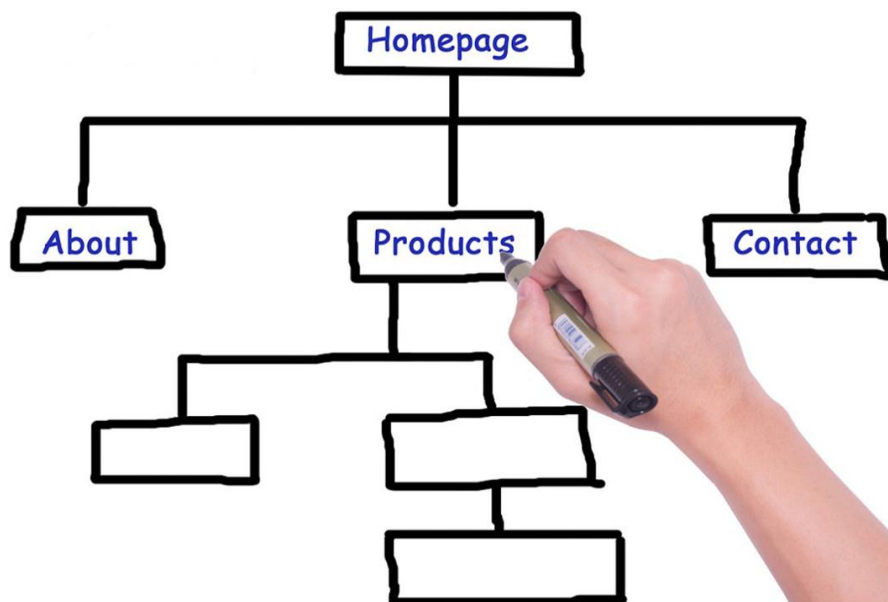
2 - تحلیل وظایف و کاربران: باید کاربرانی که قرار است با سیستم کار کنند (End-userها)

شناسایی شوند و نحوه کار آن‌ها با سیستم نیز تحلیل شود. برخی سؤالات که در این گام باید پاسخ داده شود عبارتند از:

- a. کاربر انتظار دارد که سیستم چه کاری انجام دهد؟
- b. سیستم در چه جایگاهی در برنامه و فعالیت روزانه کاربر قرار دارد.
- c. کاربر نهایی چقدر از نظر فنی، زیرک و دانا است و چه سیستم‌های مشابهی را قبلاً استفاده کرده است؟
- d. رابط کاربری با چه رنگ و لعابی به چشم این کاربر زیبا جلوه می‌کند؟

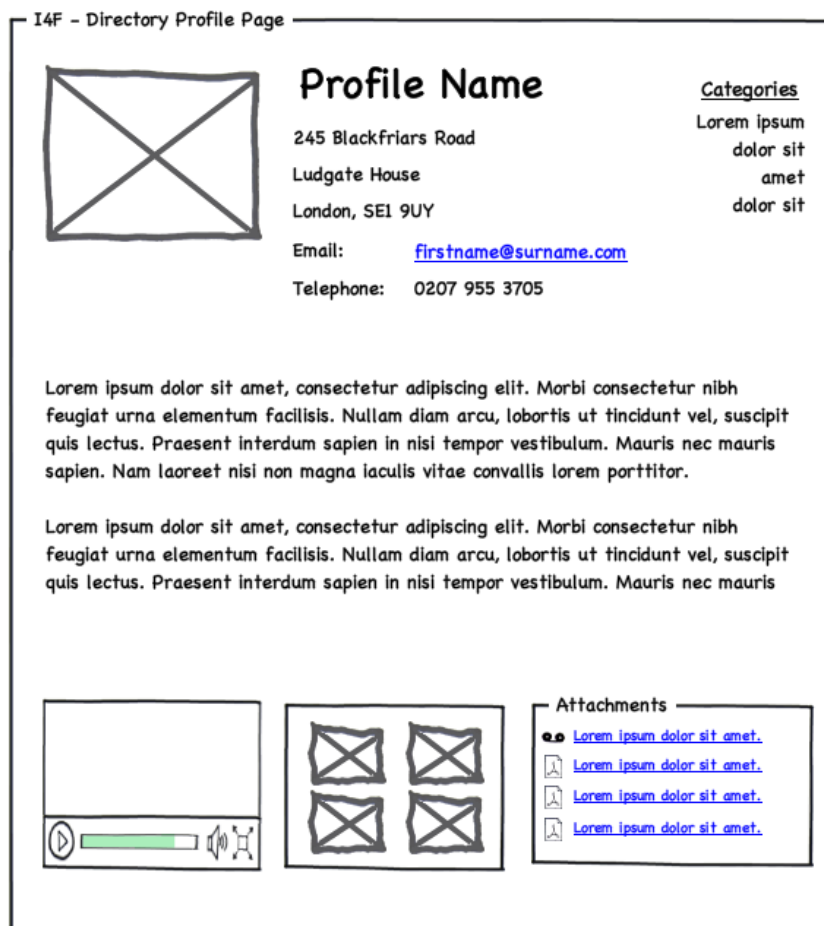
3 - معماری اطلاعات¹: در این گام باید فرایند و جریان اطلاعات سیستم توسعه داده شود. (مثلاً

برای یک وب‌سایت می‌توان یک نمودار درختی از اجزای مختلف طراحی کرد)



¹ Information Architecture

4 - نمونه‌سازی (ماکت‌سازی)^۱: در این گام یک نمونه بصری اولیه^۲ از سیستم طراحی می‌شود؛ چه روی کاغذ و چه روی صفحات لمسی و...



created with Balsamiq Mockups - www.balsamiq.com

5 - ارزیابی کاربردپذیری: از یک نفر ارزیاب بخواهید که رابط کاربری را ارزیابی کند. این کار ارزان‌تر از پیاده‌سازی تست کاربردپذیری (که در ادامه بحث می‌شود) است. این کار می‌تواند در فرایند توسعه نرم‌افزار و از طریق ارزیابی ماکت یا توصیفات سیستم^۳ صورت گیرد. برخی روش‌های ارزیابی کاربردپذیری عبارتند از:

a. Cognitive Walkthrough (ارزیابی گام به گام شناختی): این روش بر روی کارهایی که باید سیستم انجام دهد متمرکز می‌شود (آیا برای یک کاربر جدید انجام کارها با این سیستم آسان است؟) در حالی که مثلاً ارزیابی اکتشافی، سیستم را به صورت یک «کل» می‌بیند. (مثلاً

¹ Prototyping

² Wire-frame

³ System specifications

ممکن است با ارزیابی اکتشافی به این نتیجه برسیم که یک سیستم مدیریت آموزش، برای محیطی در سطح دانشگاه طراحی نشده و برای مؤسسات آموزشی مناسب است اما ارزیابی شناختی تأیید کند که این سیستم برای دانشگاه ما که دانشگاه بزرگی نیست کاربرپذیر است)

b. Heuristic Evaluation (ارزیابی اکتشافی): این روش از یک سری استدلال‌های حس‌ششمی و غیرمستدل برای ارزیابی استفاده می‌کند. (مثال: ممکن است یک نفر بگوید: «این طراحی به دل من نمی‌نشیند» و هیچ دلیل منطقی نیز برای آن نداشته باشد؛ این یک نمونه ارزیابی اکتشافی است)

c. Pluralistic Walkthrough (ارزیابی گام به گام جمعی): در این روش گروهی از افراد انتخاب می‌شوند و سپس یک سناریو در کار با سیستم به آن‌ها محول می‌شود؛ آن‌ها پس از انجام آن سناریو دور هم جمع می‌شوند و درباره مشکلات مربوط به کاربرپذیری سیستم بحث می‌کنند. (ممکن است بسیار از مشکلات در صحبت‌های این افراد کشف شود)

6 - آزمون کاربرپذیری¹: در این گام، نمونه‌های اولیه بر روی یک کاربر واقعی تست می‌شود. این کار معمولاً از طریق تکنیکی به نام «پروتکل بلند فکر کن»² اتفاق می‌افتد که در آن از کاربر خواسته می‌شود که در حین تجربه کار با سیستم درباره افکار خود صحبت کند.

7 - طراحی واسط کاربری گرافیکی: در این گام، رابط کاربری گرافیکی³ نهایی سیستم طراحی می‌شود. طراحی کنترل‌پنل‌ها و نماها، رابط‌های کنترل‌شونده با صدا که با تعاملات صوتی درگیر است و طراحی واسط‌های مبتنی بر حرکات کاربر که با حرکات بدن کاربر درگیر است در این گام اتفاق می‌افتد. طراحی ممکن است بر اساس پژوهش‌ها یا نتایج گام قبلی (آزمون‌ها) انجام شود.

8 - نگهداری نرم‌افزار⁴: بعد از طراحی و توسعه رابط جدید، ممکن است برای رفع باگ‌های نرم‌افزاری یا تغییر امکانات و یا آپگرید کامل سیستم، یک سری نگهداری‌های دوره‌ای لازم باشد.

¹ Usability testing

² Think Aloud Protocol

³ Graphical User Interface (GUI)

⁴ Software Maintenance

نیازمندی‌های طراحی رابط کاربری

ویژگی‌های پویای یک سیستم در قالب مفهوم «نیازمندی‌های تعامل و گفتگو»^۱ در بخش ۱۰ استانداردهای محیط کاری (ارگونومیک) در استاندارد شماره ISO 9241 در ۷ قانون توصیف شده است. این هفت قانون درباره جنبه‌های پویای رابط کاربری و غالباً درباره «احساس» مرتبط با آن بحث می‌کند.

1 - مناسب بودن (Suitability) رابط برای کار: یعنی رابط بتواند به کاربر کمک کند که به طور کارا و مؤثر کار را تمام کند.

2 - خود توصیفی (Self-descriptiveness): یعنی هر گام از رابط باید بلافاصله برای کاربر قابل درک باشد و یا در صورت تقاضای کاربر، توضیحی درباره آن داده شود. تصویر زیر می‌تواند یک نمونه خودتوصیفی باشد که در آن فقط برای یک بار کاربر راهنمایی شده است که با کلیک روی آیات می‌تواند صدای آن را بشنود (پس از یک بار کلیک روی یک آیه، این راهنما نمایش داده نخواهد شد). همینطور دکمه‌های بالا هر کدام در اولین کلیک، یک راهنما از کاربرد آن دکمه نشان خواهند داد.



3 - قابلیت کنترل (Controllability): یعنی کاربر بتواند شرایط تعامل با رابط را آنگونه که هدف او تأمین شود، کنترل کند.

4 - مطابقت با انتظارات کاربر (Conformity): یعنی رابط باید با ویژگی‌های کاربر؛ مانند سطح دانش، تحصیلات، تجربه و قوانین پذیرفته‌شده عمومی، تناسب داشته باشد.

¹ Dialogue

5 - تحمل خطا (Error Tolerance): رابط کاربری باید با وجود خطاهای احتمالی در ورودی‌ها به سیستم، نتایج مورد انتظار را با حداقل تلاش کاربر و به درستی ارائه دهد.

6 - قابلیت شخصی‌سازی (Individualization): رابط باید متناسب با نیازهای هر کار (task)، ترجیحات^۱ هر فرد و مهارت‌های کاربر، قابل ویرایش باشد.

7 - قابلیت آموزش‌دهی (Learning): یعنی رابط باید راهنمایی‌های لازم جهت آموزش به کاربر در نحوه استفاده از سیستم را ارائه کند.

تمرین: در اینترنت جستجو کنید UI Design Evaluation Checklist و یک چک‌لیست مناسب و معتبر بیابید و آن چک‌لیست را در مورد حداقل یک نرم‌افزار تکمیل نمایید.

مفهوم «کاربردپذیری» که در استاندارد ISO 9241 تعریف شده، شامل سه مفهوم می‌شود:

1 - Effectiveness (اثربخشی): به چه میزان اهداف مورد انتظار در استفاده از کل سیستم تأمین شد؟

2 - Efficiency (بهره‌وری): منابعی که باید برای دستیابی به اهداف مورد نظر صرف شود.

3 - Satisfaction (رضایتمندی): به چه میزان کاربر کل سیستم را قابل قبول می‌داند؟

بخش ۱۲ استاندارد ISO 9241 در سه سرفصل درباره نمایش بصری اطلاعات بحث می‌کند:

1 - سازماندهی اطلاعات

a. Arrangement

b. Alignment

c. Grouping

d. Labels

e. Location

2 - اشیاء گرافیکی

a. Abbreviation

b. Color

c. Size

¹ Preferences

Shape .d
Visual Cues .e

3 - تکنیک‌های کدنویسی

a. ؟

این استاندارد هفت ویژگی برای نمایش بصری اطلاعات بیان می‌کند:

- 1 - Clarity (وضوح): محتوای اطلاعات به طور سریع و دقیق منتقل شود.
- 2 - Discriminability (قابلیت تمایز): اطلاعات نمایش داده شده باید به راحتی و با دقت قابل تمایز باشد.
- 3 - Conciseness (مختصر بودن): کاربران نباید با حجم انبوهی از اطلاعات اضافه بمباران شده باشند.
- 4 - Consistency (ثبات): رابط باید دارای یک طراحی واحد (unique) باشد که با انتظارات کاربر تطابق دارد.
- 5 - Detectability (قابلیت تشخیص): توجه کاربر به اطلاعات مورد نیازش هدایت می‌شود.
- 6 - Legibility (خوانایی): اطلاعات باید به سادگی قابل خواندن باشد.
- 7 - Comprehensibility (درک پذیری): معانی باید به طور واضح قابل درک، غیر مبهم، قابل تفسیر و قابل تشخیص باشند.

بخش ۱۳ استاندارد ISO 9241 در این باره بحث می‌کند که اطلاعات مربوط به راهنمایی کاربر (راهنماها) باید به راحتی از اطلاعات دیگری که نمایش داده شده قابل تمایز باشد.

راهنماها می‌تواند از طریق یکی از پنج مورد زیر ارائه شود:

- 1 - پنجره‌های بازشونده روی پنجره فعلی (Prompt): یک نوع پنجره بازشونده راهنما در تصویر زیر مشخص است؛ کاربر با کلیک روی یک دکمه یک راهنما مرتبط با آن دکمه مشاهده کرده است:



دو نوع پنجره بازشونده قابل تصور است:

a. **Implicit Prompt** یا **Generic Prompts**: یعنی یک پنجره راهنما نمایش داده شود که

به طور کلی امکانات سیستم را آموزش می‌دهد.

b. **Explicit Prompt** یا **Specific Prompt**: به صورت خاص بر اساس کلیک کاربر روی

هر یک از امکانات، یک راهنما نمایش می‌دهد و یا یک **Tour** برای گزینه‌های سیستم طراحی شده است.

2 - بازخورد لحظه‌ای نسبت به عملکرد کاربر

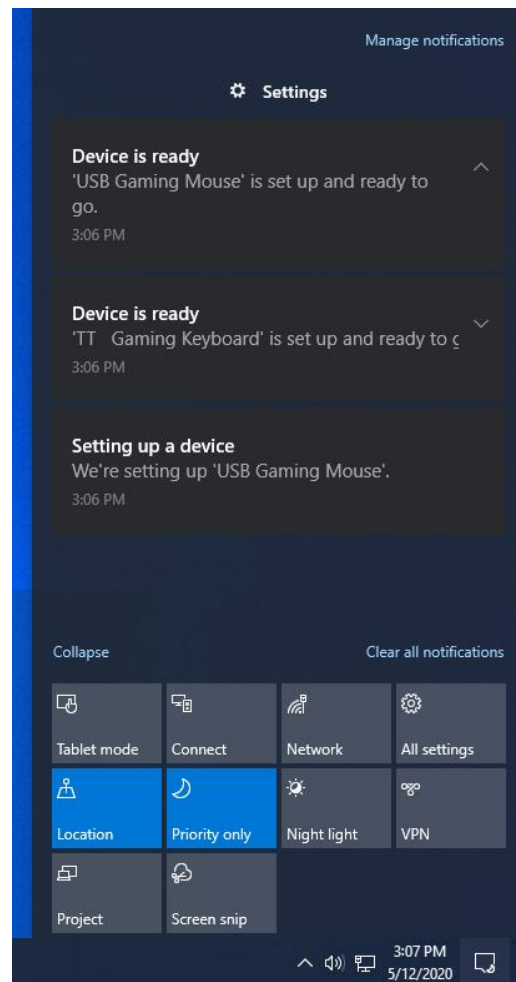
مثال: وقتی کاربر نمره‌ی یک دانشجو را با اعداد فارسی وارد می‌کند، پیامی بدهیم و از او بخواهیم اعداد را به انگلیسی وارد کند.

3 - نمایش لحظه‌ای وضعیت برنامه، اجزای سخت‌افزاری و نرم‌افزاری سیستم و فعالیت‌های کاربر

مثال: نمایش یک آیکون که کیفیت اتصال به سرور را مشخص می‌کند و یا نمایش پیغامی در محیط‌های چت؛ به این معنی که یکی از کاربران در حال تایپ کردن است.

4 - مدیریت خطاها؛ شامل پیشگیری از خطا، اصلاح خطا و فراهم کردن امکان مدیریت خطاها و پیام‌ها توسط کاربر.

مثال: بخش **Action Center** در ویندوز ۷ به بعد می‌تواند یک نمونه خوب از این مورد باشد.



5 - راهنمای آنلاین مرتبط با مشکلات ایجاد شده توسط کاربر یا توسط سیستم

پژوهش‌ها در زمینه طراحی رابط کاربری

طراحی رابط کاربری موضوع تحقیقات قابل توجهی از جمله در مورد زیبایی‌شناسی آن بوده است. استانداردهایی در دهه ۱۹۸۰ برای تعریف کاربردپذیری محصولات نرم‌افزاری ایجاد شده‌اند. یکی از پایه‌های ساختاری، مدل مرجع رابط کاربری IFIP بوده است. این مدل چهار بُعد را برای ساختار رابط کاربری پیشنهاد می‌کند:

- 1 - بُعد ورودی/خروجی (ظاهر = Look)
- 2 - بُعد محاوره‌ای (احساس = Feel)
- 3 - بُعد فنی یا عملکردی (دسترسی به ابزارها و خدمات)
- 4 - بُعد سازماندهی (پشتیبانی از تعاملات و ارتباطات با دیگران)

این مدل تا حد زیادی بر توسعه استاندارد بین المللی ISO 9241 که الزامات طراحی رابط برای قابلیت استفاده را توصیف می کند، تأثیر گذاشته است. تمایل به درک مسائل مربوط به رابط کاربری منجر به ایجاد ابزارهای نمونه سازی سریع رابط کاربری گرافیکی (GUI Rapid Prototyping) شد که ممکن است شبیه سازی قانع کننده ای از برنامه اصلی را ارائه دهند.

لیستی از این ابزارها در مطلب زیر موجود است:

<https://medium.theuxblog.com/11-best-prototyping-tools-for-ui-ux-designers-how-to-choose-the-right-one-c5dc69720c47>

دسترس‌پذیری (Accessibility)

دسترس‌پذیری به این معنی است که صفحات وب را برای همه، حتی افراد دارای معلولیت در دسترس قرار دهید.

Internet IS for everyone, but it won't be unless WE make it so.

Vint Cerf, 7. april 1999

اینترنت برای همه است؛ البته برای همه نخواهد بود مگر اینکه ما آن را برای همه (دسترس‌پذیر) کنیم.

توجه: هنگامی که در رشته کامپیوتر صحبت از معلولیت می‌شود لزوماً به معنی ناتوانی دائمی نیست. برای مثال کسی که اکنون در حین رانندگی است و نمی‌تواند به صفحه نمایش نگاه کند، از نگاه ما یک معلولیت بصری موقتی دارد. یا کسی که در یک جمع شلوغ قرار دارد؛ معلولیت شنوایی موقتی دارد؛ کسی که در جلسه‌ای است و نمی‌تواند صحبت کند؛ معلولیت بیانی موقتی دارد.

بنابراین، مستندات مربوط به طراحی رابط کاربری برای معلولان می‌تواند برای همه کاربران مورد استفاده قرار گیرد.

تاریخچه کوتاهی از Accessibility

1995: Dr. Cynthia

Dr. Cynthia Waddel published [a web design accessibility standard](#) for the City of San Jose's Office of Equality Assurance. One of the requirements in the standard was that all images must have alternative text. This is one rule that is still valid today. Another requirement was that all pages should support text browsers. That is not something we aim for any longer, even though text browsers still exist.

Even though her standard became an important starting point for other well known standards and guidelines, she is better known for the accessibility testing software [Cynthia Says](#).

1996: Bobby

The accessibility testing software Bobby was released a couple of years before Cynthia Says. Bobby worked much the same way many of the current testing tools are working. HTML code was tested against a set of rules, and

the results was a list of issues. Instead of a browser extension, that did not exist back then, you uploaded a local HTML file.

1997: Web Accessibility Initiative (WAI)

The Web Accessibility Initiative started as a W3C project in 1996. This initiative is mostly known for the Web Content Accessibility Guidelines (WCAG). They also provide a variety of strategies and resources for web accessibility.

1999: Web Content Accessibility Guidelines (WCAG) 1.0

Tim Berners-Lee founded the World Wide Web and wanted everyone to have equal access to it. W3C released 14 guidelines and 65 checkpoints about web accessibility. Many of them are still a part of the WCAG. We will get into more details about WCAG in a later module.

2001: Wave

Another accessibility testing tool. First released by Dr. Len Kasday and then taken over by WebAIM in 2003. Still a popular tool by many developers and accessibility experts.

2005: Screen reader VoiceOver

Many blind people use a screen reader, a tool that transforms information on a screen to speech. VoiceOver is a native screen reader for Apple products. Today [VoiceOver is the most popular screen reader for mobile devices](#) and the third most popular on desktop and laptop. VoiceOver was not the first screen reader, not even close. It is a part of this brief web accessibility history, because of the dominant market share on mobile devices. VoiceOver is only for Apple products.

2007: Screen reader NVDA

Another popular screen reader was released as an alternative to expensive screen readers like Jaws. In 2019 it became [the most common primary screen reader for desktop and laptops](#). NVDA is only for Microsoft Windows computers.

2008: WCAG 2.0

The second version of the guidelines was an updated version of WCAG 1.0. It categorized the guidelines into four principles: perceivable, operable, understandable and robust.

2018: WCAG 2.1

This is the most used guidelines today, and the current version. This is what organizations try to conform to. The updated version had new guidelines about touch screens and mobile devices, as well as more guidelines about cognitive disabilities.

نزدیک به ۸ میلیارد نفر در جهان وجود دارد. طبق آمار سازمان جهانی بهداشت، بیش از یک میلیارد نفر معلول وجود دارد. یعنی حدود ۱۵ درصد جمعیت. دسترسی به اطلاعات یک حقوق بشر است.

دسته‌بندی ناتوانی‌ها

ناتوانی‌ها به پنج دسته تقسیم می‌شوند:

- 1 - ناتوانی در شنوایی (Auditory)
a. برای افراد با ناتوانی شنوایی متن فایل‌های صوتی (transcripts) و زیرنویس کردن ویدئوها (Caption) فراموش نشود.
- 2 - ناتوانی در هوش، یادگیری و عصبی (Cognitive, learning and neurological)
a. برای افرادی که چالش‌هایی در درک اطلاعات دارند، ساختار واضح، برچسب‌های مداوم راهنما، لینک‌های مناسب برای درک بیشتر و زبان بیان ساده مورد استفاده قرار گیرد.
- 3 - ناتوانی فیزیکی (Physical)
a. برای افرادی با نقص اندام یا عدم توانایی کنترل ماهیچه‌ها و اعضای حسی و... از Assistive Technologies مانند تشخیص صوت (تایپ گفتاری) و امکان تنظیم اندازه فونت پشتیبانی کنید.
- 4 - ناتوانی بیانی (Speech)
a. برخی افراد توانایی صحبت ندارند، خدمات خود را تنها مبتنی بر صحبت قرار ندهید.
- 5 - ناتوانی بصری (Visual)

a. برای افراد با ناتوانی در دید، مطمئن شوید که متن‌ها قابلیت بزرگنمایی دارند، کنتراست کافی به کار گرفته‌اید، از TTS = Text To Speech (تبدیل متن به گفتار) استفاده کرده‌اید و...

ناتوانی‌های دائمی، موقتی و موقعیتی

بحث Accessibility تنها مربوط به ناتوانی‌های دائمی نیست.

لمس

- شخصی که یک دست بیشتر ندارد، ناتوانی دائمی دارد.
- شخصی که دستش شکسته، ناتوانی موقتی دارد.
- والدی که طفلش را در یک دست گفته، ناتوانی موقعیتی دارد.

دید

- یک شخص نابینا، ناتوانی دائمی دارد.
- کسی که چشمش آب‌مروارید دارد، ناتوانی موقتی دارد.
- کسی که در حال رانندگی است و امکان نگاه به صفحه نمایش ندارد، ناتوانی موقعیتی دارد.

شنوایی

- یک شخص ناشنوا، ناتوانی دائمی دارد.
- شخصی که عفونی در گوش دارد، ناتوانی موقتی در شنوایی دارد.
- شخصی که در یک جمع شلوغ قرار دارد، ناتوانی موقعیتی دارد.

صحبت

- یک شخص لال، ناتوانی دائمی دارد.
- شخصی که التهاب حنجره دارد، ناتوانی موقتی دارد.
- شخصی که توانایی صحبت به یک زبان خاص را ندارد، ناتوانی موقعیتی دارد.

سیستم برای کاربران است؛ ما کاربر نیستیم

به یاد داشته باشید که ما کاربر سیستم نیستیم. اکثر اوقات ما طراحان و توسعه‌دهندگان، فکر می‌کنیم چون خودمان به راحتی می‌توانیم از سیستم طراحی شده توسط خودمان استفاده کنیم، دیگران نیز می‌توانند به راحتی از آن استفاده کنند.

اثر اجماع کاذب (اثر درک اشتباه) False-Consensus Effect

اثر اجماع کاذب اولین بار توسط Ross و Greene و House در سال ۱۹۷۷ تعریف شد. آن نشان دادند که اکثر افراد فکر می کنند دیگران نیز در موقعیت مشابه، مانند خودشان رفتار می کنند.

The false-consensus effect was first defined in 1977 by Ross, Greene, and House. They showed that unlike scientists, “layperson psychologists” (that is, all of us who are put into the position to guess how others would behave) tend to overestimate how many people share their choices, values, and judgments, and perceive alternate responses as rare, deviant, and more revealing of the responders.

از عناصر معنایی (Semantic Elements) استفاده کنید

Semantics به این معنی است که از عنصر صحیحی در HTML استفاده شود. در HTML، یکصد و ده عنصر وجود دارد که دو سوم آنها هیچ معنایی ندارند؛ مانند div و span که هیچ محتوایی نمی تواند درباره آنها برداشت کرد. مثال هایی از تگ های معنایی عبارتند از: form، table، header، footer، nav، article و..

یک نمونه استفاده مناسب از تگ ها در سایت Uber

Get in the driver's seat and get paid

Drive on the platform with the largest network of active riders.

[Sign up to drive](#)

[Learn more about driving and delivering](#)



به کدها دقت کنید که چه مناسب انتخاب شده اند:

```
<h2>Get in the driver's seat and get paid</h2>
<p>Drive on the platform with the largest network of active riders.</p>
<a href="/signup/drive/">Sign up to drive</a>
<a href="/en/drive/">Learn more about driving and delivering</a>

```

ممکن بود یک طراح برای دکمه از تگ button استفاده کند؛ در حالی که تگ button باید زمانی استفاده شود که قرار است به رخدادی پاسخ داده شود؛ زمانی که قرار است یک تعامل به صفحه دیگری پیمایش کند، باید از لینک و تگ a استفاده کرد.

Accessibility Landmarks

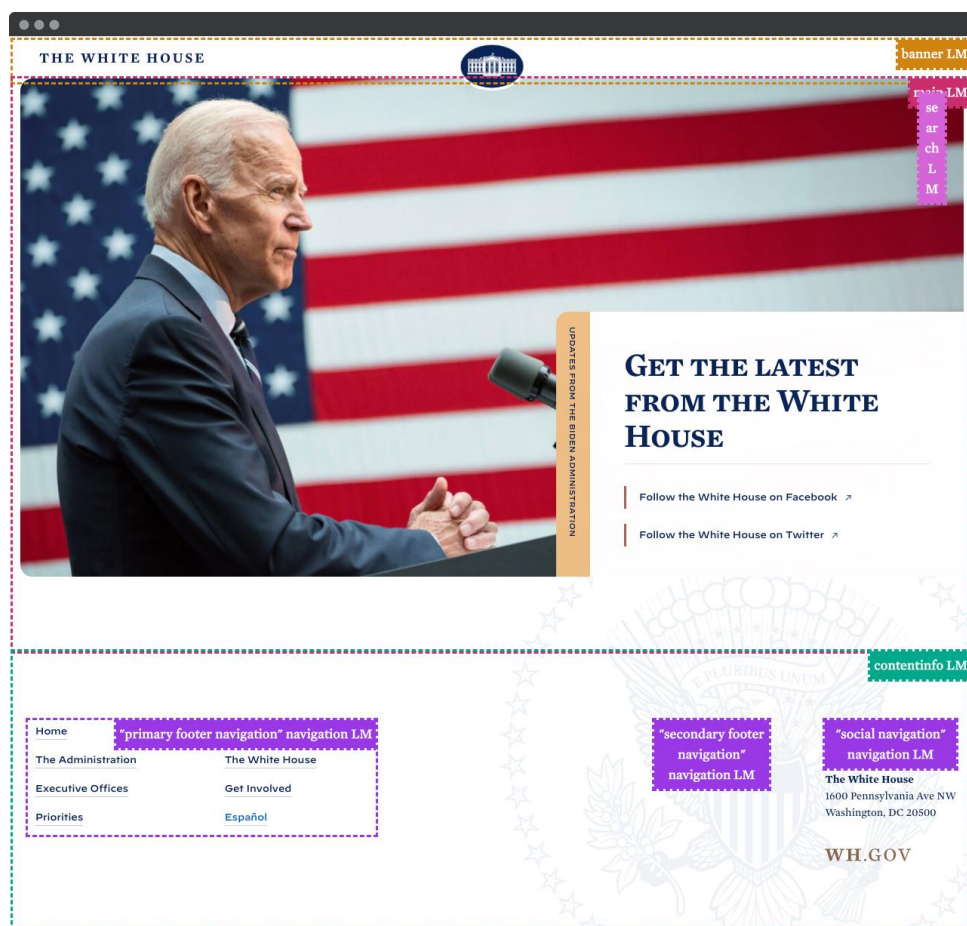
امکان landmark به Screen Reader ها اجازه می‌دهد که به بخش‌های خاصی از صفحه پرش کنند و اطلاعاتی به افراد نابینا درباره آن بخش ارائه دهند.

استفاده از تگ‌های معنایی مانند

- `<header>`
- `<nav>`
- `<main>`
- `<aside>`
- `<section>`
- `<footer>`

یکی از راه‌های پیاده‌سازی landmark در صفحات وب است.

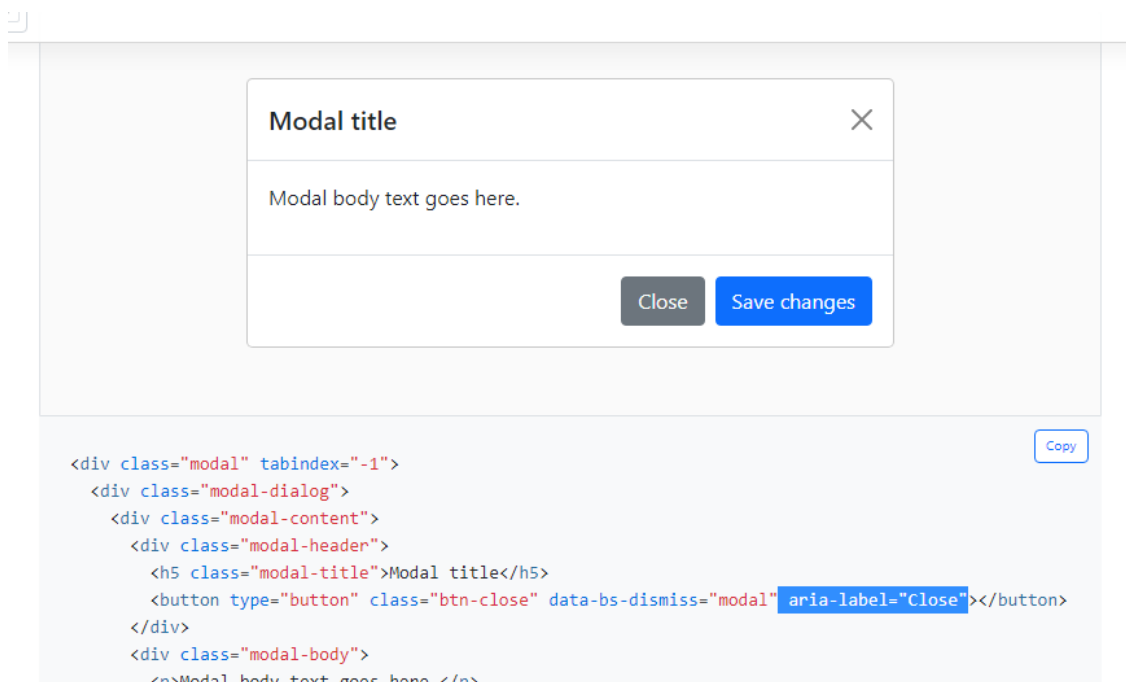
نمونه‌ای از استفاده از landmark ها در صفحه اول سایت کاخ سفید که با کمک سایت accessibilityinsights.io بررسی شده است:



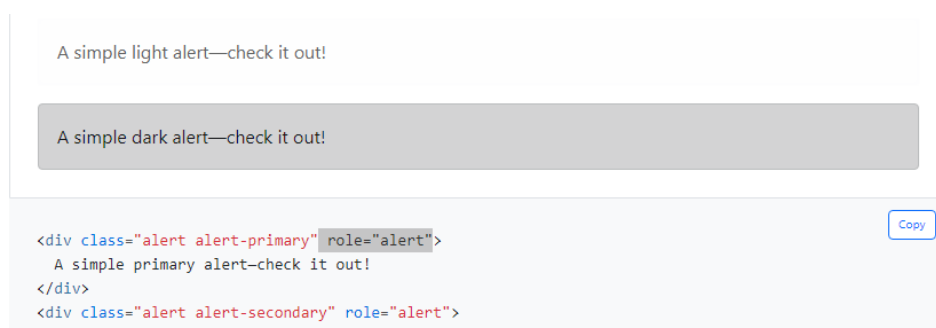
برای اینکه بهتر درک کنید که افراد نابینا صفحات سایت شما را چگونه می‌بینند، افزونه Screen Reader را روی مرورگر کروم خود نصب کنید. همینطور افزونه Accessibility Insights که توسط مایکروسافت طراحی شده به شما به عنوان طراح، کمک می‌کند که صفحاتی طراحی کنید که بهتر برای افراد نابینا قابل شناسایی و پیمایش باشد.

واژه ARIA مخفف Accessible Rich Internet Applications است که قوانینی دارد که به Screen Readerها قابلیت درک بهتر صفحه را می‌دهد.

برای نمونه در Bootstrap گاهی صفاتی که با aria- آغاز می‌شوند را می‌بینید:



یکی دیگر از راه‌های اعمال قوانین aria استفاده از صفت role است. مثال:



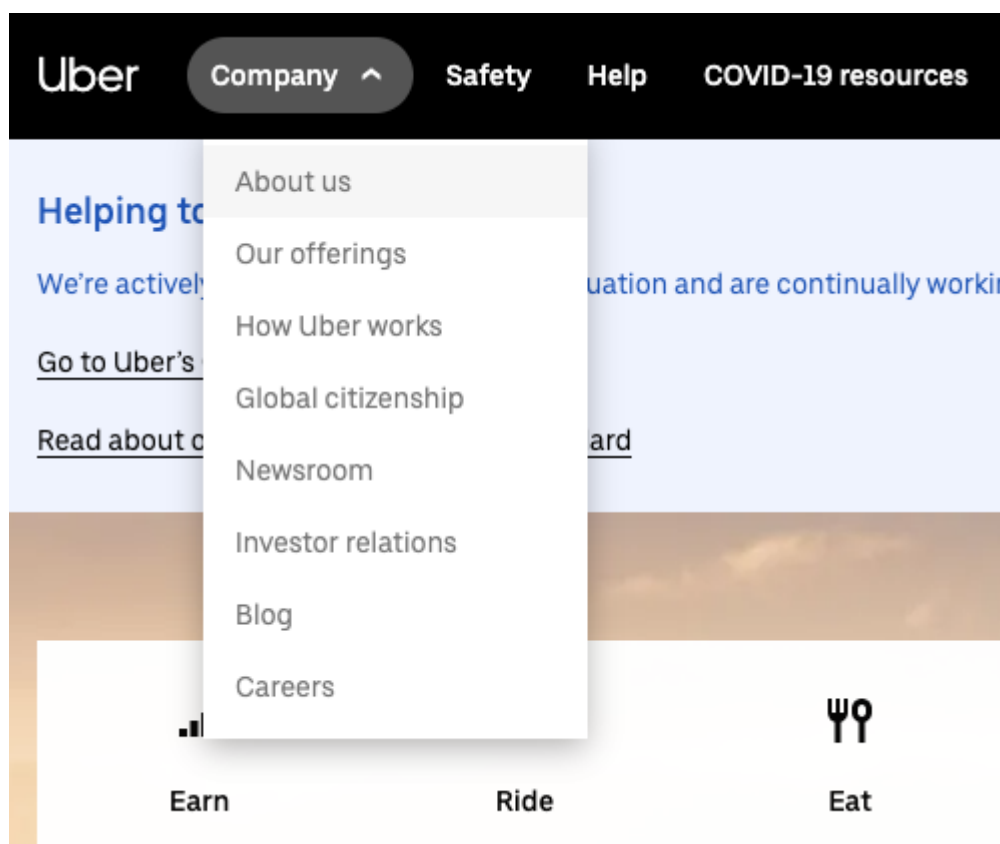
یک قانون مهم این است که از تگ‌های landmark مانند nav و main فقط یک بار در هر صفحه استفاده کنید. اما اگر به هر دلیلی نیاز شد که بیش از یک بار استفاده شود؛ باید با صفت aria-label به آن تگ‌ها برچسب بزنید. مثلاً اگر یک تگ nav برای دکمه‌های شبکه‌های اجتماعی گذاشته‌اید باید به آن aria-label="social" اضافه کنید.

دکمه‌ها و لینک‌ها

دکمه‌ها و لینک‌ها یکی از مهم‌ترین بخش‌های صفحه هستند؛ چرا که کاربران باید با آن بخش‌ها تعامل کنند. استفاده درست از دو تگ a و button می‌تواند به دستیارهای دیجیتالی افراد ناتوان کمک بیشتری کند.

نکته مهم این است که: از تگ button زمانی استفاده می کنیم که قرار است یک عملیات در همین صفحه فعلی اتفاق بیفتد اما از تگ a زمانی استفاده می کنیم که قرار است کاربر به یک صفحه جدید پیمایش کند.

مثال: در منوهای پیمایش تگی که زیرمنو را باز می کند باید button باشد و تگی که کاربر را به صفحات زیرمنوها هدایت می کند باید a باشد.

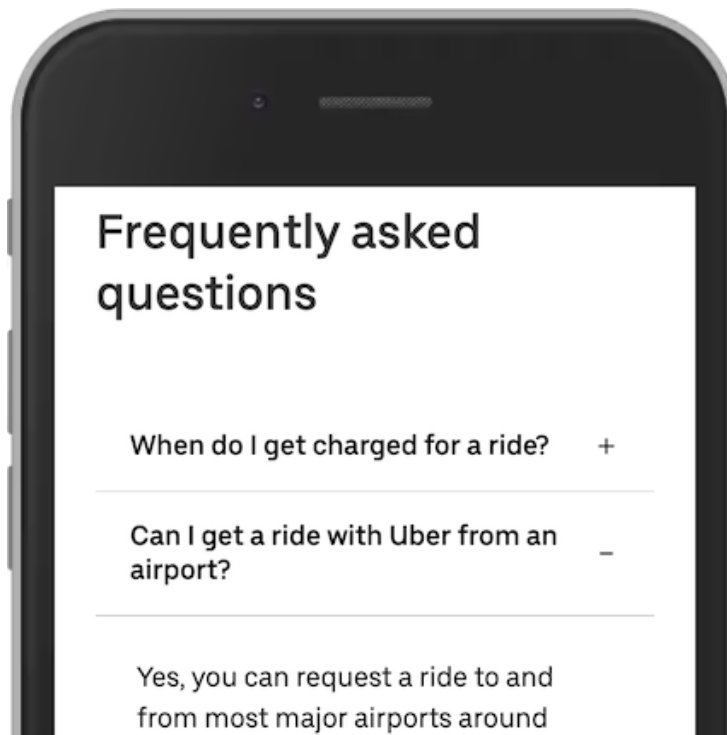


اگر مجبور شدید که از یک عنصر دیگر به جای عناصر معنایی استفاده کنید؛ باید به آن عنصر role و name و value بدهید. مثال: در تصویر زیر از تگ div به عنوان دکمه + و - برای باز و بسته کردن یک محوطه استفاده شده است. کد آن div به صورت زیر است:

```
<div role="button" aria-expanded="false">When do I get charged for a ride?</div>
```

نام عنصر div است؛ role آن button است و value آن برابر با aria-expanded=false

که به Screen Reader می فهماند که این محوطه، بسته (collaped) است و با کلیک روی این دکمه باز (expand) می شود.



Accessibility Color Contrast

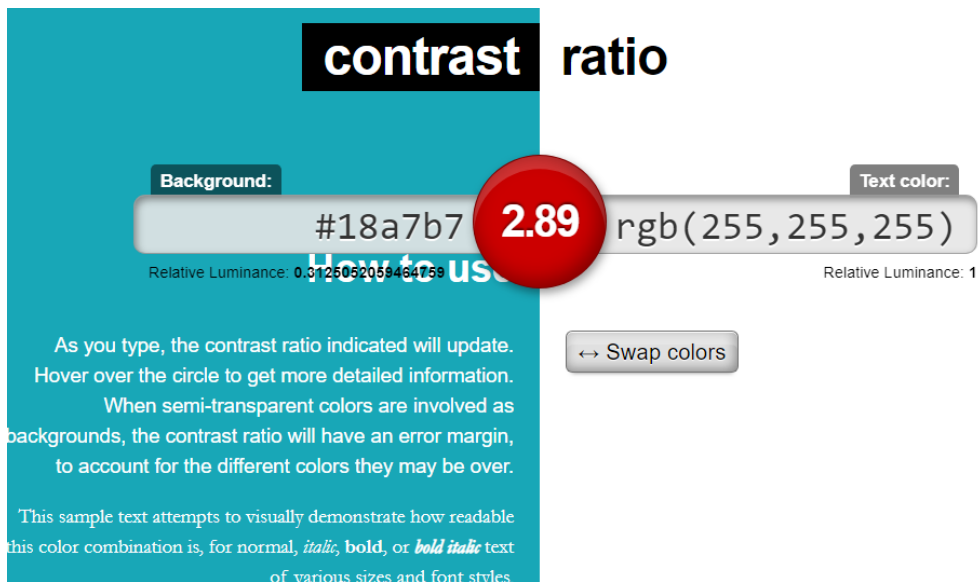
نوشته‌ها و عناصر گرافیکی باید از تضاد (contrast) کافی برخوردار باشند تا خوانایی بهتری داشته باشند؛ این موضوع به ویژه برای افرادی با مشکلات بینایی (مانند کسانی که نور را چند برابر می‌بینند) مهم‌تر نیز می‌شود.

تضاد عناصر نسبت به رنگ زمینه آن‌ها (background) اندازه‌گیری می‌شود و این مفهوم، نرخ تضاد یا Contrast Ratio نامیده می‌شود. یک نوشته سفید روی بک‌گراند سیاه، تضاد ۲۱ را به خود اختصاص می‌دهد و یک نوشته سفید روی بک‌گراند سفید، تضاد ۱ می‌گیرد.

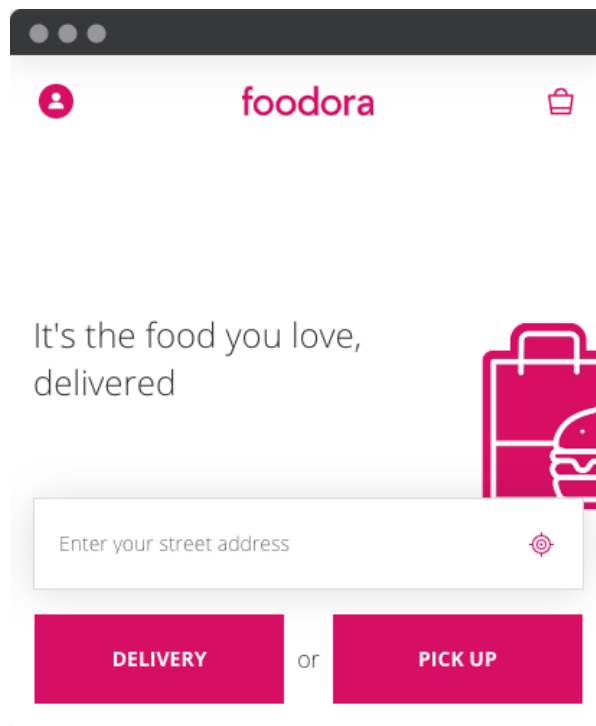
اپل در یک پژوهش نشان داده است که بهتر است تضاد بین 4.5 تا 7 باشد¹.

سایت contast-ratio.com می‌تواند در انتخاب رنگ متن و بک‌گراند با تضاد مناسب به شما کمک کند:

¹ <https://developer.apple.com/design/human-interface-guidelines/business-chat/visual-design/color/>



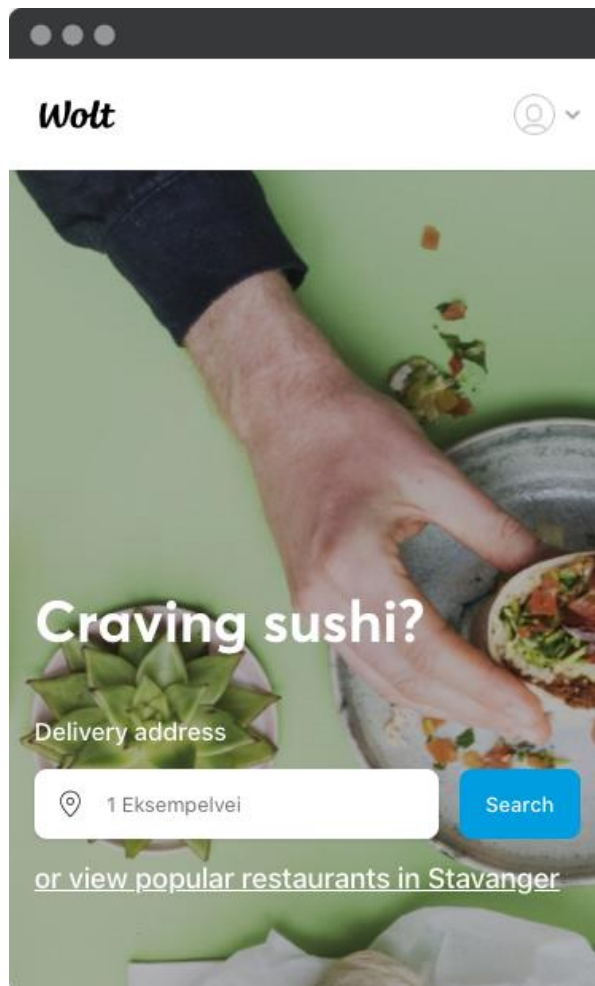
یک نمونه از طراحی خوب با کنتراست حدود ۵:



متن بر روی عکس

در مورد متن روی تصاویر اگر تصویر، روشن است، روشن ترین نقطه تصویر ملاک قرار می گیرد و اگر تیره است، تیره ترین نقطه.

مثلاً در تصویر زیر که روشن به حساب می آید، رنگ سفید نوشته با روشن ترین نقطه بک گراند، نرخ تضاد ۲.۳۵ دارد که کم به حساب می آید.



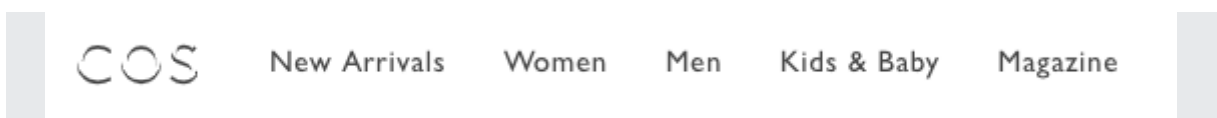
یک بهبود برای این تصویر می‌تواند این باشد که نوشته با یک رنگ ساده (solid) هایلایت شود.

حالات مختلف عناصر

عناصر ممکن است حالات مختلفی داشته باشند؛ برای مثال: حالت عادی، حالتی که موس روی آن‌ها قرار می‌گیرد (hover)، حالت غیرفعال (disabled)، در حالت تمرکز (focused) و...

اولاً باید نرخ تضاد در همه حالات رعایت شود. ثانیاً برای اینکه حالات مختلف عنصر از هم قابل تشخیص باشد باید نرخ تضاد بین دو حالت حداقل ۳ باشد. (مثلاً حالت عادی و حالت hover باید تفاوت حداقلی ۳ داشته باشند)

مثال: این منو در حالت عادی تضاد مناسبی دارد:



اما وقتی موس روی Women قرار می‌گیرد، نرخ تضاد به ۲.۸۴ کاهش می‌یابد:

COS

New Arrivals

Women

Men

Kids & Baby

Magazine

بحث تا:

https://www.w3schools.com/accessibility/accessibility_color_meaning.php