



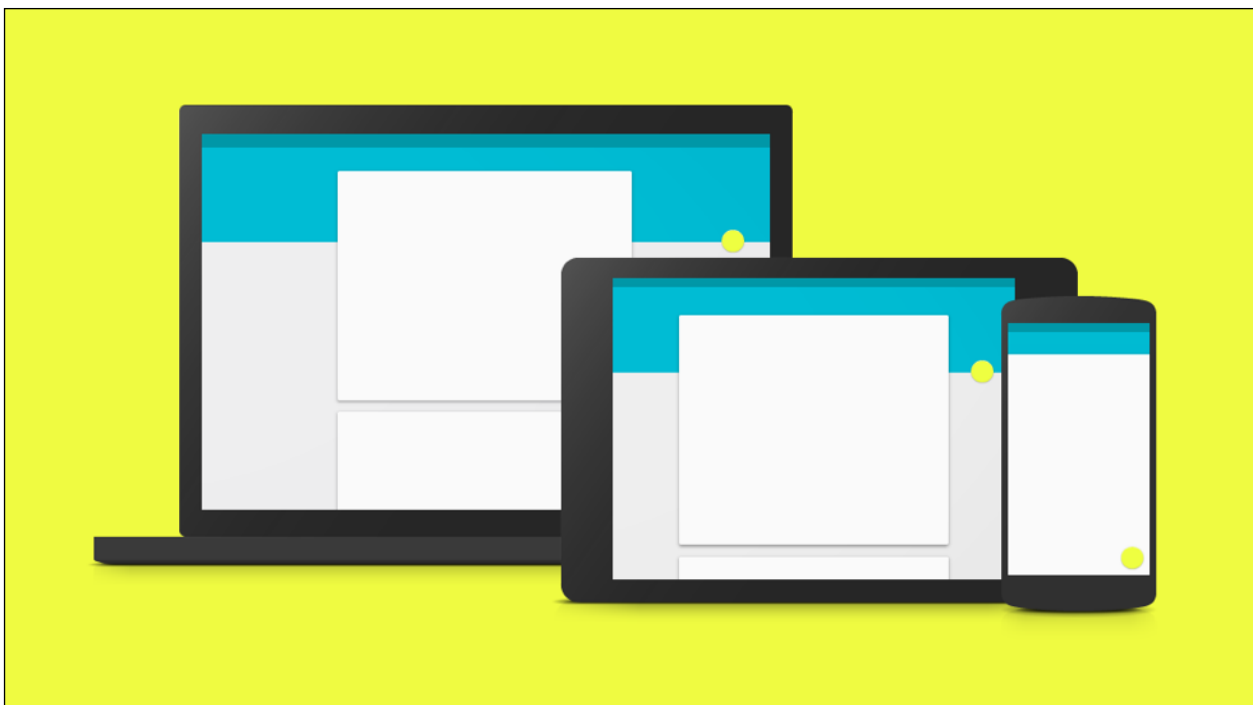
Google

Material Design

非官方设计规范指导手册 1.0

目录

第一章 概述	2	5.10 菜单	161
第二章 动画	5	5.11 进度和动态	194
2.1 真实的动作	5	5.12 滑块	200
2.2 交互响应	8	5.13 消息栏和弹出消息	205
2.3 有意义的转场动画	12	5.14 副标题	211
2.4 打动用户的细节	17	5.15 开关	213
第三章 样式	18	5.16 切换标签	217
3.1 色彩	18	5.17 文本框	226
3.2 字体排版	26	5.18 工具提示	244
3.3 图标	38	第六章 模型	247
3.4 图像	42	6.1 选择	247
第四章 布局	56	6.2 手势	251
4.1 原则	72	6.3 浮动操作按钮	259
4.2 标尺与参考线	85	6.4 设置	273
4.3 结构	98	6.5 图像处理	276
第五章 组件	102	6.6 搜索	278
5.1 底部卡片	102	第七章 可用性	284
5.2 按钮	106	第八章 资源下载	288
5.3 卡片	123	尾言	290
5.4 纸片	133		
5.5 提示框	136		
5.6 分割线	143		
5.7 网络	147		
5.8 列表	159		
5.9 列表控制	174		

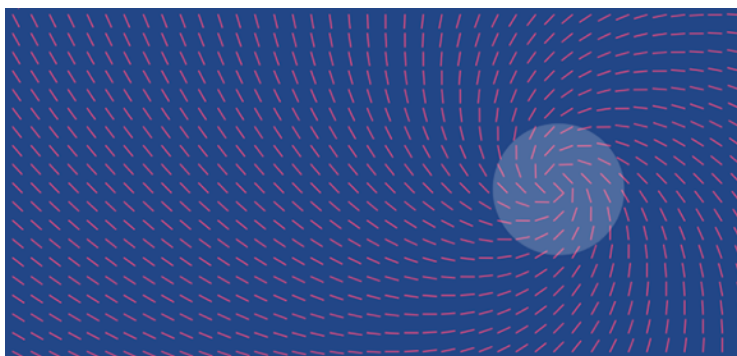


概述

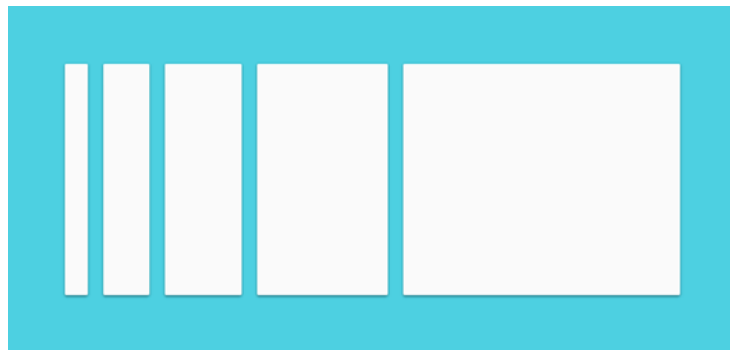
我们挑战自我，为用户创造了崭新的视觉设计语言。与此同时，新的设计语言除了遵循经典设计定则，还汲取了最新的科技，秉承了创新的设计理念。这就是原质化设计(Material Design)。这份文档是动态更新的，将会随着我们对 Material Design 的探索而不断迭代、升级。

目标

我们希冀创造一种新的视觉设计语言，能够遵循优秀设计的经典定则，同时还伴有创新理念和新的科技。



我们希望创造一种独一无二的底层系统，在这个系统的基础之上，构建跨平台和超越设备尺寸的统一体验。遵循基本的移动设计定则，同时支持触摸、语音、鼠标、键盘等输入方式。



设计原则

实体感就是(通过设计方式来表达)隐喻

通过构建系统化的动效和空间合理化利用，并将两个理念合二为一，构成了实体隐喻。与众不同的触感是实体的基础，这一灵感来自我们对纸墨的研究，但是我们相信，随着科技的进步，应用前景将不可估量。

实体的表面和边缘提供基于真实效果的视觉体验，熟悉的触感让用户可以快速地理解和认知。实体的多样性可以让我们呈现出更多反映真实世界的设计效果，但同时又绝不会脱离客观的物理规律。

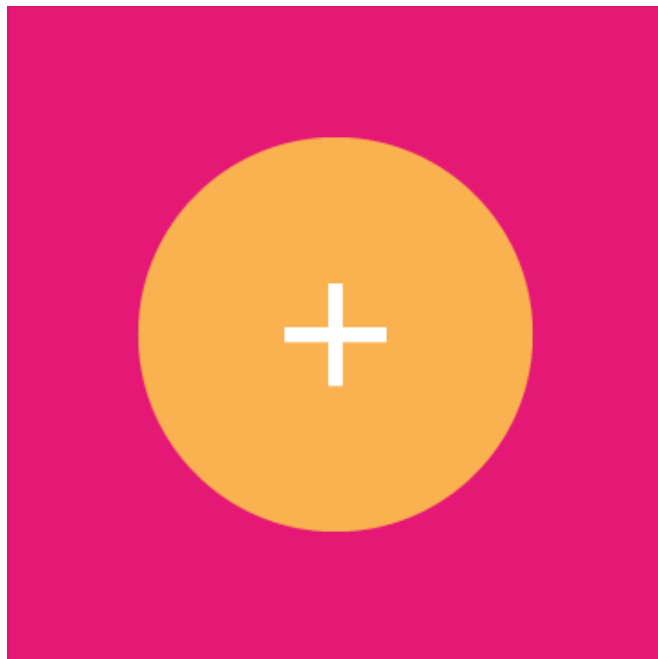


光效、表面质感、运动感这三点是解释物体运动规律、交互方式、空间关系的关键。真实的光效可以解释物体之间的交合关系、空间关系，以及单个物体的运动。

鲜明、形象、深思熟虑

新的视觉语言，在基本元素的处理上，借鉴了传统的印刷设计——排版、网格、空间、比例、配色、图像使用——这些基础的平面设计规范。在这些设计基础上下功夫，不但可以愉悦用户，而且能够构建出视觉层级、视觉意义以及视觉聚焦。精心选择色彩、图像、选择合乎比例的字体、留白，力求构建出鲜明、形象的用户界面，让用户沉浸其中。

Material Design设计语言强调根据用户行为凸显核心功能，进而为用户提供操作指引。



有意义的动效

动画效果(简称动效)可以有效地暗示、指引用户。动效的设计要根据用户行为而定，能够改变整体设计的触感。

动效应当在独立的场景呈现。通过动效，让物体的变化以更连续、更平滑的方式呈现给用户，让用户能够充分知晓所发生的变化。

动效应该是有意义的、合理的，动效的目的是为了吸引用户的注意力，以及维持整个系统的连续性体验。动效反馈需细腻、清爽。转场动效需高效、明晰。



2.1 真实的动作

真实的动作（Authentic Motion）

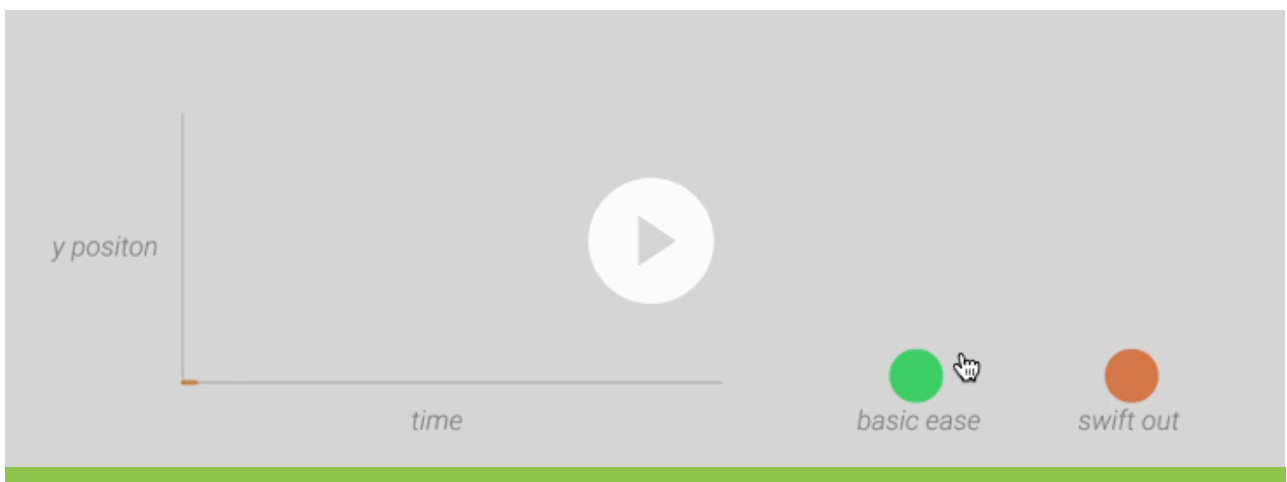
感知一个物体有形的部分可以帮助我们理解如何去控制它。观察一个物理的运动可以告诉我们它轻还是重，柔性还是刚性，小还是大。在material design设计规范中，动作不止是呈现着它美丽的一面，它还意味着在空间中的关系、功能以及在整个系统中的趋势。

质量和重量

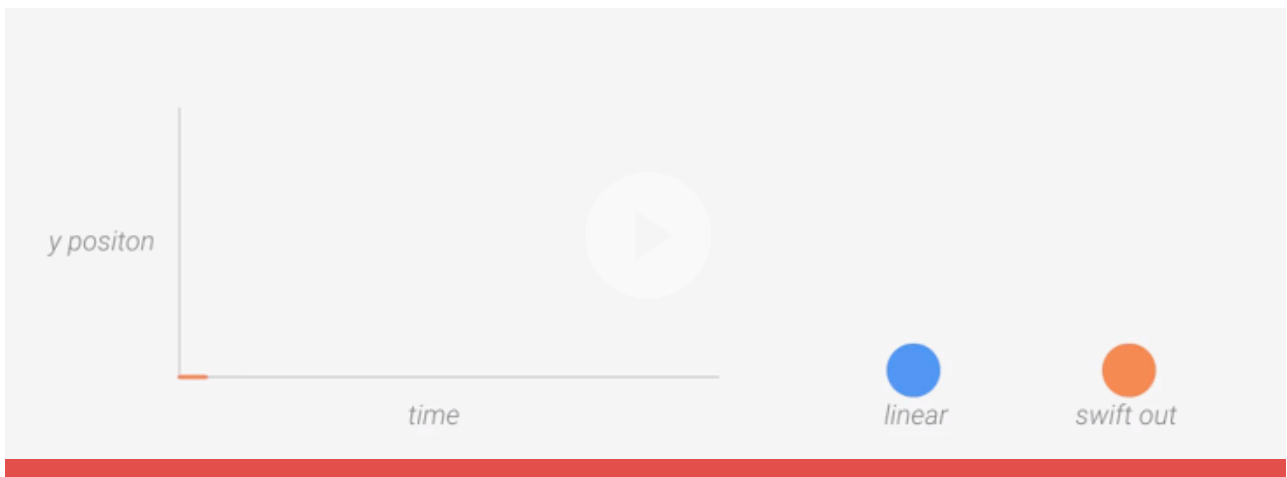
物理形体具有质量，当受到外力驱使时能够运动。因此，物体不能突然开始或结束运动。突然开始运动或突然结束运动的动效是生硬的，显得不自然，无法符合用户的预期，也无法取悦用户。

最佳案例

Material Design中动效设计的关键是保证物体运动的物理感，同时不牺牲物体运动的优雅感、简约感、美感，让物体的运动充满魔力，打造无缝式的用户体验。下面是一些动效设计规范，通过这些规范，可以理解如何用动效展示理念。



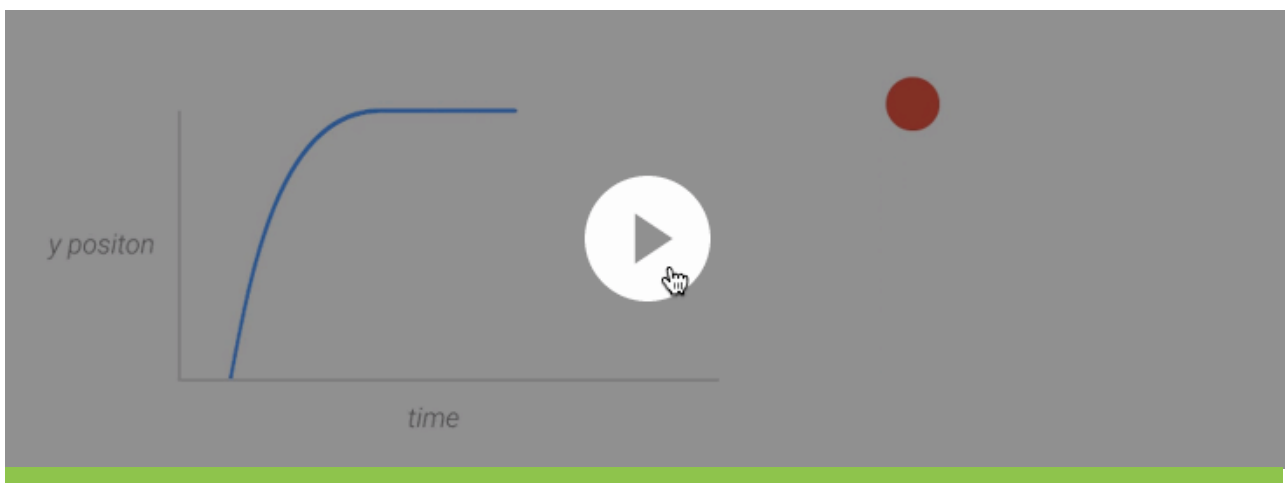
正确：灵活的加速、温和的减速，这样的动效让用户感到自然且愉悦



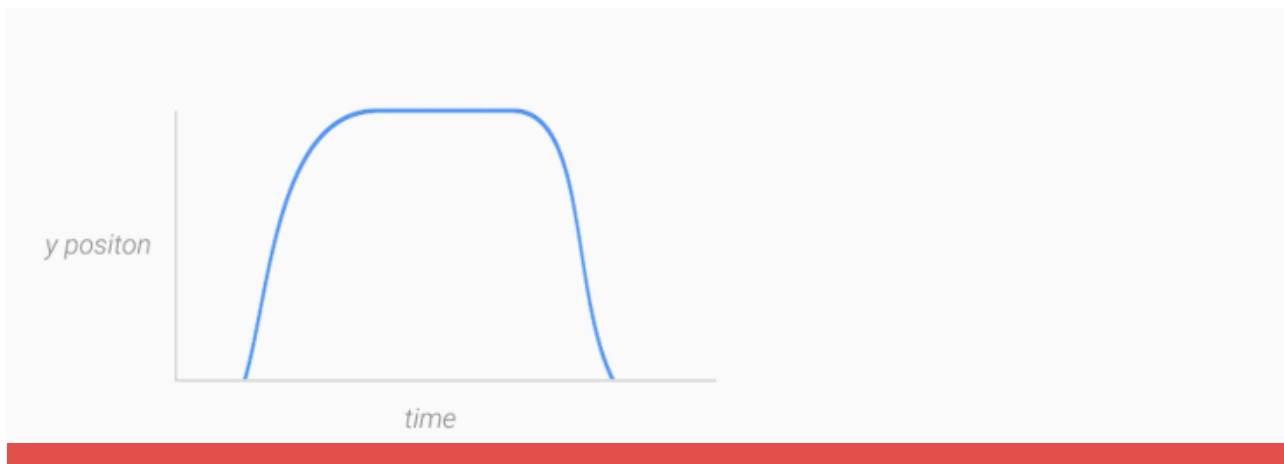
错误：线性的运动让人感到机械生硬。观察运动曲线，我们可以看到在运动开始和结尾处，运动速率突然的改变，这让人感到非常的突兀、不真实

特殊案例：进场、退场

当物体进场时，确保物体进场时处于最大速率。这样的运动便显得非常自然：因为物体在进场前便开始了运动，而不是进场时才开始运动。同样的，当物体退场时，要确保物体处于最大速率，而不是减速。在进场时缓入，在退场时缓出，这样用户的注意力会集中在动效上。然而，大多数情况下，这不是设计者想要的结果。



正确：进场和退场时的速率为最大速率，能够营造出自信感十足的动效



错误：在进场时加速，退场时减速。这样用户就会因速率的变化而分心

适当调整

不是所有的物体都以同样的方式运动。轻量化/小型物体可以很快的完成加速和减速，以为内他们质量很小，很小的外力就可以产生加速度的变化。大型/较重的物体需要更多的时间加速，才能到达最大速度。我们需要思考一下这些物理规律应该如何应用到UI元素上，并思考用何种运动方式表达设计。

2.2 交互响应

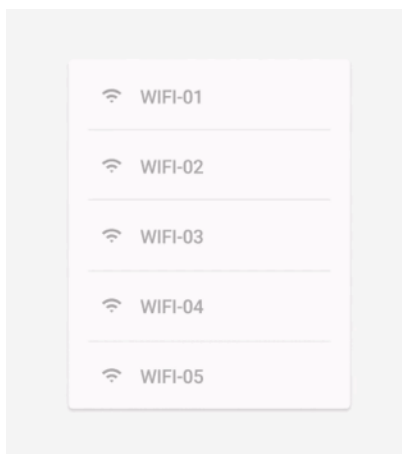
响应式交互能让用户信任，并且吸引他们。当用户操作一个美观且符合常理的应用时，他们会感到满意甚至很高兴。那是一种经过深思熟虑、有目的、非随机的而且可以带有轻微异想天开但不会让人分心的交互。

在material design中，应用是响应式的并且渴望用户操作的：

- 触控、语音输入、键盘、鼠标是首要考虑的输入模式
- 虽然UI元素是有形的，但是他们被限制在屏幕里面（电脑或者移动设备的屏幕），视觉元素和动效能减少这种割裂，让用户能够立即感知自己的操作。

响应式交互把一个应用从简单展现用户所请求的信息，提升至能与用户产生更强烈、更具体化交互的工具。

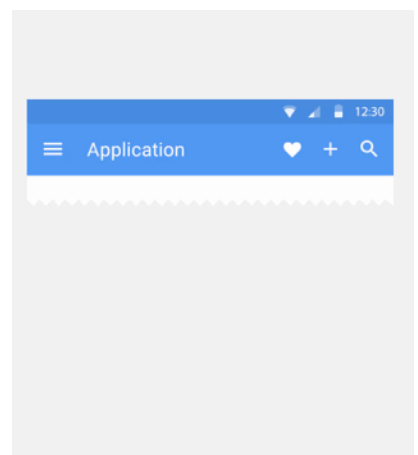
案例：



表面层反馈



物体本身的反馈



径向效果

表层响应

接收到输入事件，如点击屏幕 接收到输入事件，系统会立即在交互的触点上绘制出一个可视化的图形让用户感知到：如在点击屏幕时、使用麦克风时，或者键盘输入时，会出现类似于墨水扩散那样的视觉效果形状。

触控涟漪是这种触摸效果的核心视觉机制。在进行触摸事件时，设备能清晰而及时地让用户感知触摸按钮和语音输入时的变化。

最佳案例

在输入事件对应的位置上，例如：在手指的点击位置或者讲话的时候，屏幕上麦克风的图标位置会出现一个扩散的视觉元素。



触控涟漪-按压/释放



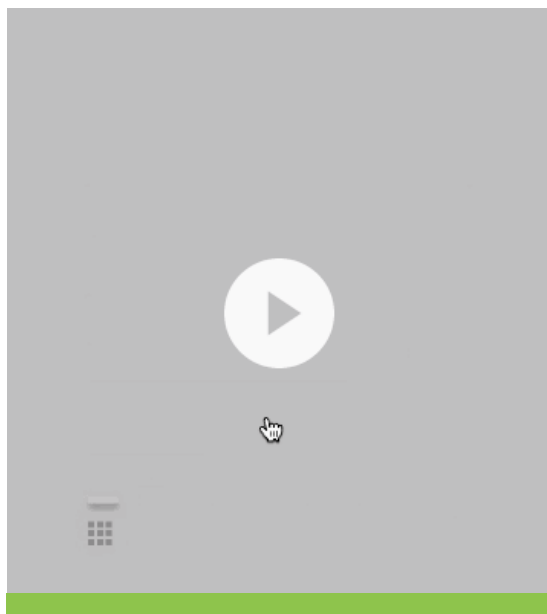
触控涟漪-拉进

元素响应

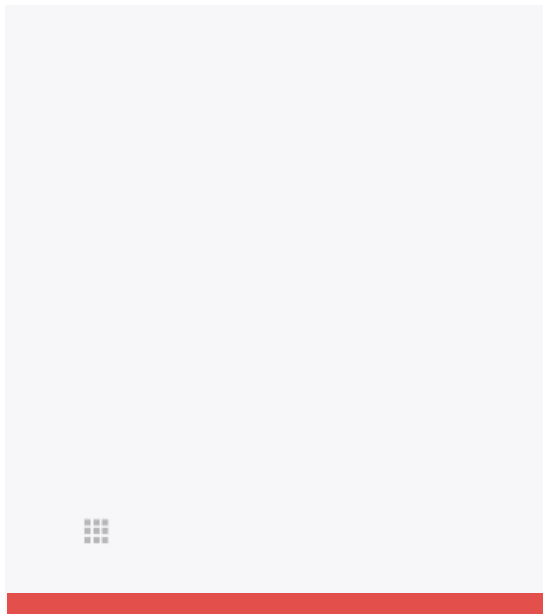
和表层响应的触控涟漪一样，每个元素本身也能做出交互响应，物体可以在触控或点击的时候浮起来，以表示该元素正处于激活状态。用户可以通过点击、拖动来生成、改变元素或者直接对元素进行处理。

最佳案例

以触控点为原点展开新元素的动画，当用户操作生成一个新元素时，元素的动画应该基于触控点展开。



正确：物体从触控点开始出现，视觉上将这一元素和触控点联系在一起



不正确：卡片元素从屏幕正中央出现，用户无法把触控点与卡片元素联系起来

点击浮动

当卡片元素或可分离元素被激活时，应该浮起以表明正处于激活状态。



径向响应

所有的用户交互行为中都会有一个中心点，他们想通过该中心点来达到自己的目的。

作为用户关注的中心点，应该绘制一个明显的视觉效果来让用户清晰地感知自己的输入（触摸屏幕、语音输入等）。在用户的操作中心点应该形成一个像涟漪一样逐渐发散开的径向动效响应。

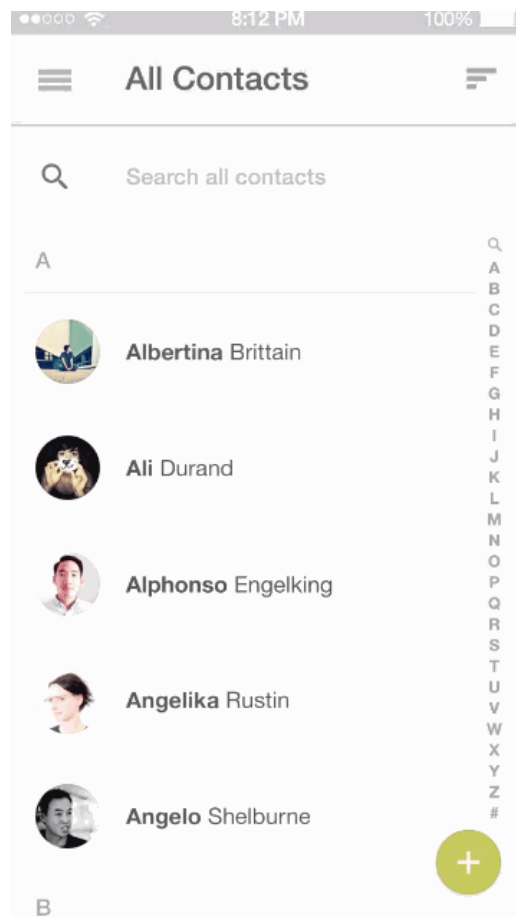
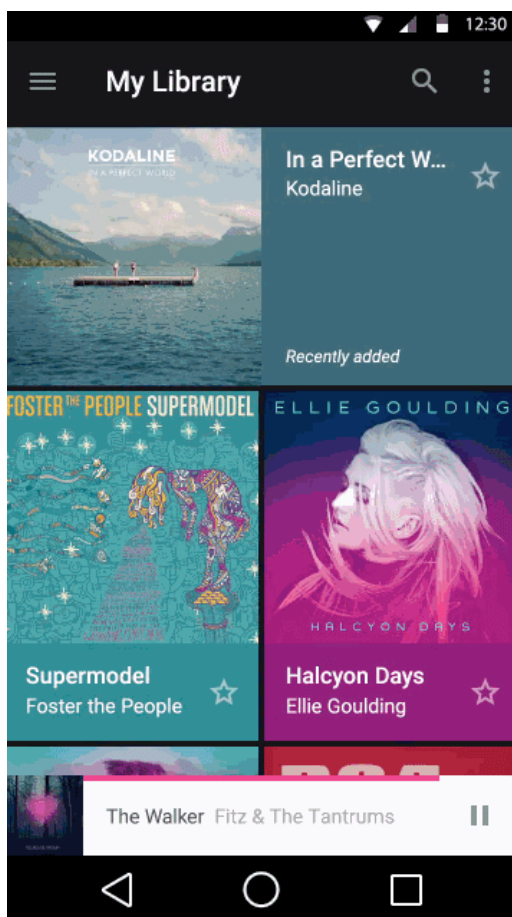
所有输入都有中心点，涟漪效果应从触控点、语音时屏幕上的语音图标、键盘输入时具体的按键点击位置上展开。

这些输入动作产生时都应该在中心点形成一个视觉上的关联，从中心点展开一连串动作产生的涟漪效果。

2.3有意义的转场动画

对于普通用户来说，是关注一个应用本身、还是更关注这个应用的元素从A点到B点转变的过程，这种选择往往有些难。谨慎编排的动画可以在有多步操作的过程中有效地引导用户的注意力；在版面变化或元素重组时避免造成困惑；提高用户体验的整体美感。动画设计不仅应当优美，更应当服务于功能。

案例：



视觉连贯性

在两个视觉效果不同的页面之间的转场应该平滑、轻快，更重要的是使用户感觉清晰而非困惑。一个好的转场可以四两拨千斤，让用户清楚地了解他们应该关注哪里。每一个转场应该包含以下三类元素：

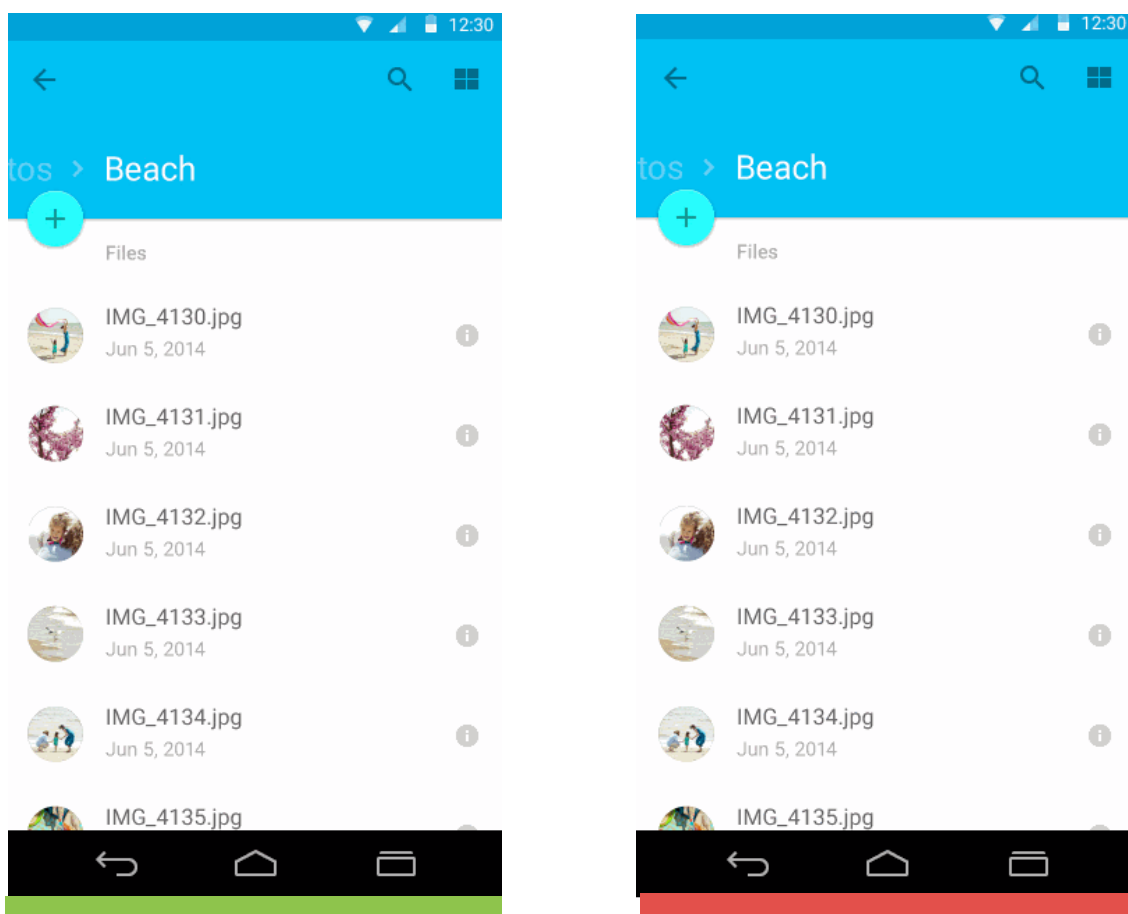
- 新入元素(Incoming elements): 完全新的元素需要有新手引导，从已有场景中转变而来的元素需要重新被识别
- 淡出元素(Outgoing elements)：与当前场景不相关的元素应当以恰当的方式被移除
- 通用元素(Shared elements)：指的是那些从转场开始到结束都没有发生变化的元素。可以是细微至单个图标，也可以是显著到占据屏幕的图片展示

思考点

当需要设计动画时，需要考虑以下：

- 用户的注意力应该被如何引导？什么元素和动画能辅助实现这个目标？新入元素，淡出元素和通用元素在这个转场中应该怎样被强调或弱化？
- 在设计场景时考虑到转场，并且尽量通过色彩和通用元素在不同场景转化间建立视觉联系
- 审慎地添加动画，思考如何移动一个特定的元素，从而使这个转场更明晰并使人愉悦

案例：



正确：恰当的时候使用纸片叠加方式。一般情况下，新入的纸片元素应当滑入视线。但字体元素(ink element)除外，他们应该以淡入的方式出现。不提倡全屏的淡入淡出，但也比瞬间切屏要好。

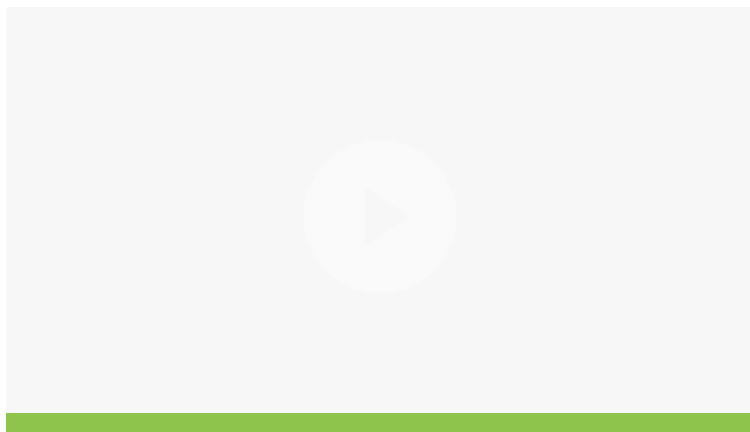
不正确：避免瞬间切屏(hard cut)。瞬间切屏过于突兀，并且会导致用户很难理解这个转场。

富有层次的时间感

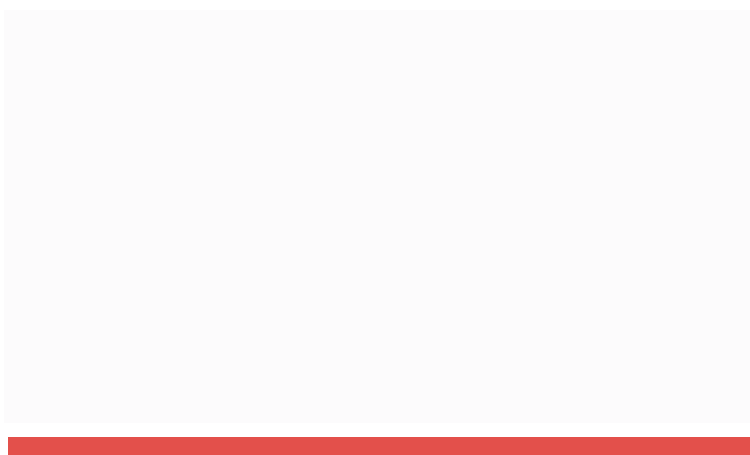
在建立转场的时候，对于元素移动的顺序和时机都要详加考虑。要确保这个动画能使信息的展示具有层次感。也就是说，它能引导用户的关注力，将最重要的内容传递给用户。

然而，这并不是说最重要的东西先动，最不重要的东西就后动。元素转场的时序要平滑并且避免脱节的感觉。

案例：



正确：运动要错开，富有层次感，来指引用户的视觉轨迹



不正确：所有元素同时运动，没有暗示出重点元素。如果所有元素同等重要，那么可以考虑将他们编组，共同构建更大范围的动画效果。

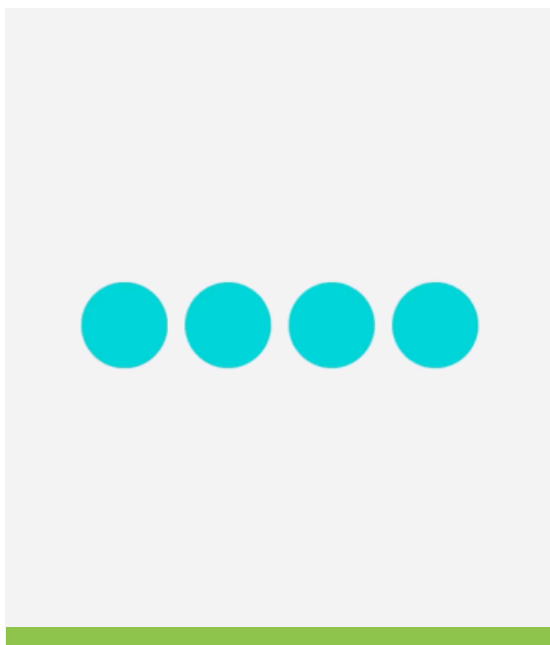
连贯的编排

由于转场元素在整屏范围里移动，他们需要以协调的方式运动。起到引导视觉焦点作用的元素，其整个移动过程都要有意义、有秩序。随机的动画会分散注意力。一个编排好的应用也能够给用户提供学习的时机。如果转场的所有元素都很好的协调，用户对于这个应用的理解也会增强。他们「理解」这个应用，不会因为动画而无所适从。

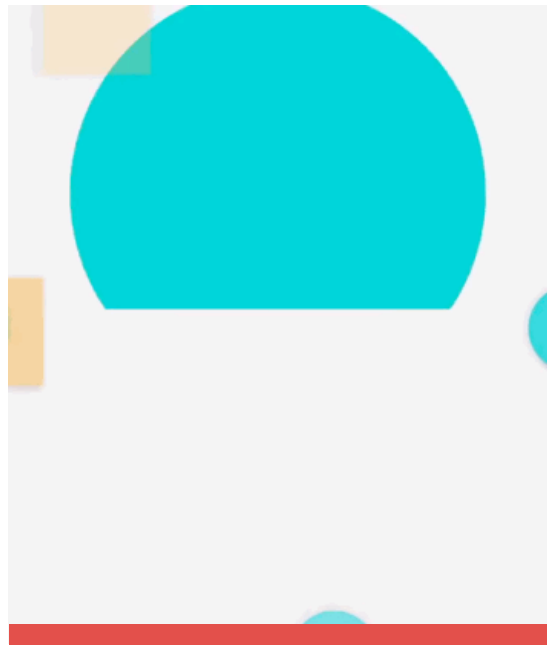
最佳案例

- 避免线性运动，除非是沿轴线的运动，或者多个元素一致朝某个点运动时
- 确保元素的运动要和整体转场效果在物理规律上契合。多个元素运动时避免杂乱的运动感和重叠的运动轨迹。
- 考虑深度、层级关系
- 试着画出屏幕上所有运动元素的运动轨迹，看看轨迹是否是美丽且足有有序的。并观察运动轨迹是否清晰。
- 要保证进场元素和退场元素空间感的一致性。

案例：



正确：通过有序的动效吸引用户的注意



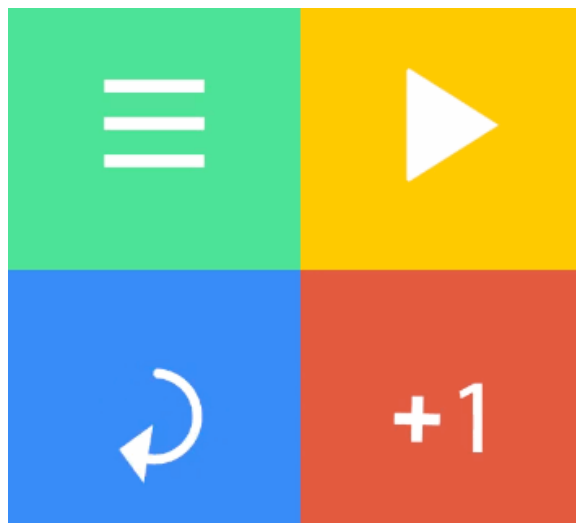
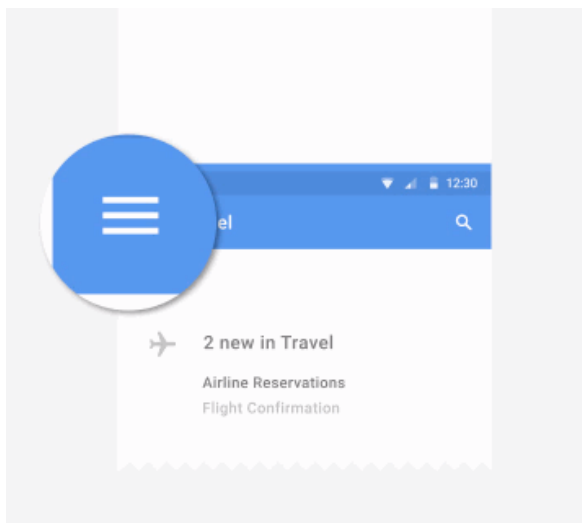
不正确：物体的进场和退场时运动方向非常随机。因动效的无序、不一致而混淆用户。

2.4 打动用户的细节

动画可以存在于应用程序的所有组件和扩展中，从细小的图标到核心的场景转换和动作，所有元素共同构建出一个拥有无缝体验、美观且功能强大的应用。

动画最基本的使用场景是过度效果，但哪怕是最基本的动画，只要恰到好处并足够出色，同样能打动用户。例如一个菜单图标变成一个箭头或者是播放控制按钮，这种服务间的无缝切换不仅仅能让用户感知，更是让完美的细节和精湛的设计充满你的应用。用户真的会感受到这些小细节。

细节动画的演示：



3.1 色彩



色彩从当代建筑、路标、人行横道以及运动场馆中获取灵感，由此引发出大胆的颜色表达激活了色彩，与单调乏味的周边环境形成鲜明的对比。强调大胆的阴影和高光。引出意想不到的且充满活力的颜色。

UI调色板

调色板以一些基础色为基准，通过填充光谱来为Android、Web和iOS环境提供一套完整可用的颜色。基础色的饱和度是500。

Red		Pink		Purple	
500	#e51c23	500	#e91e63	500	#9c27b0
50	#fde0dc	50	#fce4ec	50	#f3e5f5
100	#f9bdbb	100	#f8bbd0	100	#e1bee7
200	#f69988	200	#f48fb1	200	#ce93d8
300	#f36c60	300	#f06292	300	#ba68c8
400	#e84e40	400	#ec407a	400	#ab47bc
500	#e51c23	500	#e91e63	500	#9c27b0
600	#dd191d	600	#d81b60	600	#8e24aa
700	#d01716	700	#c2185b	700	#7b1fa2
800	#c41411	800	#ad1457	800	#6a1b9a
900	#b0120a	900	#880e4f	900	#4a148c
A100	#ff7997	A100	#ff80ab	A100	#ea80fc
A200	#ff5177	A200	#ff4081	A200	#e040fb
A400	#ff2d6f	A400	#f50057	A400	#d500f9
A700	#e00032	A700	#c51162	A700	#aa00ff

Deep Purple		Indigo		Blue	
500	#673ab7	500	#3f51b5	500	#5677fc
50	#ede7f6	50	#e8eaf6	50	#e7e9fd
100	#d1c4e9	100	#c5cae9	100	#d0d9ff
200	#b39ddb	200	#9fa8da	200	#afbfff
300	#9575cd	300	#7986cb	300	#91a7ff
400	#7e57c2	400	#5c6bc0	400	#738ffe
500	#673ab7	500	#3f51b5	500	#5677fc
600	#5e35b1	600	#3949ab	600	#4e6cef
700	#512da8	700	#303f9f	700	#455ede
800	#4527a0	800	#283593	800	#3b50ce
900	#311b92	900	#1a237e	900	#2a36b1
A100	#b388ff	A100	#8c9eff	A100	#a6baff
A200	#7c4dff	A200	#536dfe	A200	#6889ff
A400	#651fff	A400	#3d5afe	A400	#4d73ff
A700	#6200ea	A700	#304ffe	A700	#4d69ff

Light Blue	
500	#03a9f4
50	#e1f5fe
100	#b3e5fc
200	#81d4fa
300	#4fc3f7
400	#29b6f6
500	#03a9f4
600	#039be5
700	#0288d1
800	#0277bd
900	#01579b
A100	#80d8ff
A200	#40c4ff
A400	#00b0ff
A700	#0091ea

Cyan	
500	#00bcd4
50	#e0f7fa
100	#b2ebf2
200	#80deea
300	#4dd0e1
400	#26c6da
500	#00bcd4
600	#00acc1
700	#0097a7
800	#00838f
900	#006064
A100	#84ffff
A200	#18ffff
A400	#00e5ff
A700	#00b8d4

Teal	
500	#009688
50	#e0f2f1
100	#b2dfdb
200	#80cbc4
300	#4db6ac
400	#26a69a
500	#009688
600	#00897b
700	#00796b
800	#00695c
900	#004d40
A100	#a7ffeb
A200	#64ffda
A400	#1de9b6
A700	#00bfa5

Green	
500	#259b24
50	#d0f8ce
100	#a3e9a4
200	#72d572
300	#42bd41
400	#2baf2b
500	#259b24
600	#0a8f08
700	#0a7e07
800	#056f00
900	#0d5302
A100	#a2f78d
A200	#5af158
A400	#14e715
A700	#12c700

Light Green	
500	#8bc34a
50	#f1f8e9
100	#dcedc8
200	#c5e1a5
300	#aed581
400	#9ccc65
500	#8bc34a
600	#7cb342
700	#689f38
800	#558b2f
900	#33691e
A100	#ccff90
A200	#b2ff59
A400	#76ff03
A700	#64dd17

Lime	
500	#cddc39
50	#f9fbe7
100	#f0f4c3
200	#e6ee9c
300	#dce775
400	#d4e157
500	#cddc39
600	#c0ca33
700	#afb42b
800	#9e9d24
900	#827717
A100	#f4ff81
A200	#eeff41
A400	#c6ff00
A700	#aeea00

Yellow		Amber		Orange	
500	#ffeb3b	500	#ffc107	500	#ff9800
50	#ffffde	50	#fff8e1	50	#fff3e0
100	#ffff9c	100	#ffecb3	100	#ffe0b2
200	#fff59d	200	#ffe082	200	#ffcc80
300	#fff176	300	#ffd54f	300	#ffb74d
400	#ffee58	400	#ffca28	400	#ffa726
500	#ffeb3b	500	#ffc107	500	#ff9800
600	#fdd835	600	#ffb300	600	#fb8c00
700	#fbc02d	700	#ffa000	700	#f57c00
800	#f9a825	800	#ff8f00	800	#ef6c00
900	#f57f17	900	#ff6f00	900	#e65100
A100	#ffff8d	A100	#ffe57f	A100	#ffd180
A200	#ffff00	A200	#ffd740	A200	#ffab40
A400	#ffea00	A400	#ffc400	A400	#ff9100
A700	#ffd600	A700	#ffab00	A700	#ff6d00

Deep Orange		Brown		Grey	
500	#ff5722	500	#795548	500	#9e9e9e
50	#fbe9e7	50	#efebe9	50	#fafafa
100	#ffccbc	100	#d7ccc8	100	#f5f5f5
200	#ffab91	200	#bcaaa4	200	#eeeeee
300	#ff8a65	300	#a1887f	300	#e0e0e0
400	#ff7043	400	#8d6e63	400	#bdbdbd
500	#ff5722	500	#795548	500	#9e9e9e
600	#f4511e	600	#6d4c41	600	#757575
700	#e64a19	700	#5d4037	700	#616161
800	#d84315	800	#4e342e	800	#424242
900	#bf360c	900	#3e2723	900	#212121
A100	#ff9e80			1000	#000000
A200	#ff6e40			1000	#ffffff
A400	#ff3d00				
A700	#dd2c00				

Blue Grey	
500	#607d8b
50	#eceff1
100	#cfd8dc
200	#b0bec5
300	#90a4ae
400	#78909c
500	#607d8b
600	#546e7a
700	#455a64
800	#37474f
900	#263238

UI颜色使用

选择你的调色板

限制颜色的数量，在众多基础色中选出三个色度以及一个强调色。强调色用于后背，可有可无。

Primary – Indigo	
500	#3F51B5
100	#C5CAE9
500	#3F51B5
700	#303F9F

Accent – Pink	
A200	#FF4081
Fallback	
A100	#FF80AB
A400	#F50057

文本、图标、分割线要设置透明度

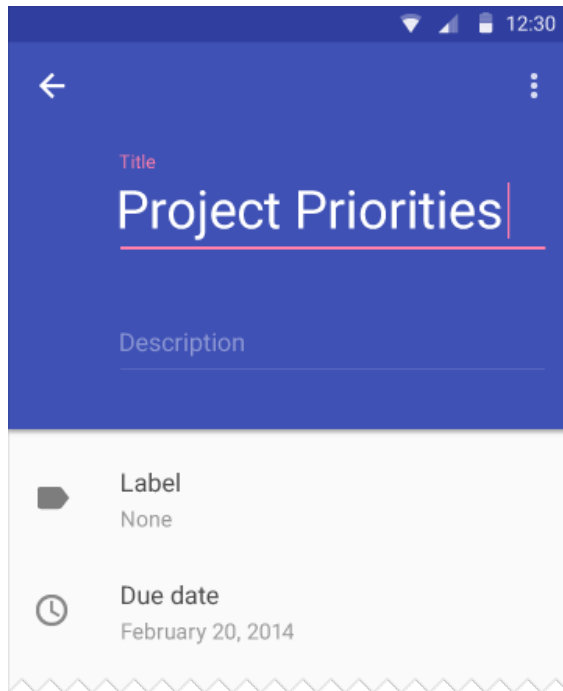
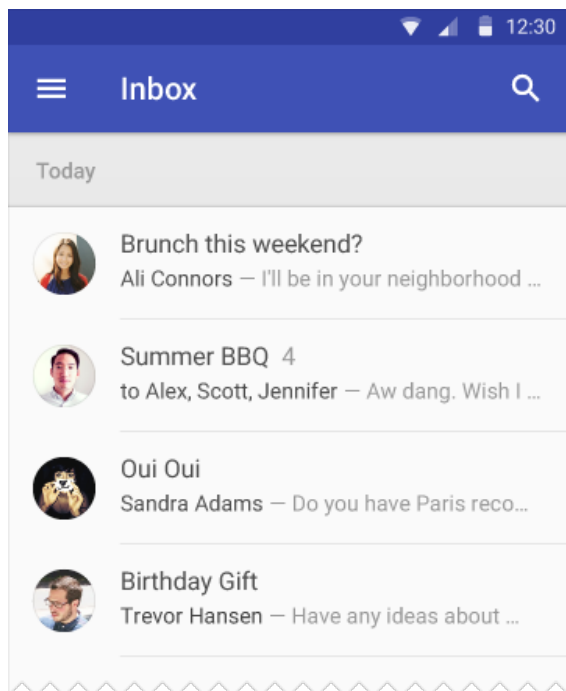
为了打造更高效的信息层级，文本要使用不同色调。白色背景中，标准的文本透明度是87% (#000000)。而次级文本/图标，不透明度应该是48% (#000000)。提示文本，处于还要更低的视觉层级，那么不透明度是24% (#000000)。

Accent – Pink	
A200	#FF4081
Fallback	
A100	#FF80AB
A400	#F50057

其他元素，比如说图标和分割线，也应该设置透明度，效果要比纯色好，这样的透明度设置（下图），能适应任何背景色。

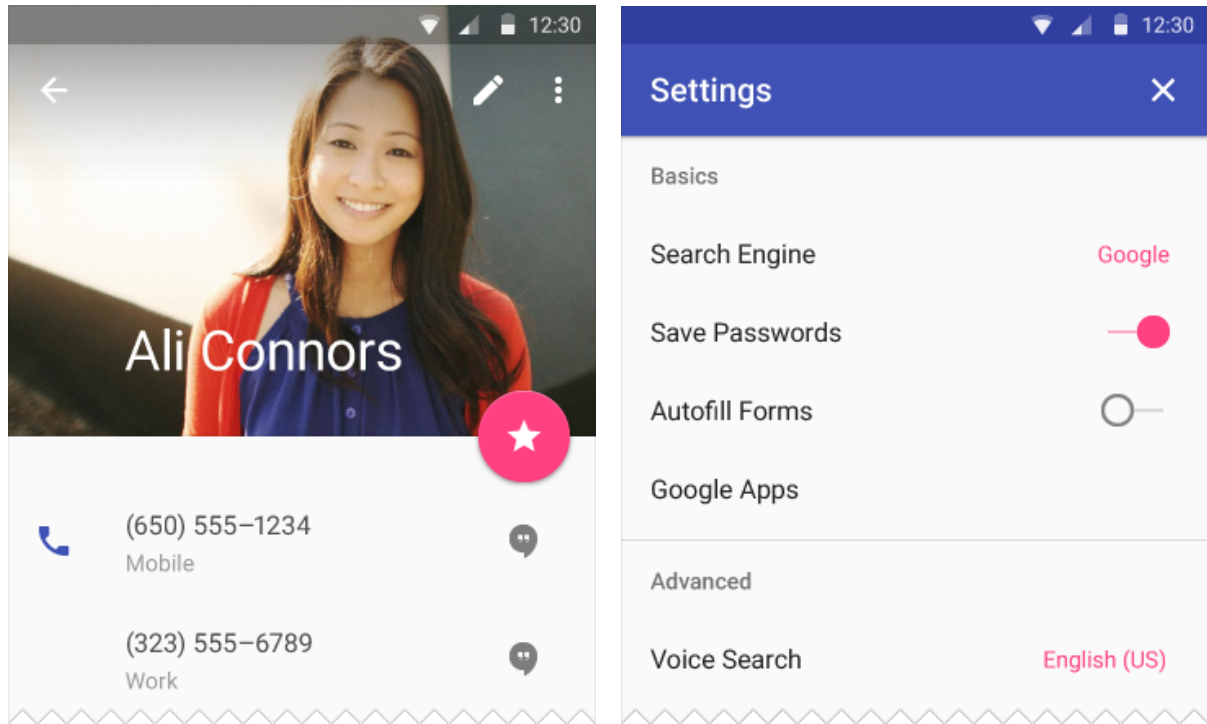
而有色背景上的黑白色文本，请参考上图的配色表，来设置透明度，提高对比感。

推荐在界面设计中，在较大块的区域大胆用色。界面中的不同元素采用不同的色彩主题。如下图，工具栏，以及较大块的区域，采用了500蓝这种主色。这是应用的主色，而状态栏采用了700蓝，比主色稍深。



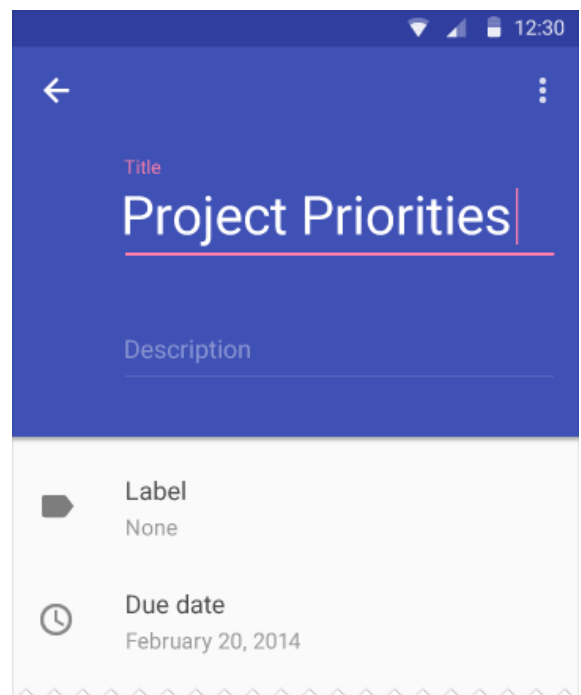
强调色

靓丽的强调色可被用来当做按钮和控件，例如开关和滑块。左对齐的选项（或是图标，或是标题）也可以采用强调色。



备用强调色

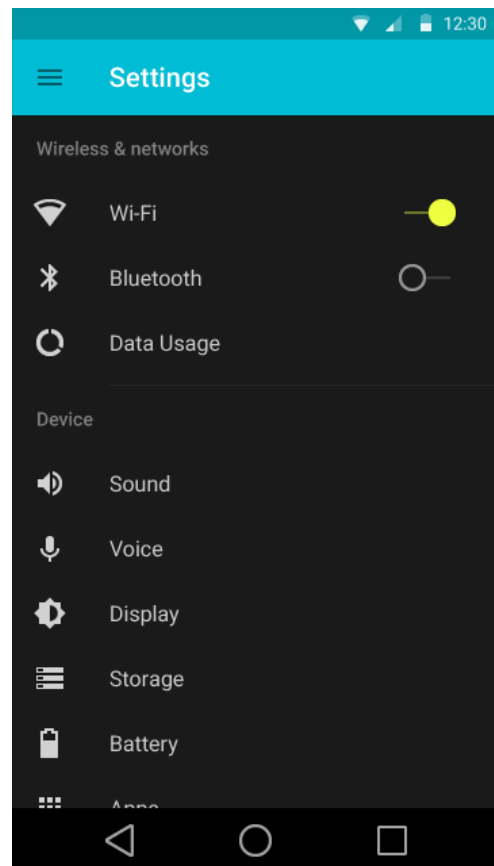
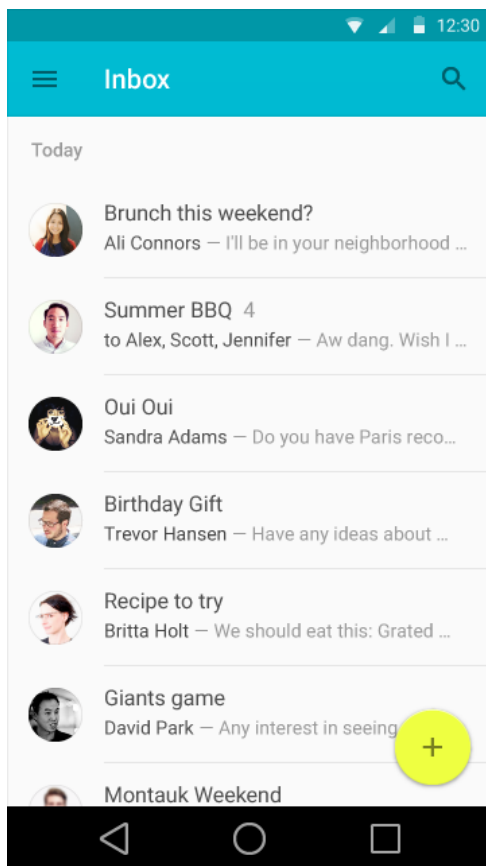
如果你的强调色相对于背景色太深或者太浅，默认的做法是选择一个更浅或者更深的备用颜色。如果你的强调色无法正常显示，那么在白色背景上会使用饱和度500的基础色。如果背景色就是饱和度500的基础色，那么会使用100%的白色或者54%的黑色。



色彩主题

主题是对应用提供一致性色调的方法。样式指定了表面的亮度、阴影的层次和字体元素的适当不透明度。为了提高应用间的一致性，提供两种主题选择：

- 明亮
- 深邃



3.2 字体排版

Roboto

Ice Cream Sandwich发布后，Roboto成为了安卓的标准字体。在这个版本中，将Roboto做了进一步全面优化，以适配更广泛的平台。它变得稍宽了一点，并进行了轻微圆化，进一步提升了清晰度，让阅读更加舒适。



标准样式

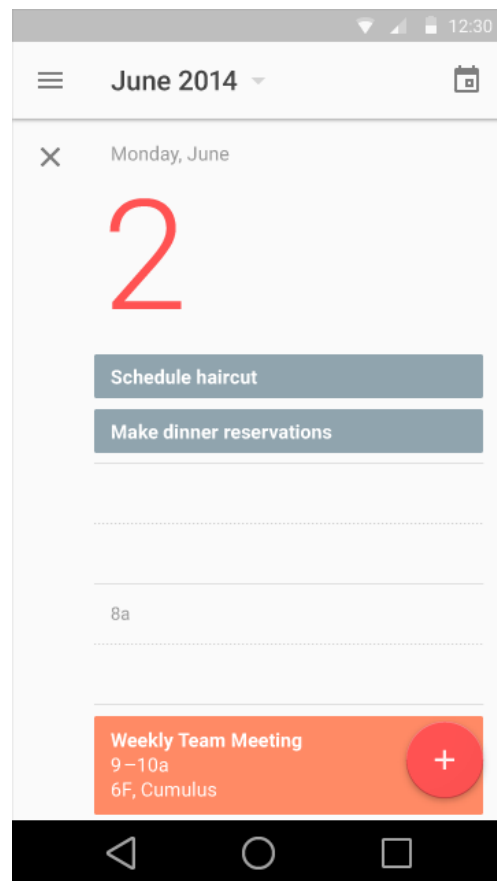
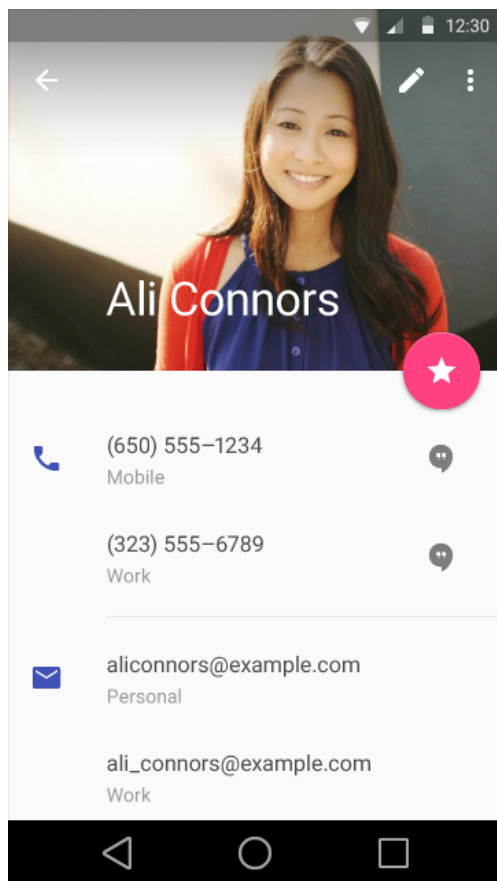
字体比例和基本样式

推荐使用的字体样式

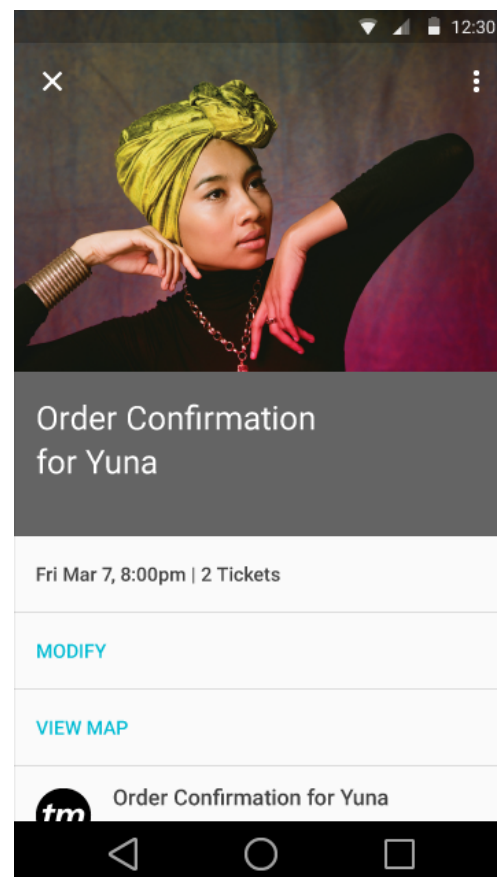
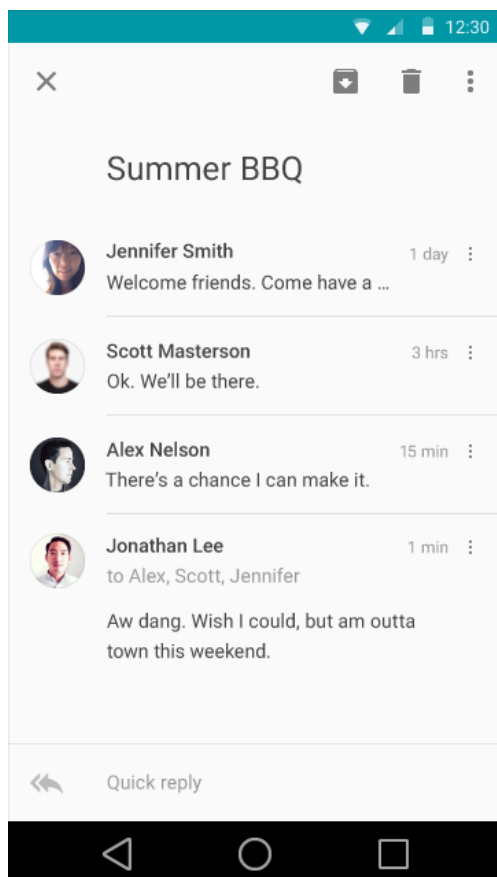
Display 4	Light 112sp
Display 3	Regular 56sp
Display 2	Regular 45sp
Display 1	Regular 34sp
Headline	Regular 24sp
Title	Medium 20sp
Subhead	Regular 16sp (Device), Regular 15sp (Desktop)
Body 2	Medium 14sp (Device), Medium 13sp (Desktop)
Body 1	Regular 14sp (Device), Regular 13sp (Desktop)
Caption	Regular 12sp
Menu	Medium 14sp (Device), Medium 13sp (Desktop)
Button	MEDIUM (ALL CAPS) 14sp

过多的字体尺寸和样式可以轻松毁掉任何一个布局。字体排版的缩放是包含了有限个字体尺寸的集合，并且他们能够与布局结构和谐的共存。最基本的样式集合就是基于12、14、16、20和34号的字体排版缩放。

这些尺寸和样式在传统应用场合中让内容密度和阅读舒适度取得平衡。字体尺寸是通过SP（可缩放像素数，scaleable pixels）指定的，让大尺寸字体获得更好的可接受度。

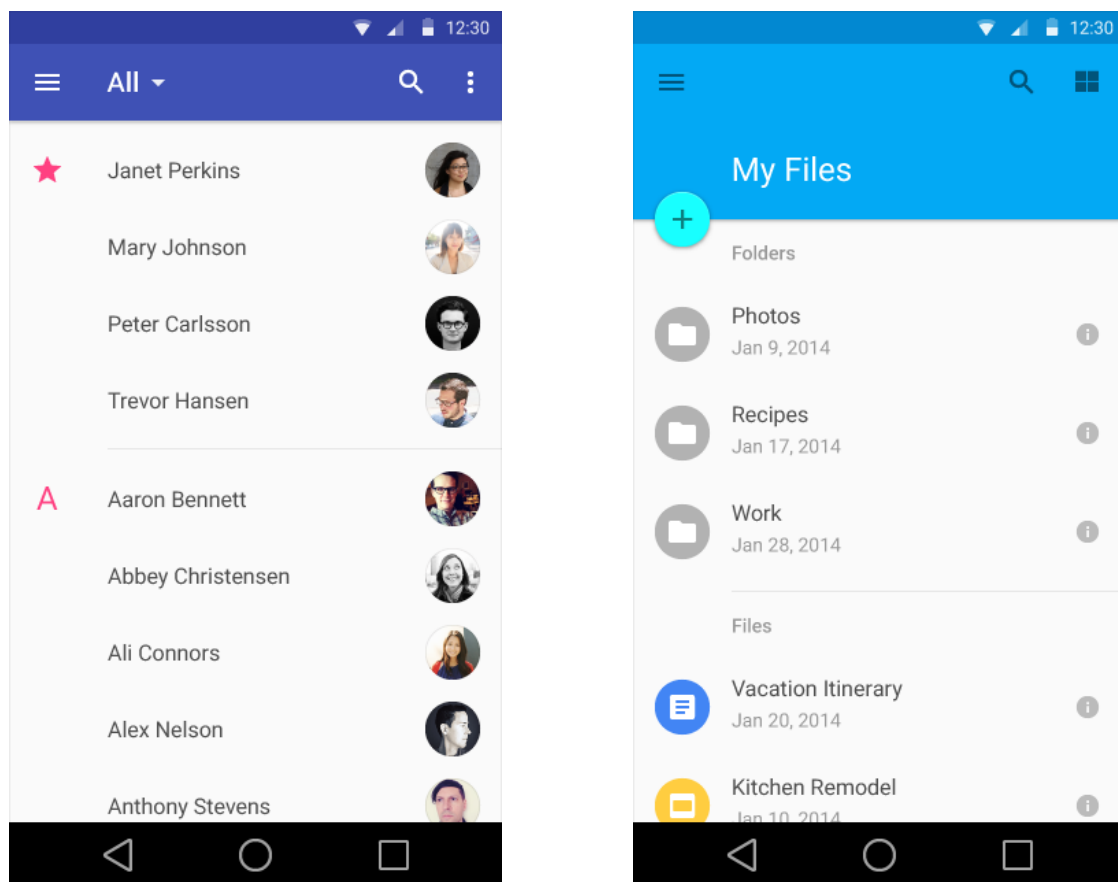


使用Display字体样式的案例



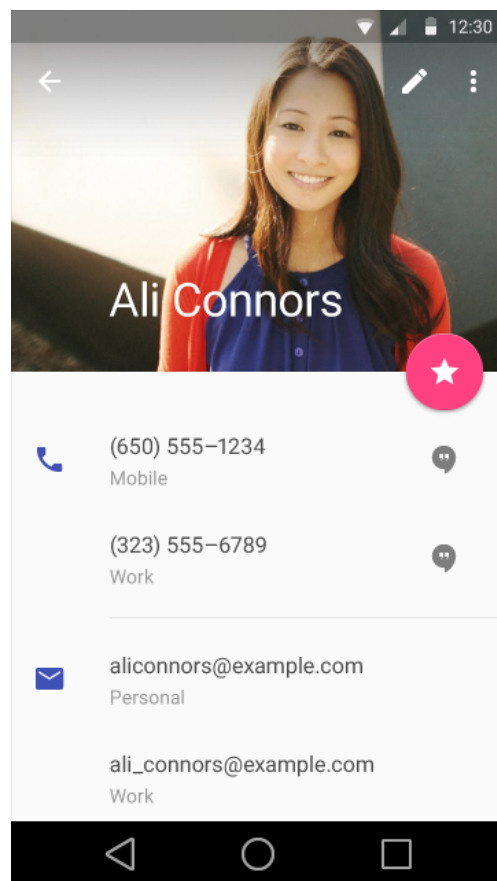
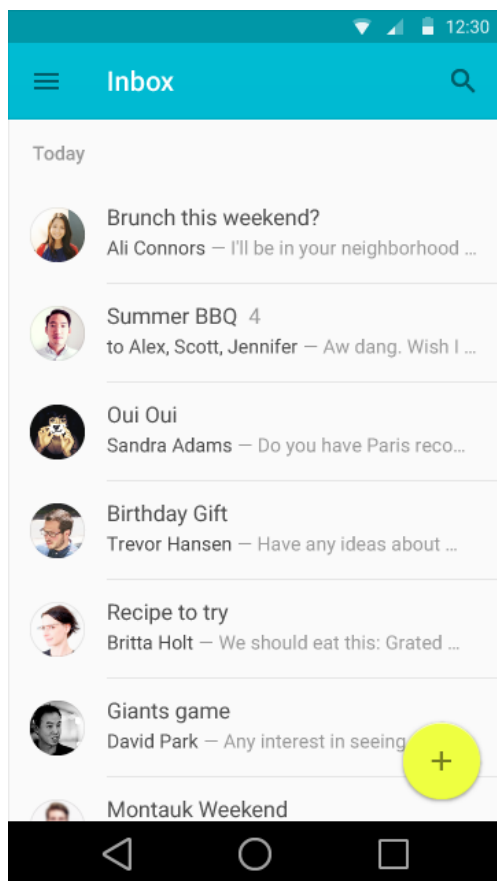
使用Headline字体样式的案例

不论其余因素如何变化，App Bar中出现的所有标题都要使用“标题”样式

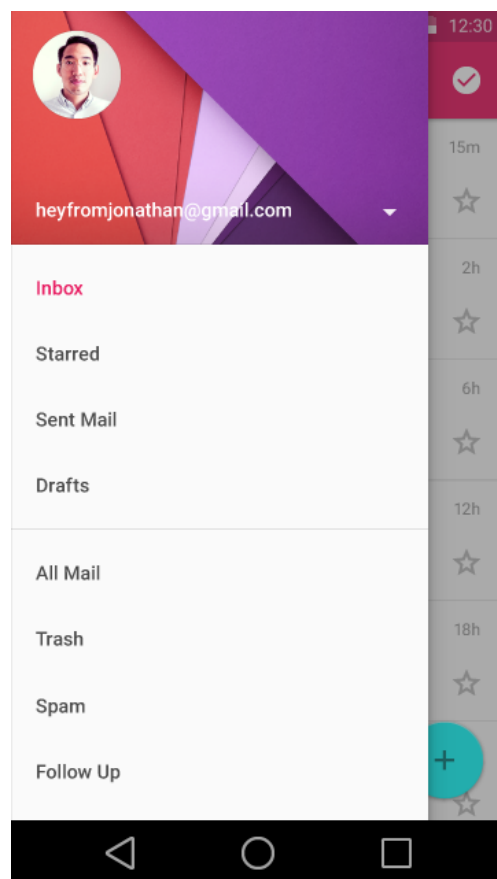
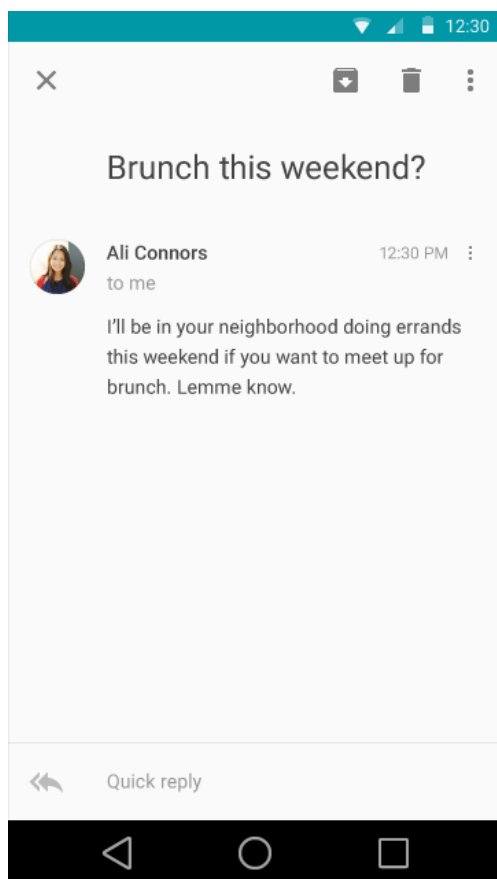


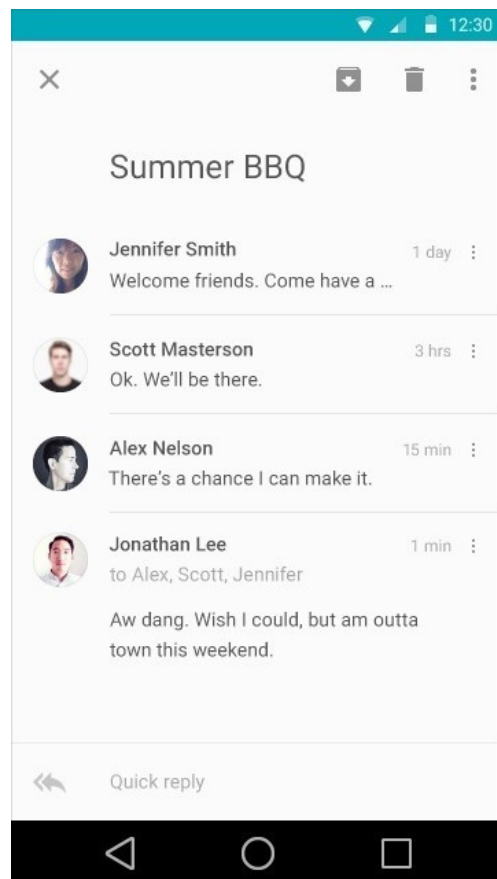
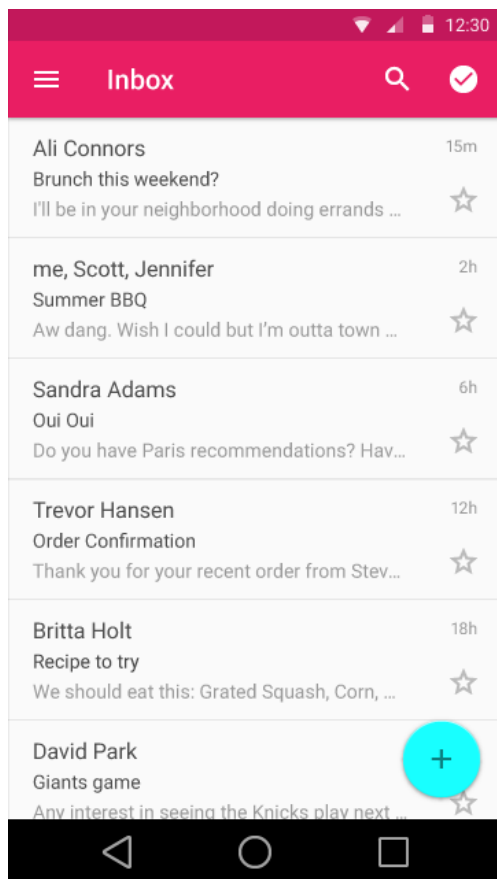
使用Title 字体样式的案例

有些场景下，会使用稍大的Subhead字体样式，而不是采用稍小的Body字体样式。在信息需要小块小块呈递的案例中可以采用这种字体样式。

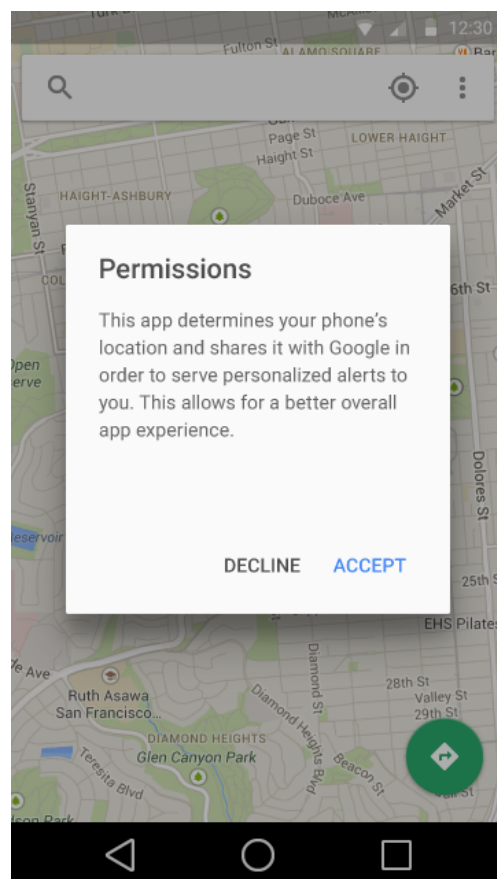
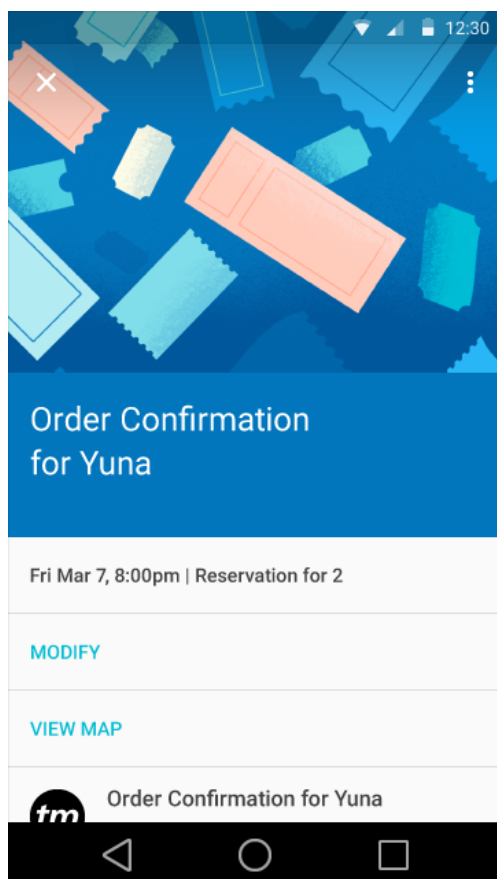


使用Subhead字体样式的案例





使用Body字体样式的案例



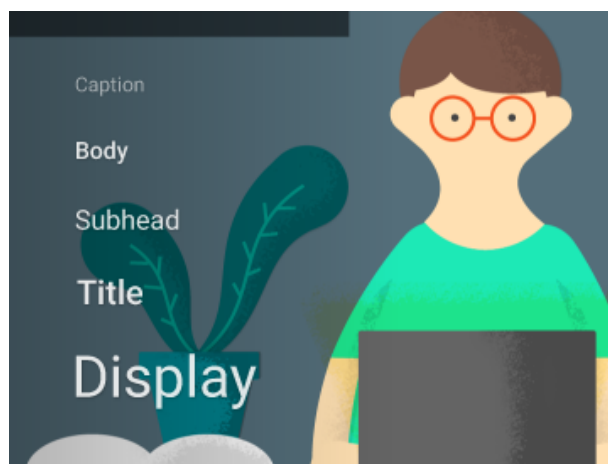
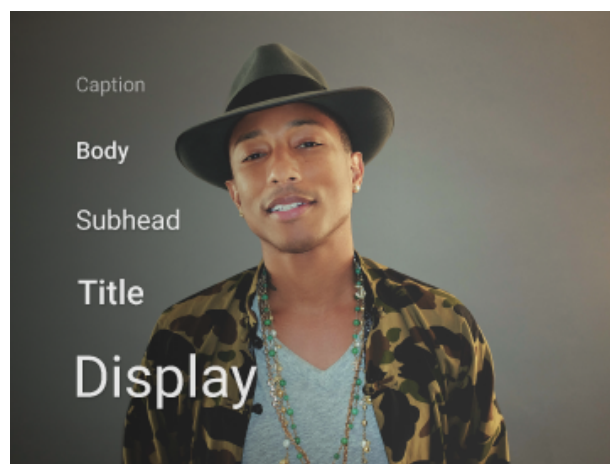
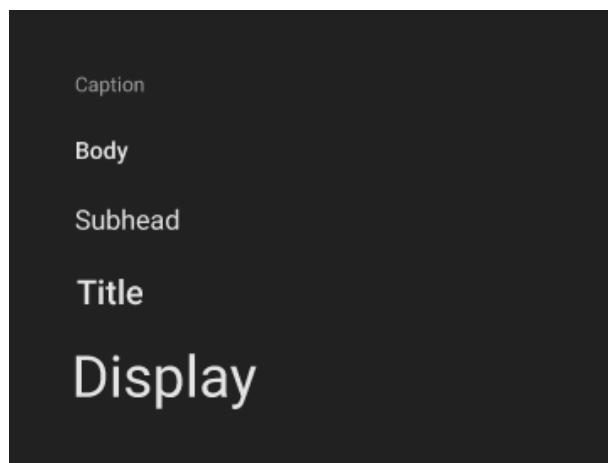
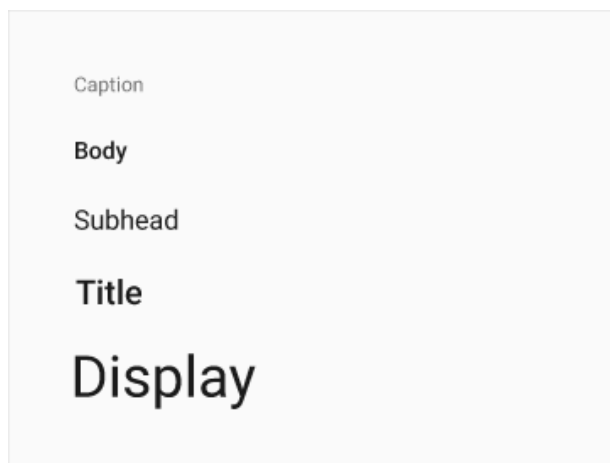
基本色/色彩对比

最基本的常识是，相同颜色的背景和文字是无法阅读的。但有些人不知道的是，带有过强对比度的文本会有些炫目，同样难以阅读。这一点在深色背景下尤其明显。

要获得良好的辨识度，文本应当满足一个最低的对比度，也就是4.5:1（依据明度计算）。7:1的对比度是最适合阅读的。

这些色彩的组合同样要考虑到某些非典型用户对于对比度的反应。

Display 4	Black 54%
Display 3	Black 54%
Display 2	Black 54%
Display 1	Black 54%
Headline	Black 87%
Title	Black 87%
Subhead	Black 87%
Body 2	Black 87%
Body 1	Black 87%
Caption	Black 54%
Menu	Black 87%
Button	BLACK 87%

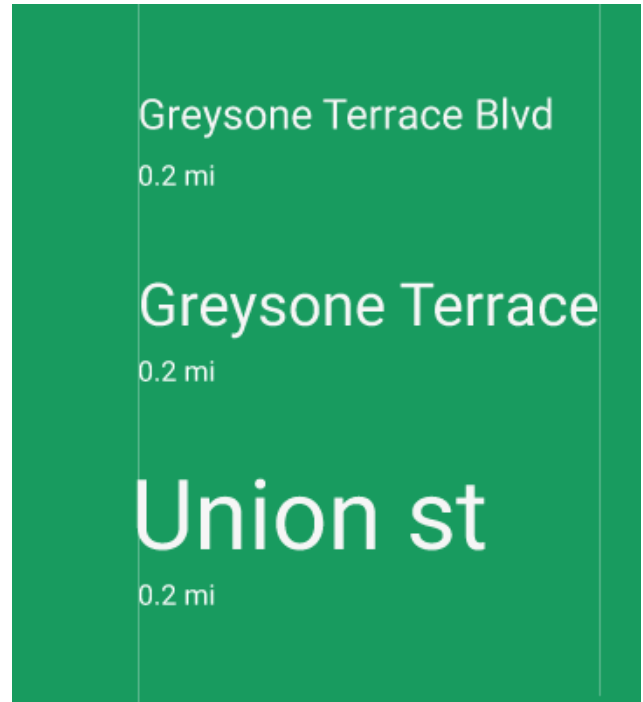


大尺寸/动态字体尺寸

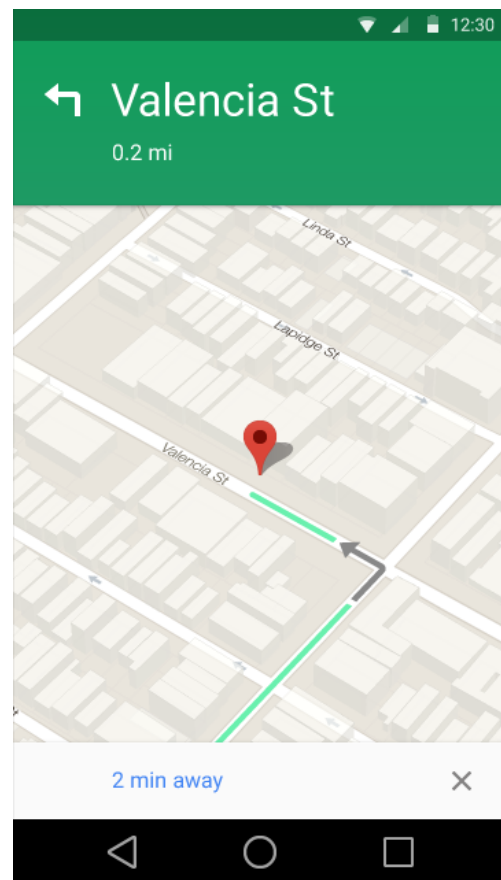
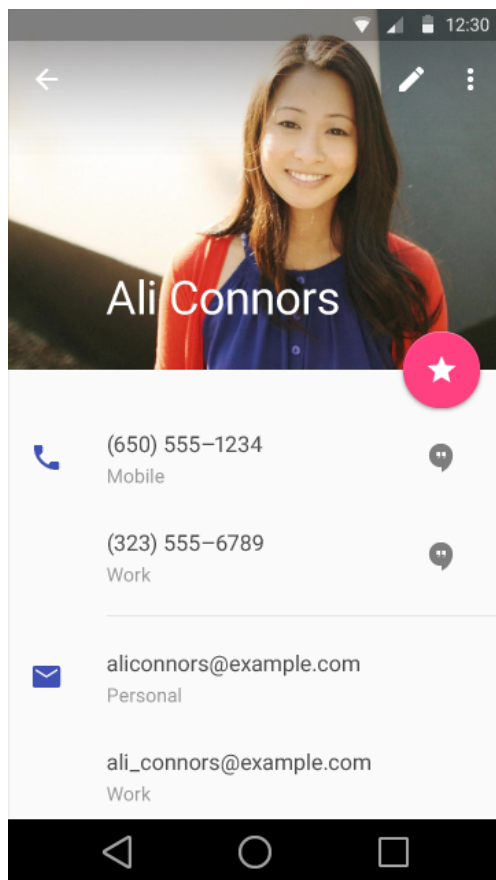
如过恰当运用，大尺寸的字体可以让应用充满趣味，区分布局，而且用户可以快速解读内容。

动态字体尺寸让大号字可以在文本长度未知的情况下保持在容器之内。动态尺寸是根据可用空间和预估的字符空间，从字体排版缩放中选择的。

除非万不得已，非常不推荐轻率地的使用小号字来适配较小的容器。



匹配案例



行高

行高要根据样式中不同的尺寸和字重而分别设置，从而提高可读性，合理设置间隔。只有 Body，Subhead，Headline，以及稍小的 Display 样式中采用自动换行。其他字体样式保持单行。

1B Display 2

Type 45sp, Leading 48pt

1A Display 1

Type 34sp, Leading 40pt

2 Headline

Type 24sp, Leading 32pt

3B Subhead 2

Type 15sp and 16sp, Leading 28pt

3A Subhead 1

Type 15sp and 16sp, Leading 24pt

4B Body 2

Type 13sp and 14sp, Leading 24pt

4A Body 1

Type 13sp and 14sp, Leading 20pt

1B Lorem ipsum
dolor sit.

3A Lorem ipsum dolor sit amet, consec-
tetuer adipiscing elit, at diam nonum-
my nibh euismod tincidunt ut laoreet
vel commodo consequat.

40dp

1A Lorem ipsum
dolor sit amet.

3A Lorem ipsum dolor sit amet, consec-
tetuer adipiscing elit, at diam nonum-
my nibh euismod tincidunt ut laoreet
vel commodo consequat.

40dp

2 Lorem ipsum dolor sit
amet, consectetuer.

4A Lorem ipsum dolor sit amet, consectetuer
adipiscing elit, at diam nonummy nibh
euismod tincidunt ut laoreet vel commo-
do consequat. Duis autem vel eum iriure

32dp

3A Lorem ipsum dolor sit amet, consectetuer
adipiscing elit, at diam.

4A Lorem ipsum dolor sit amet, consectetuer
adipiscing elit, at diam nonummy nibh
euismod tincidunt ut laoreet vel commo-
do consequat. Duis autem vel eum iriure

32dp

3B Lorem ipsum dolor sit amet, consectetuer
adipiscing elit, at diam nonummy nibh
euismod tincidunt ut laoreet vel commodo
consequat. Duis autem

4B Lorem ipsum dolor sit amet, consectetuer adipi-
scing elit, at diam nonummy nibh euismod tinci-
dunt ut laoreet vel commodo consequat. Duis
autem vel eum iriure

换行规则/连字符

Lorem ipsum dolor sit amet, consectetur adipiscing elit, at diam nonummy nibh euismod tincidunt ut laoreet vel commodo consequat. Duis autem vel eum iriure dolor.

Magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip. lobortis nisl ut aliquip ex ea commodo esse, autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie

Ex ea commodo consequat. Duis autem vel eum iriure dolor hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et. Ut wisi minim veniam, quis nostrud exerci tation ullamcorper suscipitlobortis nisl ut aliquip ex ea commodo consequat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, at 1
diam nonummy nibh euismod tincidunt ut laoreet vel 2
commodo consequat. Duis autem vel eum iriure dolor.

Magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip. lobortis nisl ut aliquip ex ea commodo, 3
autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie 4

Ex ea commodo consequat. Duis autem vel eum iriure dolor hendrerit in vulputate velit esse molestie consequat, velillum dolore eu feugiat nulla facilisis at vero eros et. Ut wisi 5
minim veniam, quis nostrud exerci tation ullamcorper suscip- 6
itlobortis nisl ut aliquip ex ea commodo consequat.

- 1 Short words hanging
- 2 Big dent
- 3 Big dent
- 4 Orphan
- 5 Big dent
- 6 Hyphenated word

正确

不正确：1.短语 2.凹进 3.凹进 4孤立 5凹进 6.连字符

每行平均字符长度

根据Baymard 学院的建议，我们考虑到了行长度对可读性的影响：

“好的阅读体验，每行大约60个字符。恰当字符数量是提高可读性的关键。

“如果行中文本太长，用户的注意力很难集中在文本上，用户不知道何处开始阅读，何处结束阅读。而且暂停后，在大块文本区域中很难继续阅读。”

“如果航中文本太短，那么用户的眼睛会来回穿梭，打破了阅读节奏感。而且会给给予阅读者压力，可能会出现跳读”

资料来源: Readability: the Optimal Line Length

Body Text

0

40

60

90

Too narrow

Lorem ipsum dolor sit amet,
consectetuer adipiscing elit,
sed diam nonummy nibh
euismod tincidunt ut laoreet.

Ideal range

Lorem ipsum dolor sit amet, consectetur adipiscing elit, aliquip sed diam
nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam nostrud
erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation velit
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Too wide

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet
dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper sus-
cipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum liriure dolor in hendrerit in vulputate
velit esse molestie consequat, vel illum dolore eu feugiat null.

字距

Display 4

Tracking -10

Display 3

Tracking -5pt

Display 2

Tracking 0pt

Display 1

Tracking 0pt

Headline

Tracking 0pt

Title

Tracking 5pt

Subhead

Tracking 10pt

Body 2

Tracking 10pt

Body 1

Tracking 10pt

Caption

Tracking 20pt

Menu

Tracking 10pt

Button

TRACKING 10PT

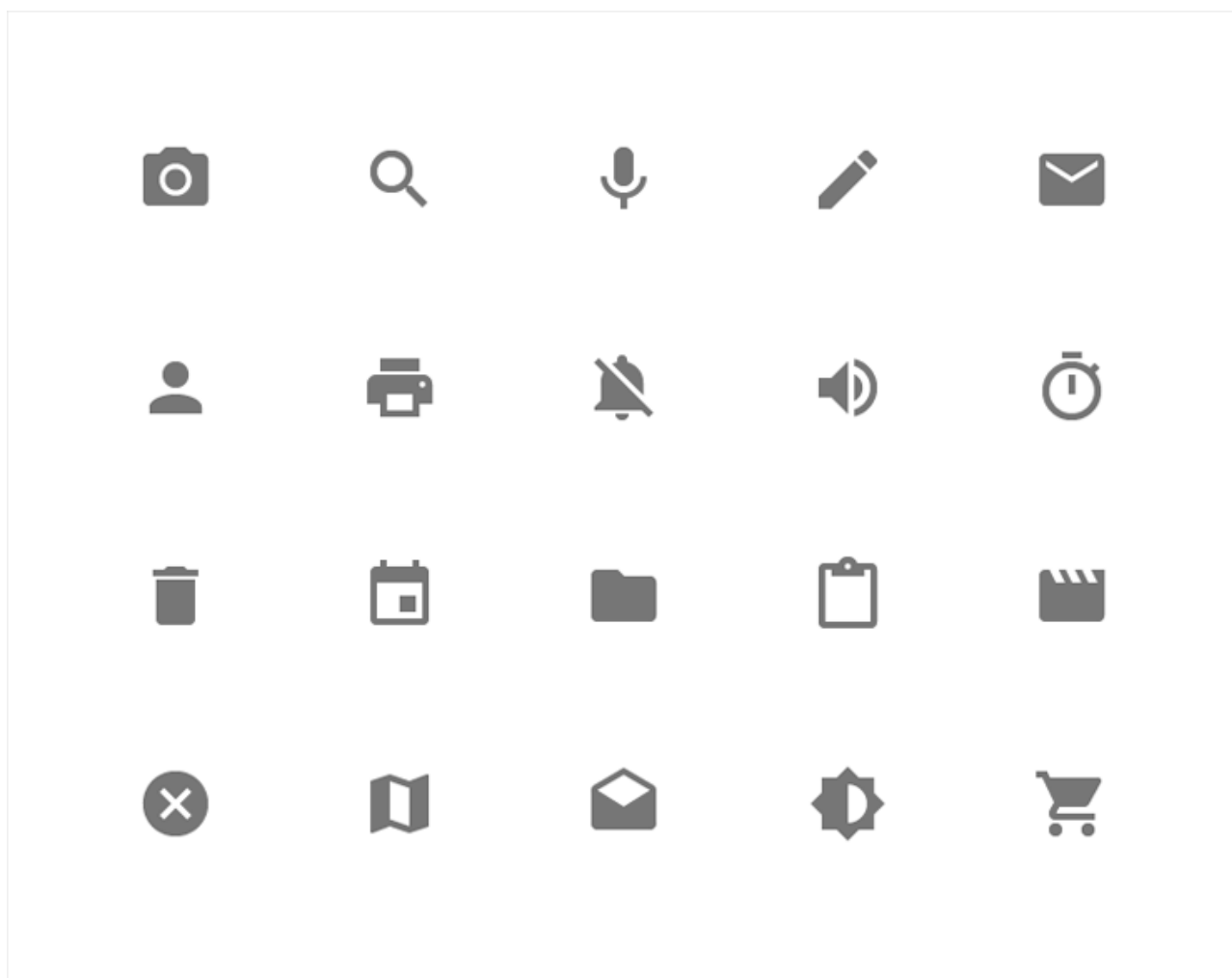
3.3 图标

系统图标

定义

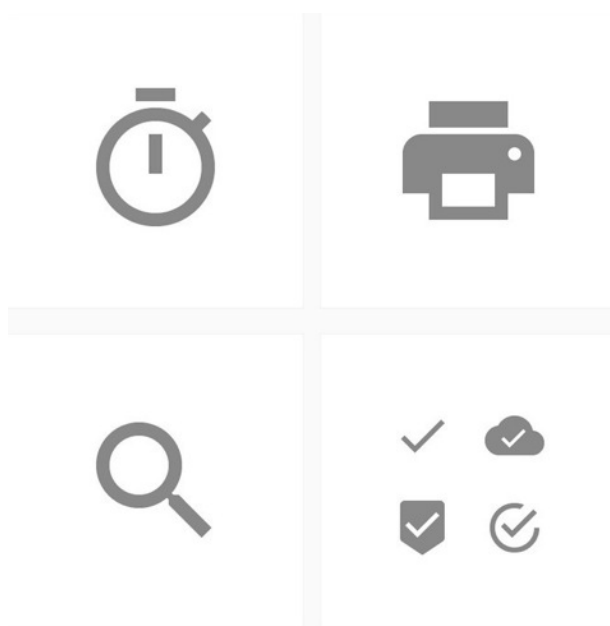
系统图标，或者UI图标，往往象征着命令、文件、设备或者目录。系统图标也可以用来代表一些操作，例如删除、打印和保存。

系统图标的设计必须简约、富有现代感、友好，有时候要让人感到惊奇。图标必须在形式上极简，回归到最本质的形式诠释理念。确保小尺寸情况下的可读性和明晰性。



设计原则

形状鲜明，多多采用几何形状，对称、且一致的图标，质量最高，这类图标的简单且鲜明



网格、比例、尺寸

绘制图标时，要灵活运用图标网格，基于网格绘制比例和关键部分。让扁平化的几何形状对齐网格，形成图标的形式感。

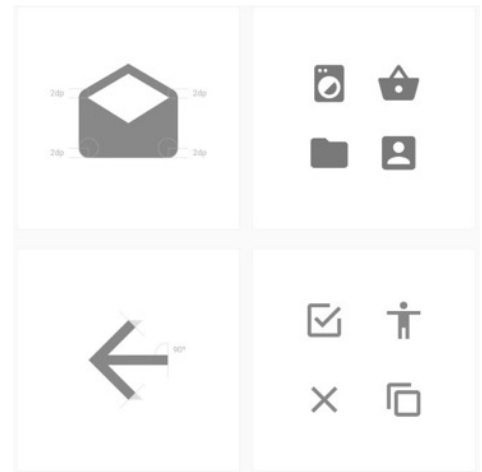
图标网格由四种基本的几何形状构成，使用特定的尺寸，确保约束设计，保证图标形状、比例、视觉一致。

在应用设计中，主要使用2种尺寸不同的图标：状态栏和上下文图标，以及工具栏和桌面图标。

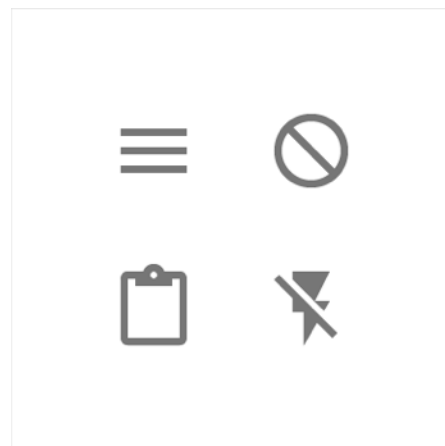
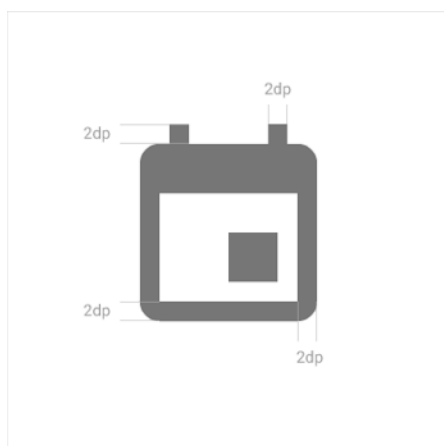


圆角

在矩形和正方形中广泛使用圆角。避免突兀和锯齿感，线图标的圆角要稍微分明一些。

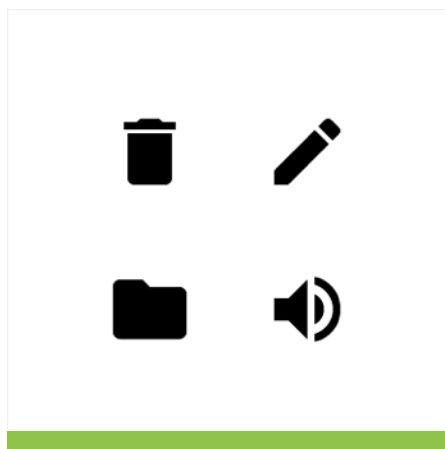


不同的系统图标尺寸需要不同的圆角，从而优化视觉一致性

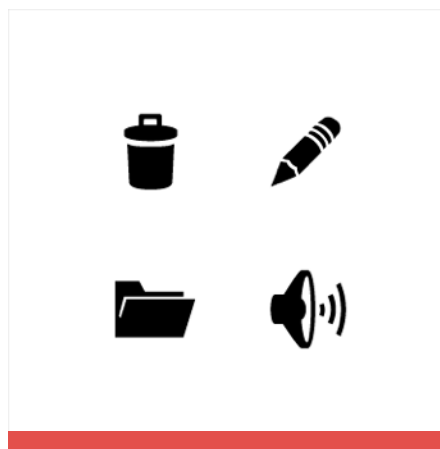


一致性

一致性很重要。在不同的应用中，尽可能使用系统图标保证一致性



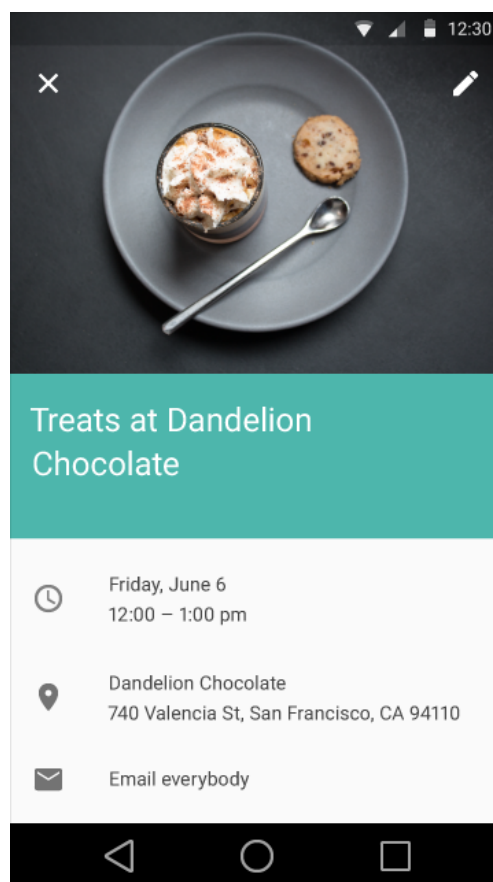
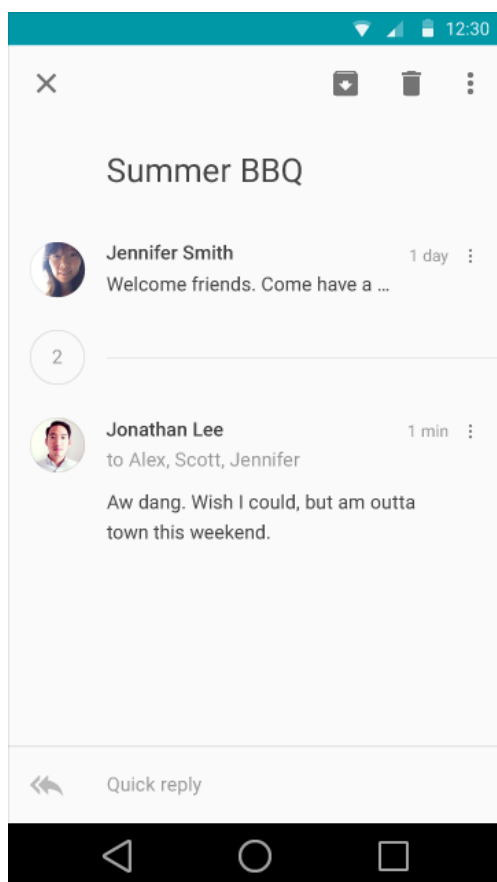
正确



不正确

应用中得图标

建议将图标放置在24dp的边界框中，视网格线而定。



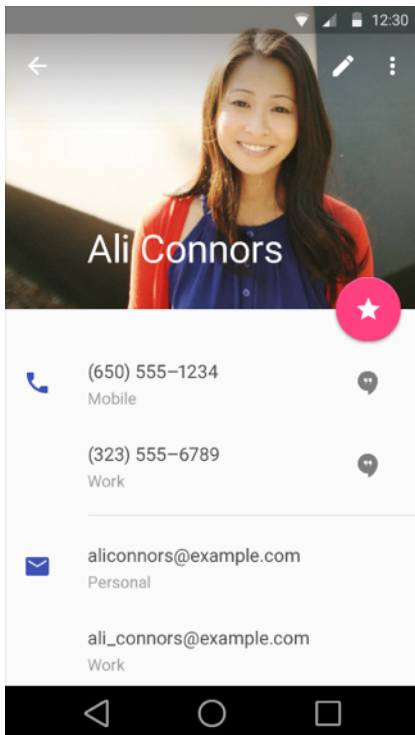
3.4 图像



在material design中，图像——无论是插画还是照片——都应该精心构造，充满魔力，而不应该让用户感到不自然、过度做作。样式风格应该是可读、令人愉悦、城市。要强调出实体感、材质感、深度感、令人惊喜的用色，以及上下文。通过图像的设计、合理利用，打造出美丽、体贴的用户界面

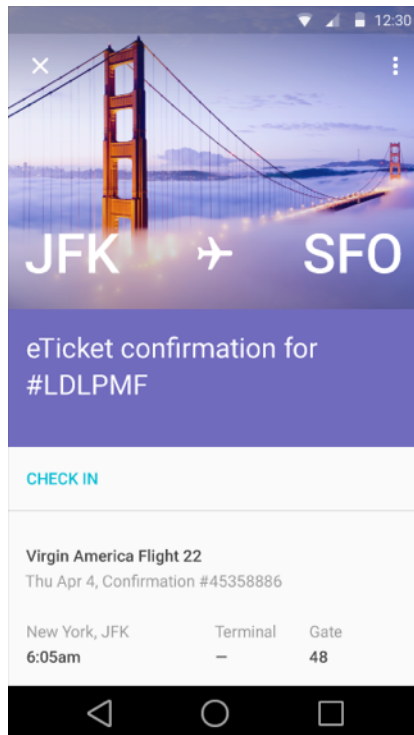
设计原则

在使用插画或图像提高使用者体验时，选择和人物相关、充满信息、令人愉悦的图像



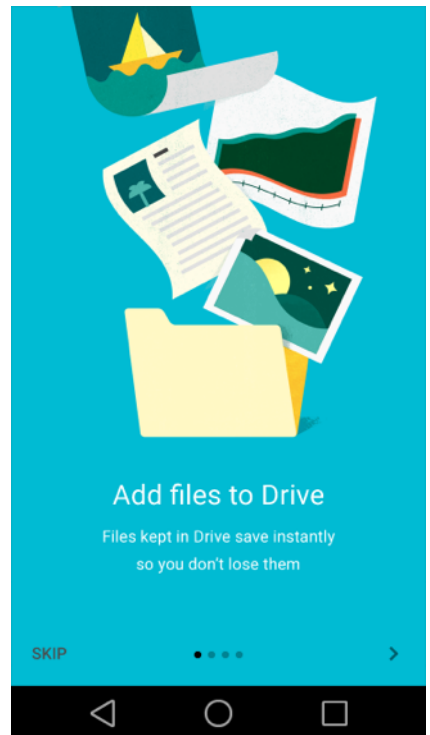
和人物相关的图像

使用人物图像可以建立情感化联系



充满信息的图像

呈递特定的信息，促进用户理解信息，打造充满智能感的界面

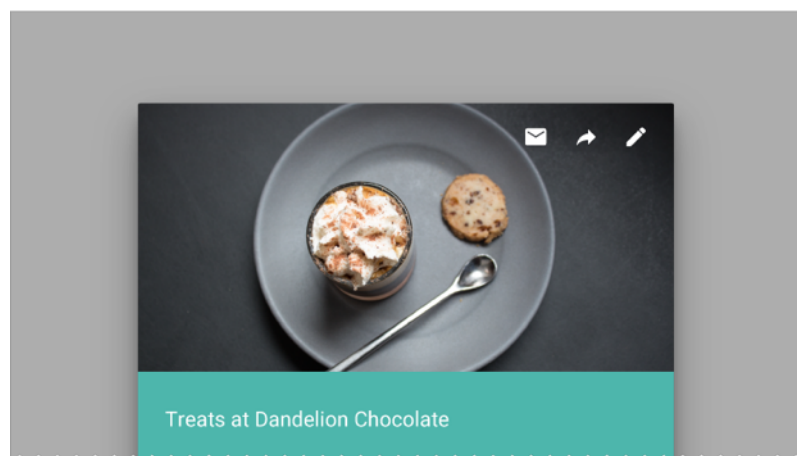


令人愉悦

用惊艳的并且和信息相关的图像来愉悦用户，让体验充满魔力

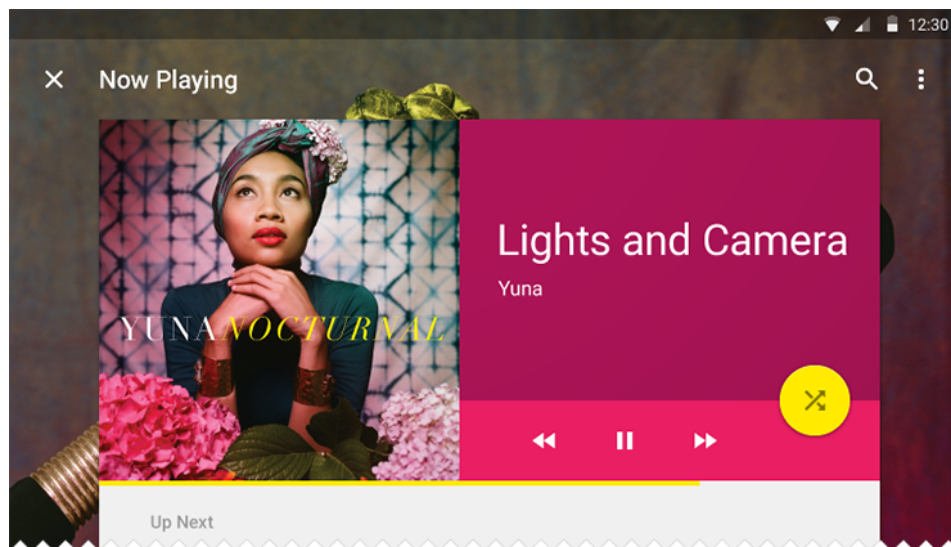
场景化价值

图像使用要符合逻辑，确保图像是动态的、和上下文场景相关。在用户需要之前提供视觉信息，进而提高用户体验



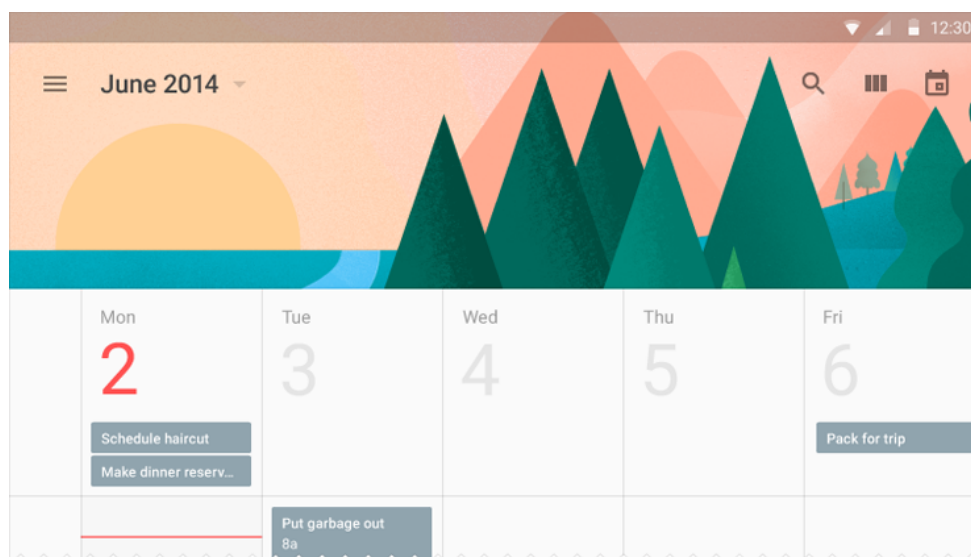
沉浸体验

不要害怕遮挡，大胆的将色彩、内容和图像整合。



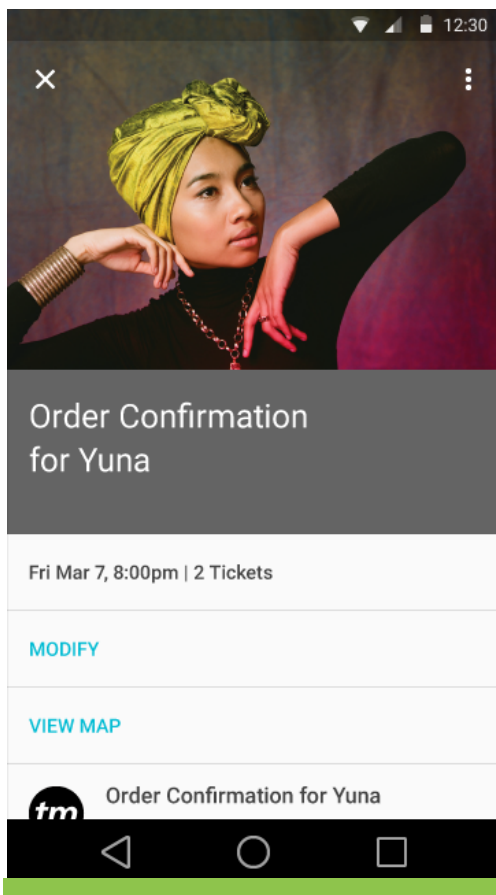
无处不在

创意性的为信息添加视觉图形，提高信息的表现力

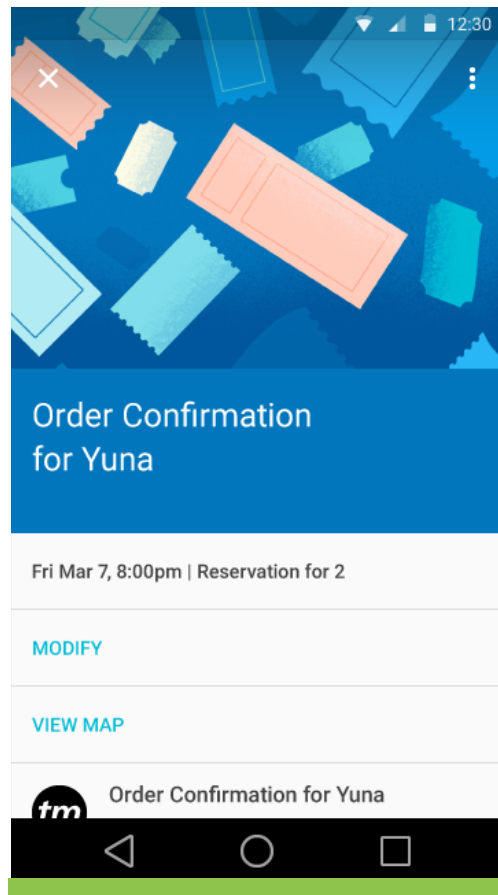


最佳案例

使用多媒体



正确：某些特定的内容，在表现时，首先可以考虑用照片表达



正确：当有多种信息时，插图可以一目了然的表达你所阐述的内容



正确：如果使用照片来传达概念，需要精选照片，考虑到照片的隐喻性。



不正确：不要使用图库里面的一些泛泛的照片

少用图库

图像的使用要与众不同，富有创意

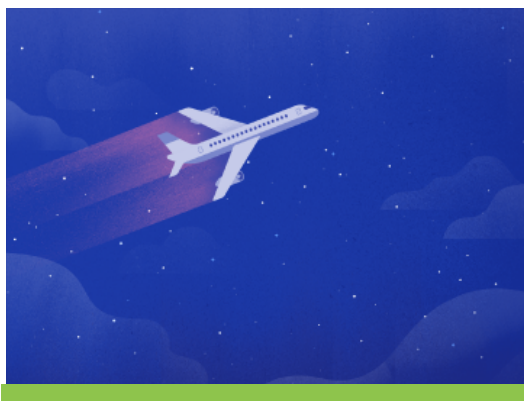
特定的内容要使用特定的照片。不要使用图库中那些照片，过于泛泛且代表性不强。图库中的照片往往很普通，且图像的解释力不够。



使用的图片要有视觉焦点

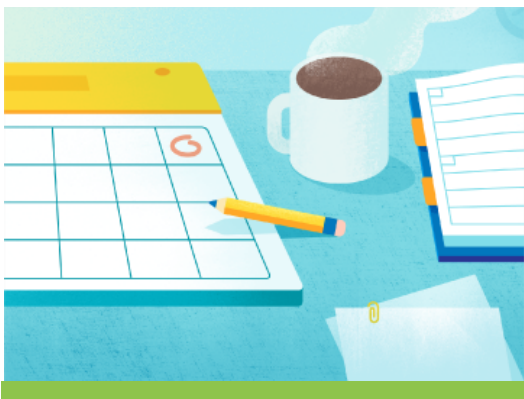
所选的图像要有明晰的视觉焦点。可以是图像中的某个元素，也可以是整体协调的构图。确保理念能够清晰传达，让用户难忘。





构建故事性

图像要能讲述出富有沉浸性的故事，同时能够提示上下文



不要过度处理

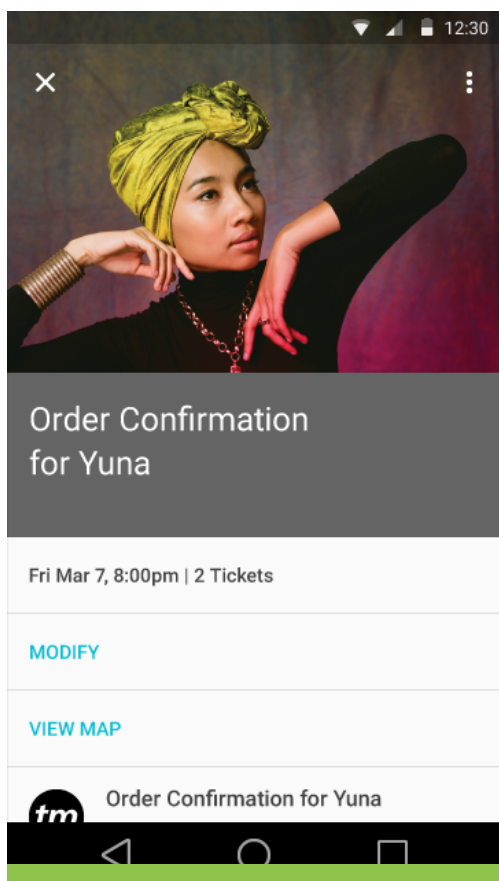
尽量位置图像的完整性。不要用滤镜和高斯模糊



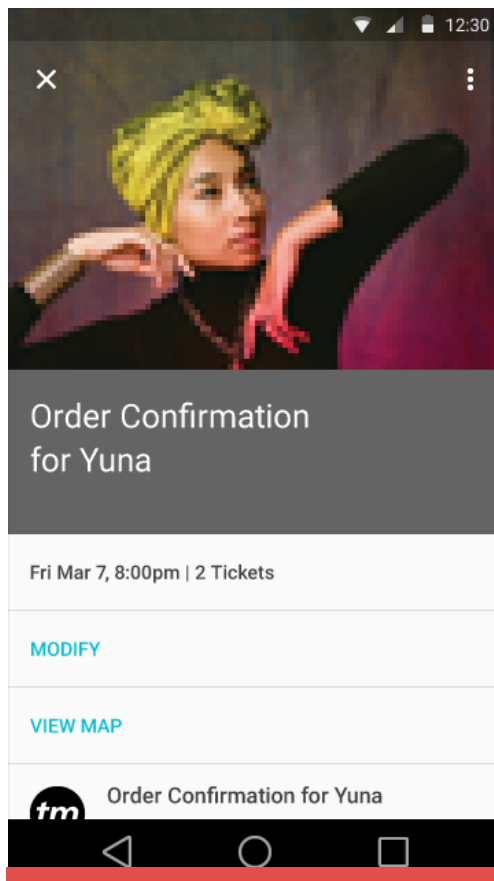
图像和界面的整合

分辨率

确保图像尺寸合适，能够满足不同平台、不同设备的显示需求。推荐使用大图，图像不能出现像素感。在不同的设备和不同的分辨率下要进行测试。



正确：尺寸合理的图像



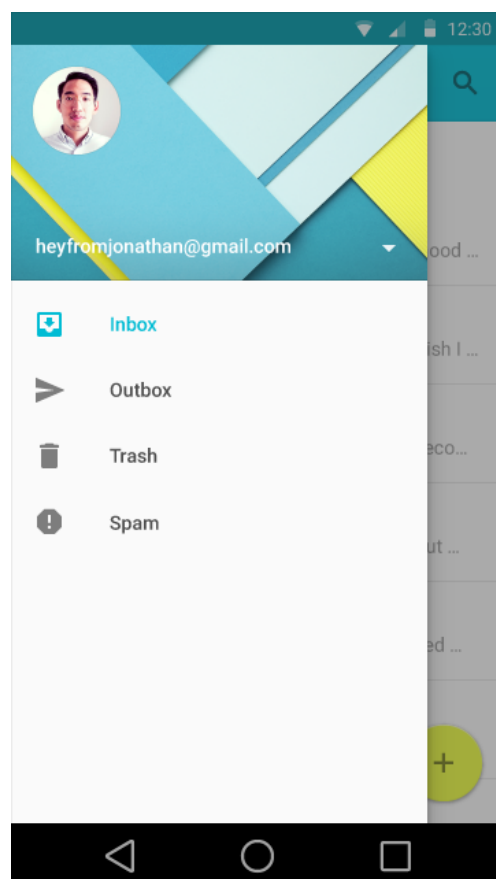
不正确：使用质量糟糕的图像

图像比例

展现图像时，可使用多种比例，用来展示视觉层级



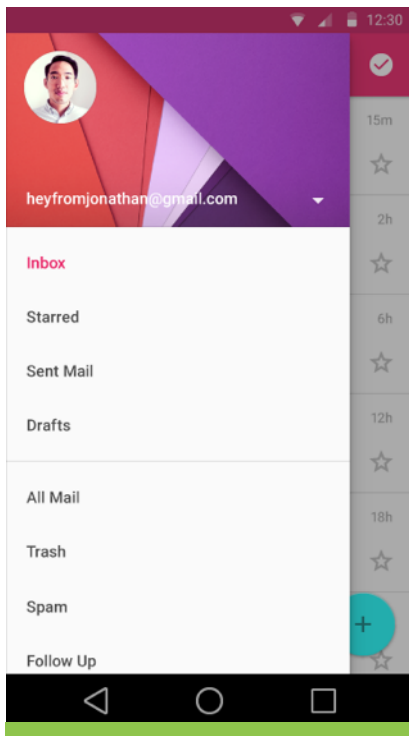
在图库场景中，缩略图尺寸要不尽相同，从而提高内容的层级感



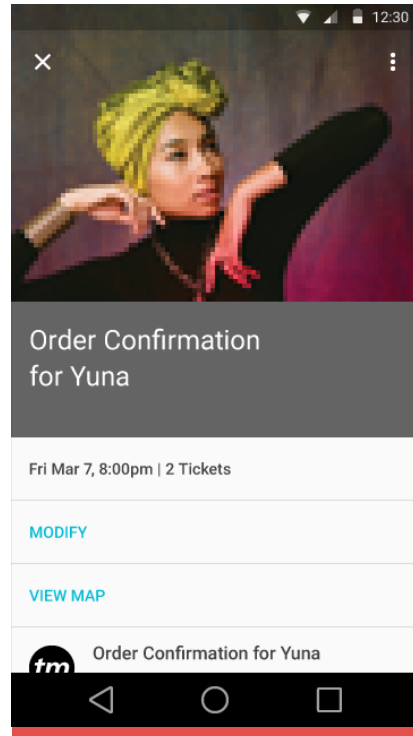
在同一生态系统下，允许有多种内容容器

文本保护

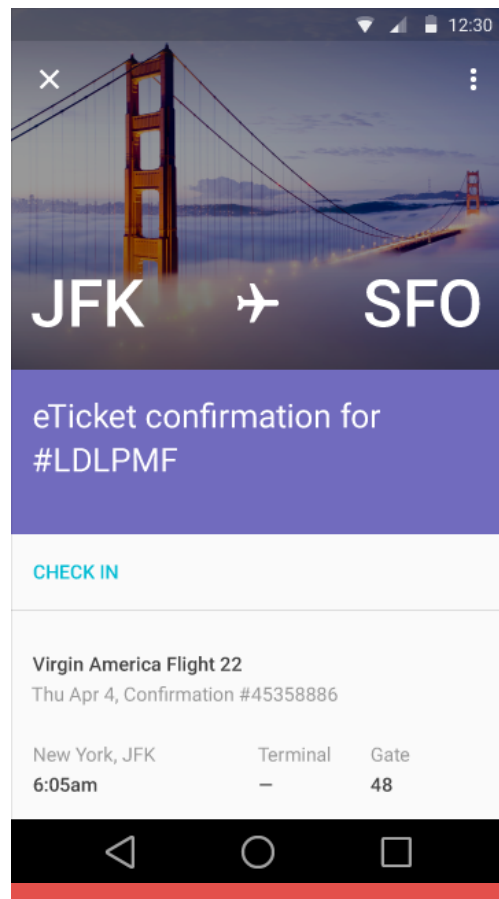
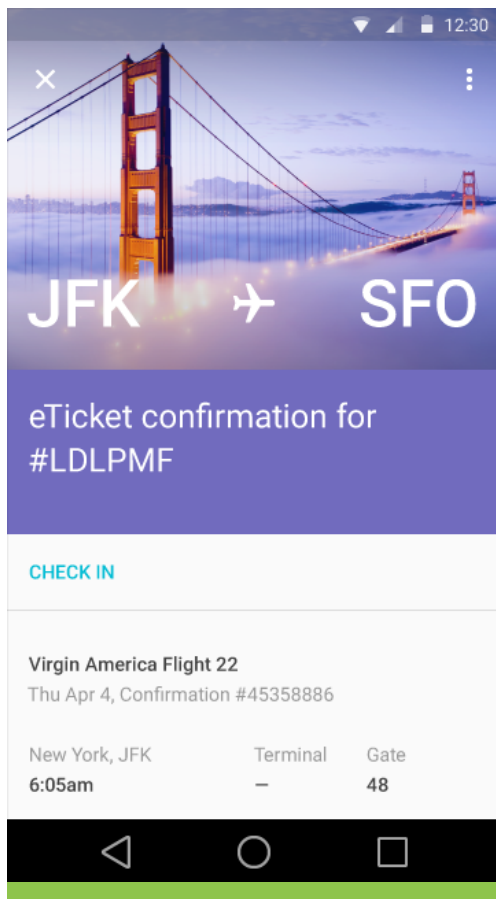
为了确保图像上文本的可读性，建议添加text protection scrim（编者：可能是对图像某些区域进行适当柔化，提高文本可读性）



正确：深色的保护区域，根据内容，理想状况下不透明度是20%-40%。浅色保护区域40%-60%

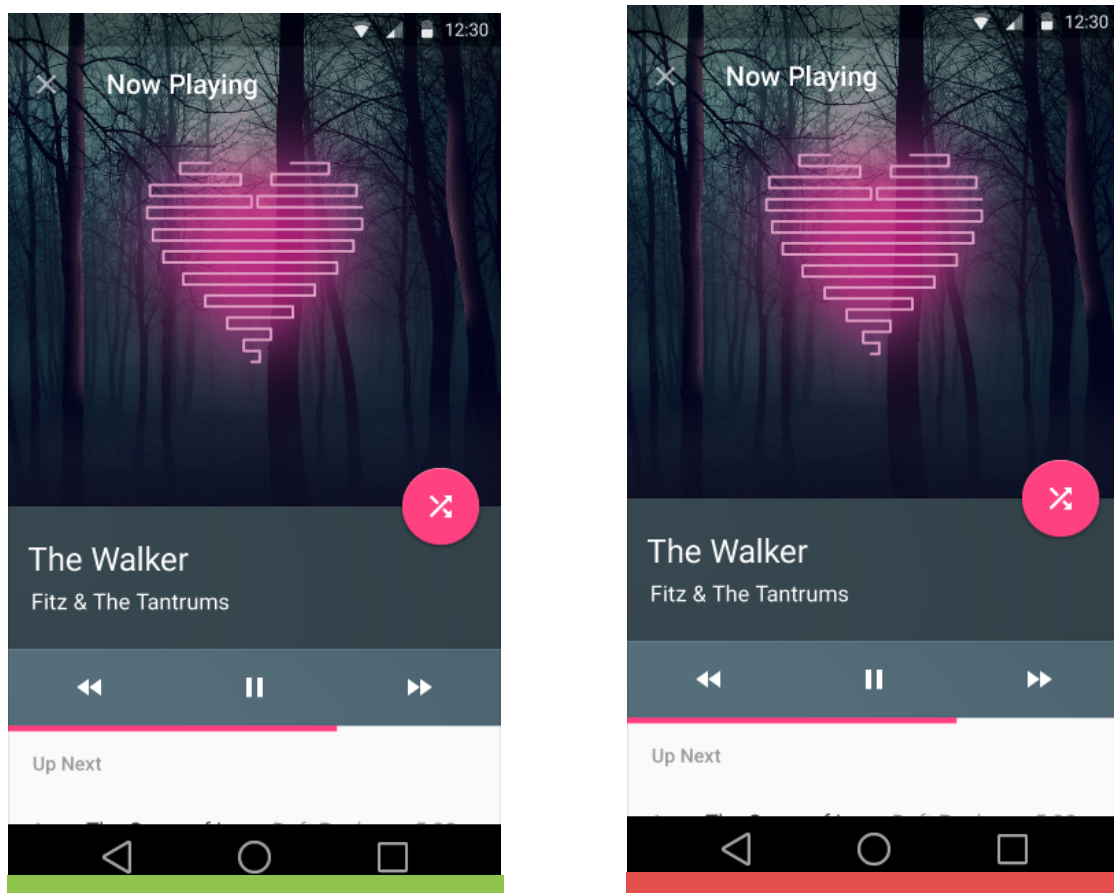


不正确：覆盖在图像之上的保护区域不宜太过明显



正确：在空间充足的情况下，给特定区域添加文本保护，而不是保护区覆盖整个图像

不正确：保护区不能扩张为整个图像



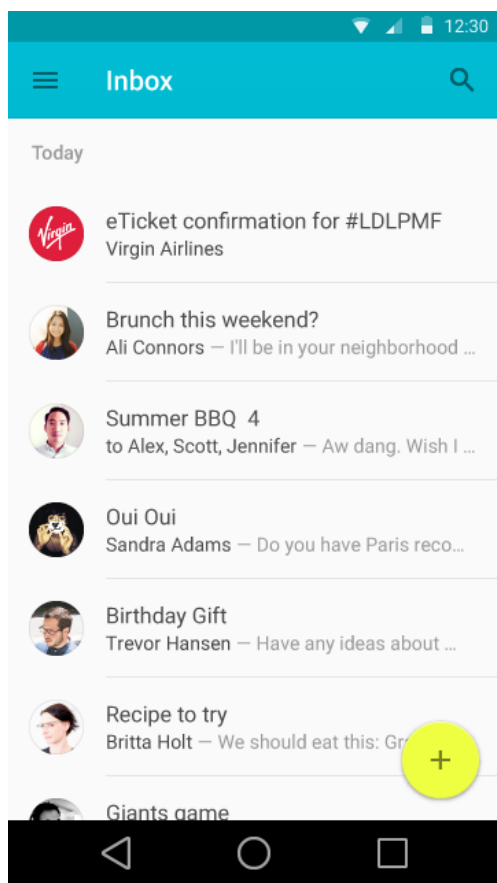
正确：色彩叠加是另外一种保护文本的方式，可被当做设计元素来运用。创建互补色叠加，可以让整体配色鲜明，或者取样自图像颜色的文本叠加，可以提高用户对内容的重视

头像和缩略图

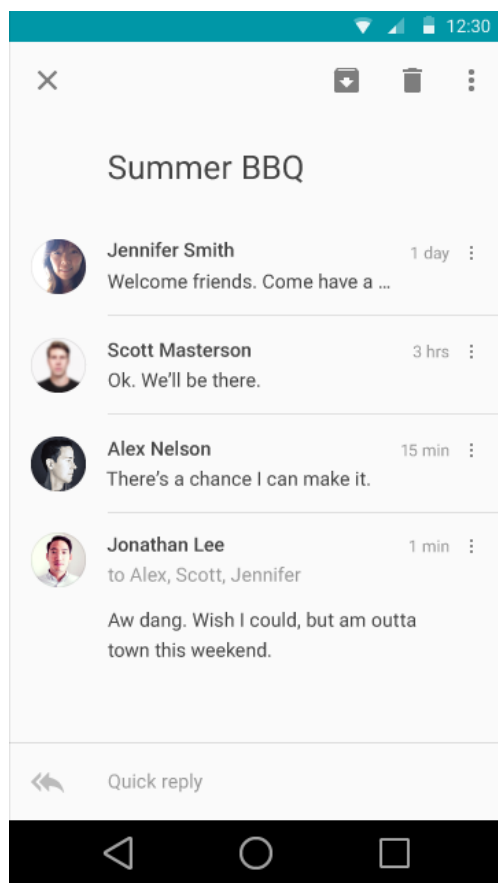
头像和缩略图能够表现真实感和具体内容。既不像照片那样具体的表现，也不像插画那样抽象的表现。大体上，头像和缩略图主要为了引导用户点击，进而进入子内容的主界面

头像可以用来代表人。在用户头像设置上，可以添加一些个性化选项，愉悦用户。当用户使用特定的Logo时，头像可以表示品牌。

缩略图可以暗指更多、更细腻的信息——这里仅提供内容的一部分，吸引用户进一步深入内容。



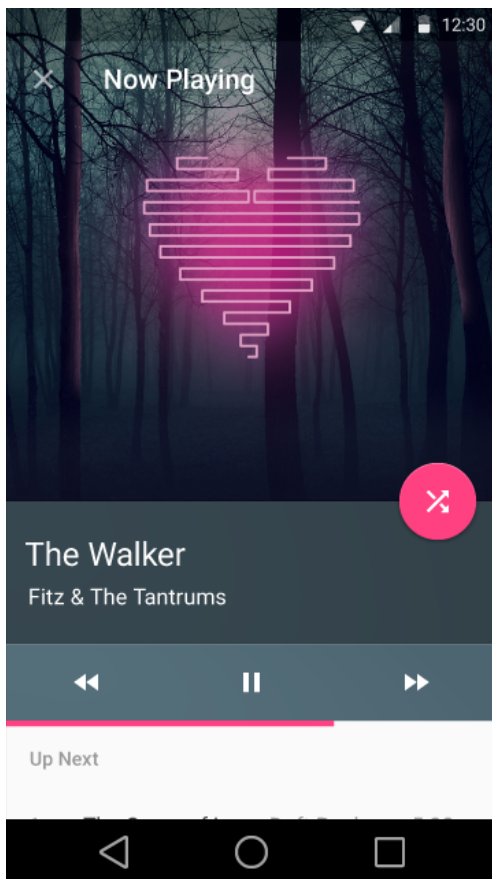
真人头像让应用更具人性化，
而且占用的空间很少



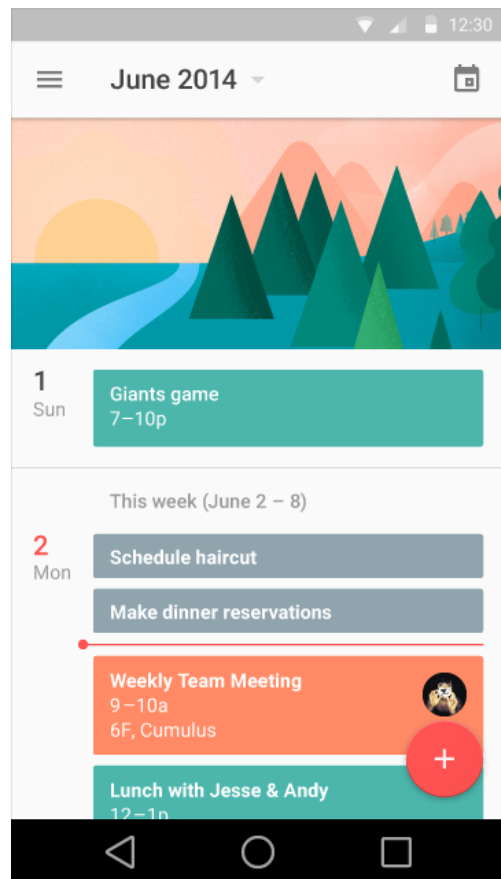
品牌头像能够在用户一瞥的瞬间
传递品牌信息，即便是缩略图的

Hero图像

Hero图像指的是在界面中具有支配地位的图像，例如屏幕顶部的Banner。主要用来吸引用户，提供和上下文相关的内容，加强品牌效应。



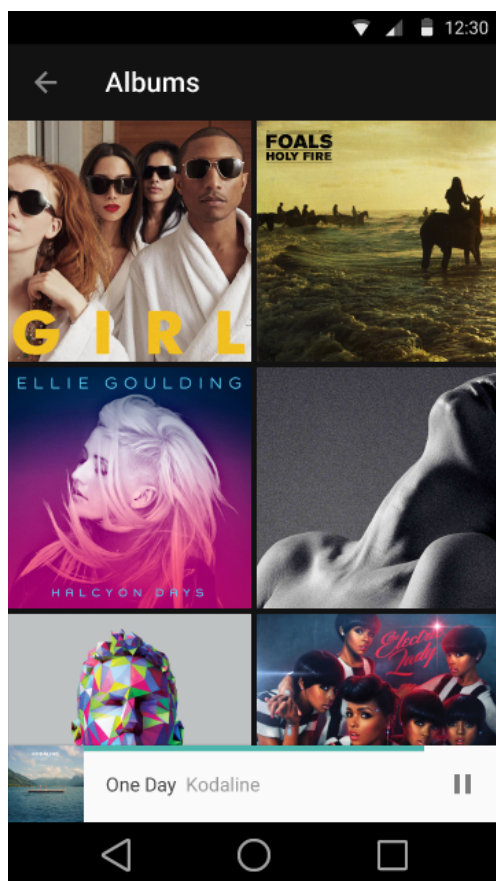
特征图像:主要指的是不同布局中，最显著、最能吸引注意力的图像



集成的Hero图像:往往能够起到协调作用，将布局中不同内容协调一致。但是此类图像不是

图库

图库图像具有极强的视觉吸引力，布局一般不规则，或是网格，或是单个图像。



照片网格



图库图像

4.1 原则

设计原则

Material design 采用了印刷设计中的一些基本工具，比如基准网格线和结构网格线。这种布局设计可以帮助视觉元素可以平滑伸缩，以达成跨平台一致的视觉体验，简化开发流程，并最终帮助设计师制作适用于各种屏幕的应用。

布局设计规范中，鼓励设计师在跨平台设计中使用使用相同的视觉元素、结构网格和整体的边距规则，在不同尺寸的屏幕上让应用有有一致的风格和体验。结构和视觉风格的一致，可以打造出跨平台、一致的使用体验，这些能够给用户提供一种熟悉感和安全感。

在探索布局细节之前，再次思考究竟什么是material design：一种基于卡片的设计。理解卡片的表现和制作的规则是非常重要的。

卡片艺术

在material design中，应用页面上的每个像素就像排列在轻薄的卡片上。卡片有着扁平的背景颜色，能够被用做一系列的用途。一个典型的布局由多个卡片张片组成。

卡片可以容纳像素和视觉元素（比如状态栏（Phone端：译者注）或者系统栏（Pad端：译者注），它们并不属于卡片。）。要让用户感觉视觉元素是印刷在卡片上的，而卡片是位于屏幕下方的。

卡片的排版

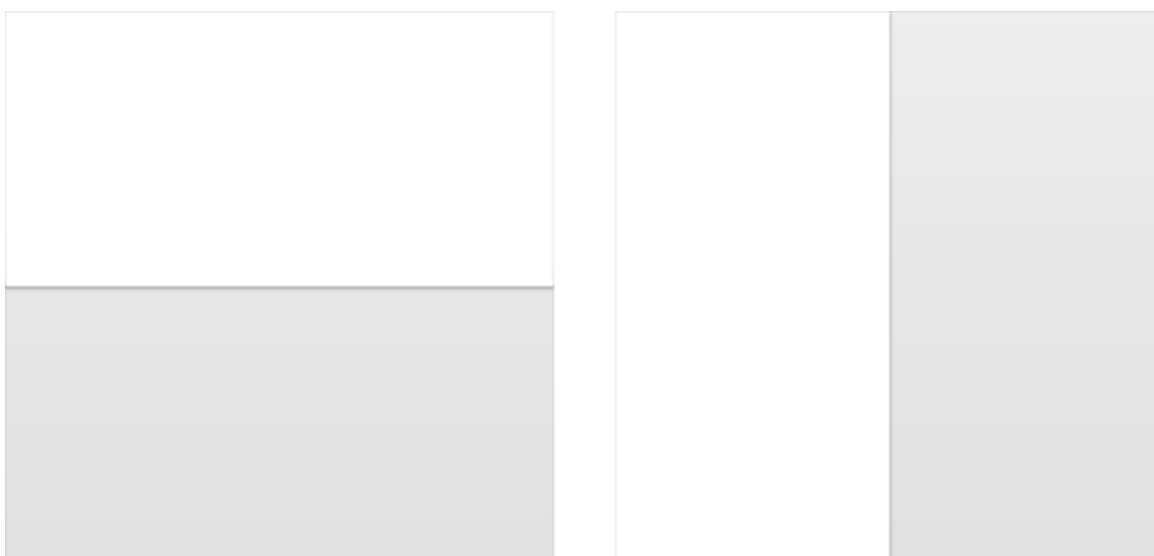
接缝，当两个卡片共用一个边长时，便出现了接缝。两个通过接缝组合在一起的卡片通常一起移动



台阶式的层级

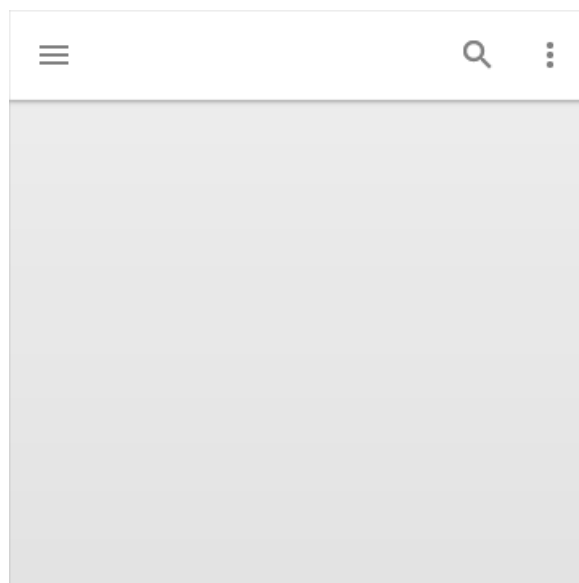
当两个卡片有着不同的Z-轴（x和y分别代表屏幕所在平面上的水平坐标和垂直坐标，而z轴则是垂直于屏幕

：译者注）次序时，这两个卡片便有了前后的层级。因此通，这两个卡片彼此独立的移动

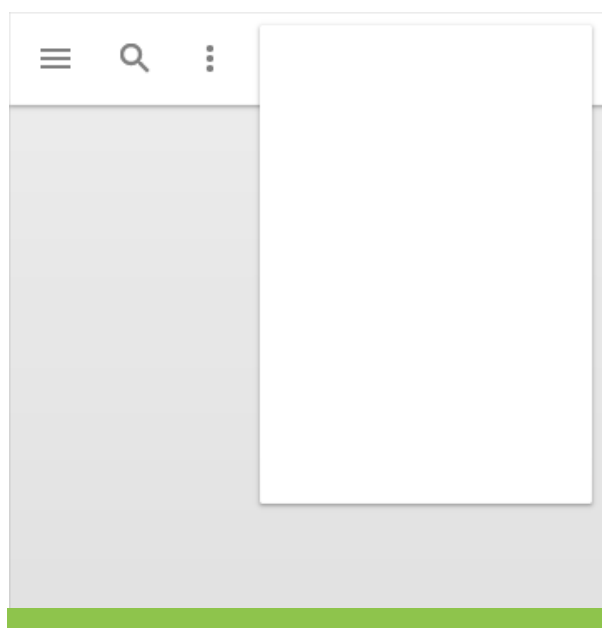


卡片 工具栏

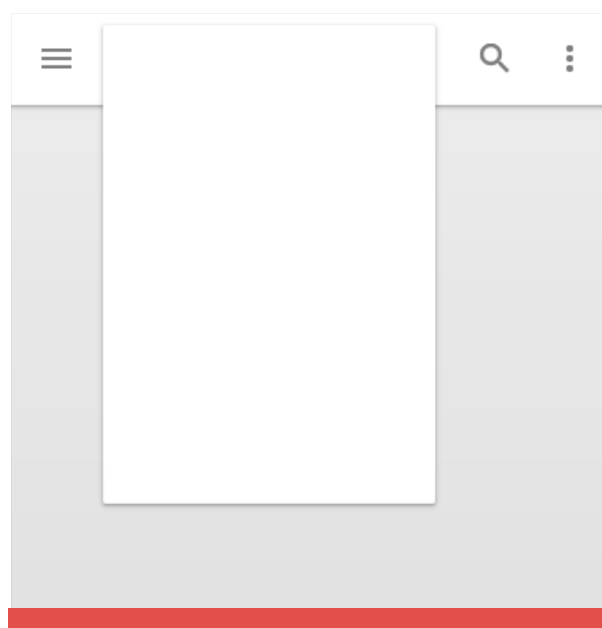
工具栏是专注于当前页面功能操作的小卡片。这些操作经常被聚集起来放置在工具栏的左侧和与右侧。与导航（抽屉菜单或者是返回箭头）相关的操作通常出现在左侧，而那些应用在当前页面内容的操作放在在右侧。



在工具栏左侧和右侧的操作所弹出式的卡片，永远不允许将原卡片割裂。然而，弹出的卡片，工具栏会限制它的宽度，不会长于整个卡片的宽度。

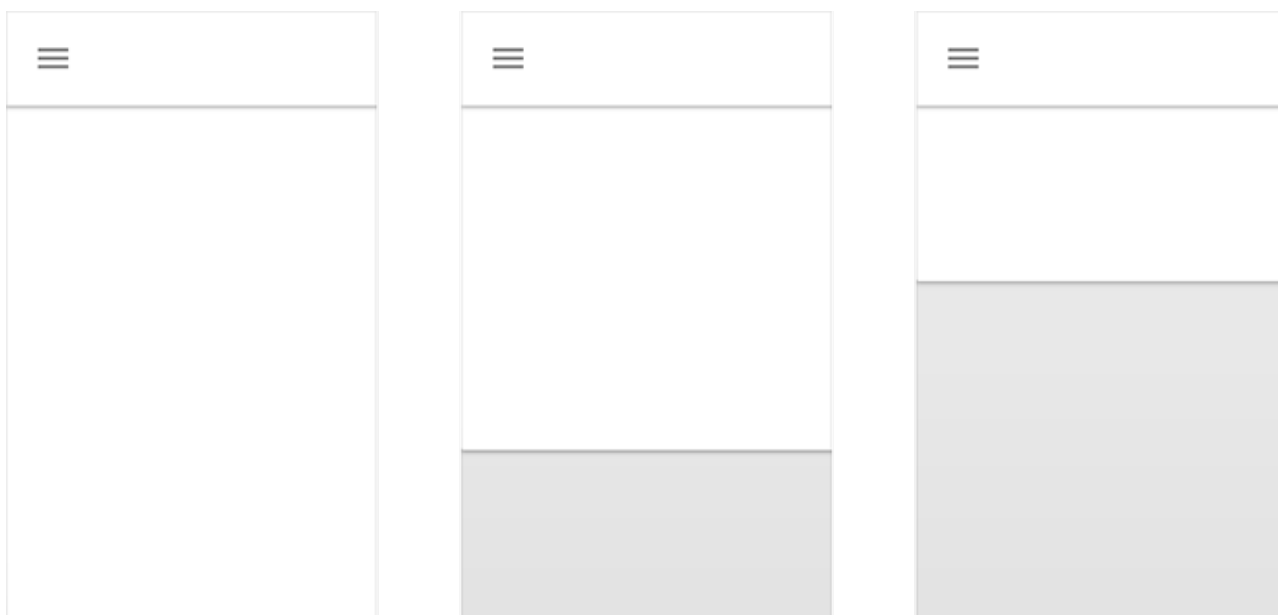


正确：弹出的卡片，被整体的长度所限制。



不正确：永远不允许弹出的卡片，将原卡片割裂

工具栏随着卡片的变化而频繁的变化，，工具栏中呈现出来的内容与卡片的内容有关。当卡片滚动到工具栏的下方时，卡片将会在某个点停止，防止卡片滑到另一侧。（可滑动的卡片到了顶点是有限制的：译者注）



工具栏也可以开始呈现出与第二个卡片接缝连接，但是拉起形成下一个动作。这种变异形式被称为瀑布流。

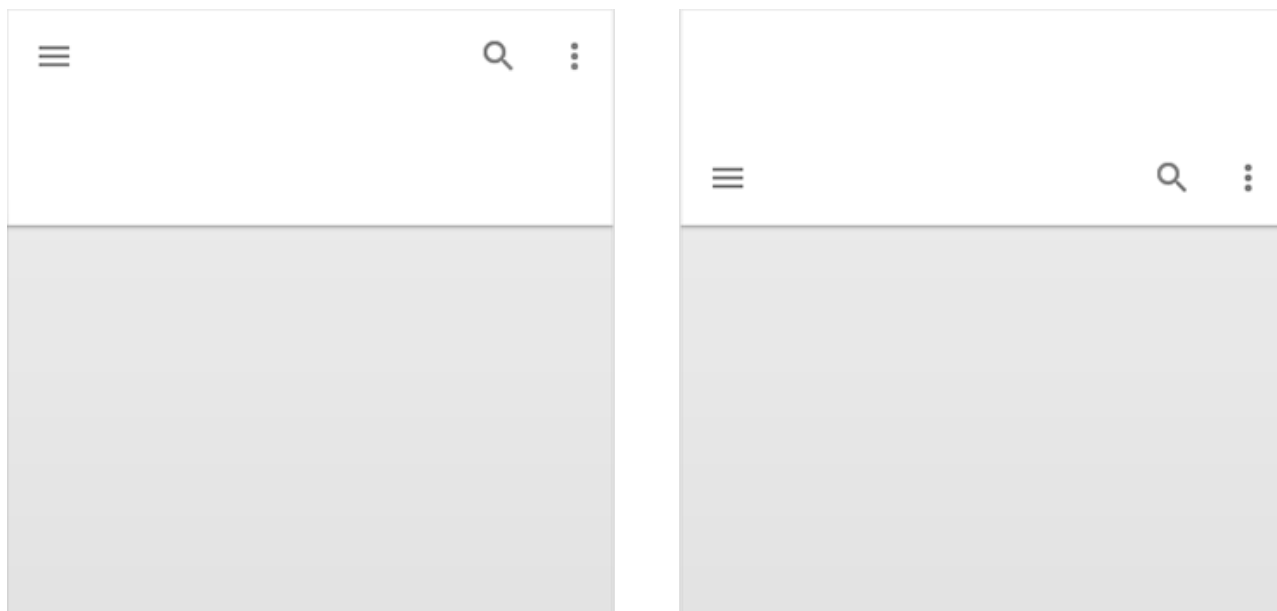


另外，工具栏可以和缝隙一起，随着两个卡片的滑动而滑出屏幕。

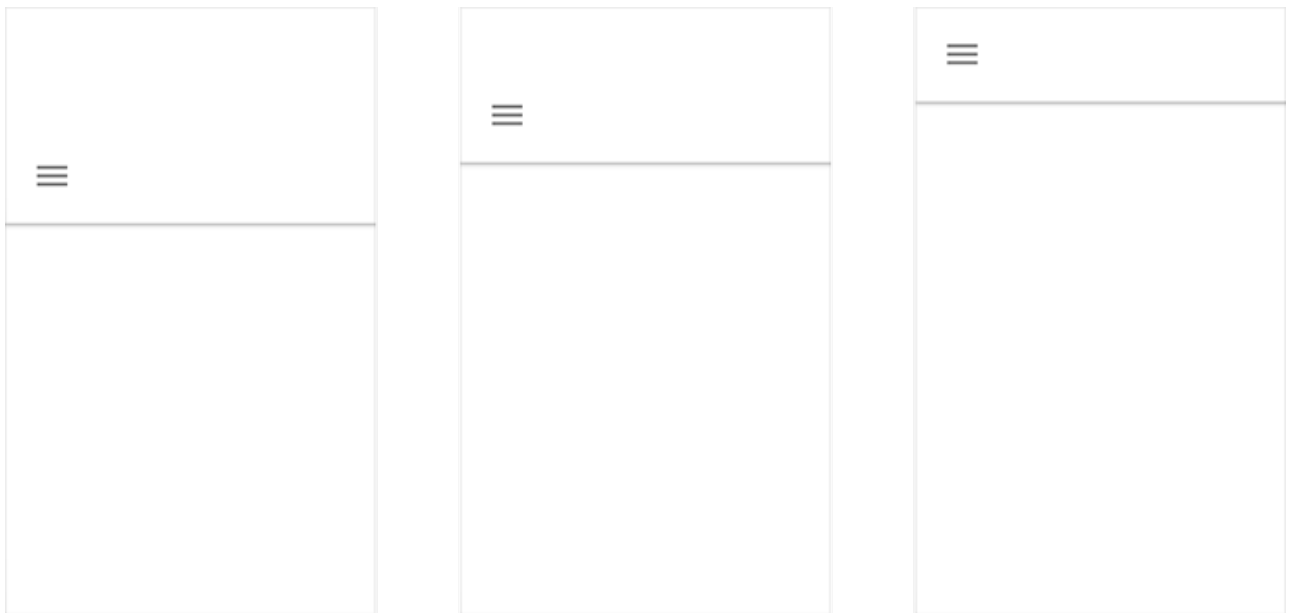
最后，第二个卡片在滑动的过程中，可以覆盖工具栏。



工具栏有标准的高度，但是也可以更高。当工具栏变高，它上面的操作可以放在顶部，也可以放在底部。



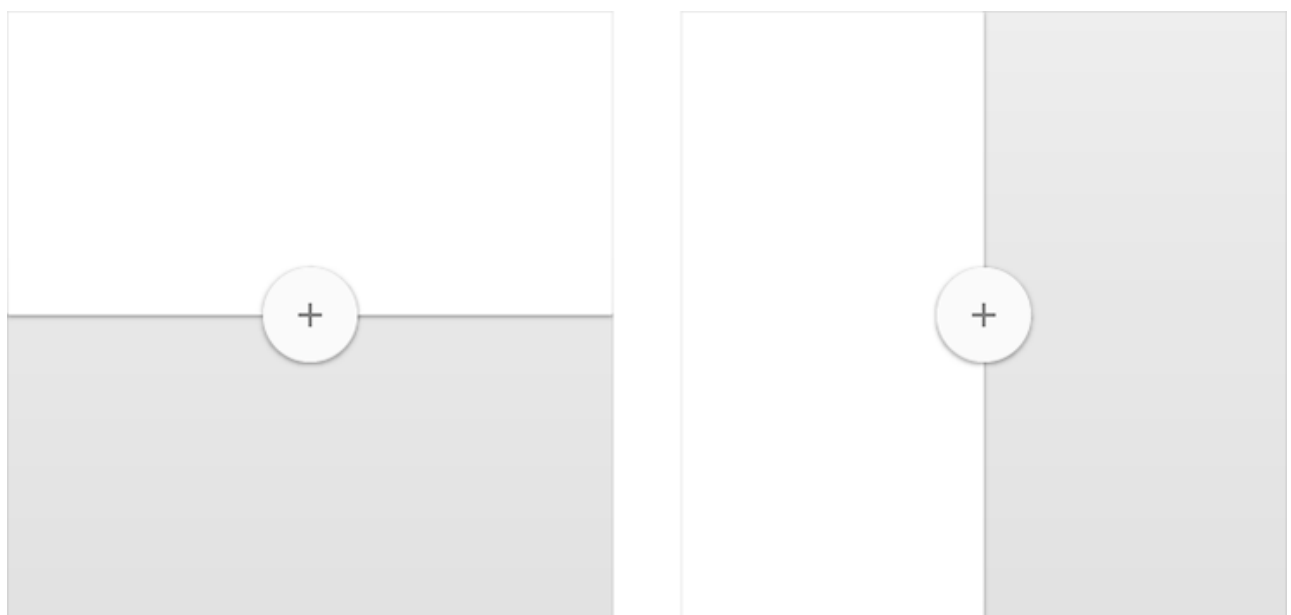
工具栏一旦被固定，可以灵活地改变它的大小。在变化的时候，它快速（有阈值与回程）的在最大值和最小值（标准值）之间变化。



悬浮按钮

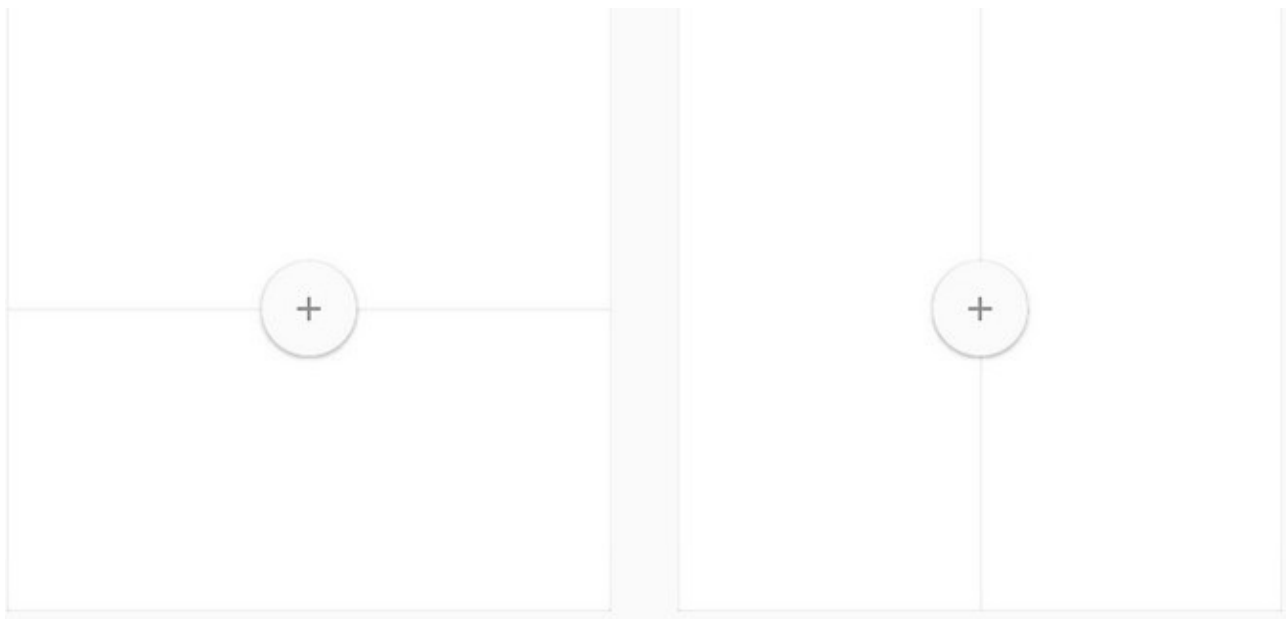
悬浮按钮指的是和工具栏分开的圆形卡片。

悬浮按钮代表着一些额外操作。如果与内容相关的话，悬停按钮可以跨越层级（有优先权限：译者注）。



悬浮操作如果与两个卡片的内容都相关的话，可以优先于接缝。

永远不要仅为了给操作一个固定的点，而引入一个接缝。



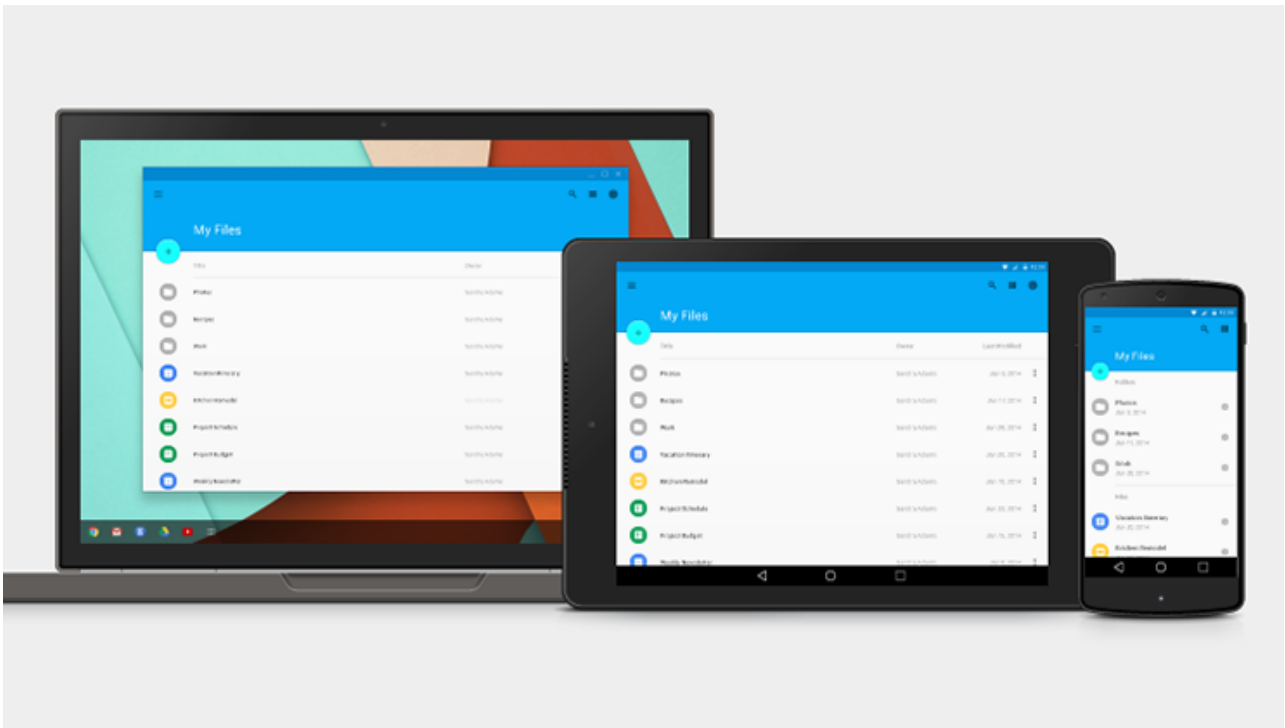
响应式设计原则

当我们建立跨设备的布局设计时，我们需要考虑固定、液态布局等一些设计策略。

下面是些应遵循的简单原则：

- 1 遵守人的常识
- 2 更大的屏幕≠更大的认知能力
- 3 线的长度应该被容易理解
- 4 考虑边角距离
- 5 遵循使用常识：允许空格，不要将工具栏固定。

在多层次的系统中使用这些策略，就像屏幕层级和卡片层级。



维度感

像素级的深度感可以通过X-Y轴体现。然而，通过Z轴体现界面的深度感，效果更佳。打造具有深度感的界面，而不应该墨守成规，让界面中的元素位置固定，这显得很死板。

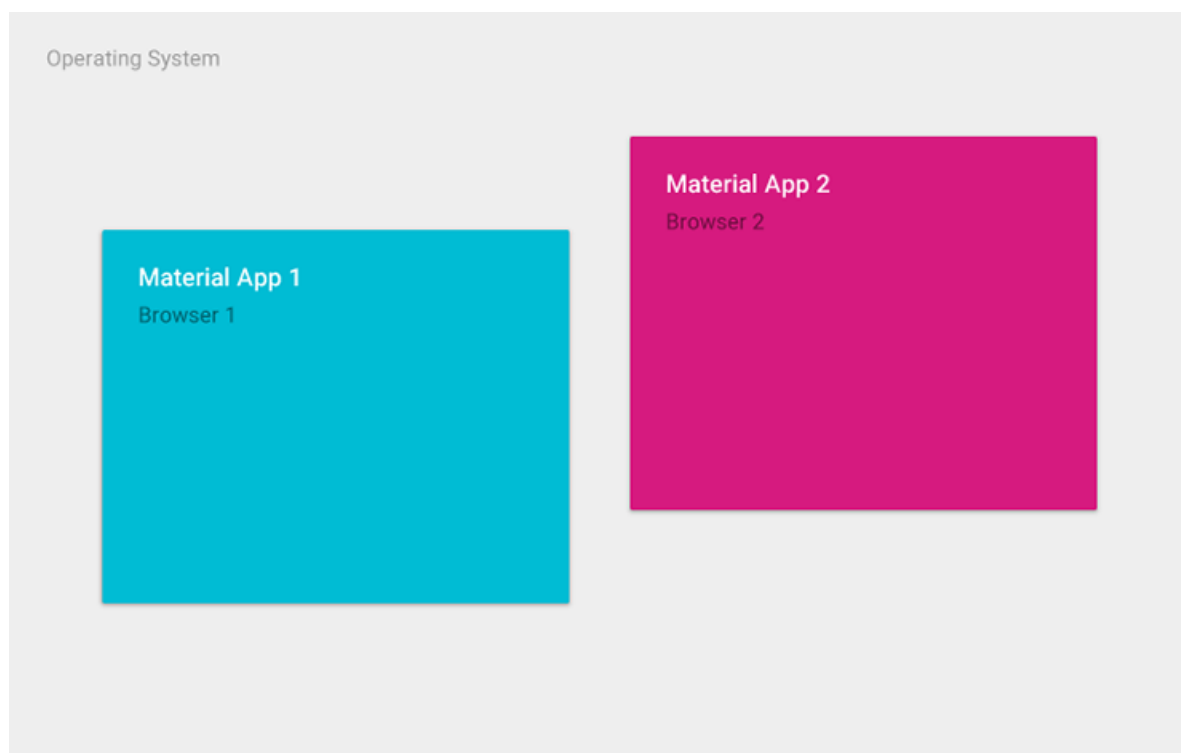
概念模型

从更高的层次上观察，每个应用其实是存在于不同空间和容器中的。

这就意味着，一个应用中的卡片不会插入到另一个应用中的卡片序列中，否则是Z轴空间的紊乱。

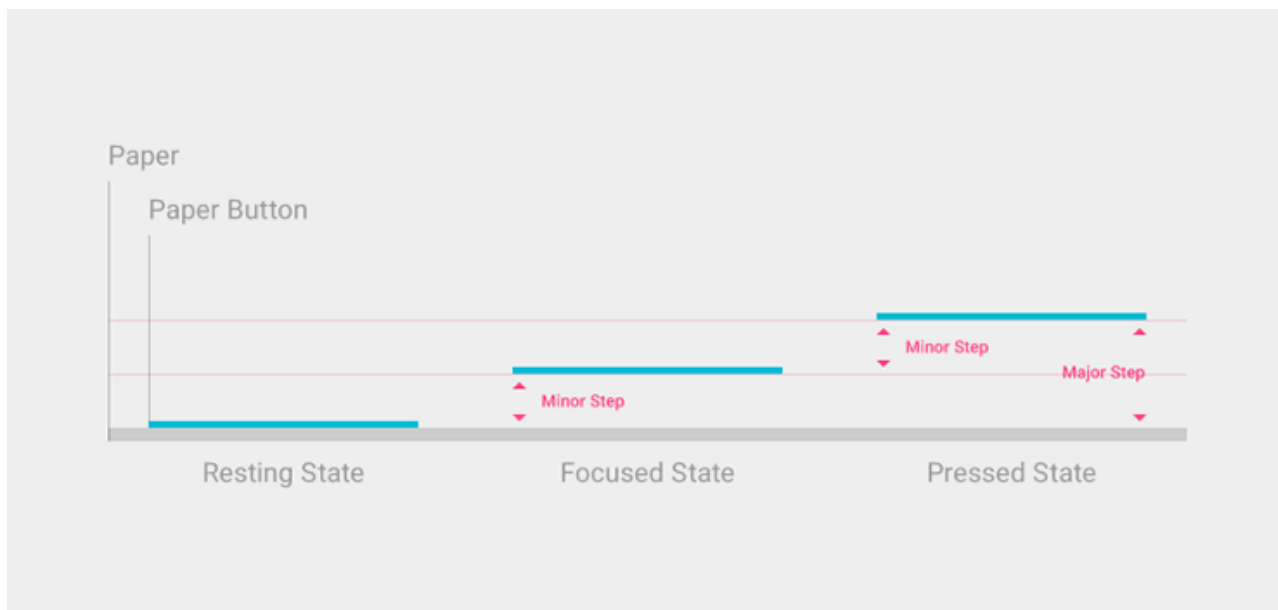
这也意味着，在一个应用中，操作和选项是隔离存在的。举例来说：在一个应用中删除一个列表不会引起另一个应用的变化。

而容器的存在，让多个应用可以同时存在于同一界面，例如浏览器中不同的标签页。



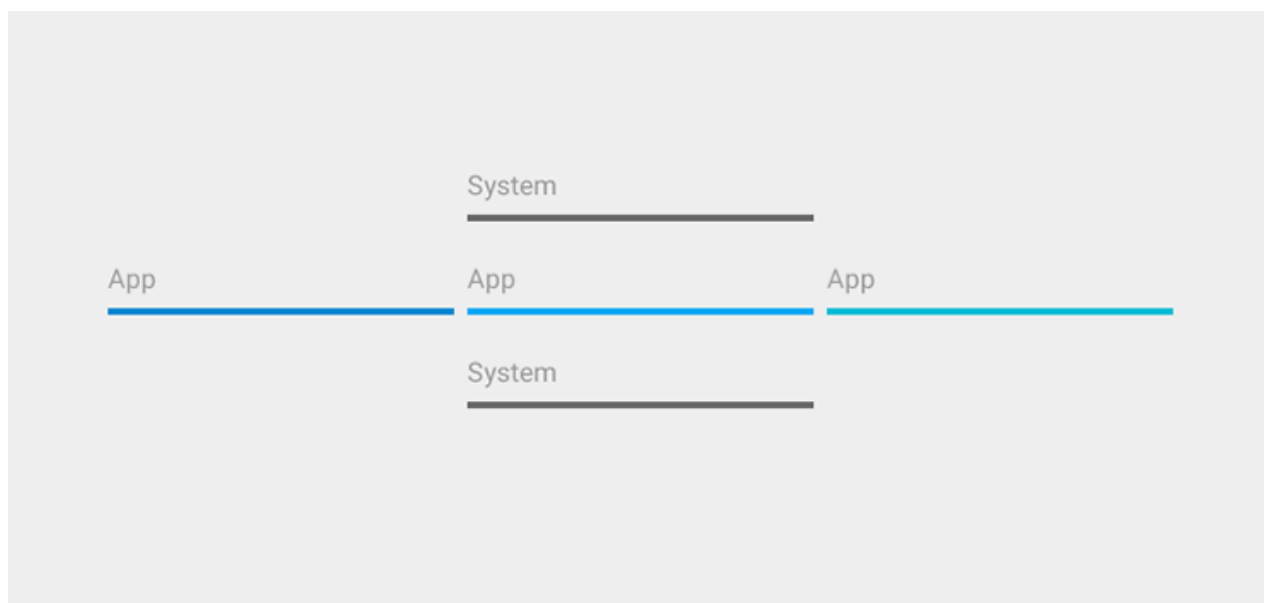
在特定应用内，大多数元素会按照Z轴，排列出主要层级和次要层级。例如：一个按钮的悬停状态是次要层级，而其按压状态是主要层级。

在应用内其他元素在Z轴的排列有着固定的优先级，这意味着它们总是出现在其他元素的上面或下面，无论这些项目沿z轴的相对位置。例如，浮动动作按钮总是出现内容和工具栏的上面，无论这个应用正在使用多少个卡片。



系统元素，比如状态栏和系统对话框，存在于一个单独的系统空间内，高于或低于所有的应用容器

根据上文，系统元素可以不呈现在应用中（比如夜间模式），但是如果存在系统空间元素，它们有更高的优先级。例如，系统对话框，永远出现在当前应用的最上层。

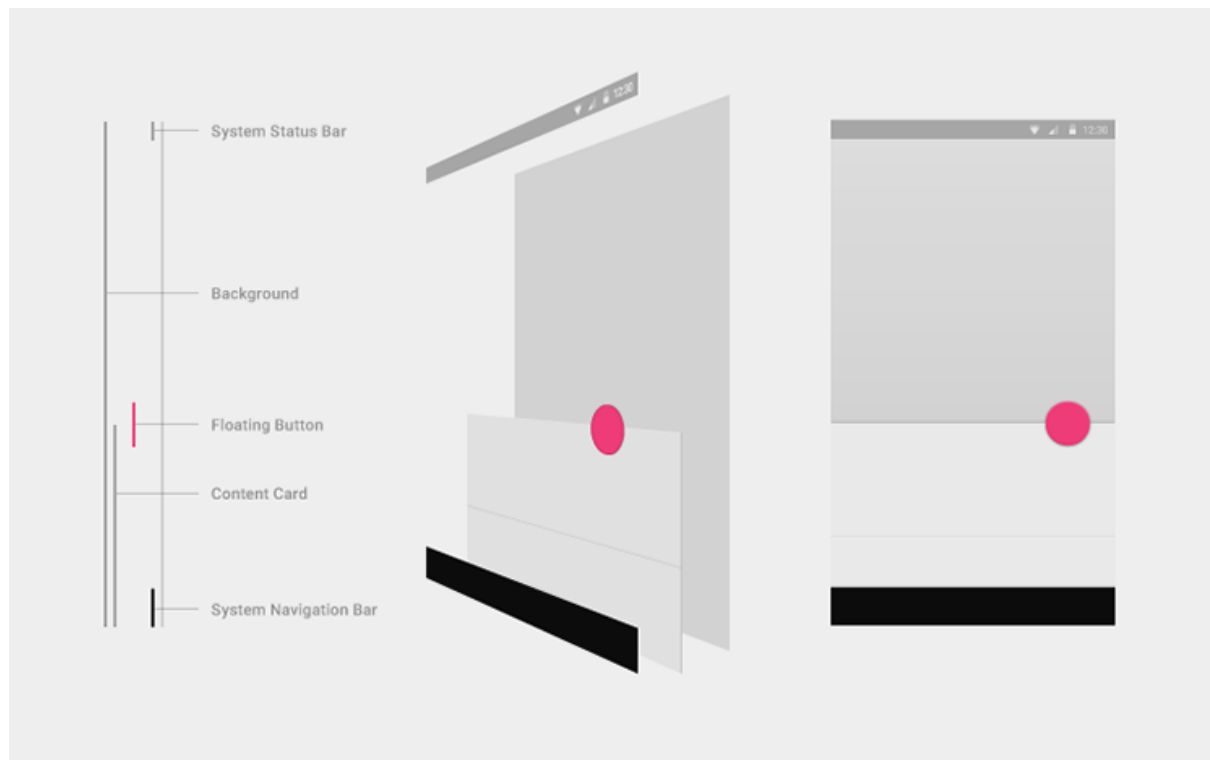


布局考量

深度感不是为了装饰而存在，而是为了功能

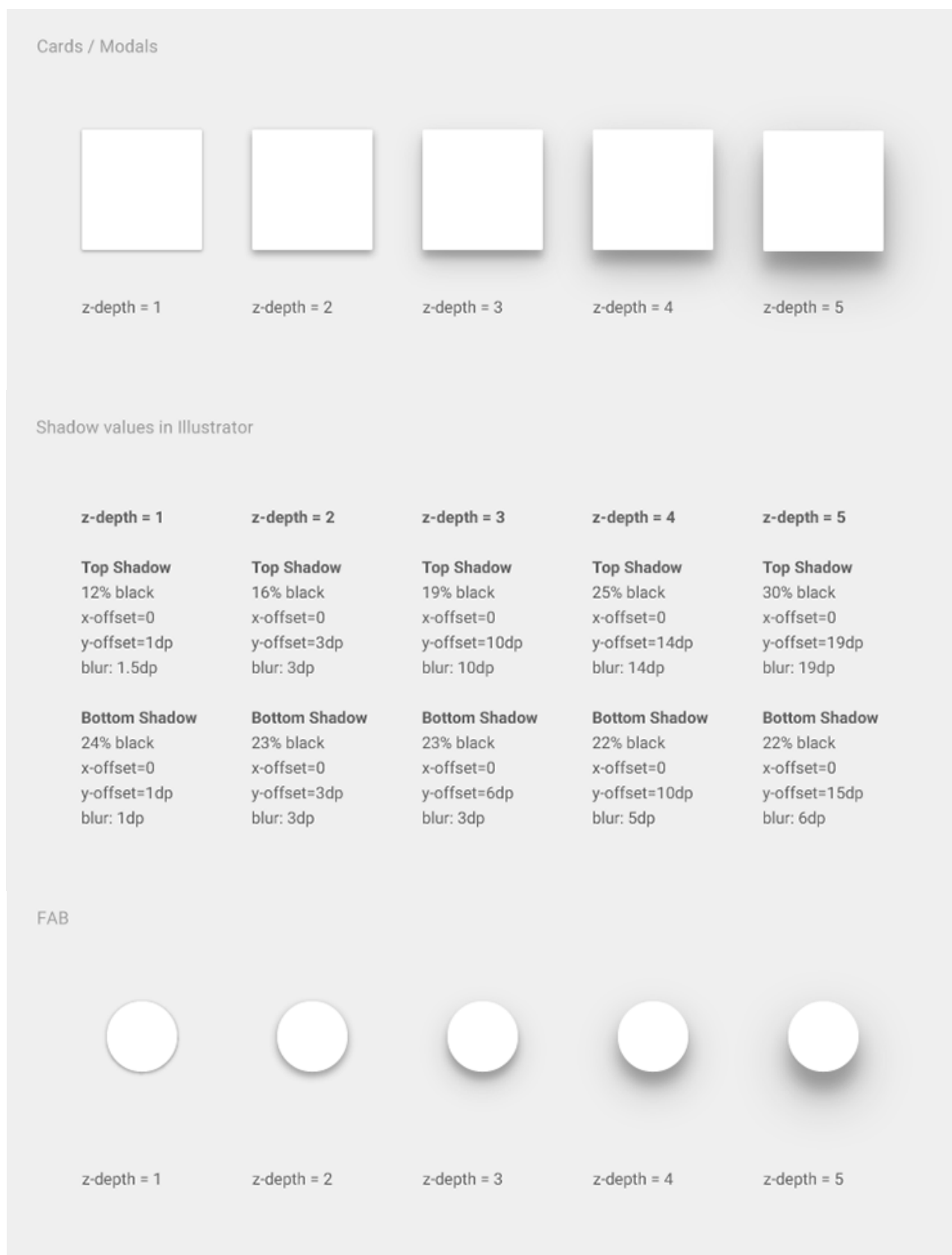
在布局时，需要审慎考虑元素在Z轴的优先级，而不是它的绝对位置。

深入思考你的应用，了解它的层次和各个元素重要性，帮助用户将精力集中在完成任务中。



阴影

阴影包括两层：顶部阴影是为了打造深度感，底部阴影是为了提升清晰度。



Shadow values in Illustrator

z-depth = 1

Top Shadow

12% black
x-offset=0
y-offset=1dp
blur: 1.5dp

Bottom Shadow

24% black
x-offset=0
y-offset=1dp
blur: 1dp

z-depth = 2

Top Shadow

16% black
x-offset=0
y-offset=3dp
blur: 3dp

Bottom Shadow

23% black
x-offset=0
y-offset=3dp
blur: 3dp

z-depth = 3

Top Shadow

19% black
x-offset=0
y-offset=10dp
blur: 10dp

Bottom Shadow

23% black
x-offset=0
y-offset=6dp
blur: 3dp

z-depth = 4

Top Shadow

25% black
x-offset=0
y-offset=14dp
blur: 14dp

Bottom Shadow

22% black
x-offset=0
y-offset=10dp
blur: 5dp

z-depth = 5

Top Shadow

30% black
x-offset=0
y-offset=19dp
blur: 19dp

Bottom Shadow

22% black
x-offset=0
y-offset=15dp
blur: 6dp

Left Nav



z-depth = 2



z-depth = 2



z-depth = 3



z-depth = 4



z-depth = 5

Shadow values in Illustrator

z-depth = 1

Top Shadow

12% black
x-offset=0
y-offset=1dp
blur: 1.5dp

Bottom Shadow

24% black
x-offset=0
y-offset=1dp
blur: 1dp

z-depth = 2

Top Shadow

16% black
x-offset=0
y-offset=3dp
blur: 3dp

Bottom Shadow

23% black
x-offset=0
y-offset=3dp
blur: 3dp

z-depth = 3

Top Shadow

19% black
x-offset=0
y-offset=10dp
blur: 10dp

Bottom Shadow

23% black
x-offset=0
y-offset=6dp
blur: 3dp

z-depth = 4

Top Shadow

25% black
x-offset=0
y-offset=14dp
blur: 14dp

Bottom Shadow

22% black
x-offset=0
y-offset=10dp
blur: 5dp

z-depth = 5

Top Shadow

30% black
x-offset=0
y-offset=19dp
blur: 19dp

Bottom Shadow

22% black
x-offset=0
y-offset=15dp
blur: 6dp

Right Nav



z-depth = 1



z-depth = 2



z-depth = 3



z-depth = 4



z-depth = 5

Shadow values in Illustrator

z-depth = 1

Top Shadow

12% black
x-offset=0
y-offset=1dp
blur: 1.5dp

Bottom Shadow

24% black
x-offset=0
y-offset=1dp
blur: 1dp

z-depth = 2

Top Shadow

16% black
x-offset=0
y-offset=3dp
blur: 3dp

Bottom Shadow

23% black
x-offset=0
y-offset=3dp
blur: 3dp

z-depth = 3

Top Shadow

19% black
x-offset=0
y-offset=10dp
blur: 10dp

Bottom Shadow

23% black
x-offset=0
y-offset=6dp
blur: 3dp

z-depth = 4

Top Shadow

25% black
x-offset=0
y-offset=14dp
blur: 14dp

Bottom Shadow

22% black
x-offset=0
y-offset=10dp
blur: 5dp

z-depth = 5

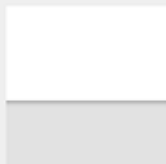
Top Shadow

30% black
x-offset=0
y-offset=19dp
blur: 19dp

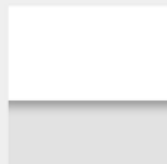
Bottom Shadow

22% black
x-offset=0
y-offset=15dp
blur: 6dp

App Bar



z-depth = 1



z-depth = 2



z-depth = 3



z-depth = 4



z-depth = 5

Shadow values in Illustrator

z-depth = 1

Top Shadow

12% black
x-offset=0
y-offset=1dp
blur: 1.5dp

Bottom Shadow

24% black
x-offset=0
y-offset=1dp
blur: 1dp

z-depth = 2

Top Shadow

16% black
x-offset=0
y-offset=3dp
blur: 3dp

Bottom Shadow

23% black
x-offset=0
y-offset=3dp
blur: 3dp

z-depth = 3

Top Shadow

19% black
x-offset=0
y-offset=10dp
blur: 10dp

Bottom Shadow

23% black
x-offset=0
y-offset=6dp
blur: 3dp

z-depth = 4

Top Shadow

25% black
x-offset=0
y-offset=14dp
blur: 14dp

Bottom Shadow

22% black
x-offset=0
y-offset=10dp
blur: 5dp

z-depth = 5

Top Shadow

30% black
x-offset=0
y-offset=19dp
blur: 19dp

Bottom Shadow

22% black
x-offset=0
y-offset=15dp
blur: 6dp

Bottom Toolbar / Sheet



z-depth = 2



z-depth = 2



z-depth = 3



z-depth = 4



z-depth = 5

Shadow values in Illustrator

z-depth = 1

Top Shadow

12% black
x-offset=0
y-offset=1dp
blur: 1.5dp

Bottom Shadow

24% black
x-offset=0
y-offset=1dp
blur: 1dp

z-depth = 2

Top Shadow

16% black
x-offset=0
y-offset=3dp
blur: 3dp

Bottom Shadow

23% black
x-offset=0
y-offset=3dp
blur: 3dp

z-depth = 3

Top Shadow

19% black
x-offset=0
y-offset=10dp
blur: 10dp

Bottom Shadow

23% black
x-offset=0
y-offset=6dp
blur: 3dp

z-depth = 4

Top Shadow

25% black
x-offset=0
y-offset=14dp
blur: 14dp

Bottom Shadow

22% black
x-offset=0
y-offset=10dp
blur: 5dp

z-depth = 5

Top Shadow

30% black
x-offset=0
y-offset=19dp
blur: 19dp

Bottom Shadow

22% black
x-offset=0
y-offset=15dp
blur: 6dp

Sticky Left Nav with Paper Canvas



z-depth = 1



z-depth = 2



z-depth = 3

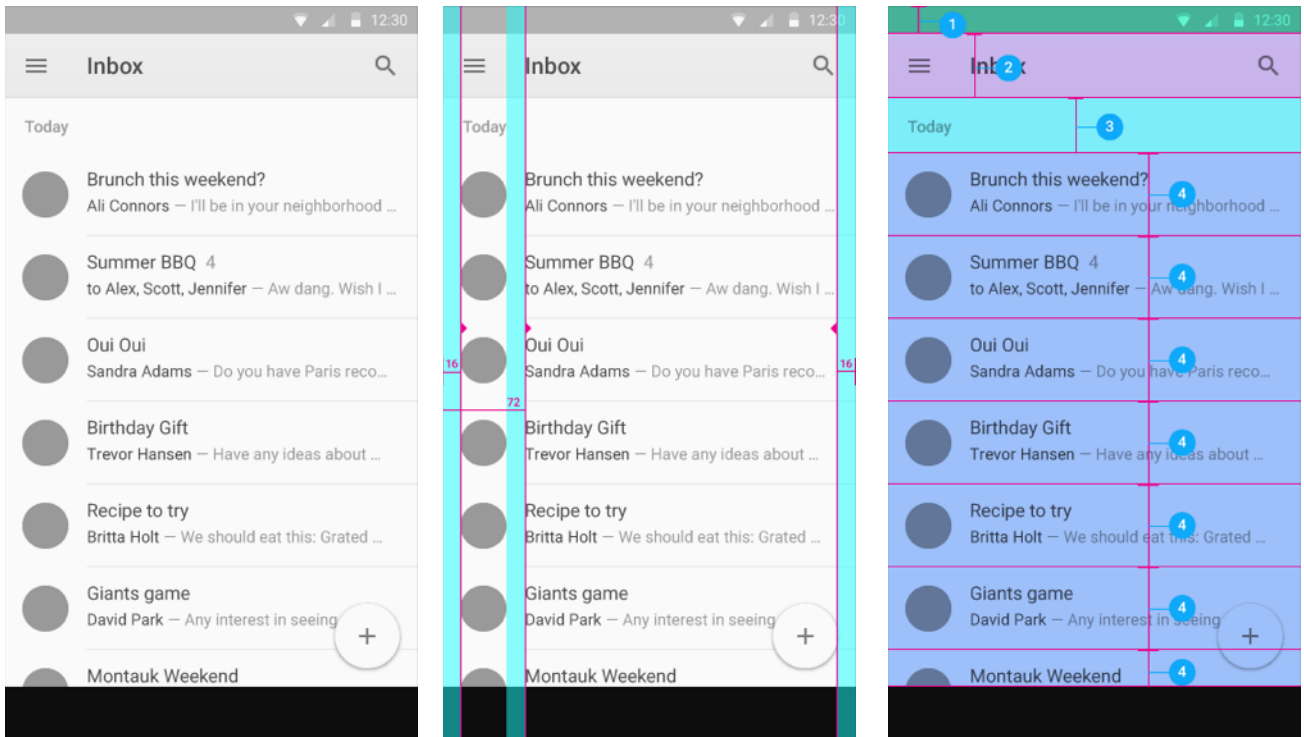


z-depth = 4

排版和间距

移动设备

移动设备布局模板包含了多种多样的屏幕和信息，这些信息描述了边框与间距如何应用于屏幕边界和元素。下面是一个有关屏幕的实例，这个实例可以通过如下的链接获得。

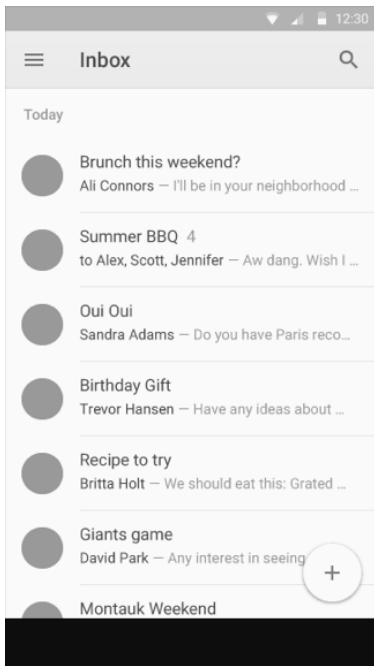


显示了图标、头像和一个2行文本的列表如何左对齐，以及一个56dp的浮动动作按钮和文本如何右对齐

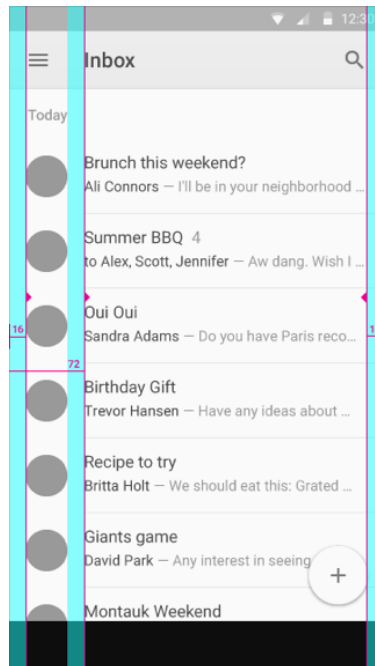
左右各有16dp的垂直边框。带有图标或者头像的内容有72dp的左边距

垂直间距

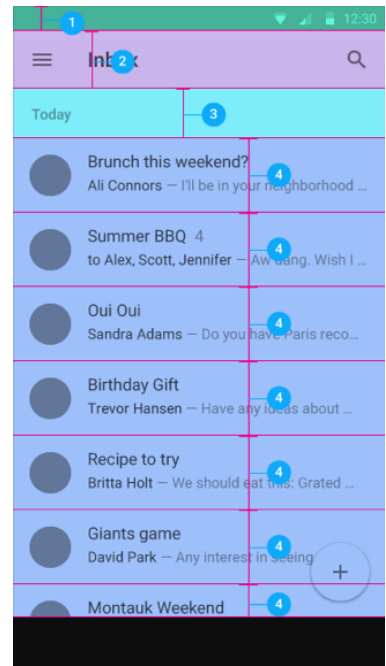
1. 24dp
2. 56dp
3. 48dp
4. 72dp



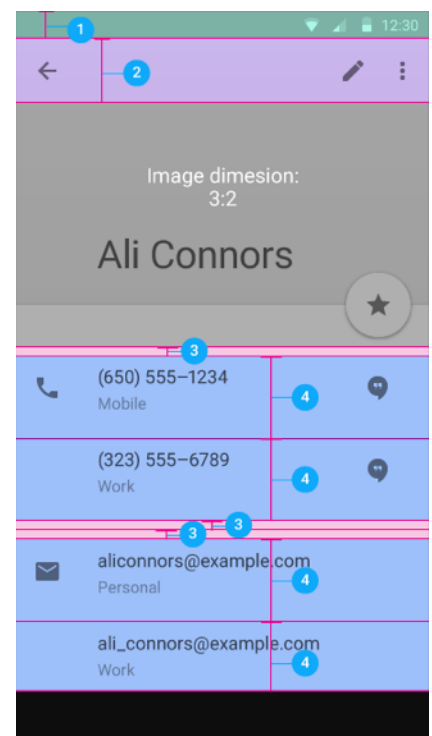
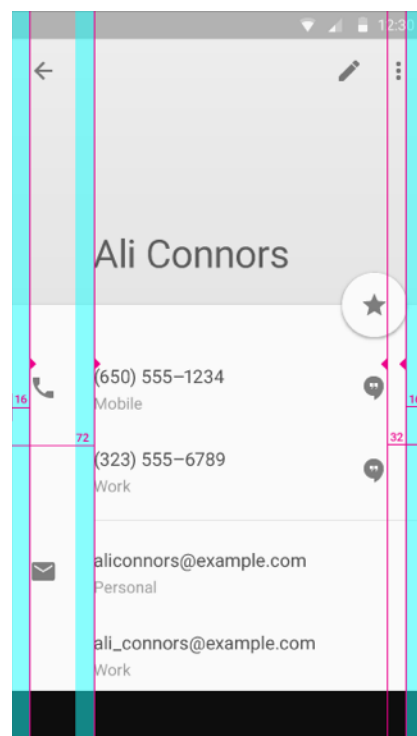
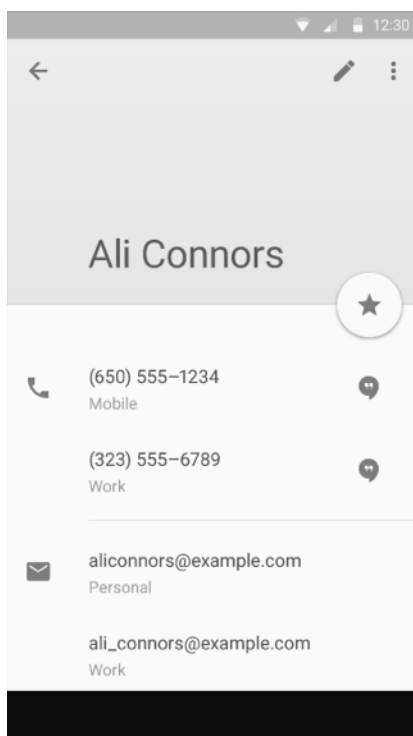
显示了图标、头像、2行
文本列表、子标题和
40dp的浮动动作按钮如
何左对齐。小图标右对齐



图标（大小图标）有
16dp的左右垂直边框。
带有图标或者头像的内容
有72dp的左边距



垂直间距
1. 24dp
2. 56dp
3. 72dp
4. 48dp
5. 8dp

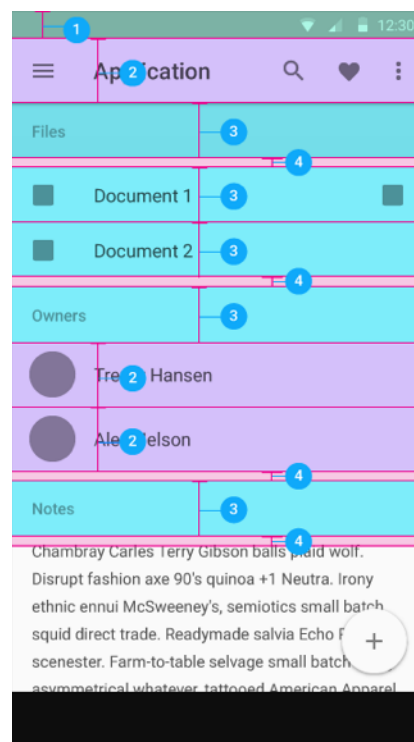
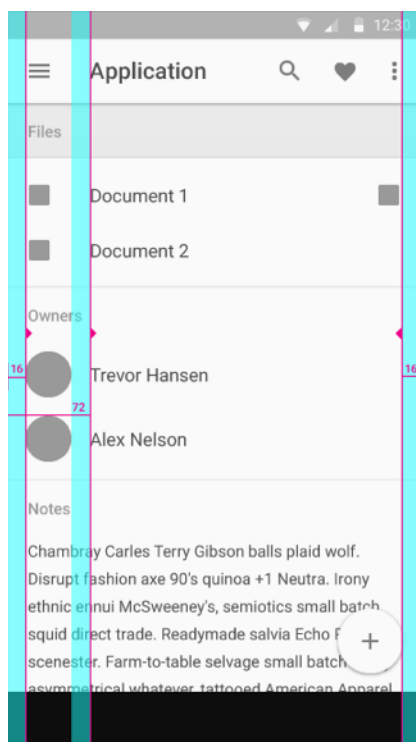
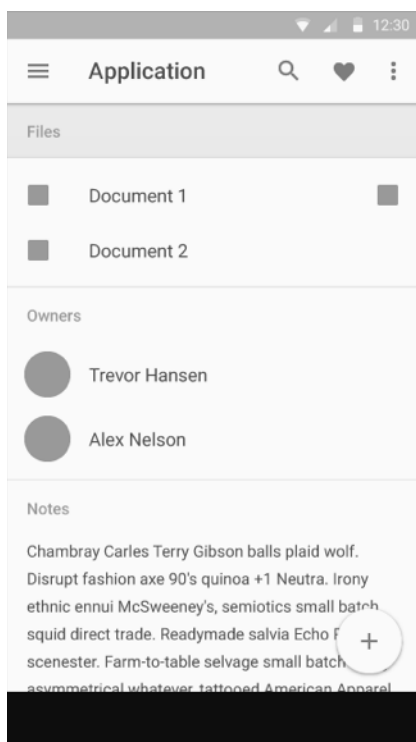


该屏幕显示图标如何左对齐，以及图标和一个56dp的浮动动作按钮如何右对齐

图标有16dp的左垂直边框。带有图标或头像的内容有72dp的左边框。另外有32dp的边框添加到了右边，与56dp的圆形浮动动作按钮的边框有重叠。这样做是允许在浮动动作按钮下的图标对齐的。

垂直间距

1. 24dp
2. 56dp
3. 8dp
4. 72dp



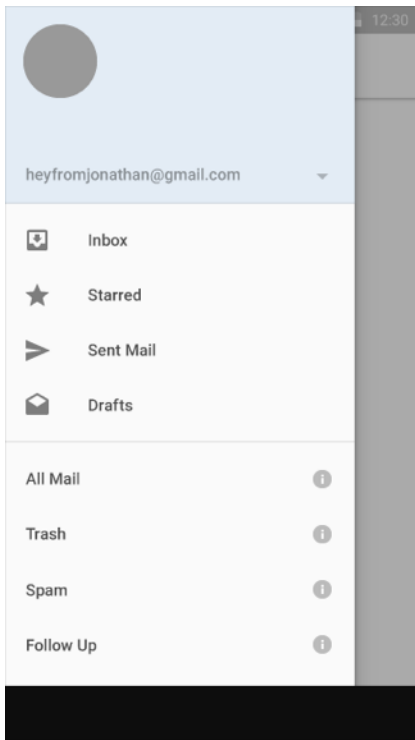
此屏幕显示了图标、头像和文本如何左对齐，浮动动作按钮、图标和文本如何右对齐

图标有16dp的左右垂直边框。带有图标或头像的内容有72dp的左边框。

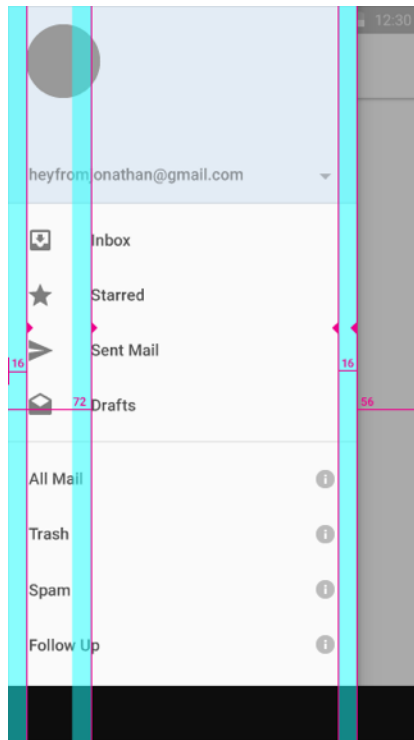
在移动设备上有16dp的水平边距。

垂直间距

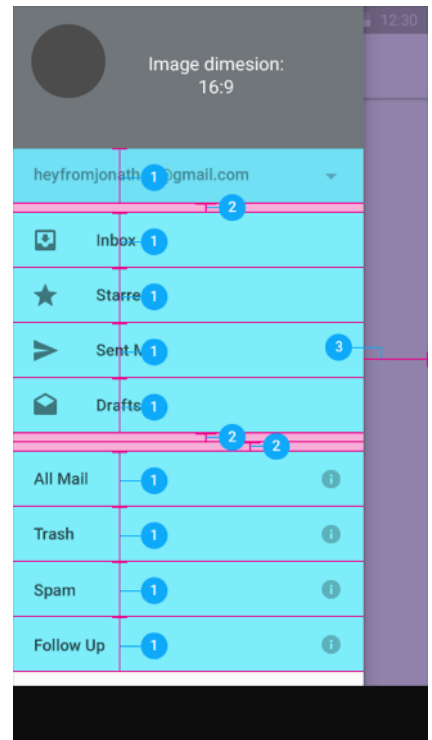
1. 24dp
2. 56dp
3. 48dp
4. 8dp



此屏幕显示了侧边导航菜单的宽度，以及图标、头像和文本如何左对齐，小图标如何右对齐



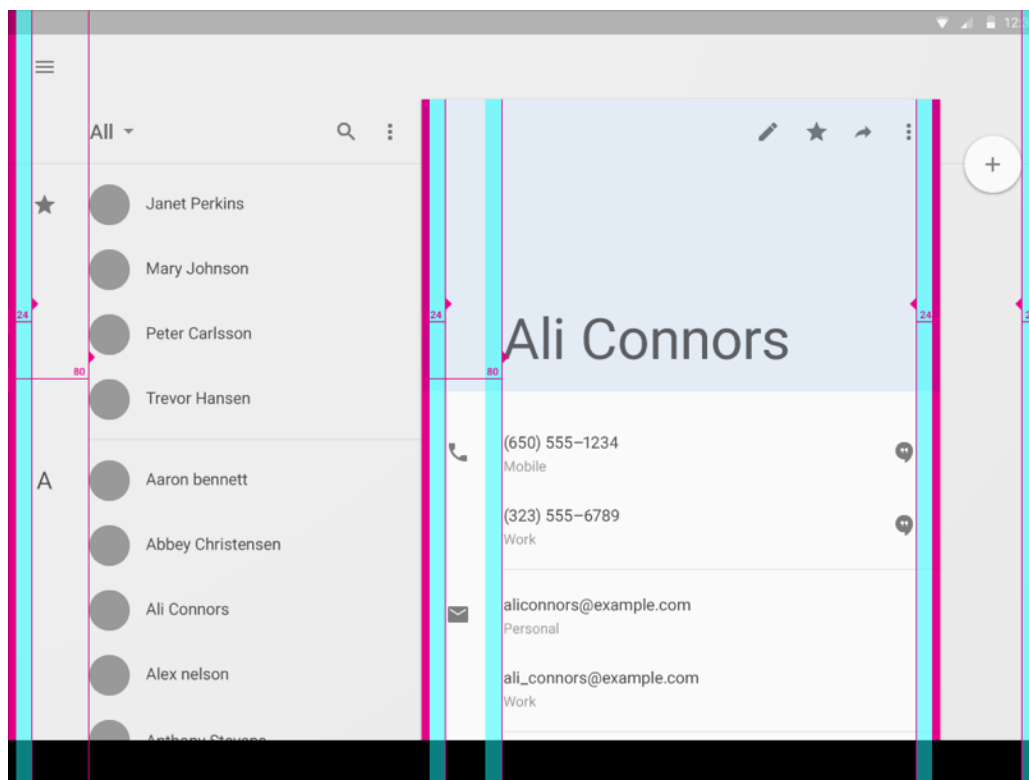
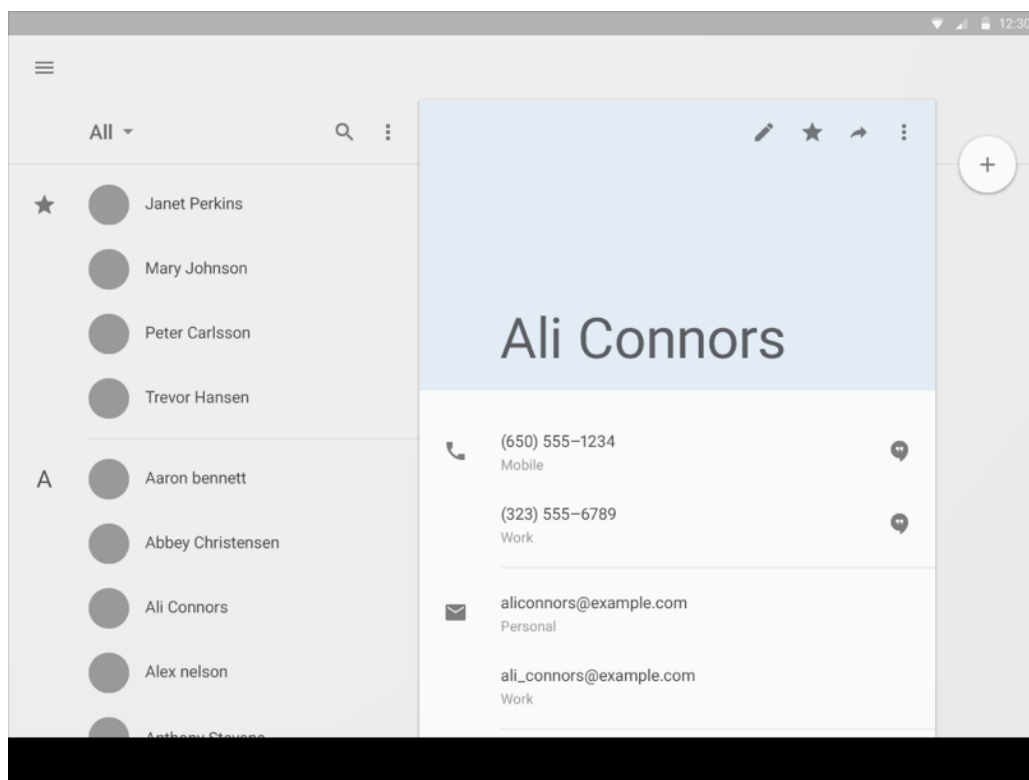
图标有16dp的左垂直边框和距侧边导航菜单16dp的右垂直边框。带有图标或者头像的内容距侧边导航菜单72dp的左垂直边框。侧边导航菜单的宽度等于屏幕的宽度减去动作条的高度，即在本例中距屏幕右侧56dp的宽
在移动设备上有16dp的水平外边距

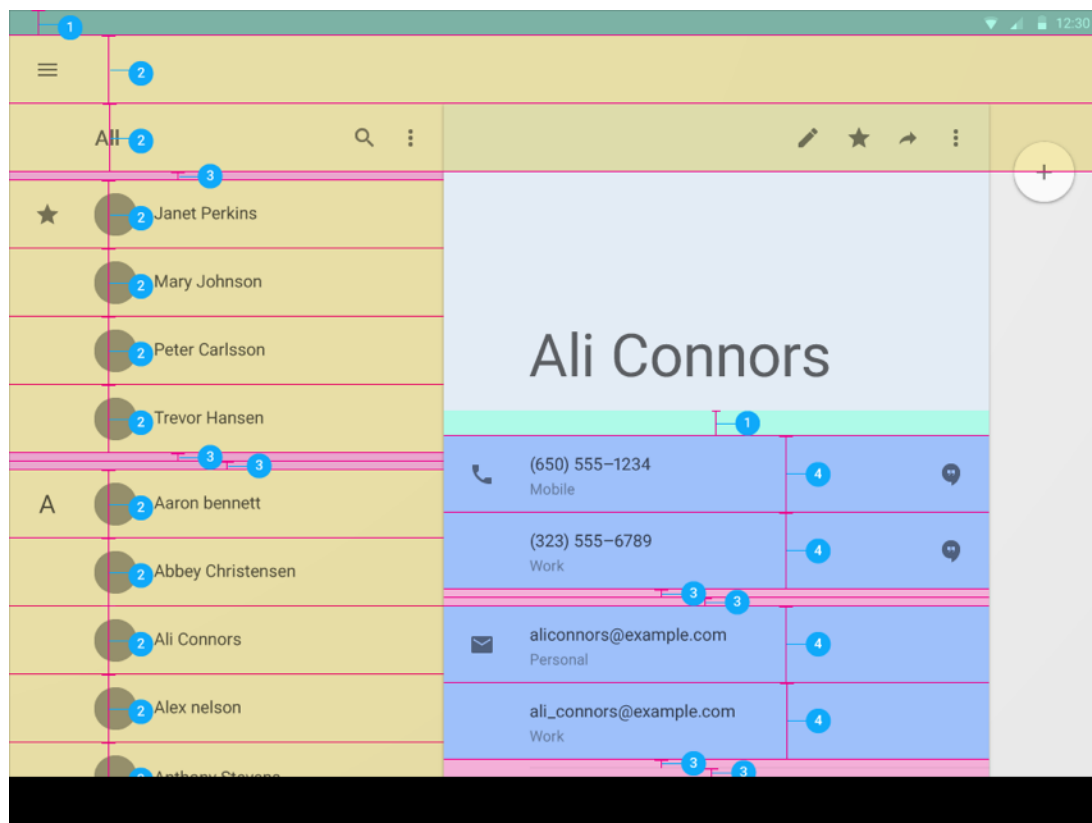


垂直间距
1. 48dp
2. 8dp
3. 56dp

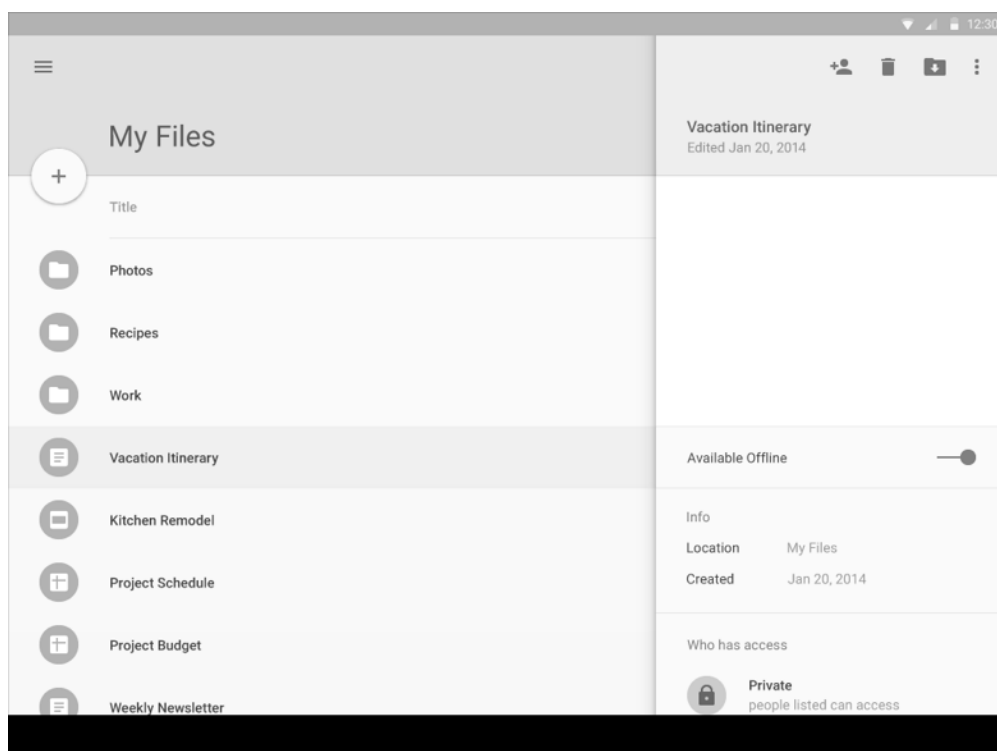
平板电脑

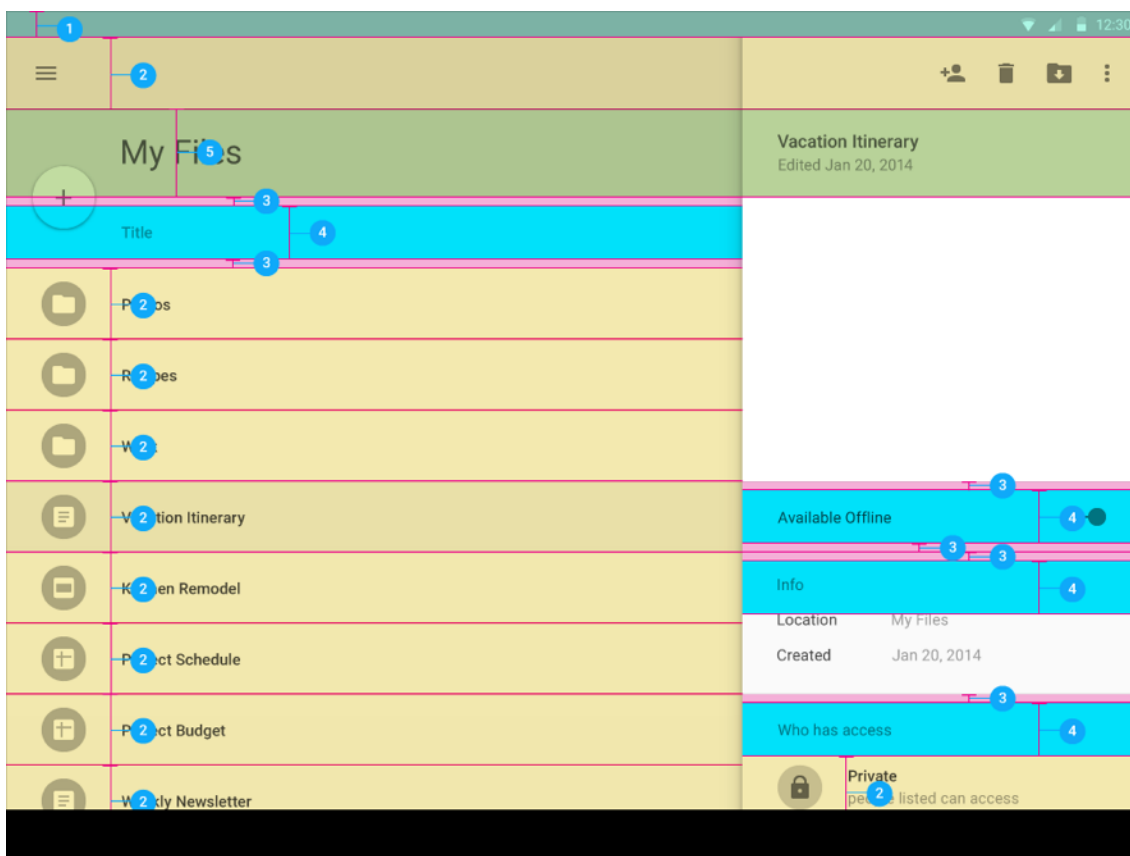
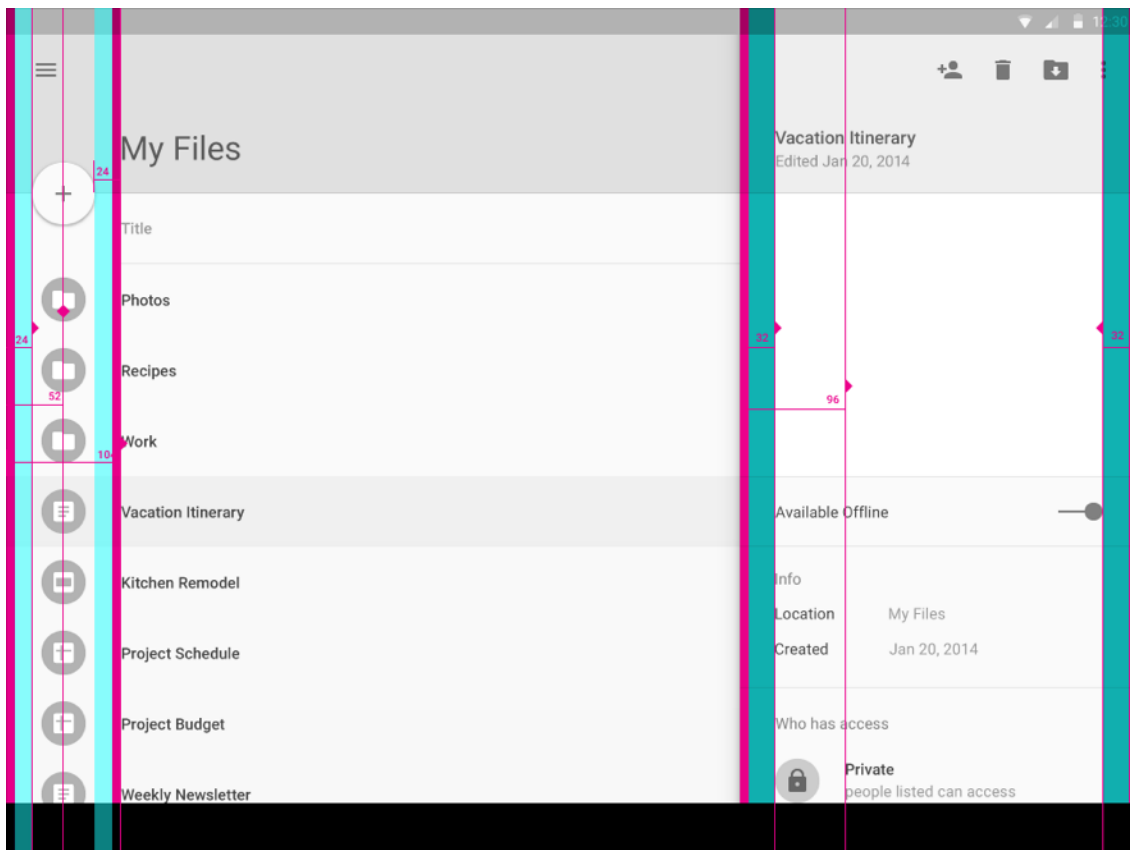
平板设备布局模板包含了14种不同的屏幕，显示了边框和边距如何应用于屏幕的边界和元素。下面的实例是14种模板的两个，可以通过下面的链接获取。





1. 24dp
2. 64dp
3. 8dp
4. 72dp

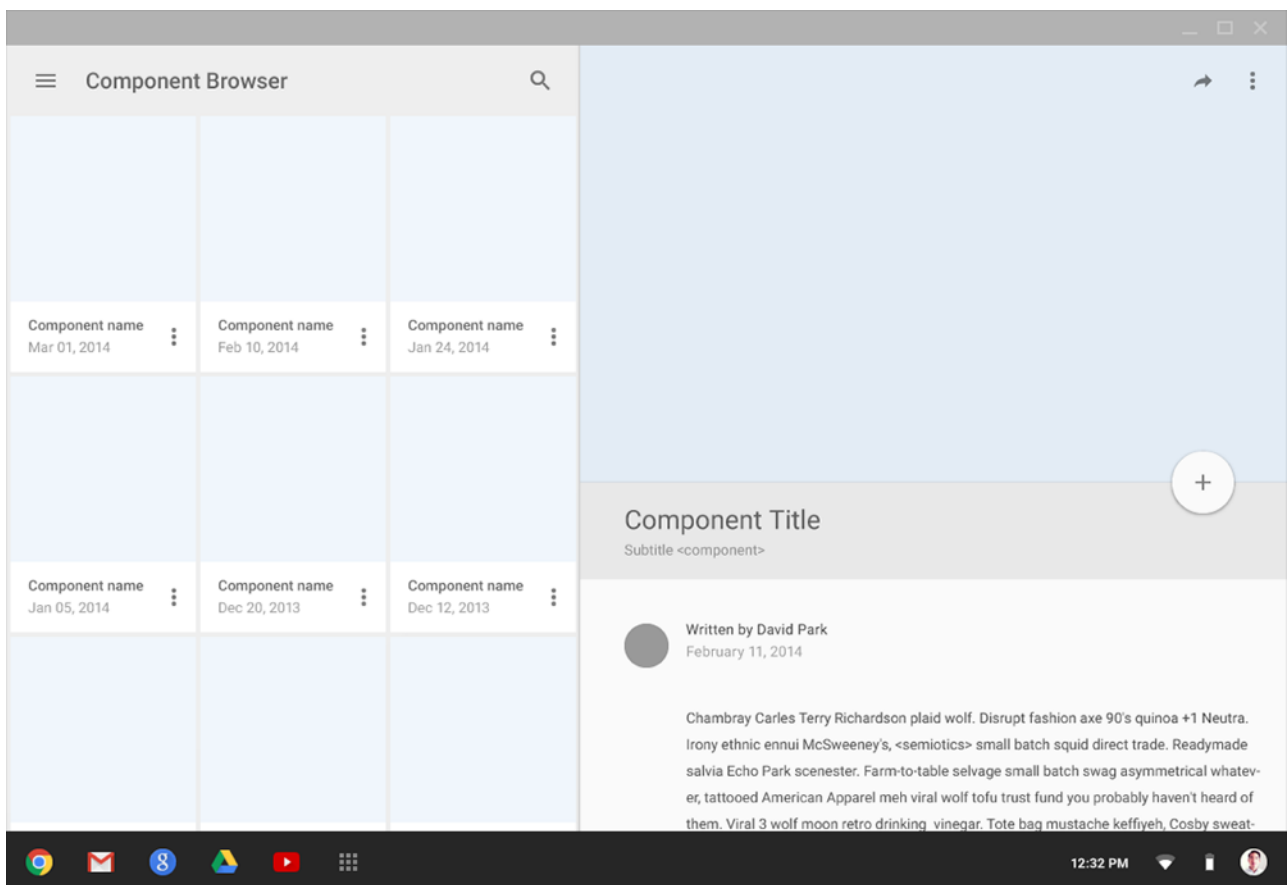


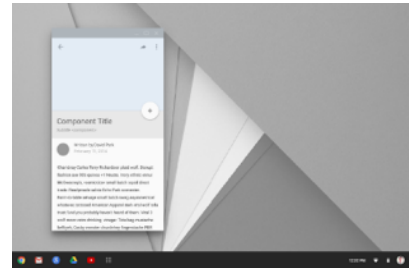
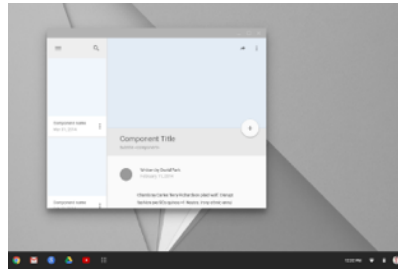
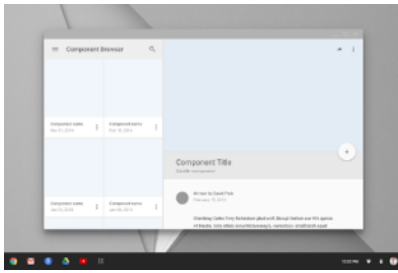


1. 24dp
2. 64dp
3. 8dp
4. 48dp
5. 80dp

桌面

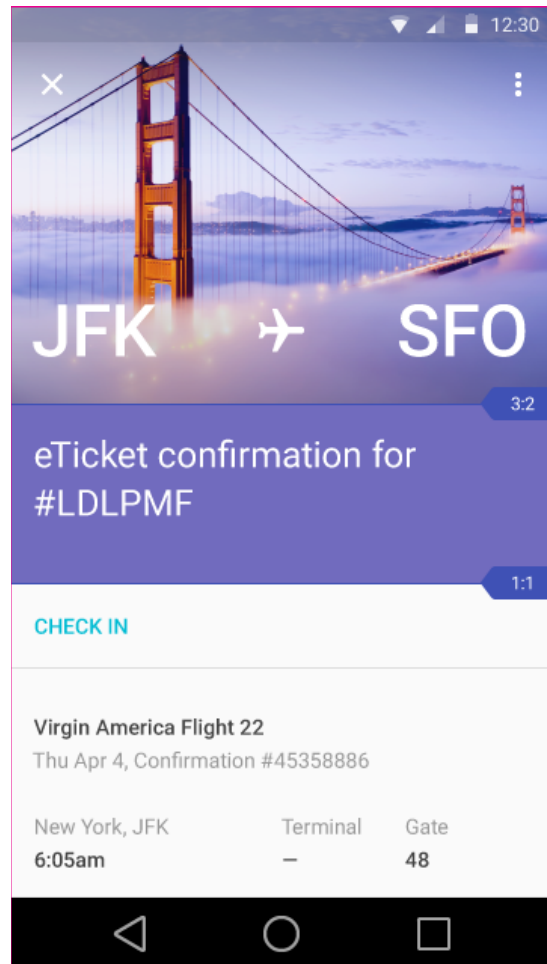
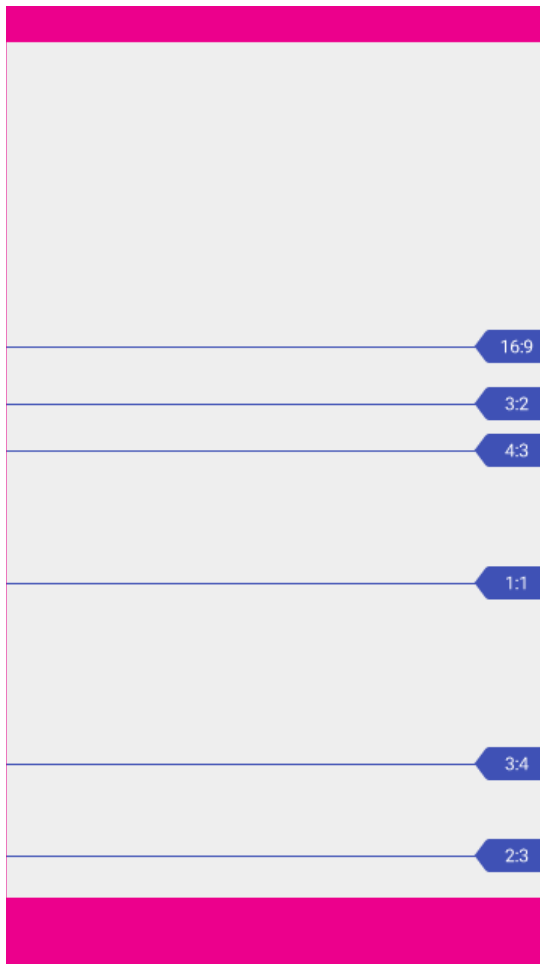
桌面应用程序布局模板包含了4个不同的屏幕，每一个屏幕右板块了4个不同的窗口尺寸，显示了边框和边距如何应用于窗口和元素。下面是一个屏幕的实例，可以通过下面所示的链接下载。基于不同的窗口尺寸，边框和边距块会继承自平板设备和移动设备的网格规则

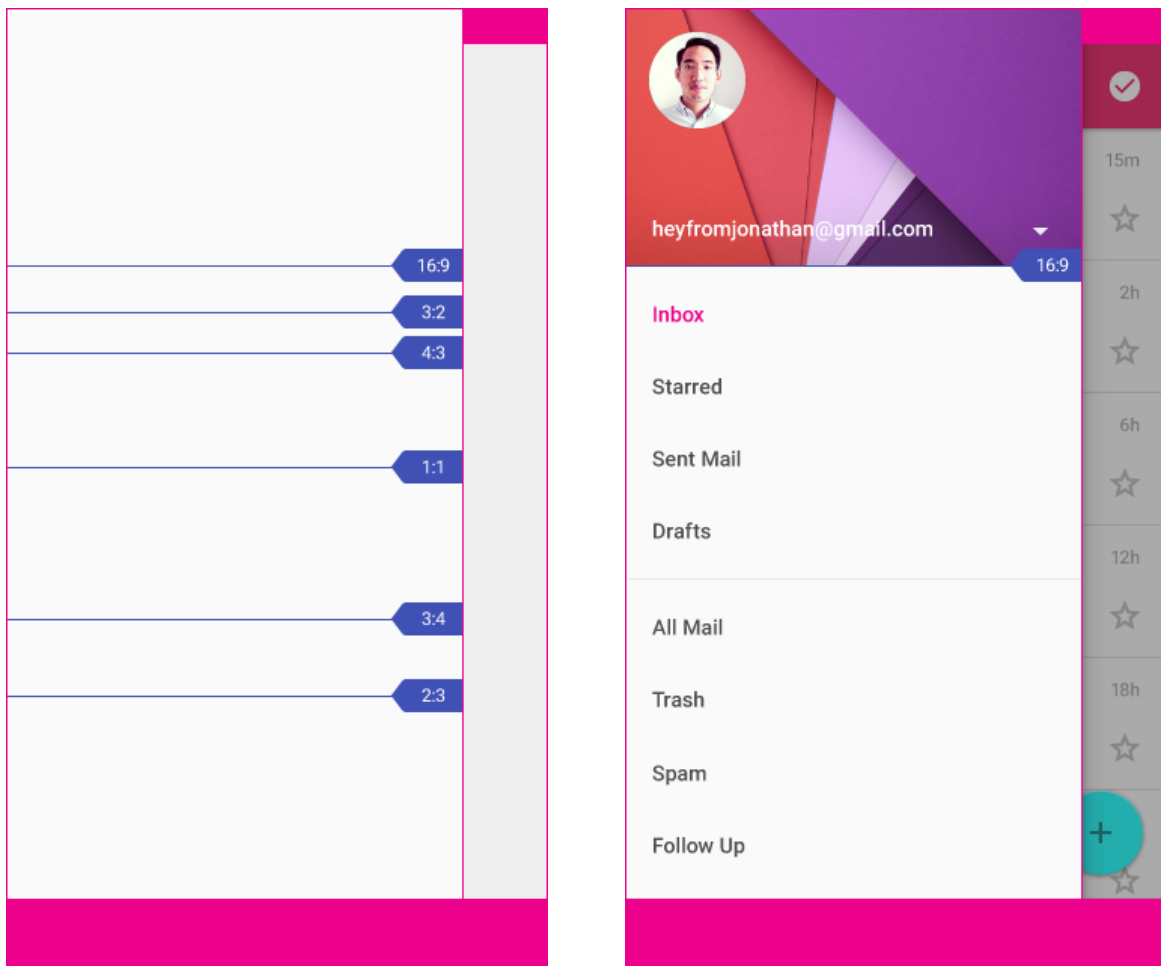




排版比例

适用于手机屏幕的宽度和UI元素，兼顾平板电脑和台式机

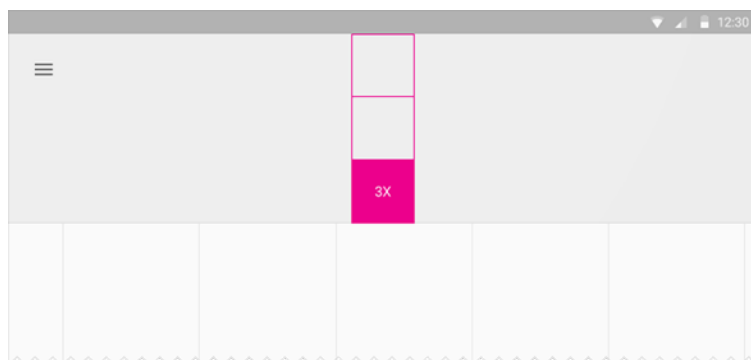


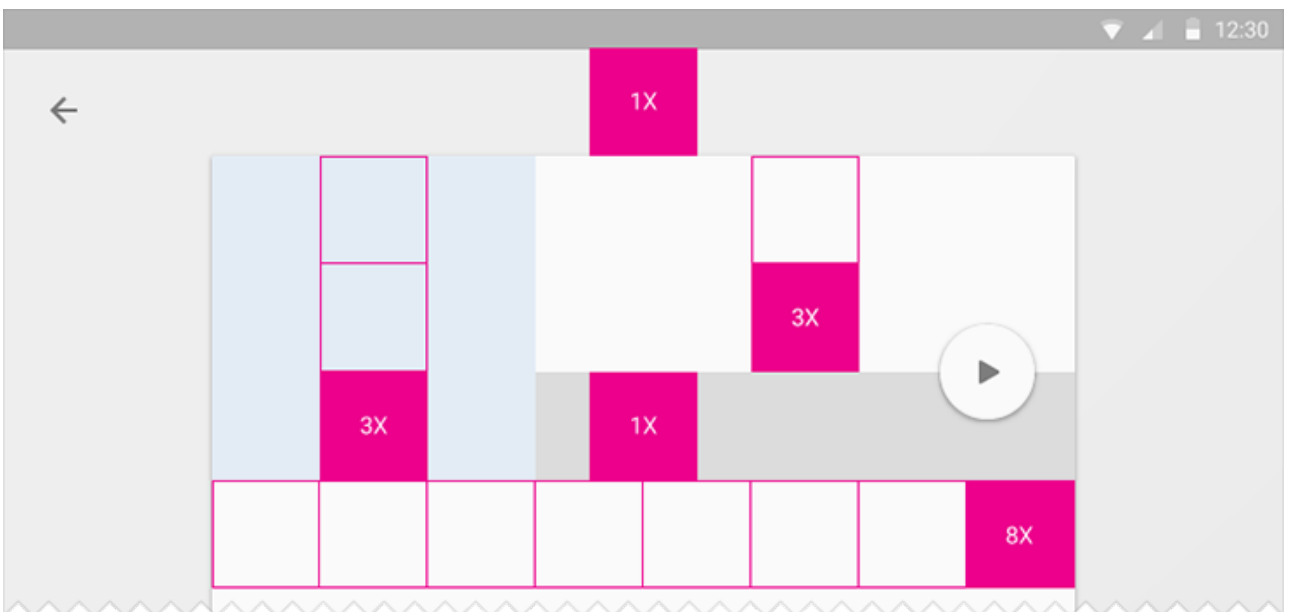
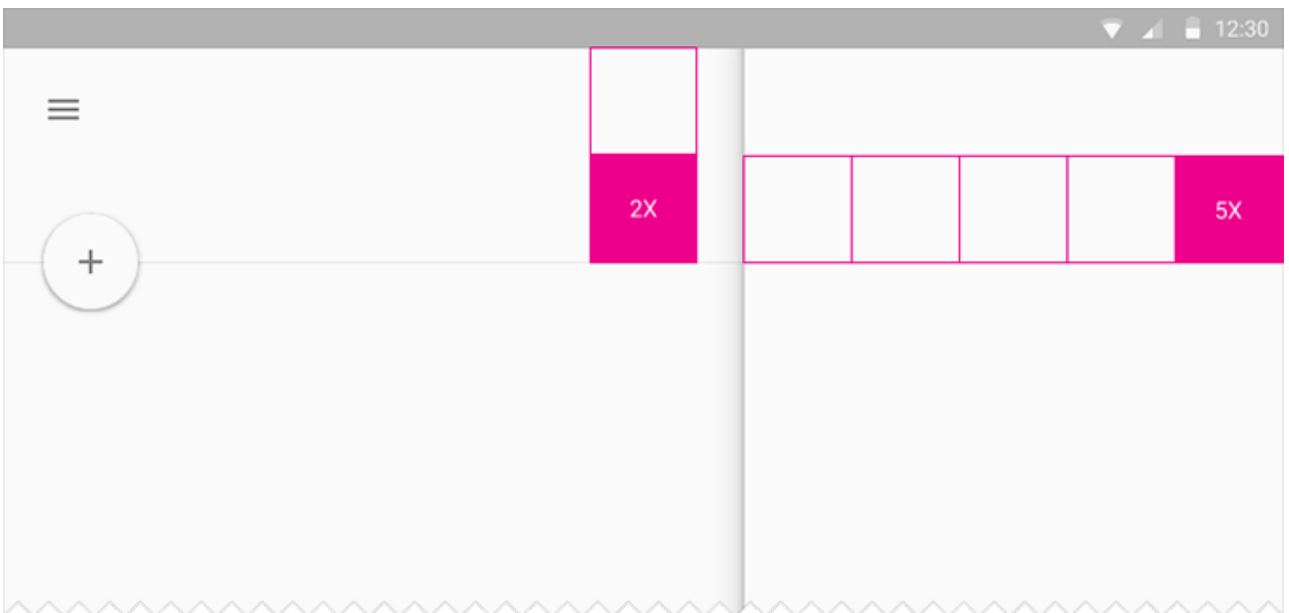
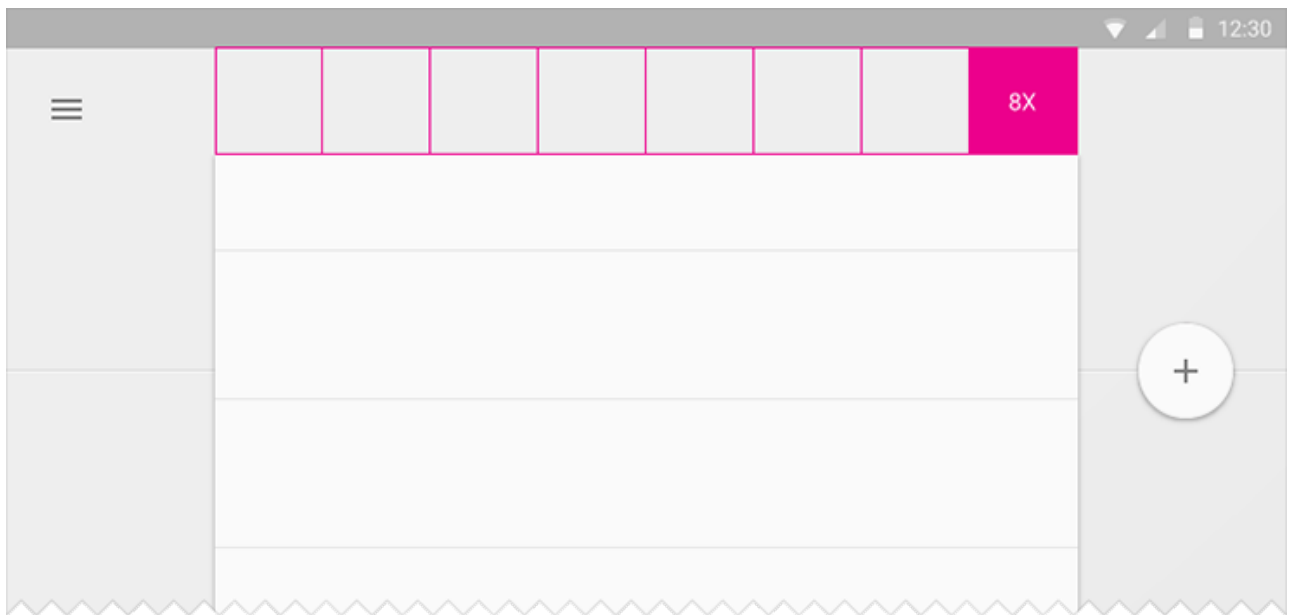


增量边框

增量边框定义了一个增量，比如动作条的高度，然后使用几倍于这个增量的数字来决定应用中其它元素的尺寸和位置。

大多数应用于桌面应用程序，有些也适用于平板设备，很少应用在移动设备中。增量的大小会根据窗口的尺寸来改变。

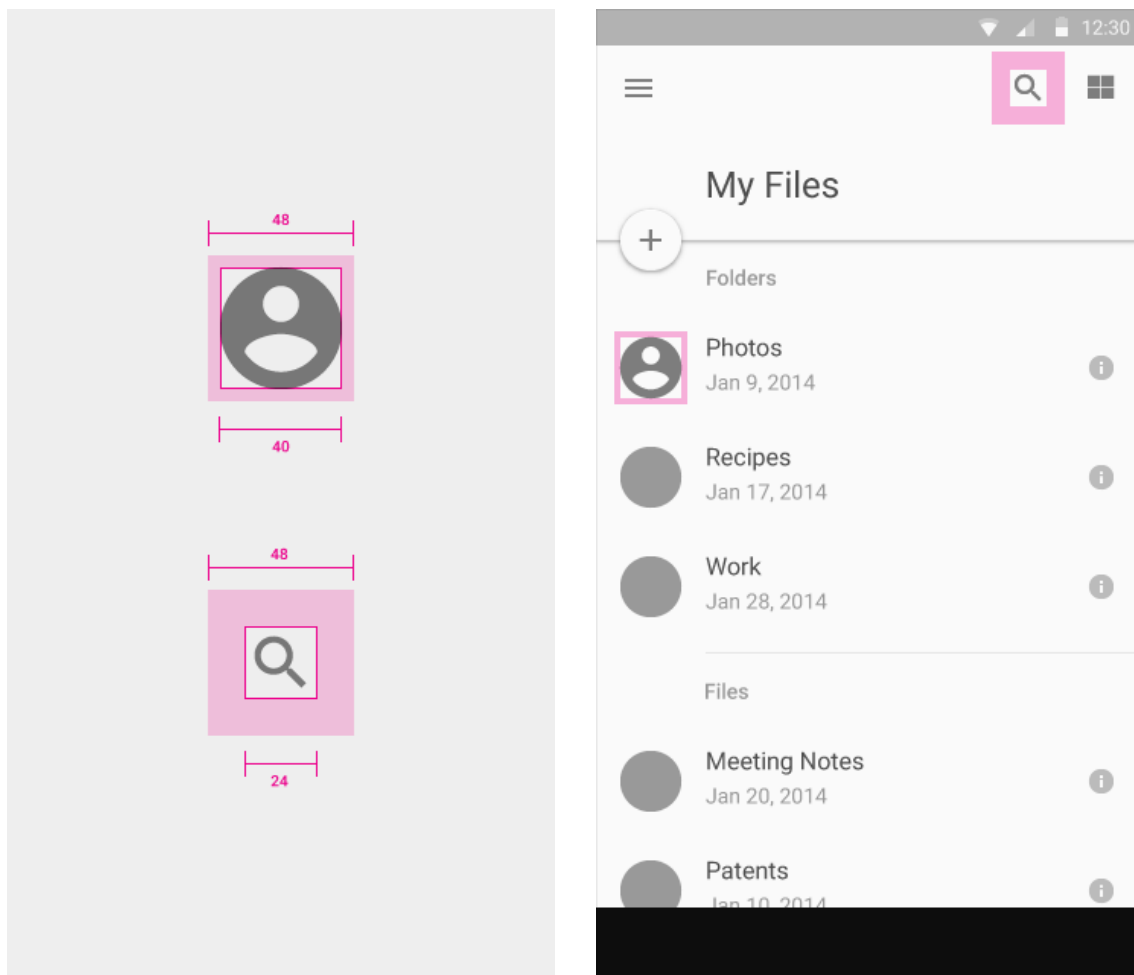




触摸目标大小

最小触摸目标大小为48dp。图标间距 (24dp) ,头像 (40dp) .

在布局时，请记住这一点，触摸区域不应该重叠。



4.3 结构

UI布局和指南

本节主要介绍我们的应用结构，包括移动端应用和桌面端应用的架构，涵盖了下面的几点指导。

应用有很多种，分别满足不同的需求：

- 只有一个界面、围绕一个主要需求而构建的应用（如计算器，相机和一些游戏）
- 主要功能是在不同的界面中切换，但是没有深度导航功能（如电话拨号记录，最近联系人，和收藏的联系人）
- 结合了一系列数据视图和深层次导航的应用（如邮件应用的不同文件夹，购物应用的不同文件夹）

自上而下

开始界面的布局需要花更多的心思。这是用户打开应用看到的第一个画面，需要能够满足新用户，兼顾老用户。

问自己这样一个问题：“一般用户最想用我的应用来干什么？”，据此构建开始界面的用户体验。

内容优先

很多优秀的应用都关注内容的呈现。尽量避免纯导航界面，，在开始界面就以内容为核心，让用户立即与内容交互，获取核心体验。选择适合数据呈现和屏幕尺寸的布局。

锚链接导航和按钮

跟工具栏一样，在应用的操作栏上可以组织排列一些重要操作的按钮。理想的导航放置区域是屏幕的顶端，包括标签的切换、左侧侧拉导航，都可以放在屏幕顶端。如果你的内容是可以搜索的，在应用操作栏中要包含搜索按钮，用户可以快速搜索，直达内容。

注重功能

当你的应用有大量的功能和操作时，关键点在于：要适度指引用户，将用户的注意力焦点转移到你想要的地方。在内容区域，要凸显重要的导航。一些特色功能可以用悬浮按钮来表示。在你的应用中不要强调那些不常用的操作路径。

顶层视图的设计策略

最顶层的界面向用户传递应用最主要的功能和内容。有些应用功能会专注内容，简化导航。而另外一些应用中，顶层界面可能会包含多个视图，务必确保这种情况下，用户可以在多视图之间高效地导航。

选择最适合您应用需求的导航的方式。

单一视图中的嵌入导航

将所有必要导航元素内联到内容中，能极大提高导航的可见性。当应用的导航架构非常简单时，可以采用这种方法。然而，当导航系统过于庞大时，大量的导航路径会占据内容的空间，且导航路径可能会全屏散布，而不是集中在某个位置，方便用户导航。

在这些情况下使用内容内部导航

- 您的应用程序的主视图非常重要，其他视图无可替代。
- 用户可以在主视图中轻松完成常见任务。
- 你认为用户会较少使用你的应用，因为不熟悉，所以需要非常明显的导航路径。

一小部分同等重要的视图界面，可采用标签选项卡来进行导航切换。

如果你应用中功能区域很少，且层级较浅，可使用在顶层界面中使用标签选项卡，只需轻轻点击或滑动便可切换界面。。然而，使用标签选项卡会占据大量空间，不过当标签选项较少时，这种导航设计模式较为高效。

在这些情况下使用标签选项卡：

- 你希望用户在应用中频繁切换界面。
- 顶层界面中所展现的标签选项卡数量有限。
- 你希望用户能够充分了解其他界面。

通过左侧的侧拉抽屉来管理更复杂的结构

左侧导航面板可以同时显示大量的导航目标。当你的应用功能围绕主界面而构建时，这种导航模式极为有效，然后将抽屉可以作为一个不经常访问的链接索引。如果你的应用程序需要从较低级别的层级跨越信息架构层级直达另一个页面时，这种快捷导航方式让用户可以更灵活操作。然而，这种导航模式并不明显，需要一定的学习成本，让用户熟悉这种导航模式。

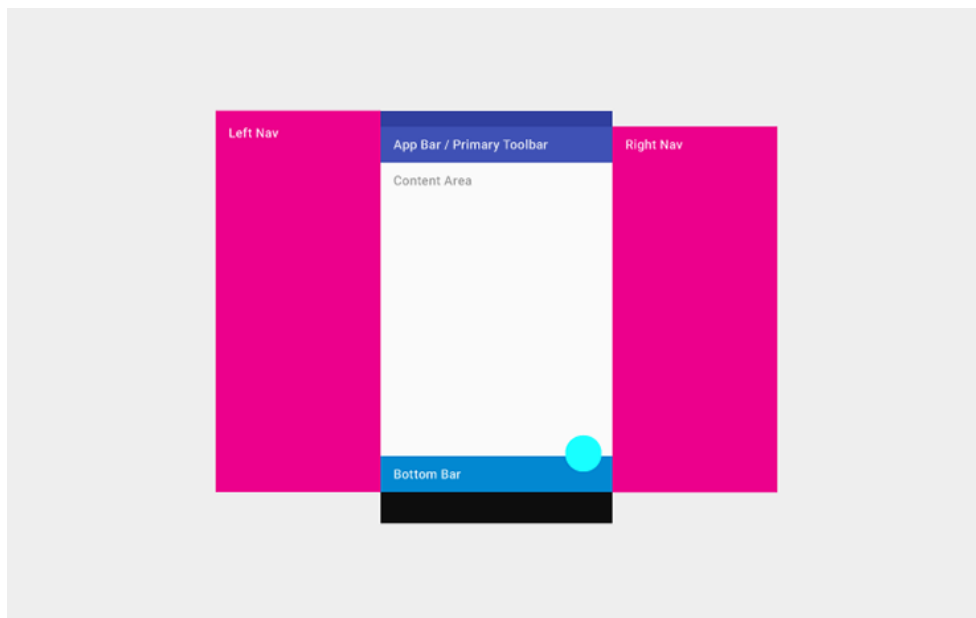
在这些情况下使用抽屉：

- 你的应用程序有大量的顶层视图
- 你希望在不直接关联的内容之间快速切换.
- 你有特别深的导航分支结构，并希望可以快速返回到应用的最顶层.
- 你想减少界面中导航元素的比重、想减少用户不常用链接，便可把他们放入到侧拉抽屉中。

不管选择何种顶层视图设计策略，都推荐内容内联导航，可以迅速的导航至相关内容。例如，推荐同一歌手的其他歌曲，根据浏览历史推荐相关产品，例如浏览一个用户发布的其他内容。这些关联导航让相关内容可以更自然的进一步呈递。

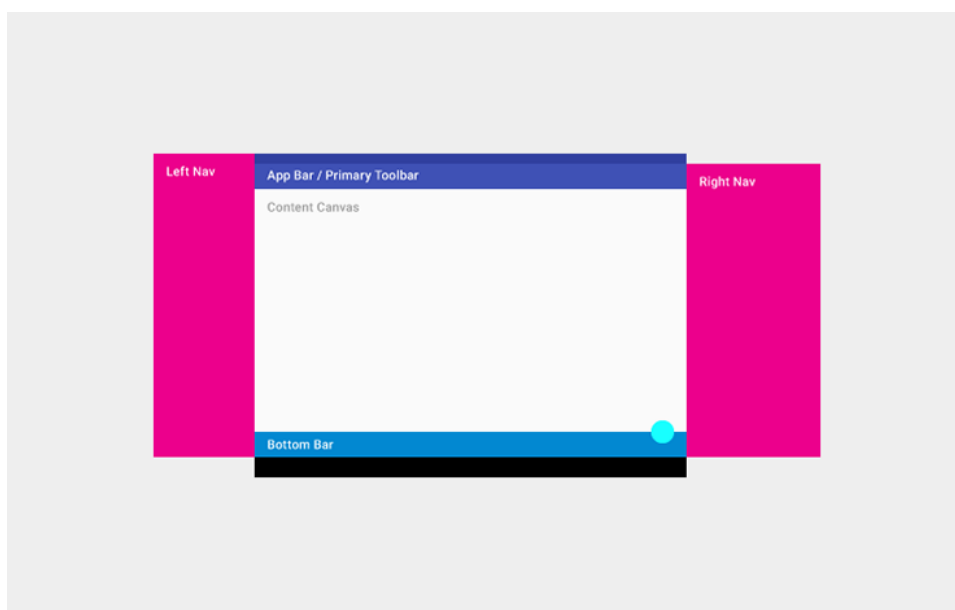
移动端

该结构包括一个永久的应用栏和悬浮操作按钮。一个附加的底部栏，里面可以添加一些附加功能和操作。提供了用户可以导航的侧拉菜单，同时侧拉菜单会覆盖其他元素结构。



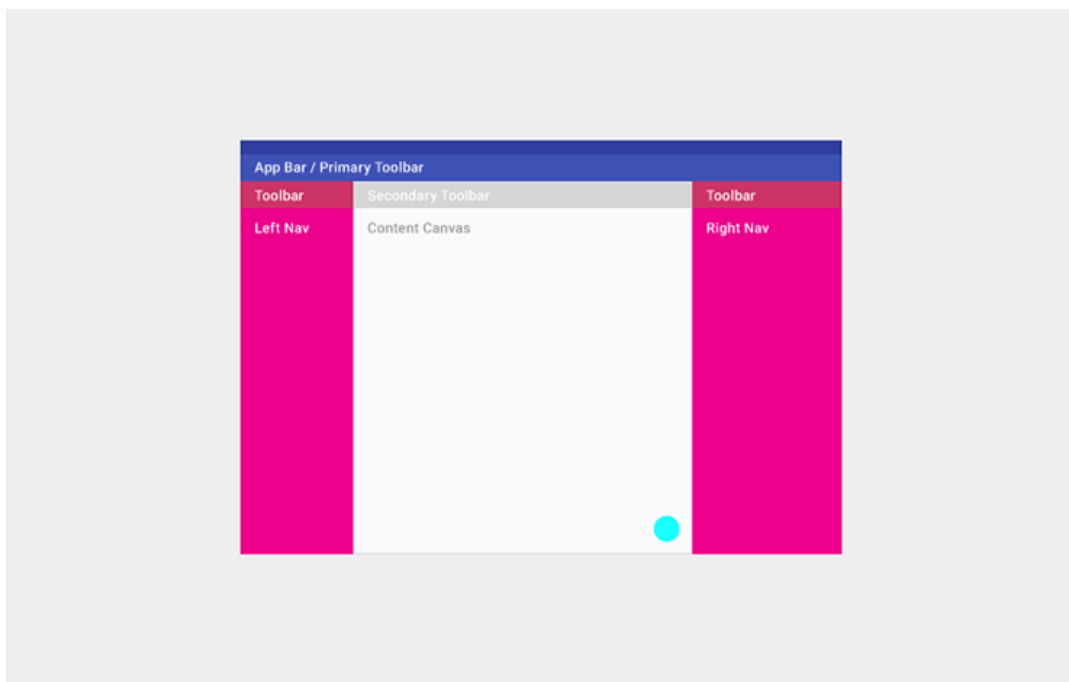
平板端

该结构包括一个永久的应用栏和悬浮操作按钮。应用栏吸收了平板电脑和手机的底部元素。一个附加的底部栏，里面可以添加一些附加功能和操作。左侧为导航菜单，用户如需访问，会覆盖所有其他结构元件。右导航菜单可以暂时访问或固定作为永久陈列。



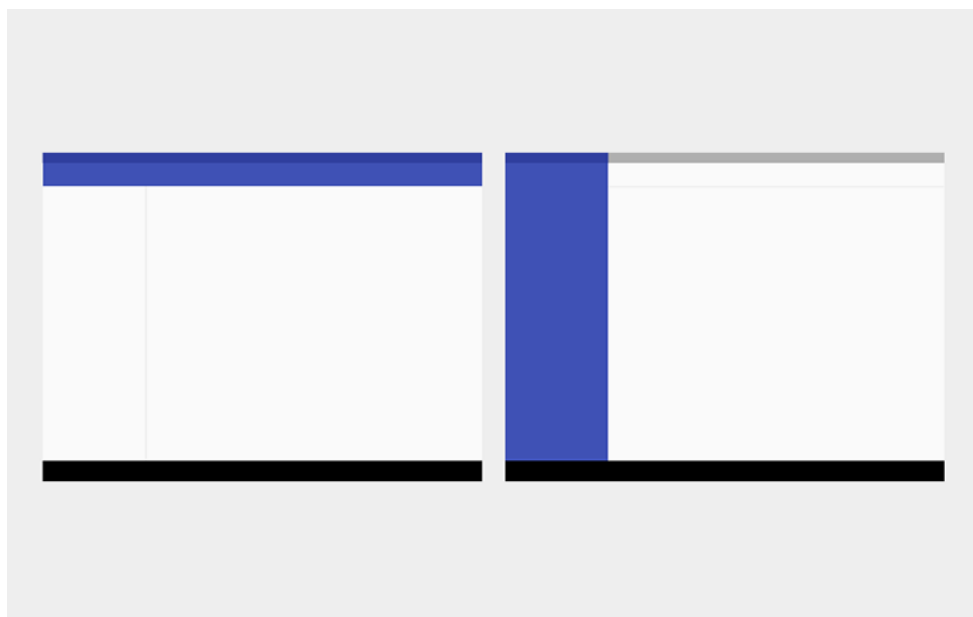
桌面端

该结构包括一个永久的应用栏和悬浮操作按钮。操作栏吸收从平板电脑和手机的底部元素。窗口控件可被归纳到操作栏中。侧面导航菜单可以暂时访问或固定作永久陈列。侧面导航菜单以及内容可以有自己的二级工具栏，显示标签选项卡，样式，或辅助操作。



UI规范

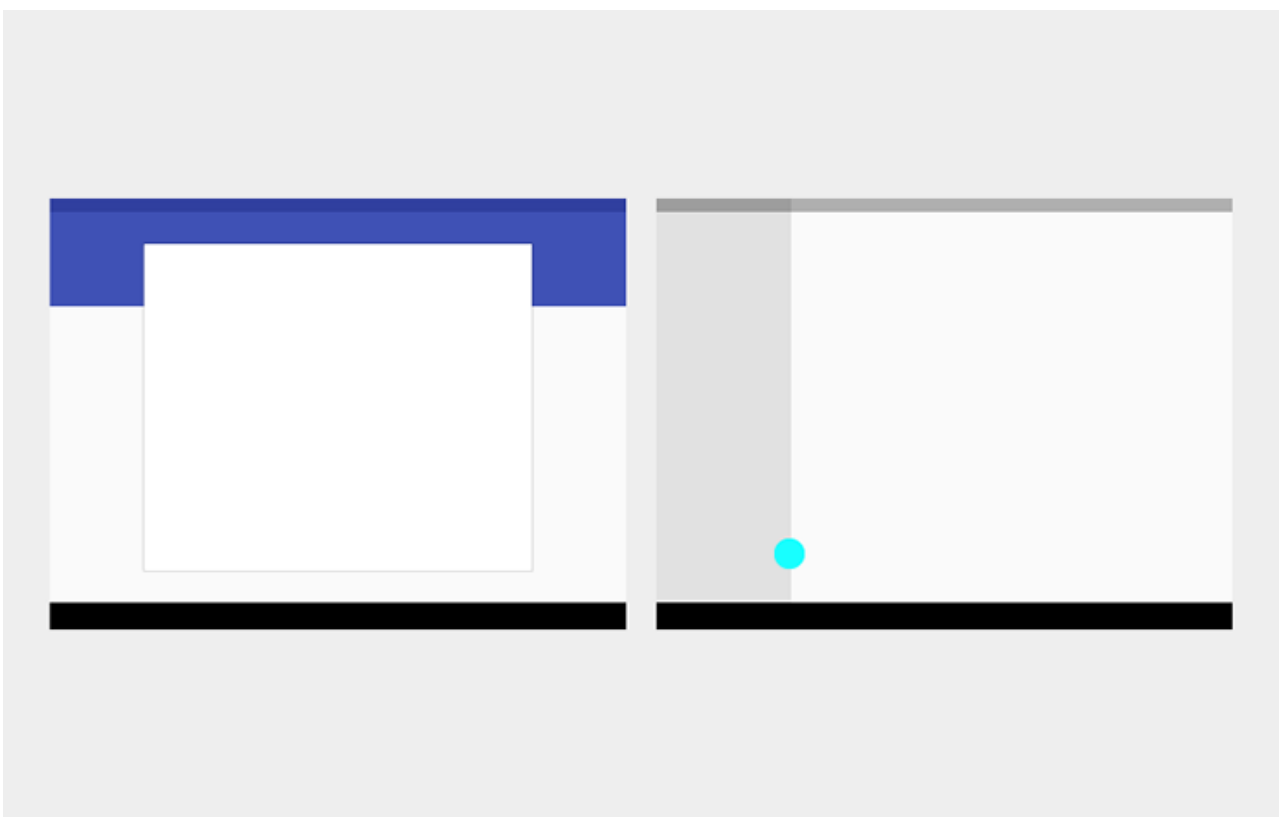
主要由水平或垂直的分割线来定义。



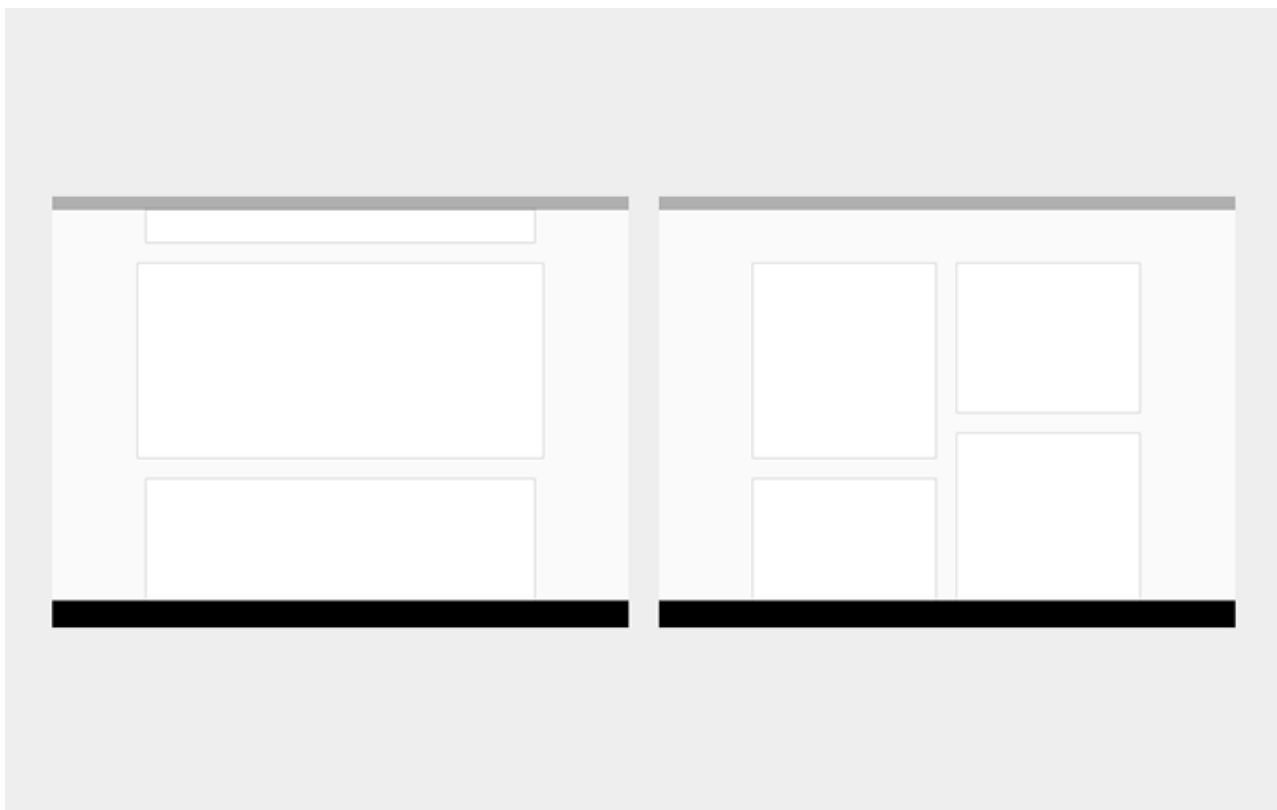
划分的区域不宜太多，否则会导致L型布局（就是页面分散）。相反，用空白字符语言（ WhiteSpace，是一种只用空白字符（空格，切换标签和回车）编程的语言：译者注）来描绘中间地带



卡片和悬浮按钮，打破页面区域

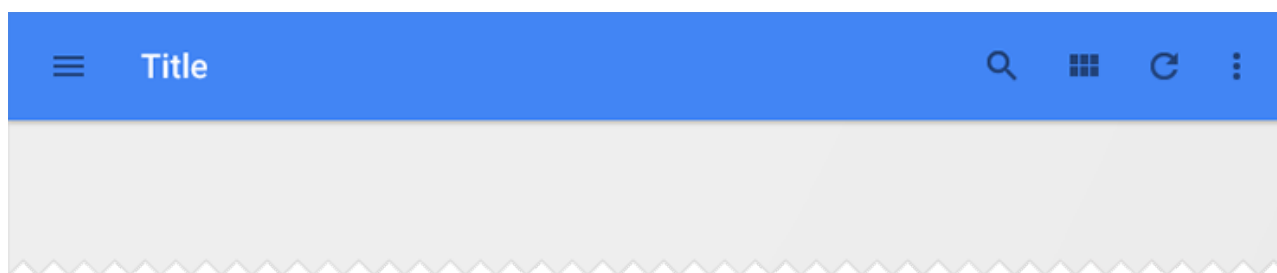


使用卡片来组织内容，可满足一些特殊的交互行为，也可以组织信息。卡片式布局需要大量的留白空间，也需要分割线来划分清晰。

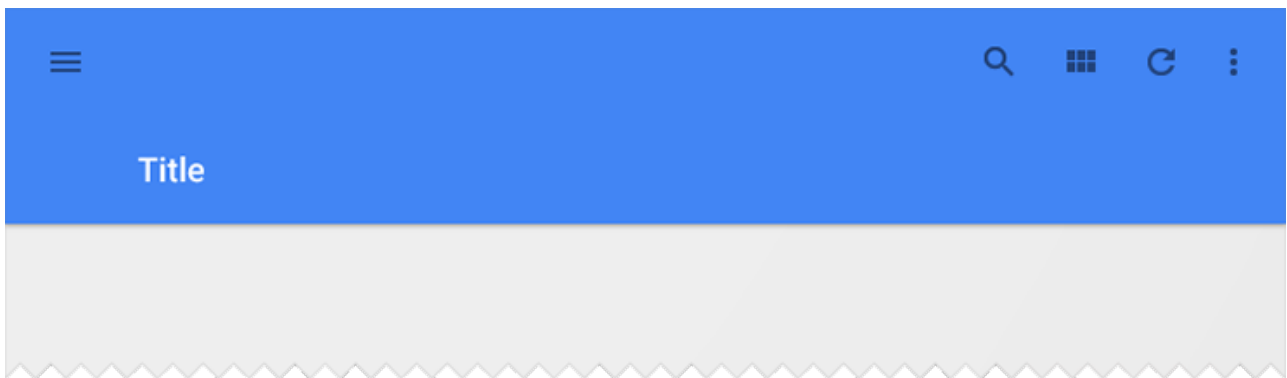


工具栏

工具栏具有多种用途，适用于应用程序的各个区域。以下是您可以使用工具栏的一些例子。



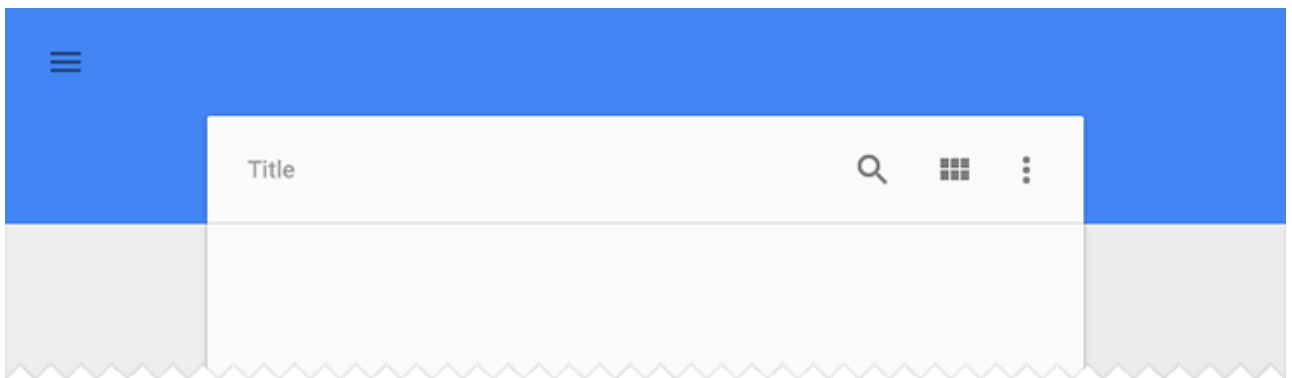
宽屏、默认高度的应用栏



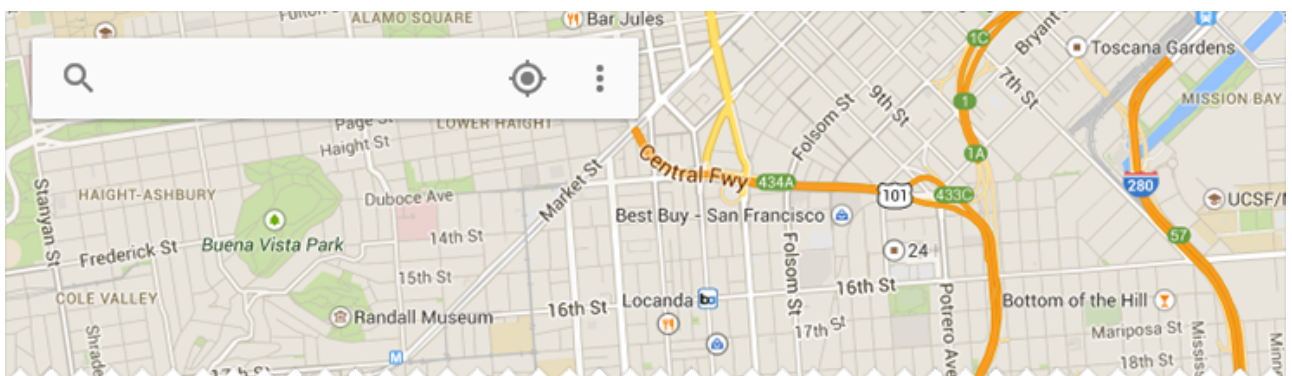
宽屏,高度扩展的应用栏 跨越多个列宽



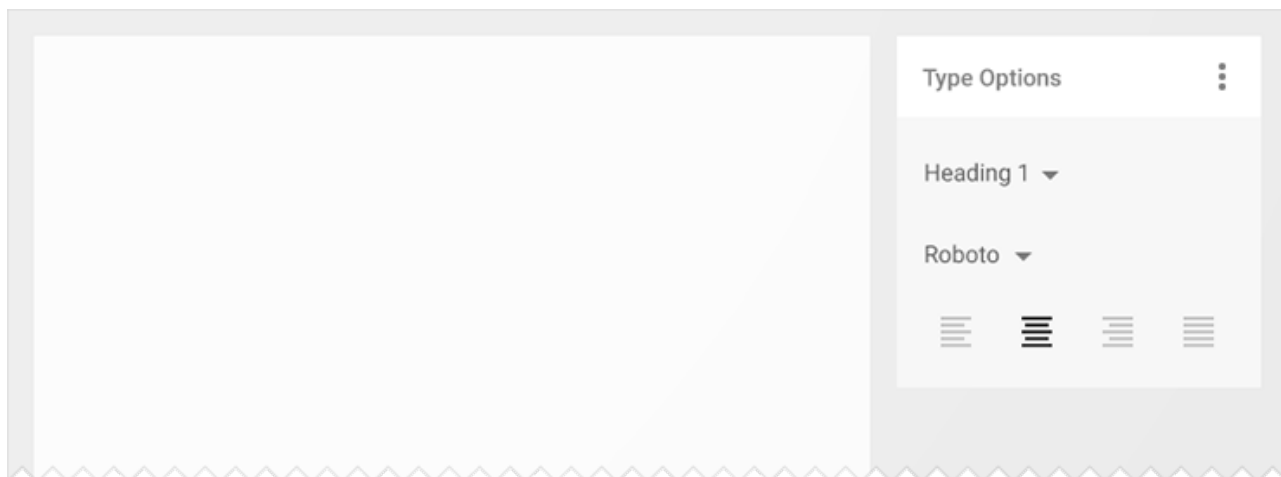
单栏宽的工具栏，可以清晰的显示出层次结构



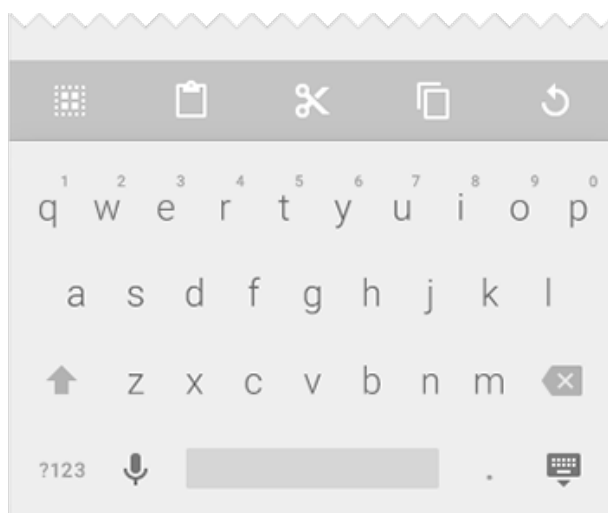
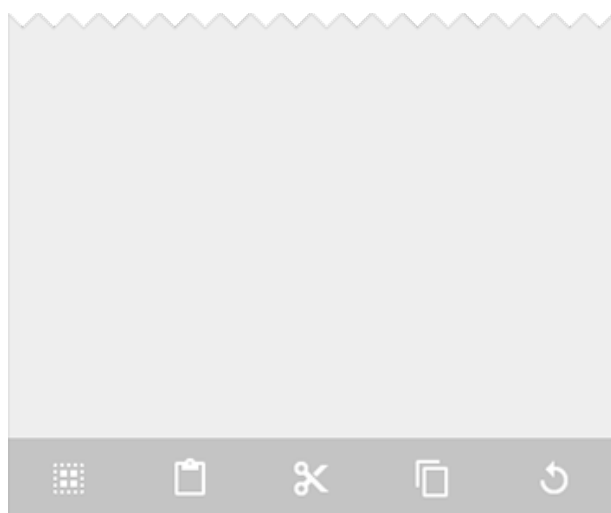
灵活的工具栏和卡片工具栏



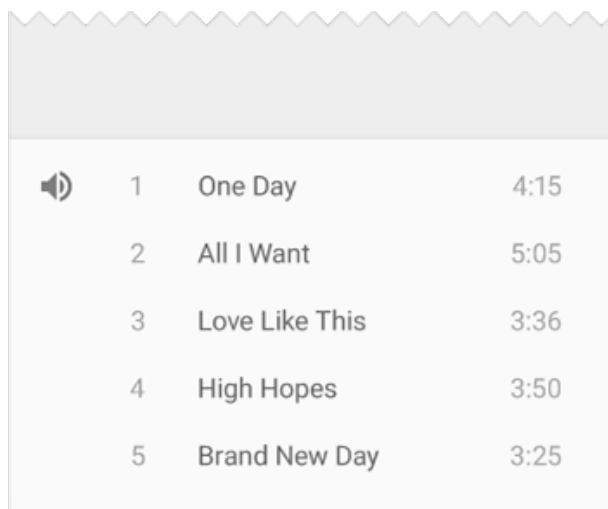
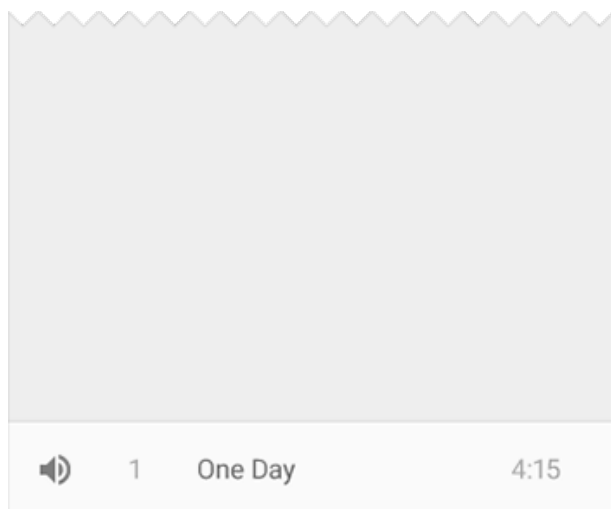
悬浮工具栏



分离的工具栏面板



底部的工具栏,,紧挨着键盘或其他底层组件



底部工具

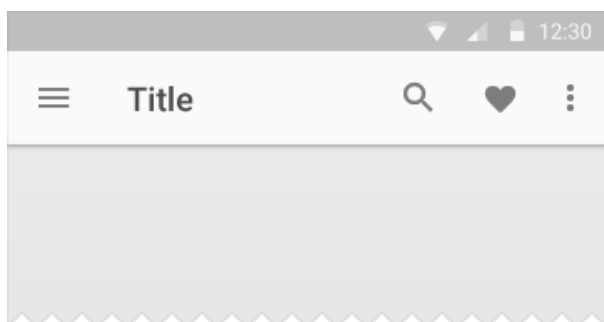
应用栏

应用栏结构

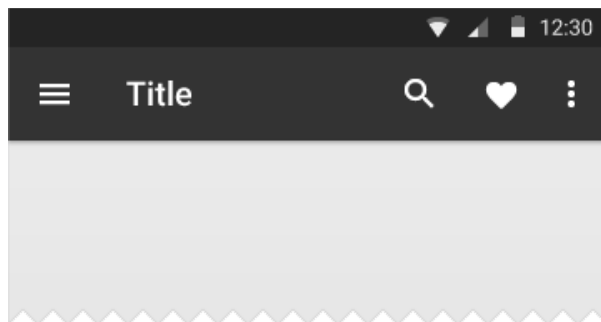
应用栏，前身为Android的操作栏，用于品牌推广，导航，搜索，和操作。

如果你的应用程序使用侧拉抽屉。那么点击Nav Icon（导航图标），便可打开侧拉，同时导航图标变为箭头。标题可以是一个应用程序的标题，页面标题或网页过滤器。

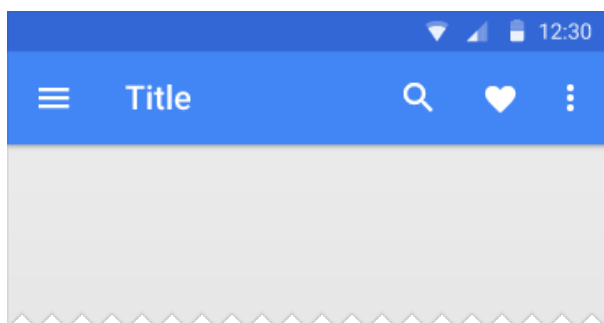
图标应该和应用相关操作有关联。点击Menu Icon（菜单图标）可出现一堆菜单项，其中包含帮助，设置和反馈。



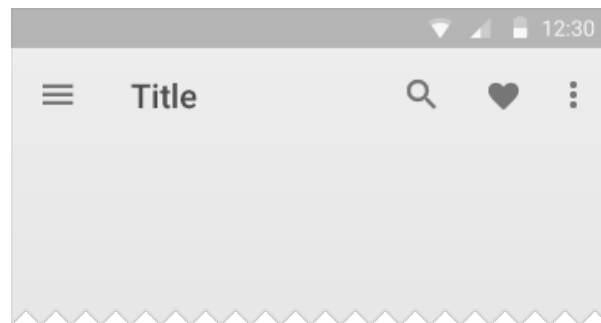
亮色



暗色



有颜色



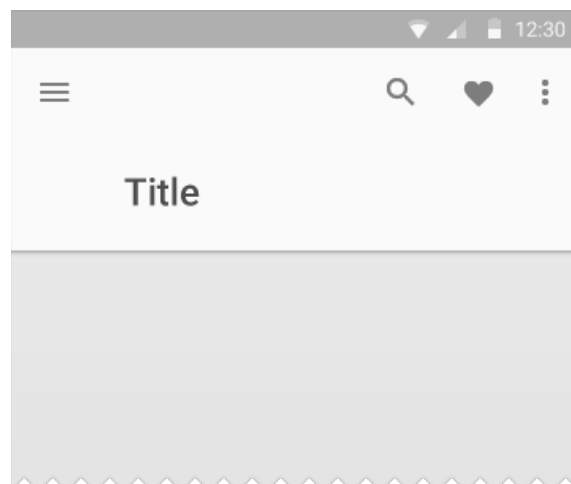
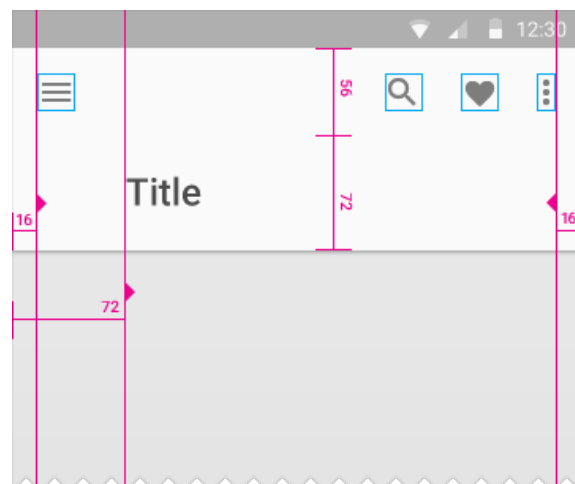
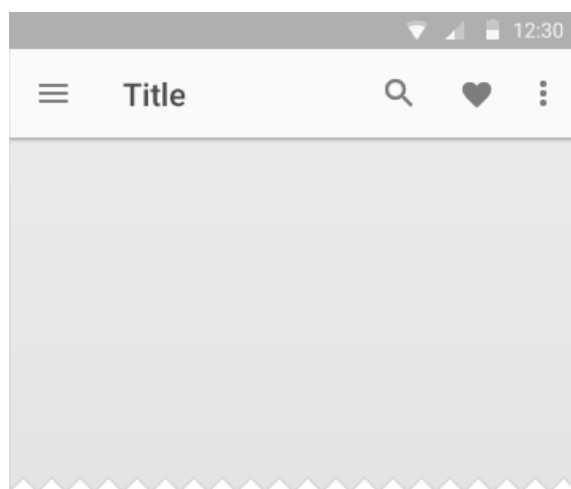
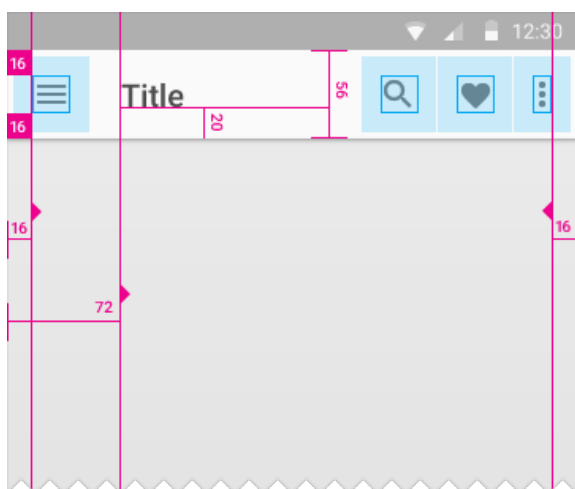
透明的

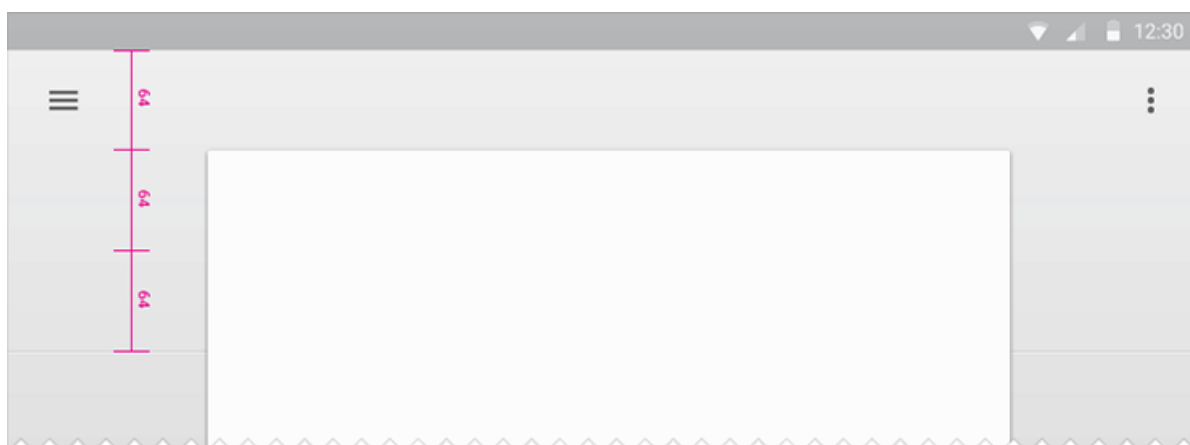
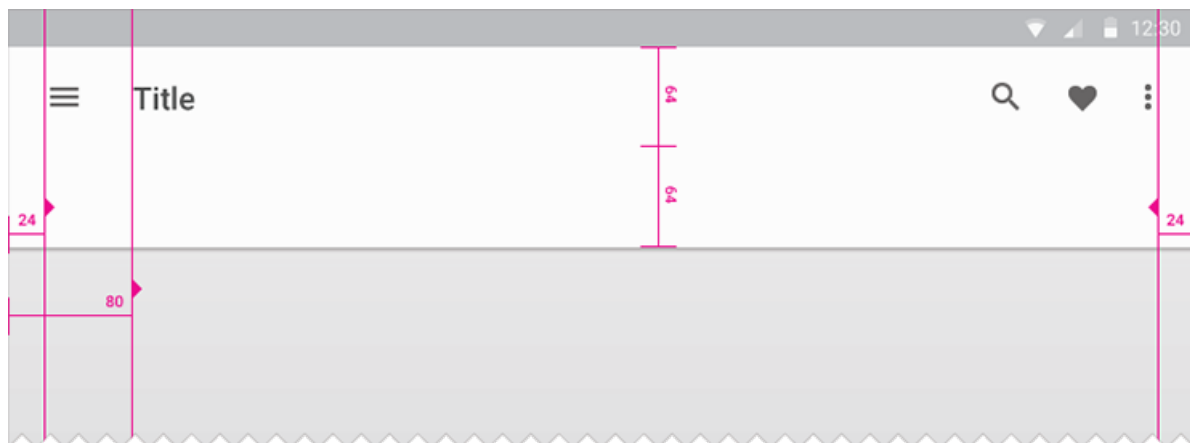
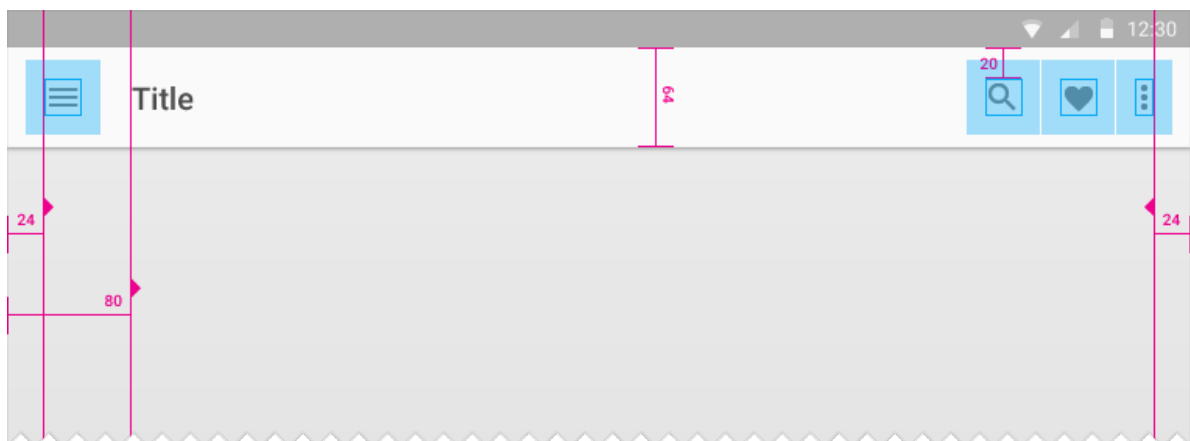
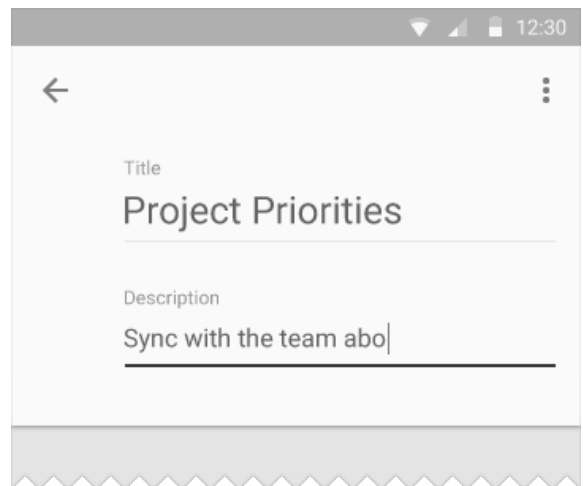
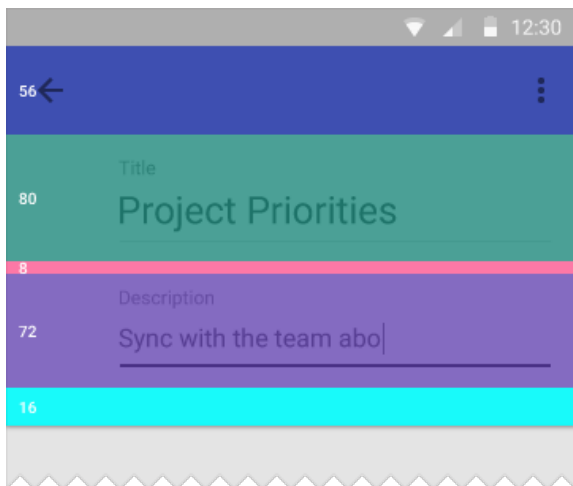
参数标准

默认高度：

- 手机竖屏: 48dp
- 手机横屏: 56dp
- 平板电脑/台式机: 64dp

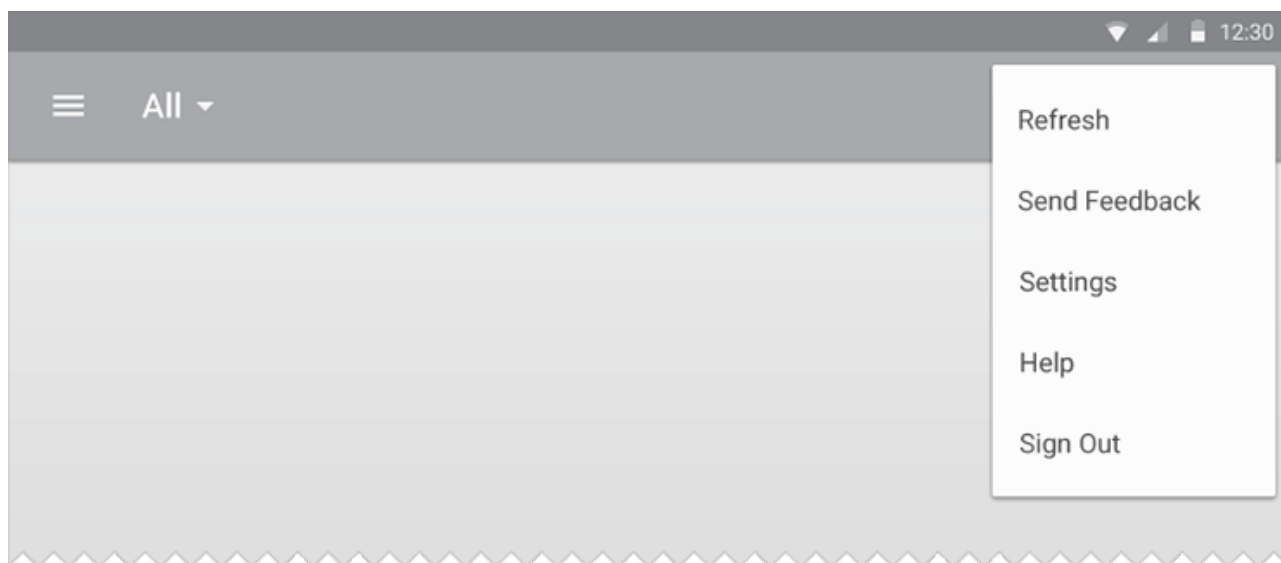
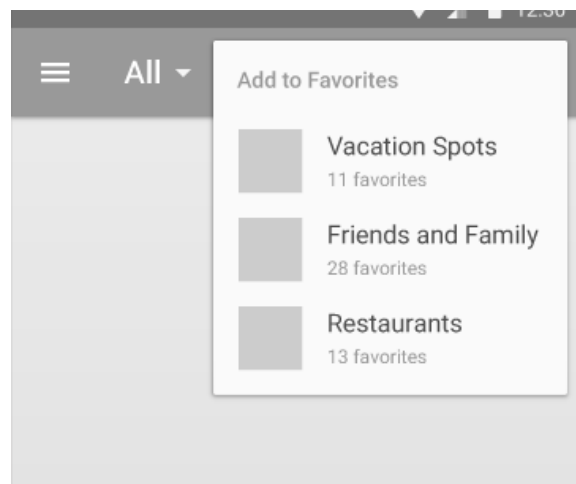
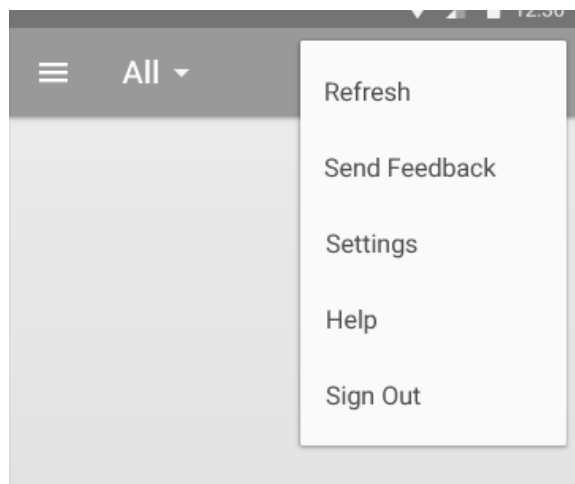
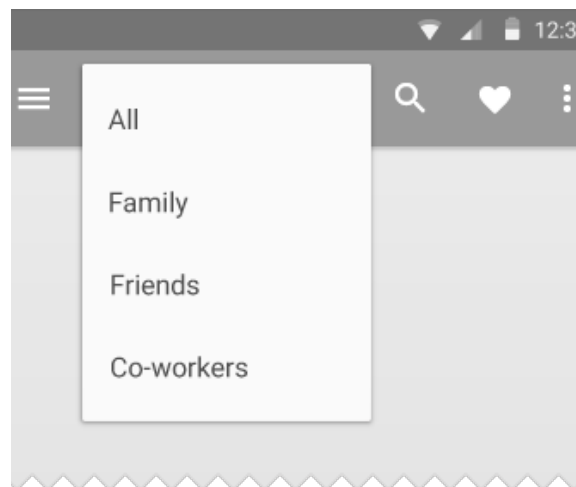
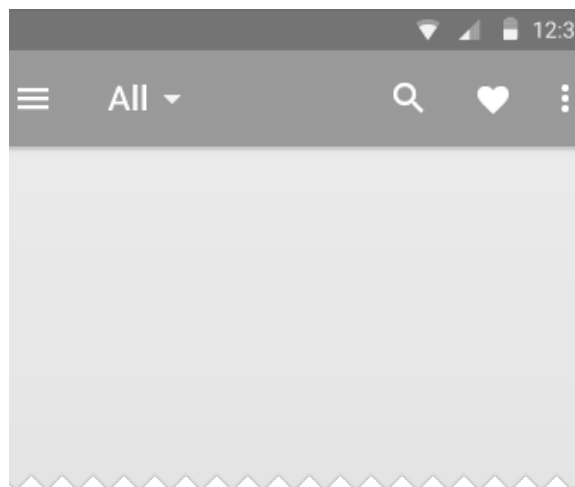
如要扩展应用栏，其高度等于默认高度加上内容增加量。





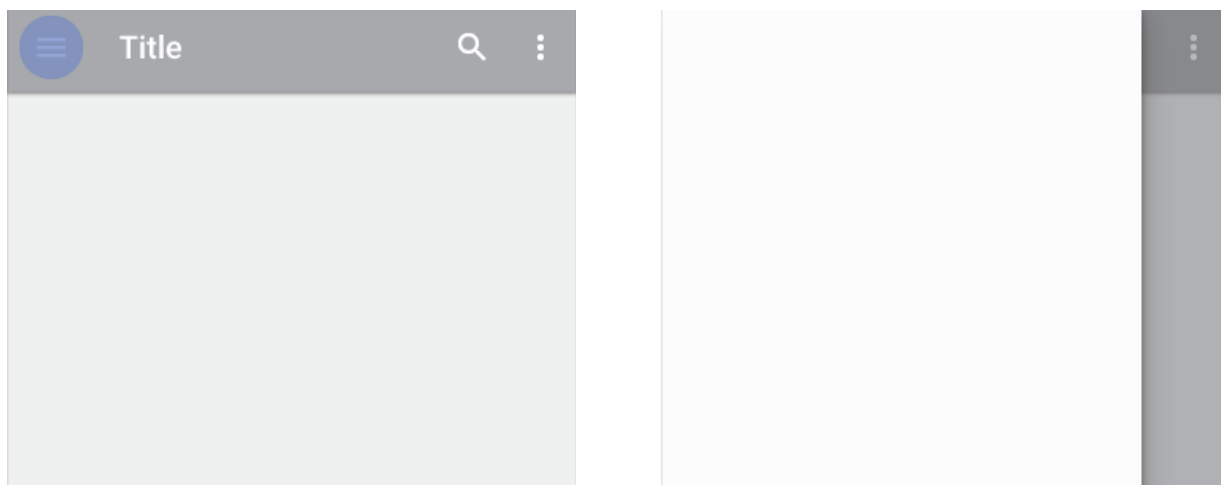
菜单

菜单是一个临时的卡片,总是覆盖应用栏,而不是作为应用栏的扩展



侧边导航

左抽屉和右导航可以固定显示，也可以暂时悬空覆盖于主界面中。在左侧侧拉菜单主要用于导航。在右边的侧拉菜单主要展现二级内容相关信息。



结构

侧滑抽屉会临时覆盖的内容。

参数标准:

移动端：侧导航宽度=屏幕宽度 - 应用栏的高度

示例:

Nexus 4: 304 dp

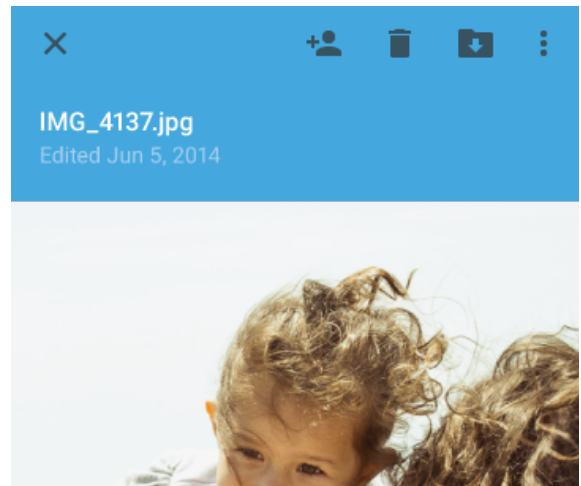
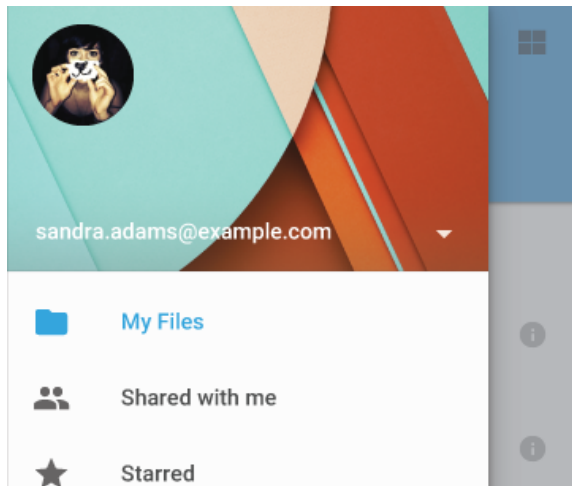
Nexus 5: 288 dp

iPhone: 264 dp

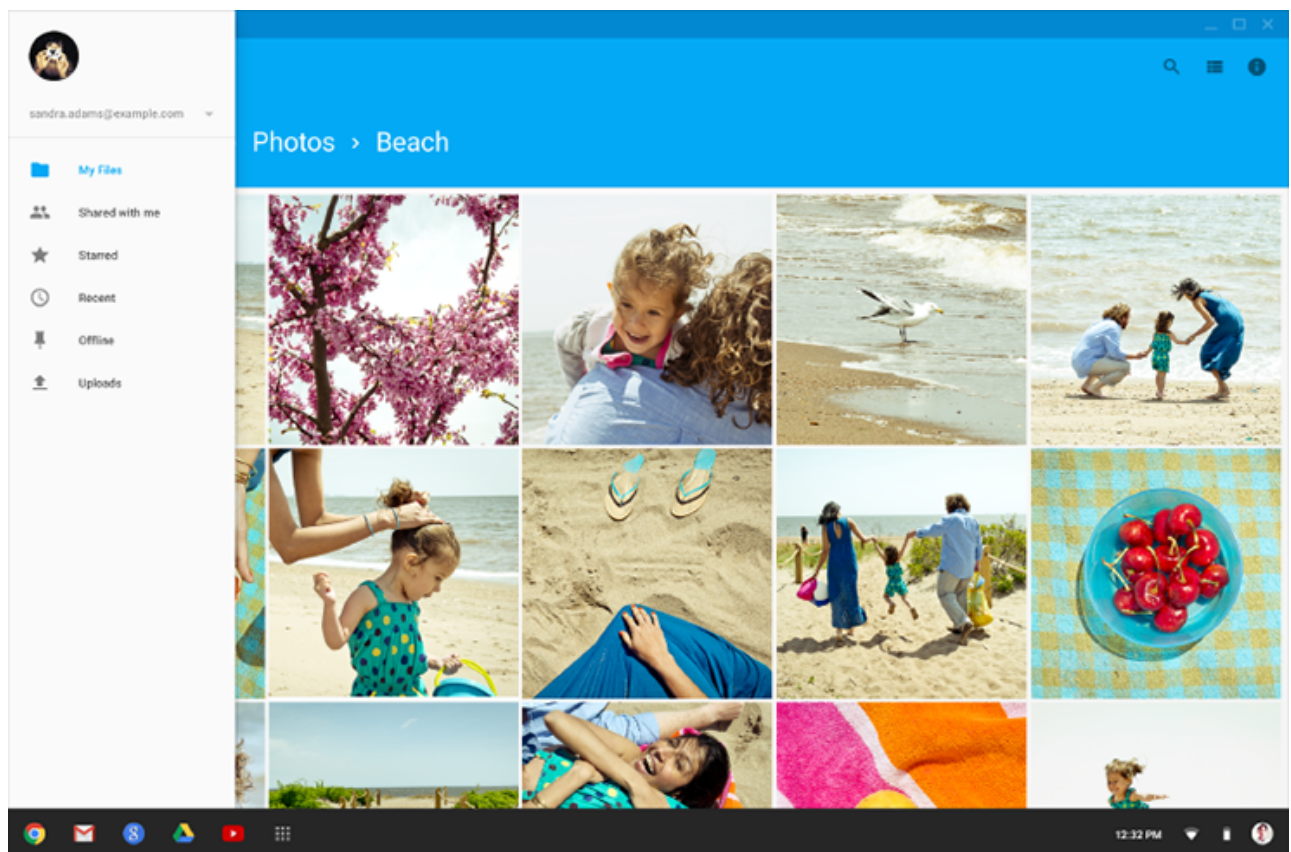
桌面：最大宽度左侧导航是400dp。右侧导航可根据内容的不同而不同。

桌面端/平板电脑：固定，宽度应与列对齐。

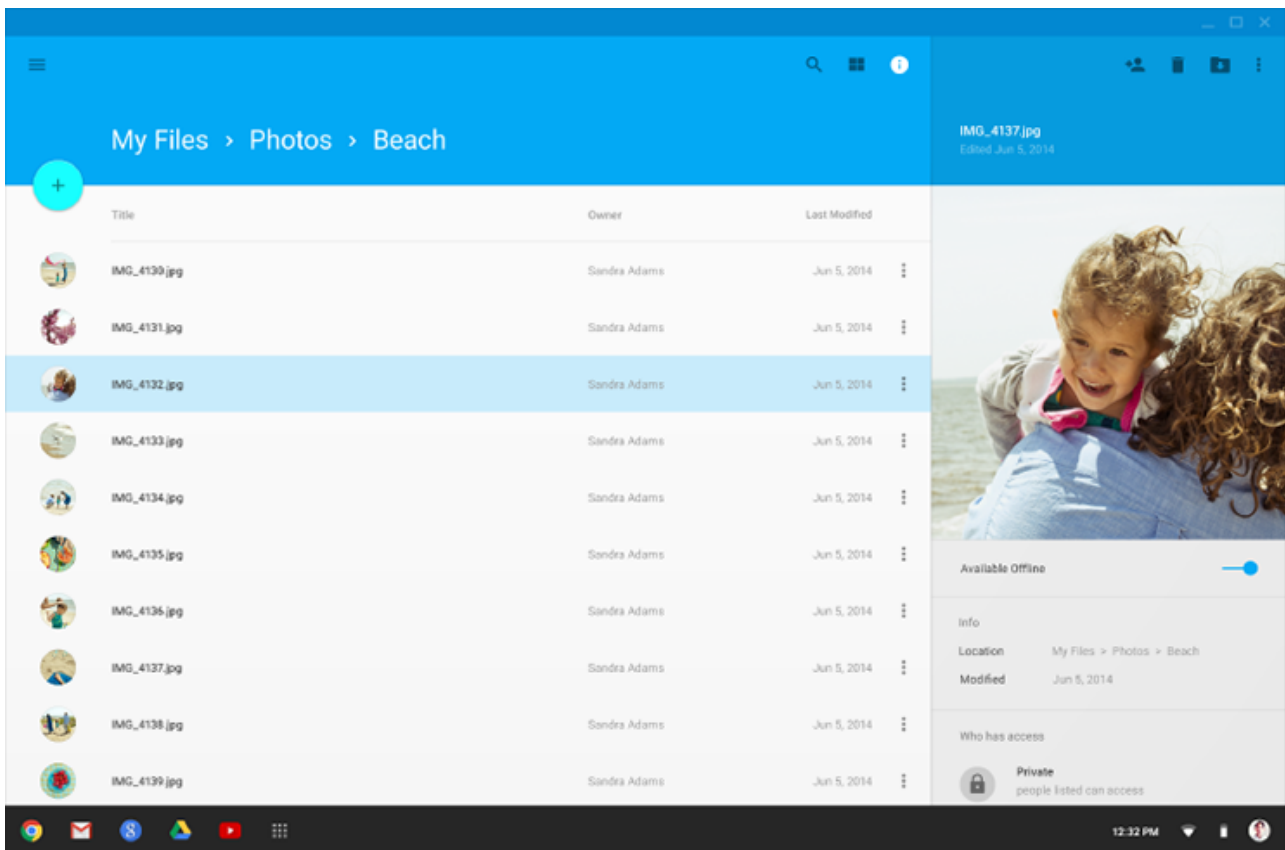
浮动最大宽度：304dp



手机侧滑



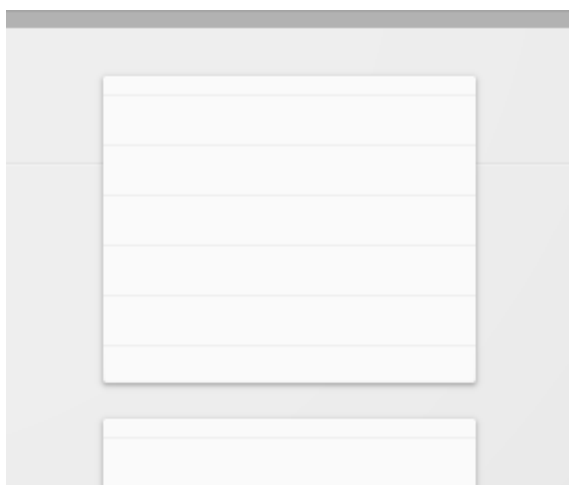
平板侧滑



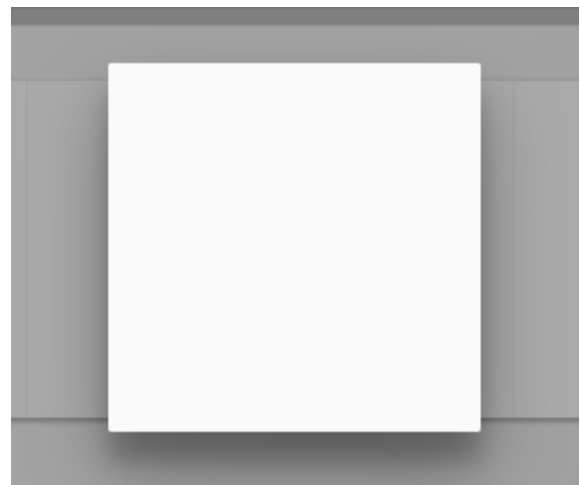
网页侧滑

基本框架

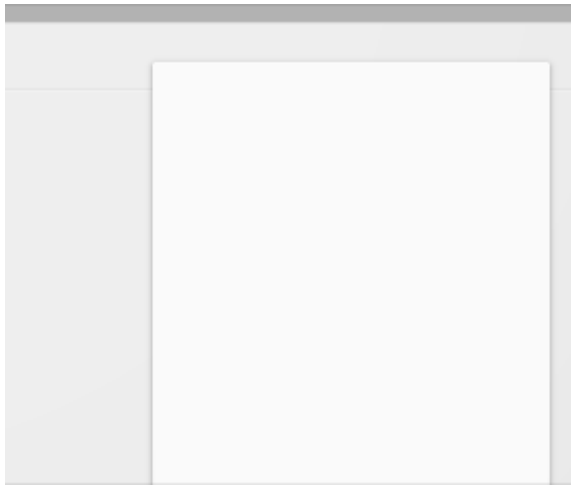
Whiteframes提供了各种结构布局的方法,分层和阴影。(均可以下载) 基本框架



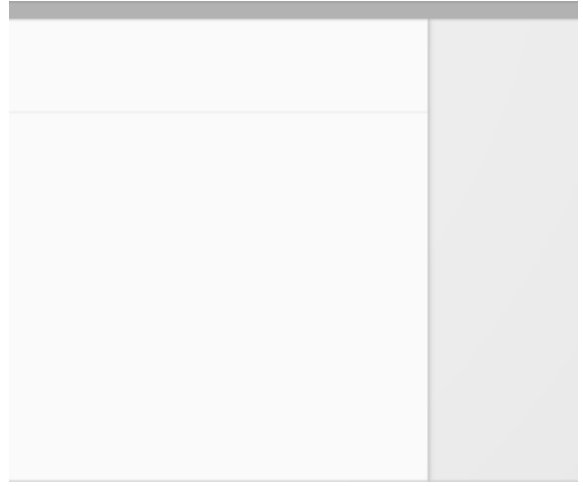
卡片内容的扩大和缩小



在移动设备中，覆盖的内容细节和聚焦的应用栏



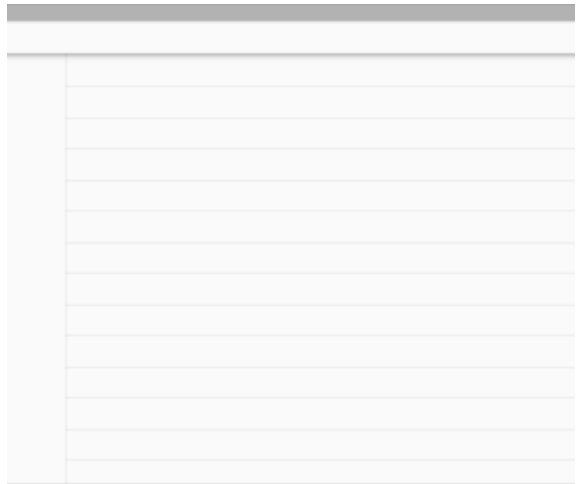
移动端，拥有多个工具栏的内容重叠
卡片和背景图像



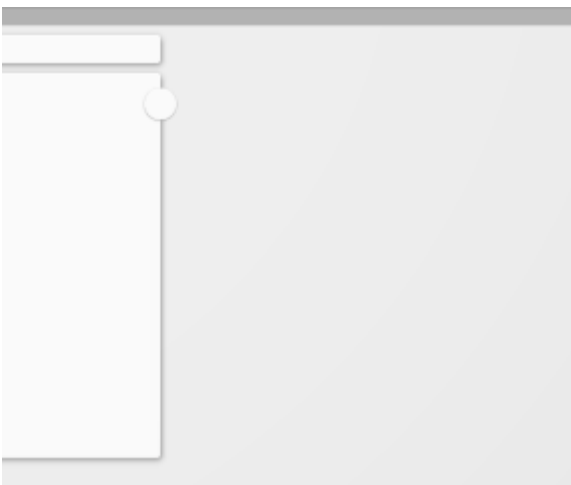
扩展应用程序栏和右侧拉抽屉



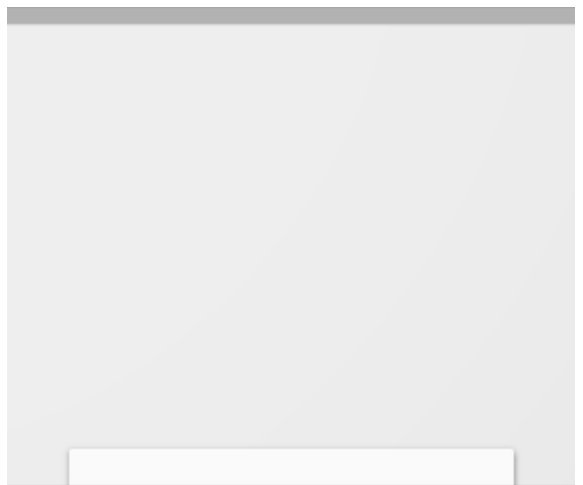
左导航抽屉和瀑布流



资源列表



嵌入搜索字段的全屏背景图像 和 卡片
式搜索结果



可扩展页脚的抽屉

5.1 底部卡片

用户可以使用一种底部卡片的导航方式，就像一张纸从屏幕底部向上滑动出来，底部卡片具有灵活性，简洁的显示和动作，不需要多余的说明。

用法

底部卡片特别适合给用户提供三个或者更多操作选择的时候，这样的操作不需要特别的说明就可以理解。但是如果只有两个或者更少的详细说明必须选择，那么就请考虑使用菜单或者对话框来代替，更合适。

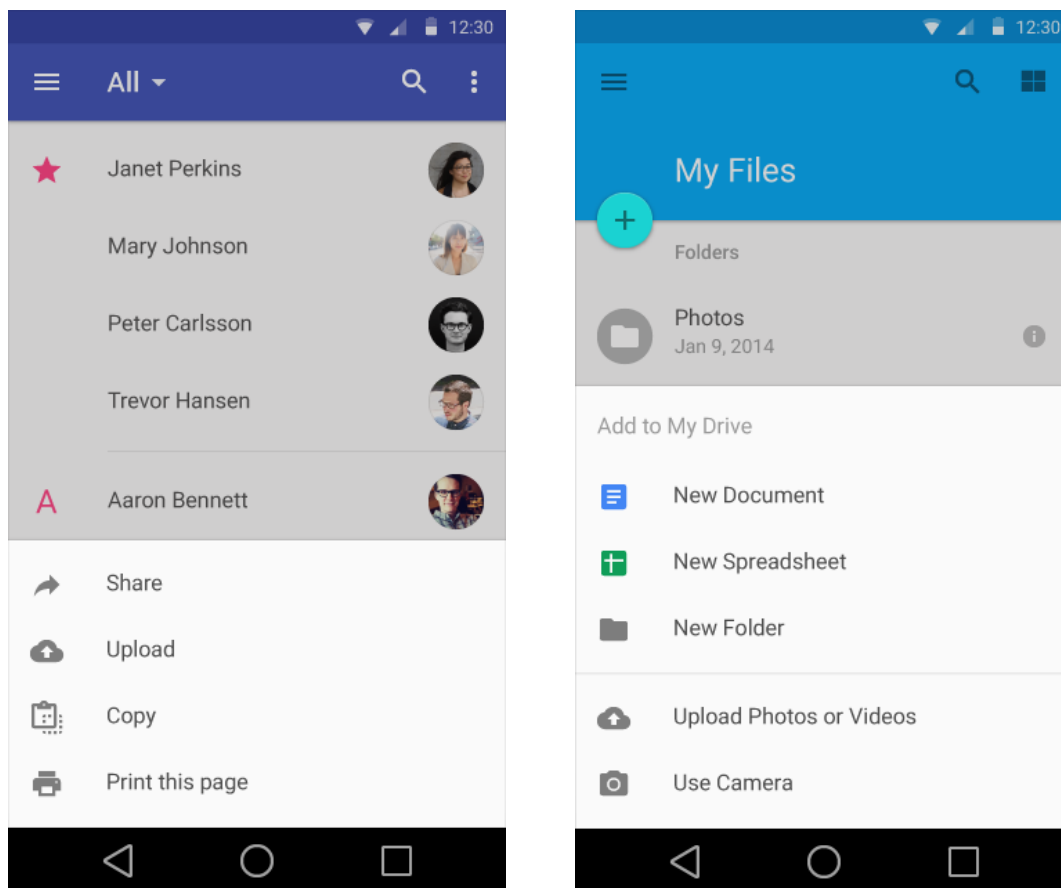
底部卡片可以使用列表式或者网络式，网络布局更加直观，容易识别。

你可以使用底部卡片显示其他程序，或者跳转过去。

内容

在标准的列表式卡片中，每个选项应该有一个文字描述和一个左对齐图标。必要时，可以使用分隔符进行逻辑分组。您还可以使用副标题或说明来解释选项含义。

底部卡片可以使用一个可以滚动的网络式标准图标+文案的链接。

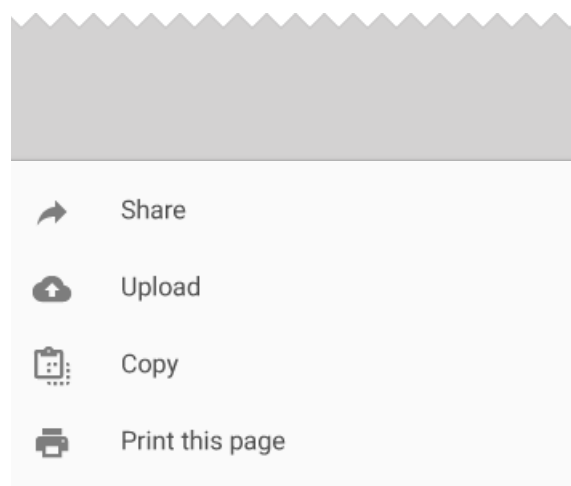
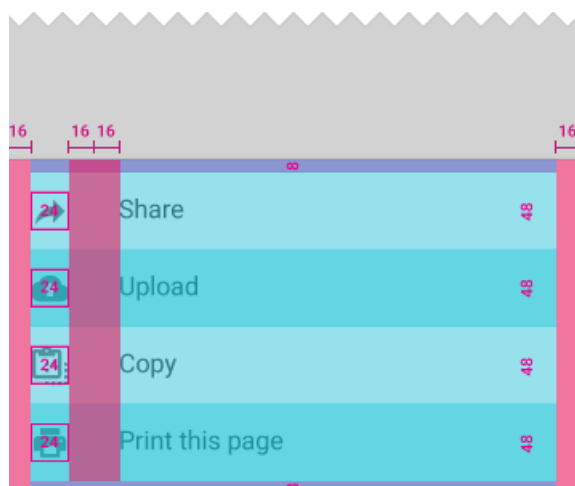


行为

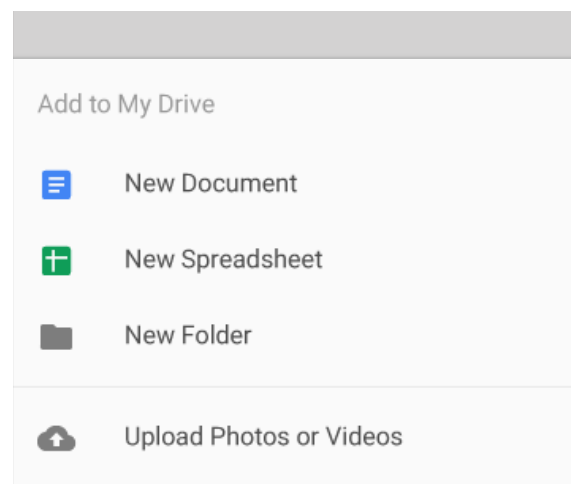
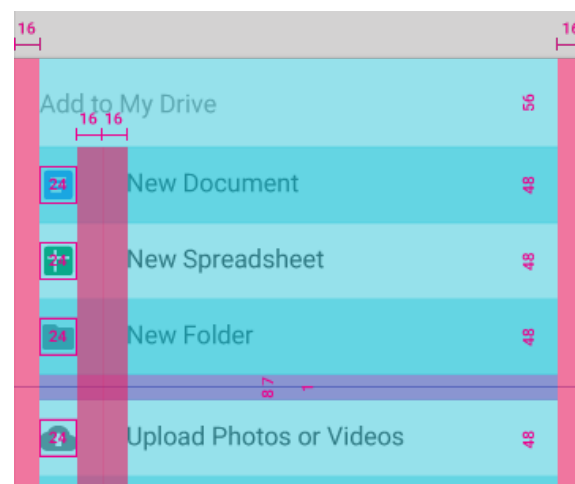
显示一个底部卡片，它应该从屏幕的底部边缘向上浮动，给用户初始的内容自然变暗。点击昏暗的区域可以退出卡片模式，纸张快速向下滑动出去。如果卡片包含初始画面中的一些操作，可以提高卡片容器的高度，并最终覆盖整个屏幕。当这个高度覆盖了整个屏幕，应该在顶部添加返回操作按钮，放在标题的左边。

规格

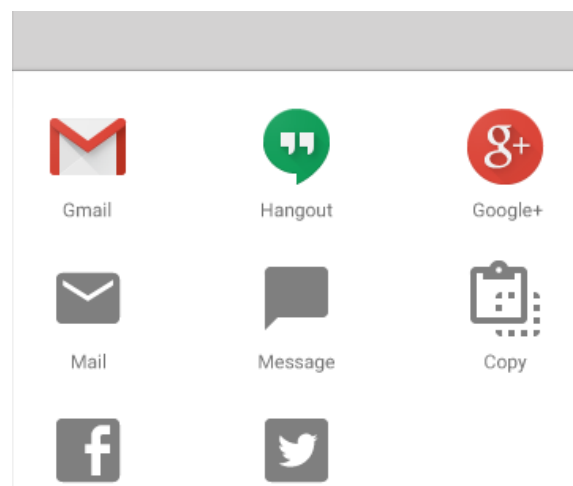
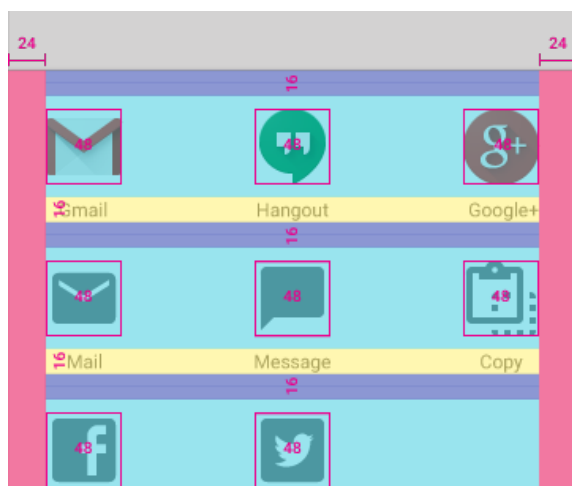
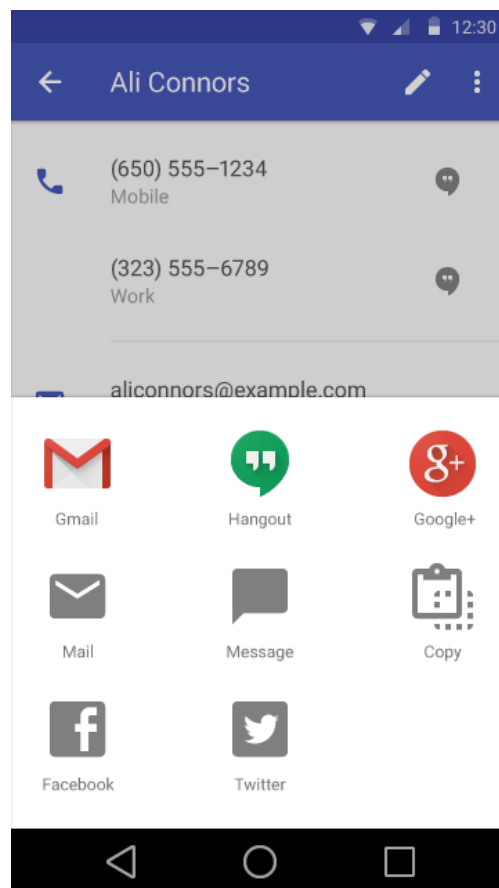
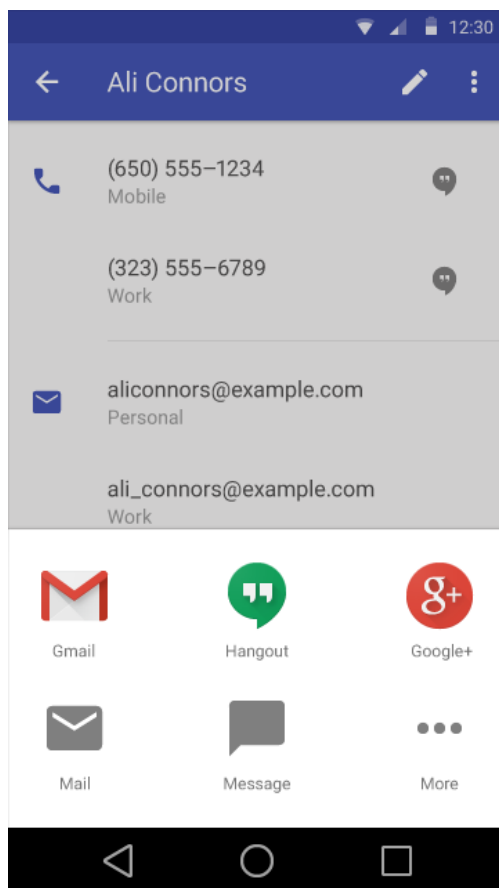
提供移动应用下列字体与颜色的规格和红色的间距区。



列表式卡片的间距



列表式卡片与顶部的间距



网路式底部卡片间距，含有一组标准的用于其他应用的操作

5.2 按钮

按钮由文字或者图案组成，文字或者图案必须能让人轻易地和点击后展示的内容联系起来。
主要的按钮有三种：

悬浮响应按钮，点击后会产生墨水扩散效果的圆形按钮。

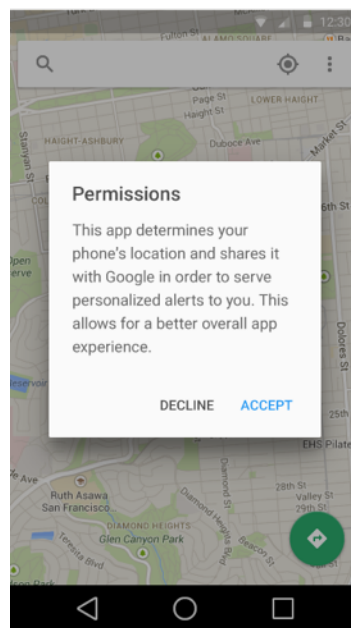
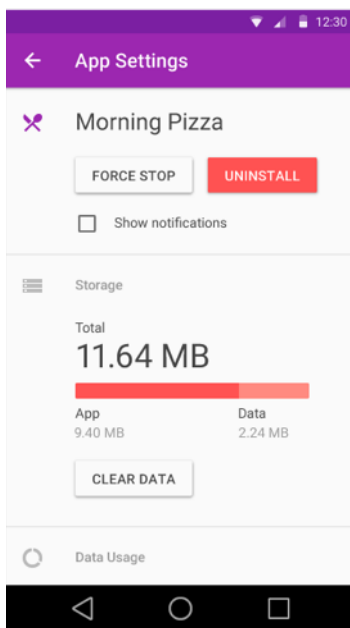
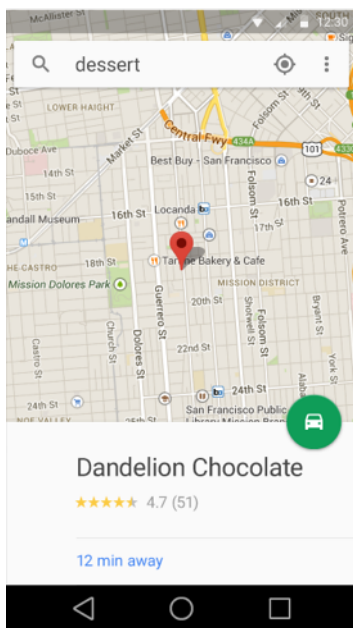
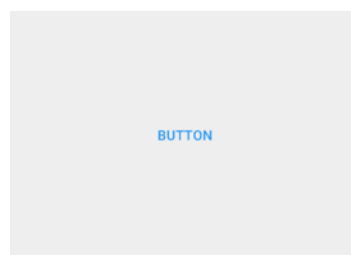
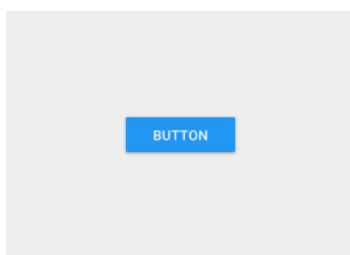
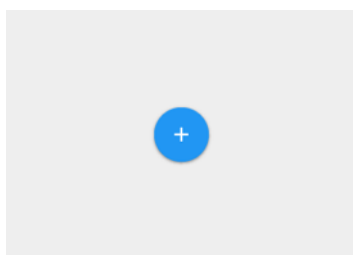
浮动按钮，常见的方形纸片按钮，点击后会产生墨水扩散效果。

扁平按钮，点击后产生墨水扩散效果，和浮动按钮的区别是没有浮起的效果。

颜色饱满的图标应当是功能性的-尽量避免把他们作为纯粹装饰用的物体。

按钮的设计应当和应用的视觉主题保持一致。

用法



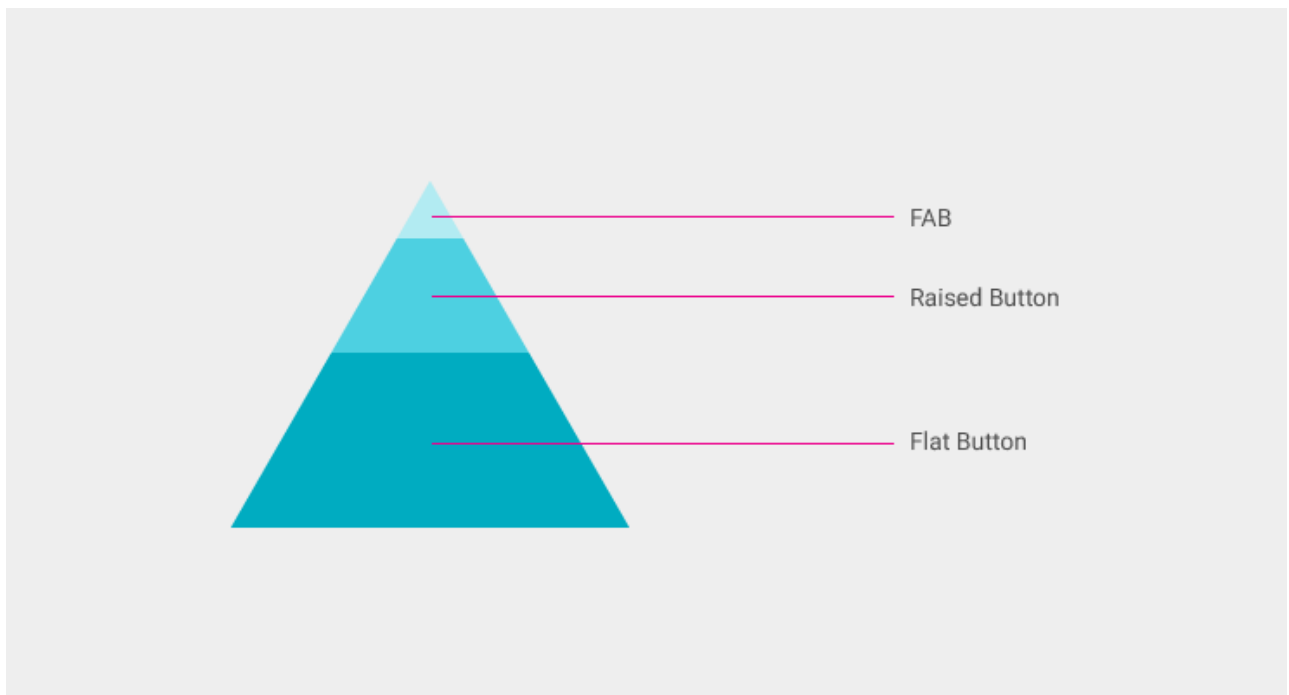
主按钮

按钮类型应该基于主按钮, 屏幕上容器的数量, 以及整体布局来选择.

首先, 审视一遍你的按钮功能: 按钮功能是不是非常重要而且应用广泛到需要用上悬浮响应按钮?

然后, 选择使用浮动按钮或者扁平按钮基于放置按钮的容器以及屏幕上层的数量。应该避免过多的层叠。

最后, 检查你的布局. 一个容器应该只使用一种类型的按钮. 只在比较特殊的情况下 (比如需要强调一个浮起的效果) 才应该混合使用多种类型按钮.



对话框中的按钮

对话框中使用扁平按钮作为主要按钮类型以避免过多的层次叠加。

Permissions

This app determines your phone's location and shares it with Google in order to serve personalized alerts to you. This allows for a better overall app experience.

DECLINE ACCEPT

Permissions

This app determines your phone's location and shares it with Google in order to serve personalized alerts to you. This allows for a better overall app experience.

DECLINE ACCEPT

36
16
16

Normal

Permissions



This app determines your phone's location and shares it with Google in order to serve personalized alerts to you. This allows for a better overall app experience.

MORE INFO

DECLINE ACCEPT

Focus/press areas

Permissions



This app determines your phone's location and shares it with Google in order to serve personalized alerts to you. This allows for a better overall app experience.

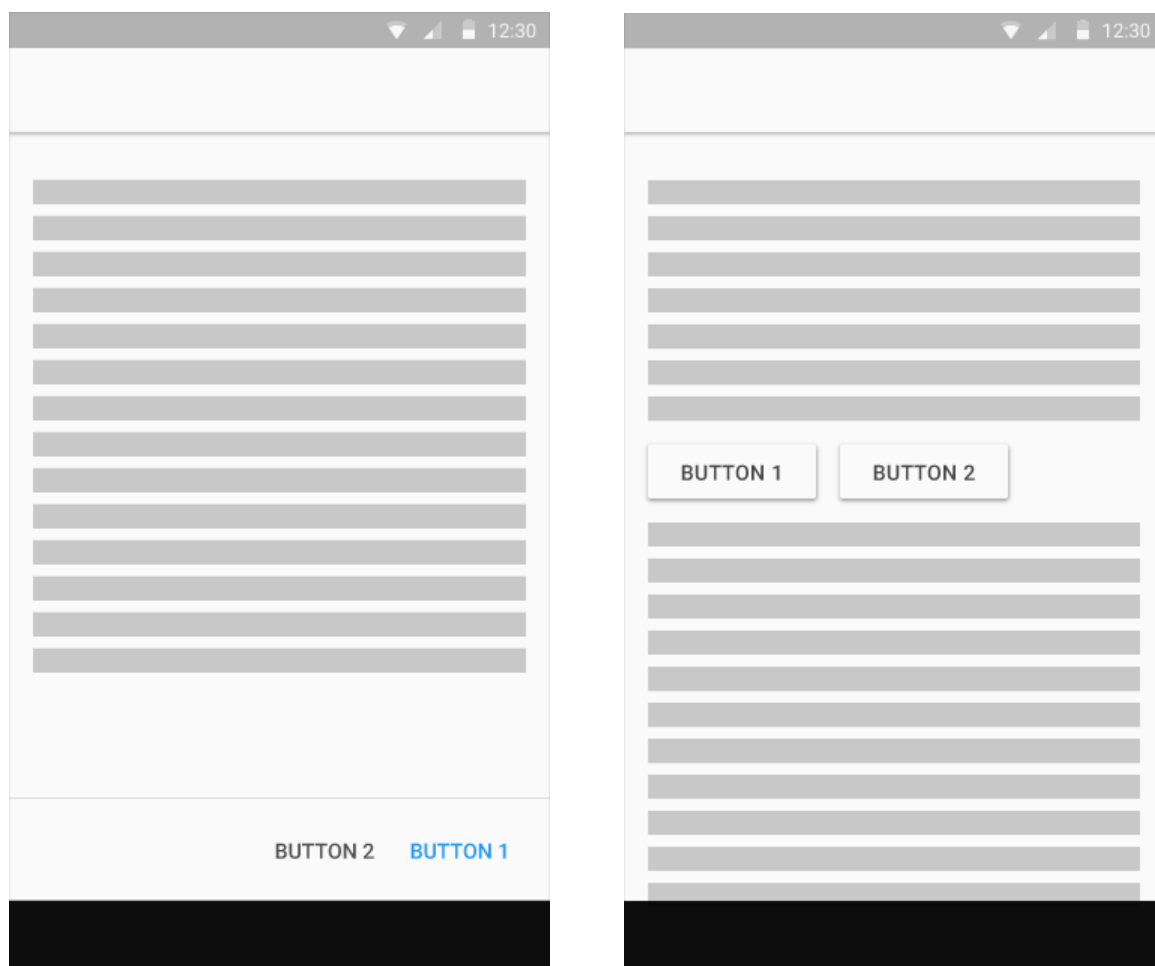
MORE INFO

DECLINE ACCEPT

16
16

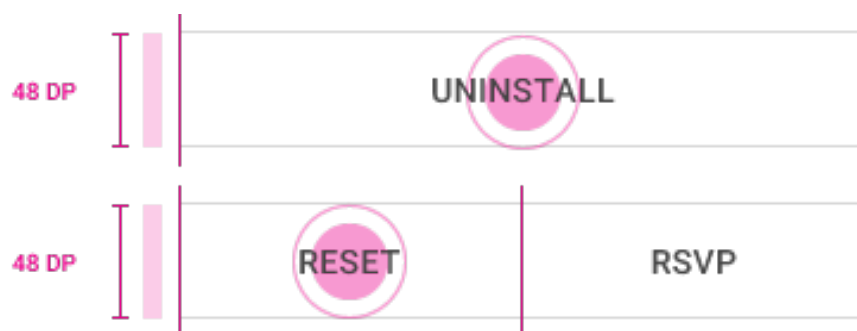
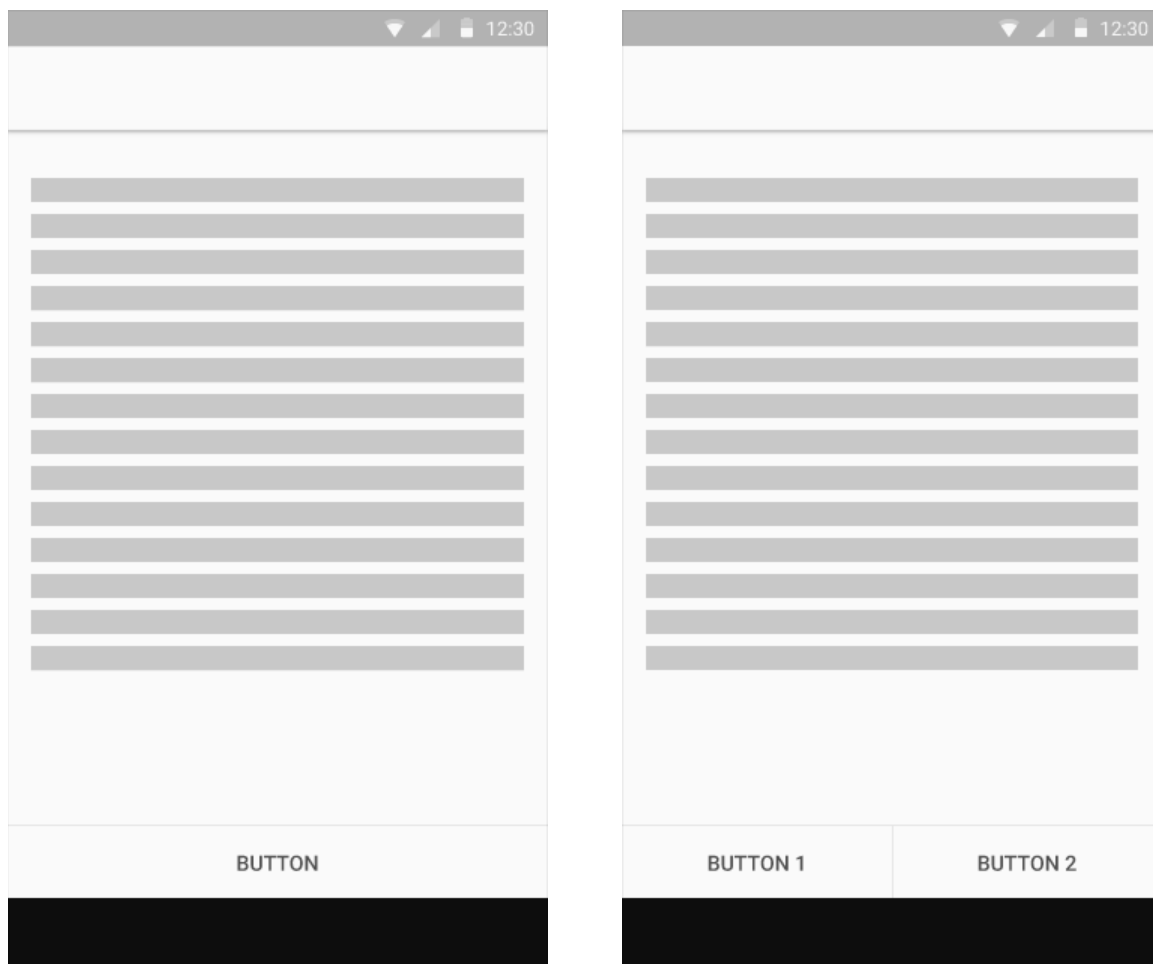
按钮内部

根据特定的布局来选择使用扁平按钮或者浮动按钮. 对于扁平按钮, 应该在内部四周留出足够的空间以使按钮清晰可见。



底部固定按钮

如果需要一个对用户持续可见的功能按钮, 应该首先考虑使用悬浮响应按钮。如果需要一个非主要, 但是能快速定位到的按钮, 则可以使用底部固定按钮。



不可在底部固定按钮的区域内使用浮动按钮。

Don't

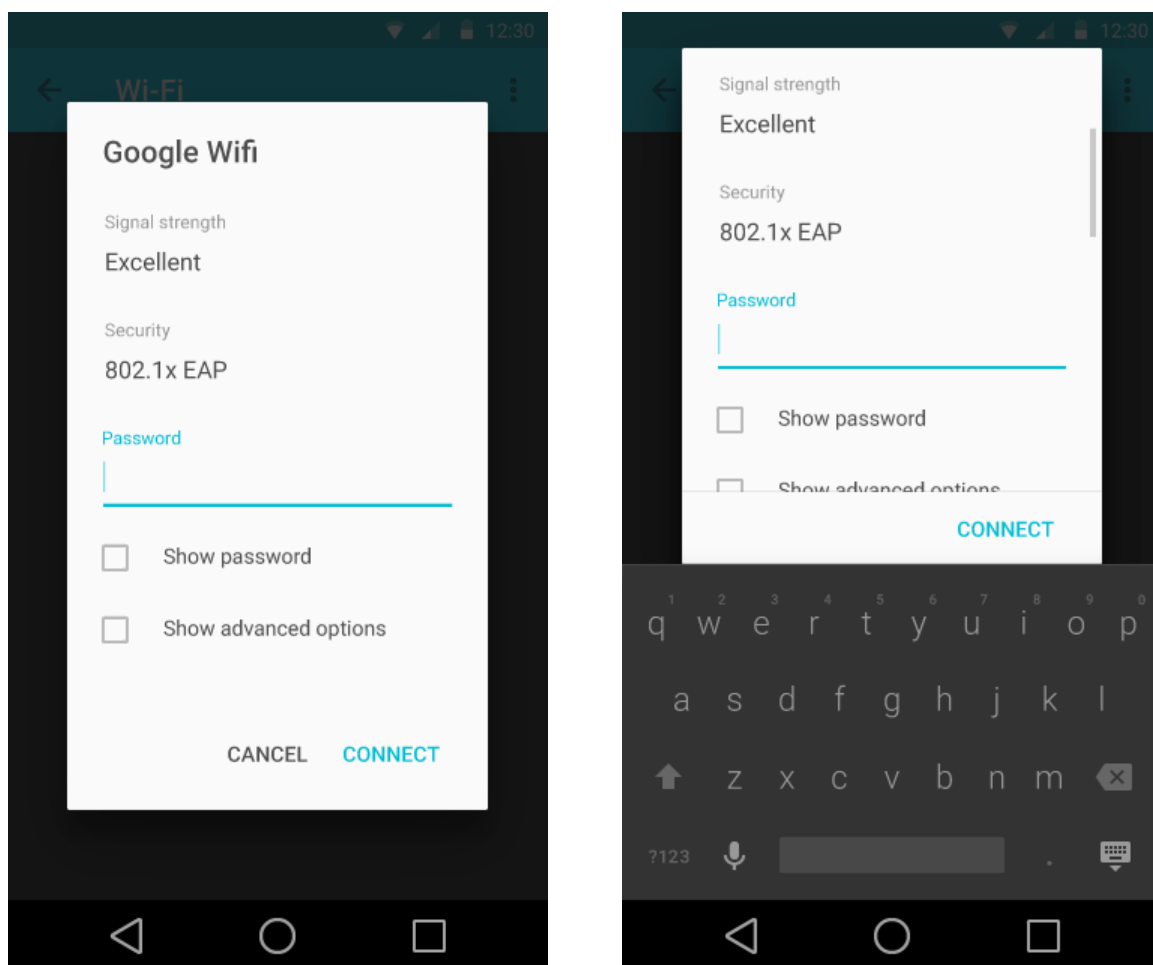
CONFIRM

Don't

SHARE

CONFIRM

底部固定按钮也可以用在内容可拉动的对话框中，前提是要加上divider

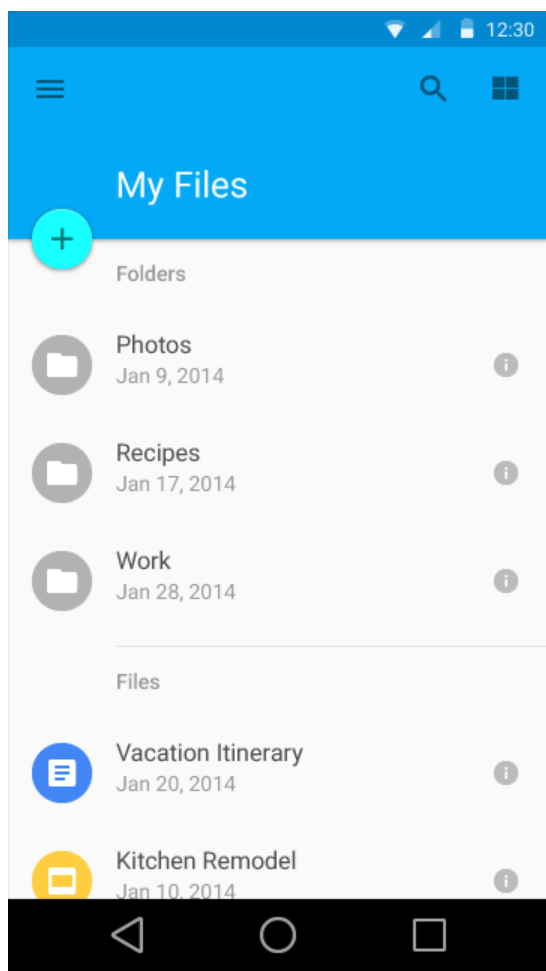
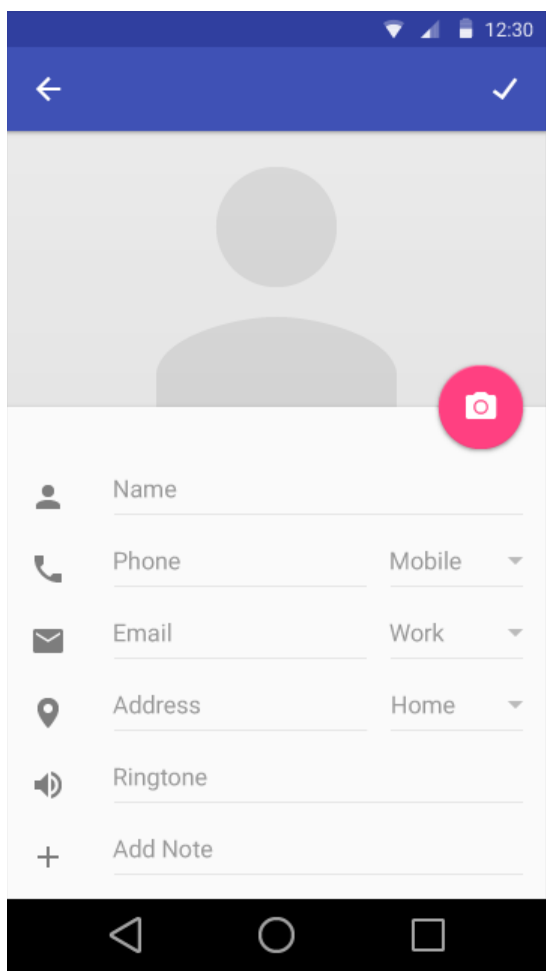


主按钮

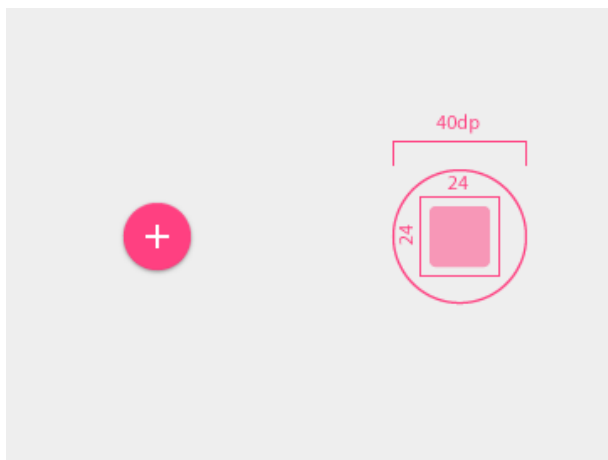
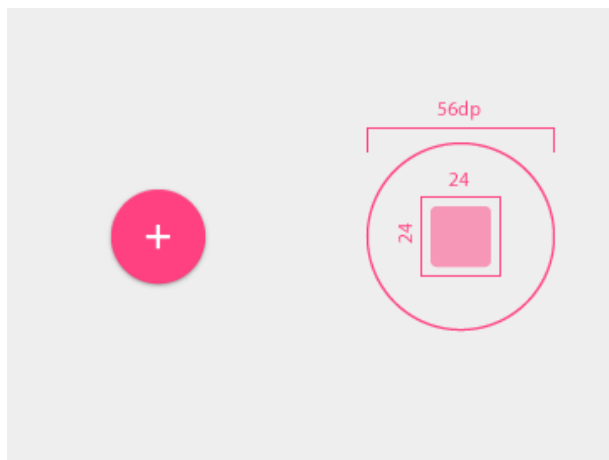
悬浮响应按钮

悬浮响应按钮是促进动作里的特殊类型。 是一个圆形的漂浮在界面之上的拥有一系列特殊动作的按钮，这些动作通常和变换、启动、以及它本身的转换锚点相关。

更多关于悬浮响应按钮的信息, 请参考 模式：促进的动作(Promoted Actions)

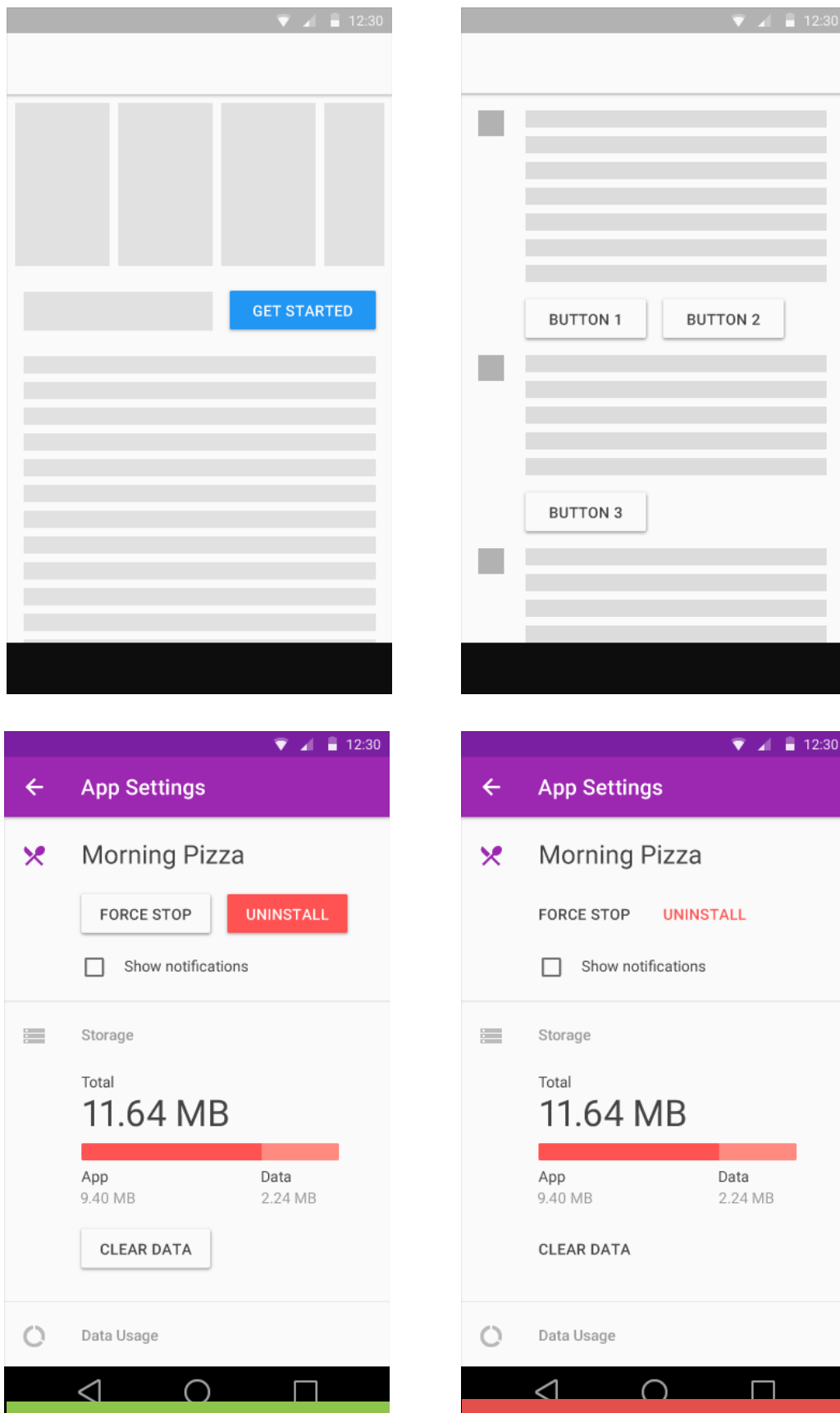


悬浮响应按钮有两种尺寸: 默认尺寸和迷你尺寸. 迷你尺寸仅仅用于配合屏幕上的其他元素制造视觉上的连续性。



浮动按钮

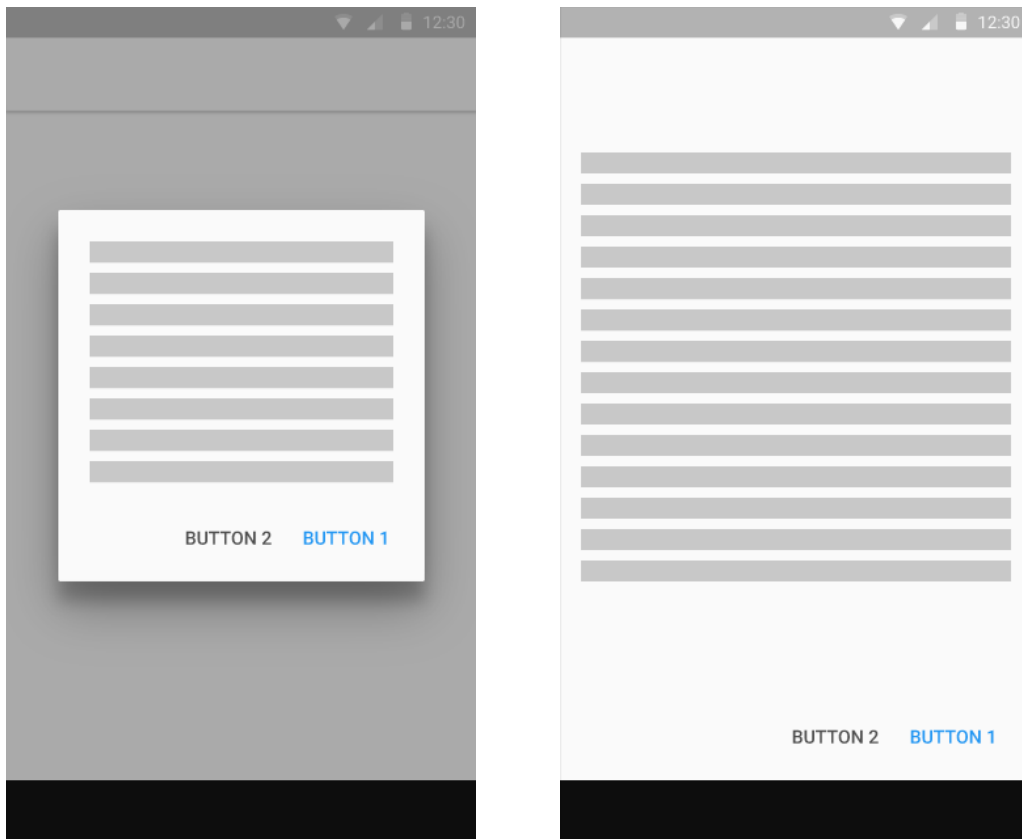
浮动按钮使按钮在比较拥挤的界面上更清晰可见。能给大多数扁平的布局带来层次感

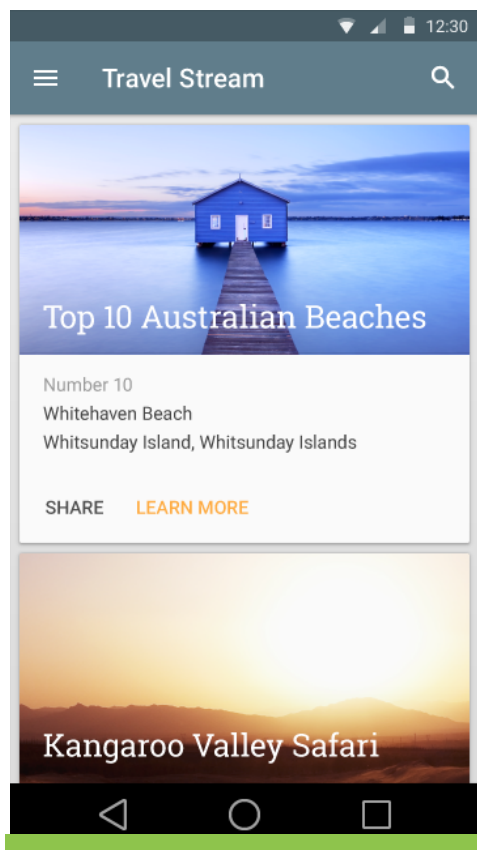


上图左侧正确，右图不正确，按钮不够明显。

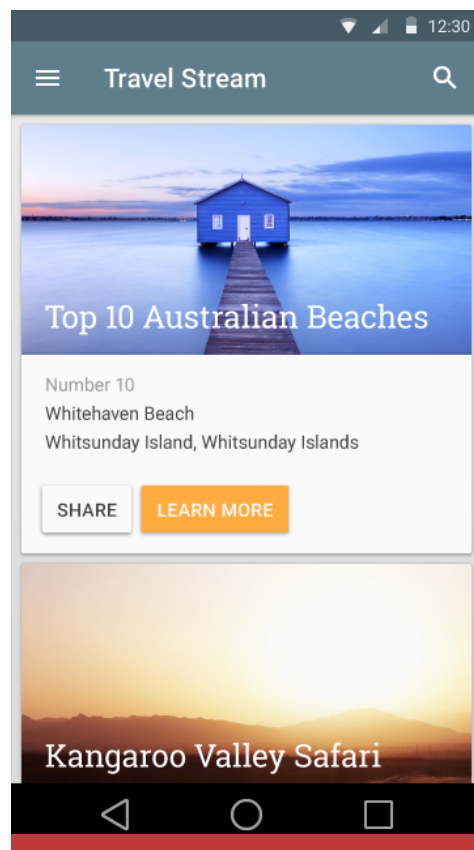
扁平按钮

扁平按钮一般用于对话框或者工具栏, 可避免页面上过多无意义的层叠。





扁平按钮的正确使用



不正确层次感太强了

扁平 and 浮动按钮状态

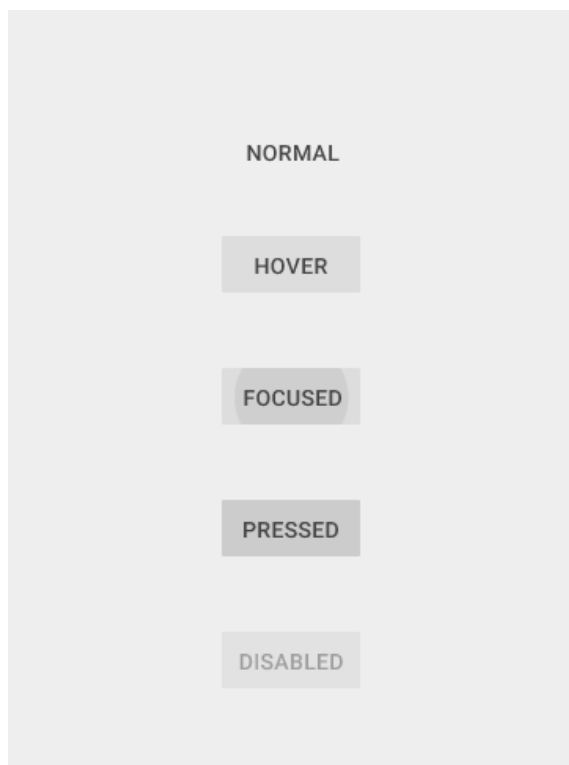
按钮状态模拟

浮动按钮看起来像一张放在页面上的纸片一点击后会浮起来并表现出色彩。

扁平按钮会一直保持和页面贴合的状态，点击后会填充颜色。

墨水效果会跟着焦点的改变从一个按钮转换到另一个按钮。聚焦状态的动画会表现出正常状态和点击状态间来回切换的过渡效果。

模拟按钮状态的时候, 可以使用图形轮换来表现动画。注意聚焦状态会一直处于动画的状态。(下面这些图并没有显示出真实的聚焦状态。)

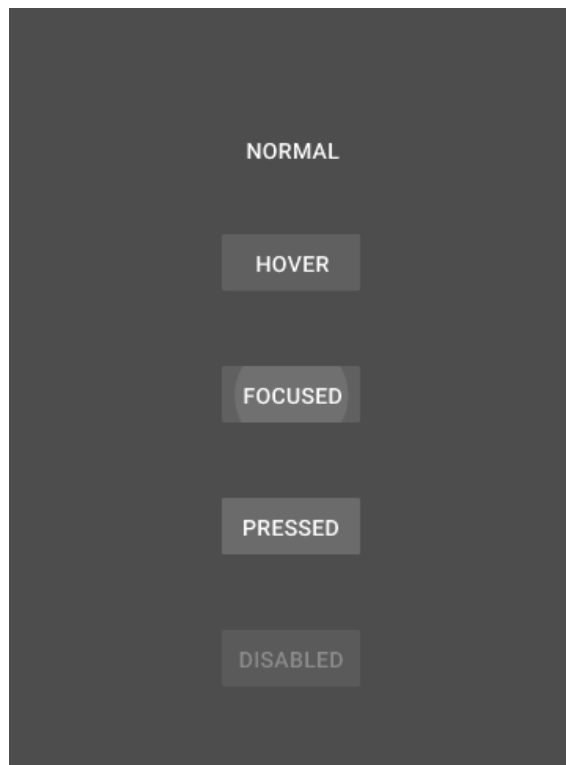


Flat Light/Light color

最小宽度: 88 dp, 高度: 36 dp

覆盖状态: 20% #999, 点击状态: 40%

#999, 不可用状态: 10% #999

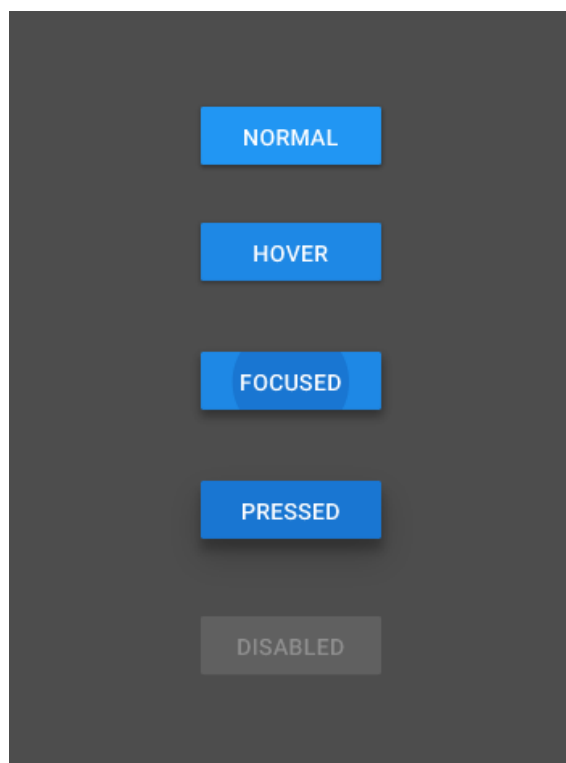
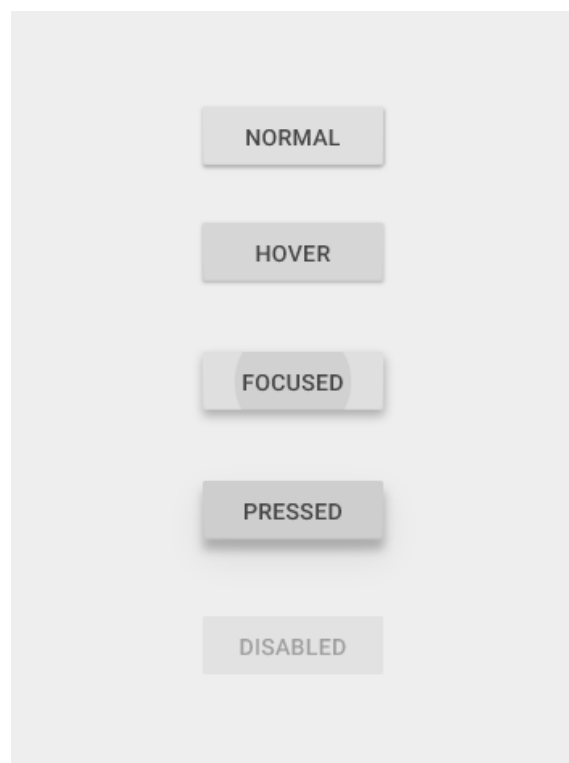


Flat Dark/Dark Color

最小宽度: 88 dp, 高度: 36 dp

覆盖状态: 15% #ccc, 点击状态: 25%

#ccc, 不可用状态: 10% #ccc



Raised Light/Light Color

最小宽度: 88 dp, 高度: 36 dp

Raised Dark/Dark Color

最小宽度: 88 dp, 高度: 36 dp

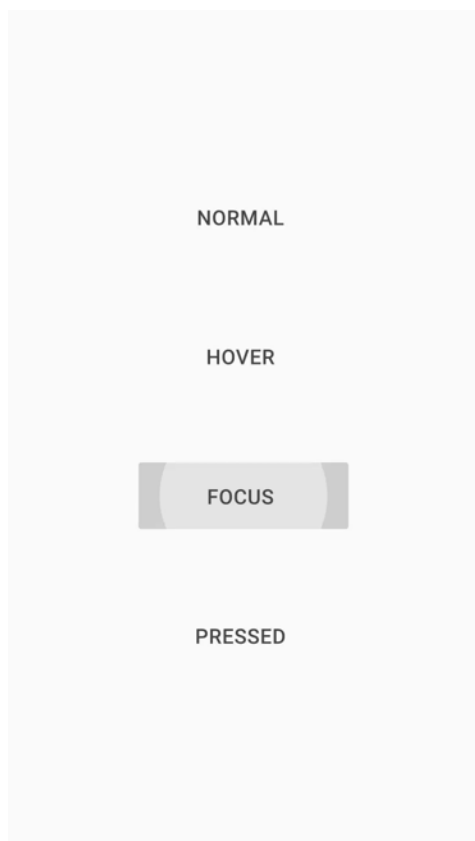
正常状态: Color 500, 覆盖状态: Color

600, 点击状态: Color 700,

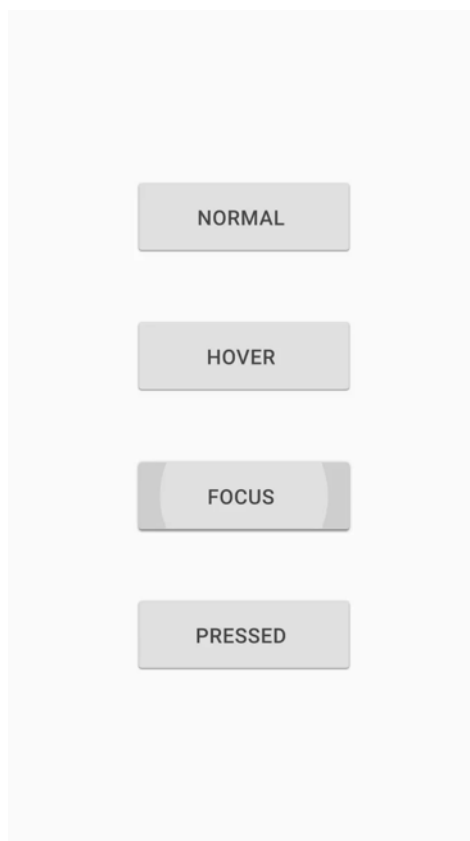
不可用状态: 10% #ccc

按钮动态效果

扁平按钮



浮动按钮

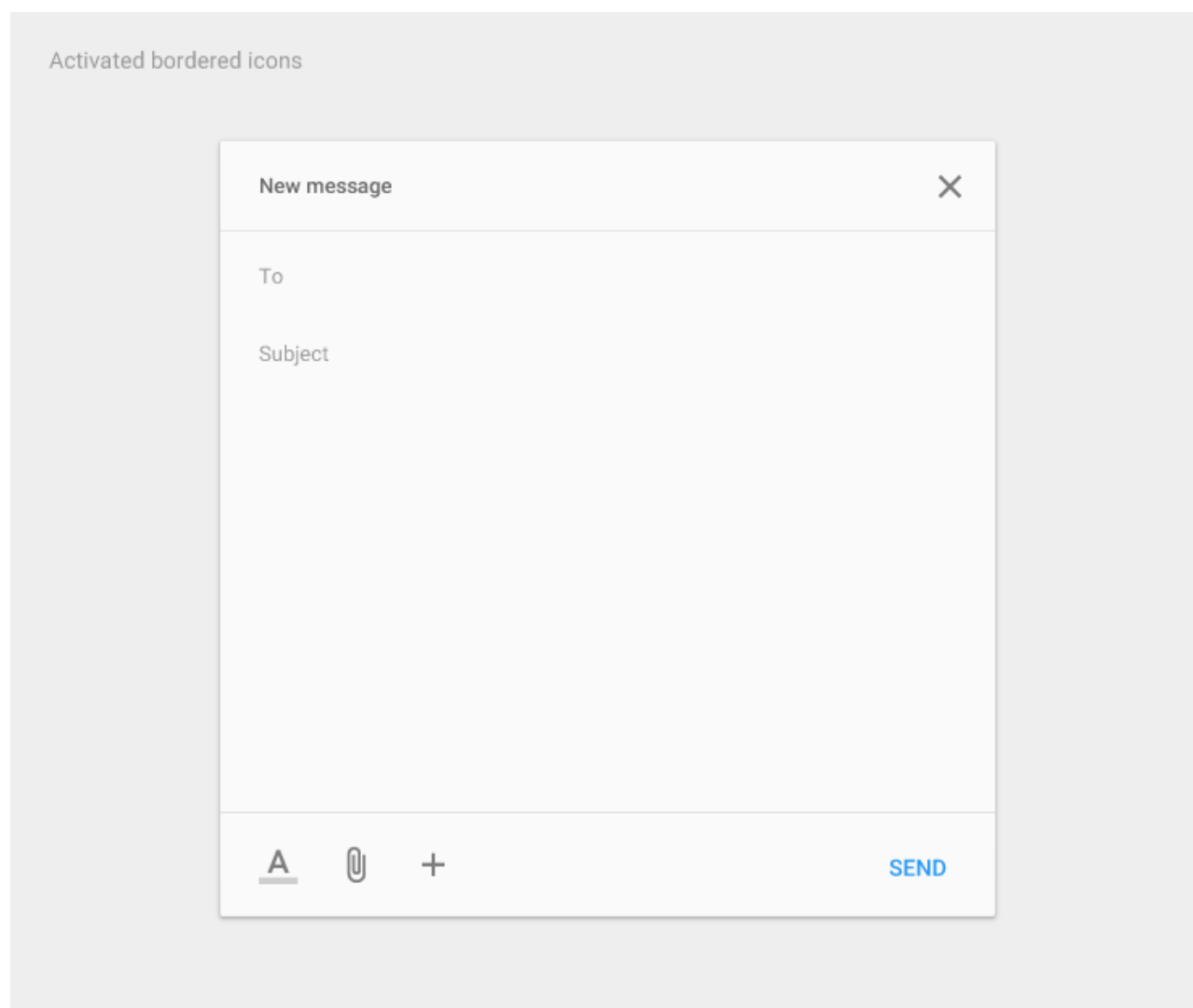
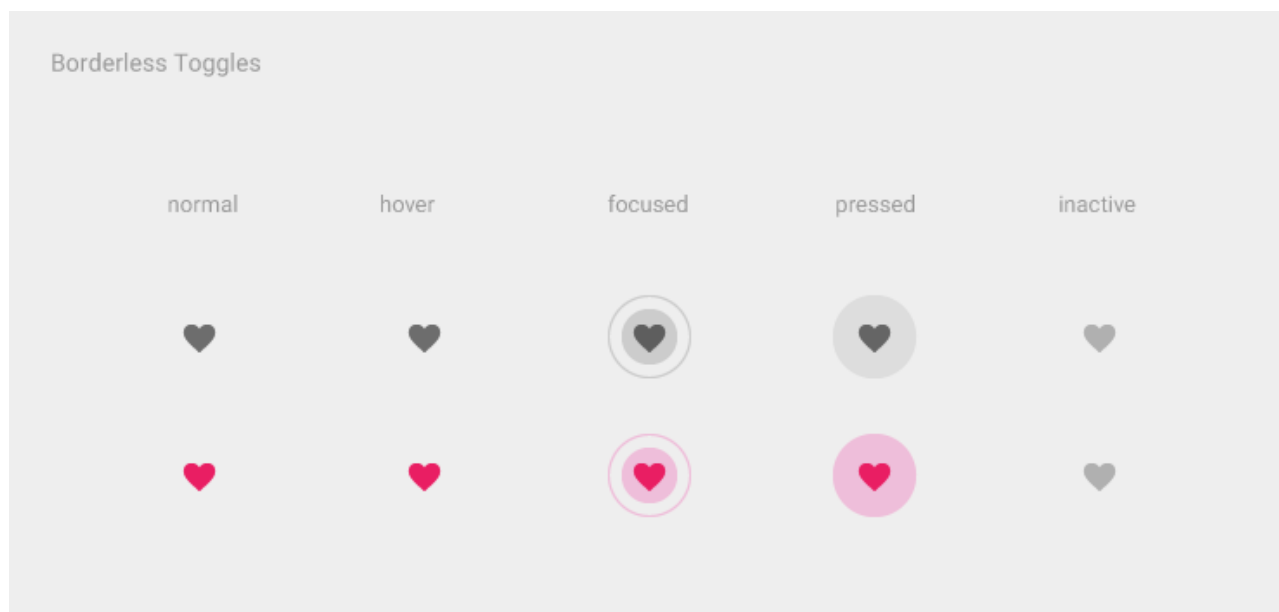


其他类型的按钮

图标开关

图标适合用在应用导航条或者工具条上，作为动作按钮或者开关。

图标开关可以在它的范围内呈现弹性或者非弹性的墨水扩散涟漪效果：



With focus area guidelines

New message

To

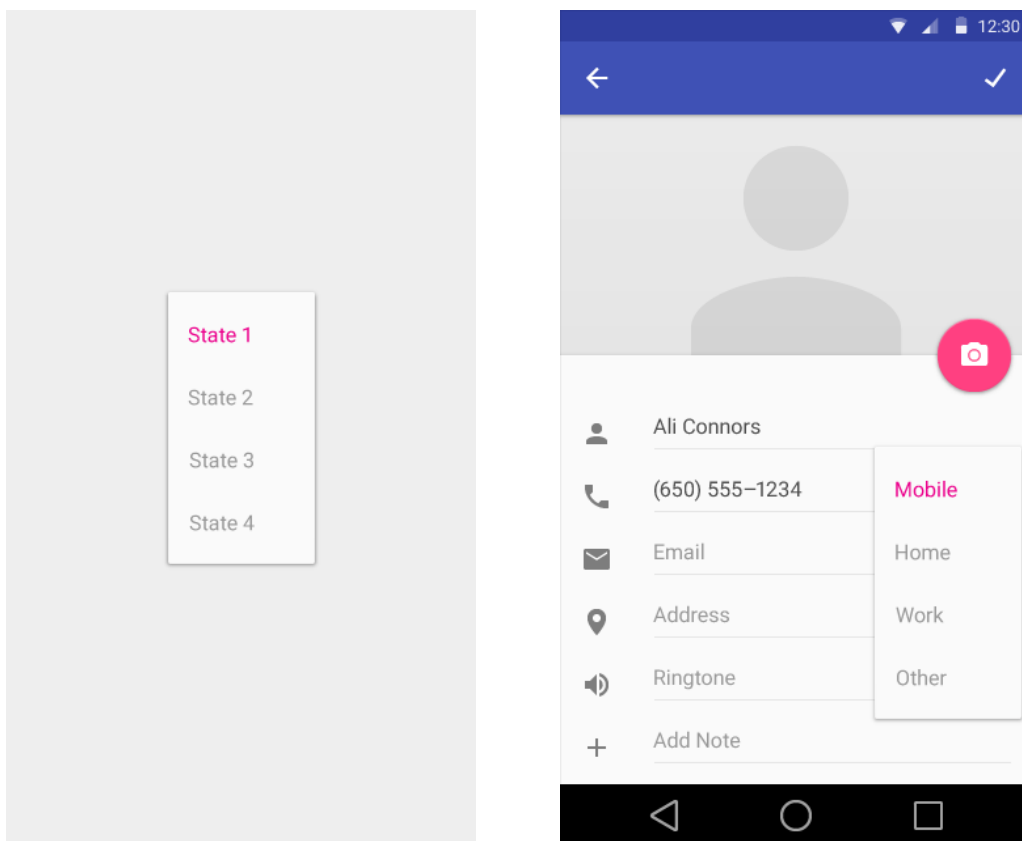
Subject

A [paperclip] + SEND

移动端下拉菜单按钮

下拉菜单按钮

下拉菜单按钮可以用来控制对象状态; 一般会有两个甚至更多的状态。按钮会显示当前状态以及一个向下的箭头—当按钮触发后, 一个包含所有状态的菜单会在按钮周围弹出 (通常都是在下方)。菜单中的状态通常会以字符、调色板、图标或者其他的形式呈现出来。点击任意一个状态将会改变按钮的状态显示。这展示的是一个常见的带有列表式菜单的下拉菜单按钮。



溢出下拉菜单按钮

这种类型的下拉菜单按钮不会显示当前状态，而是显示一个向下箭头或者一个默认菜单图标。点击后会弹出菜单。点击菜单中的任意一个选项将会引导到对应的设置页面。

分段式下拉菜单按钮

分段式下拉菜单按钮有两个区域: 当前状态和下拉箭头。点击当前状态会触发状态相应的动作。点击下拉箭头则会弹出所有状态菜单; 点击任意一个状态会改变当前的状态。

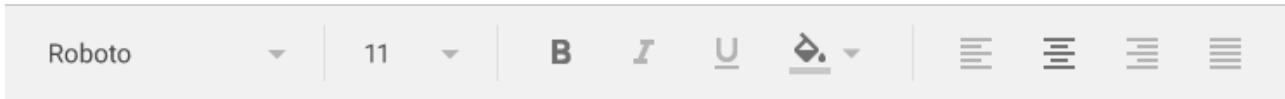
可编辑分段式下拉菜单按钮

可编辑分段式下拉菜单按钮的当前状态位置是可编辑的（例如用来选择文字大小的下拉菜单）。点击当前状态位置会触发相应的动作并且当前状态会变成可编辑。点击下拉箭头会显示所有状态。

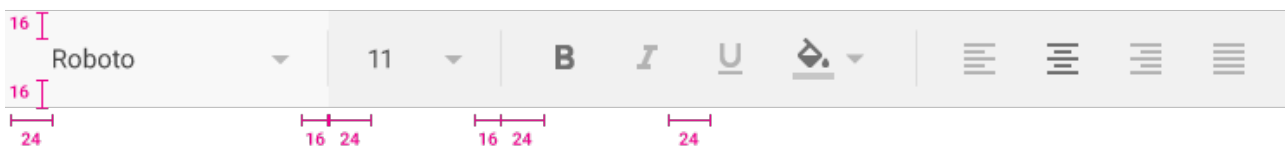
桌面下拉相关

桌面应用工具栏

Regular



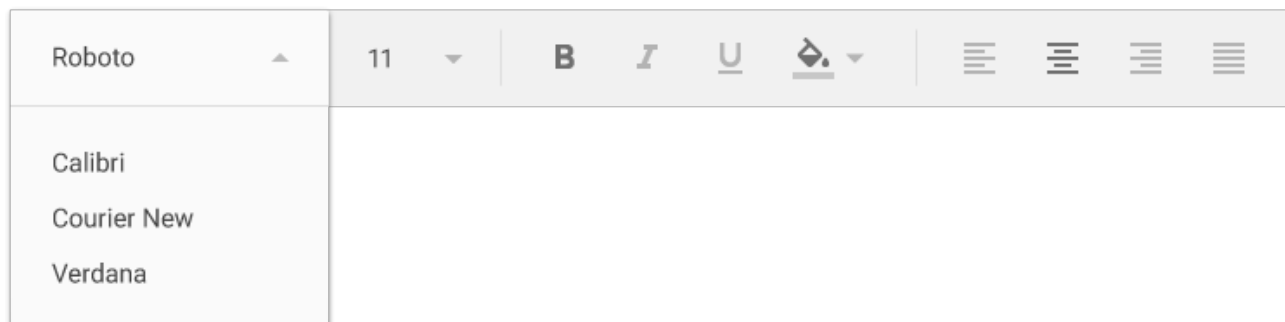
Hover



Hover



Open



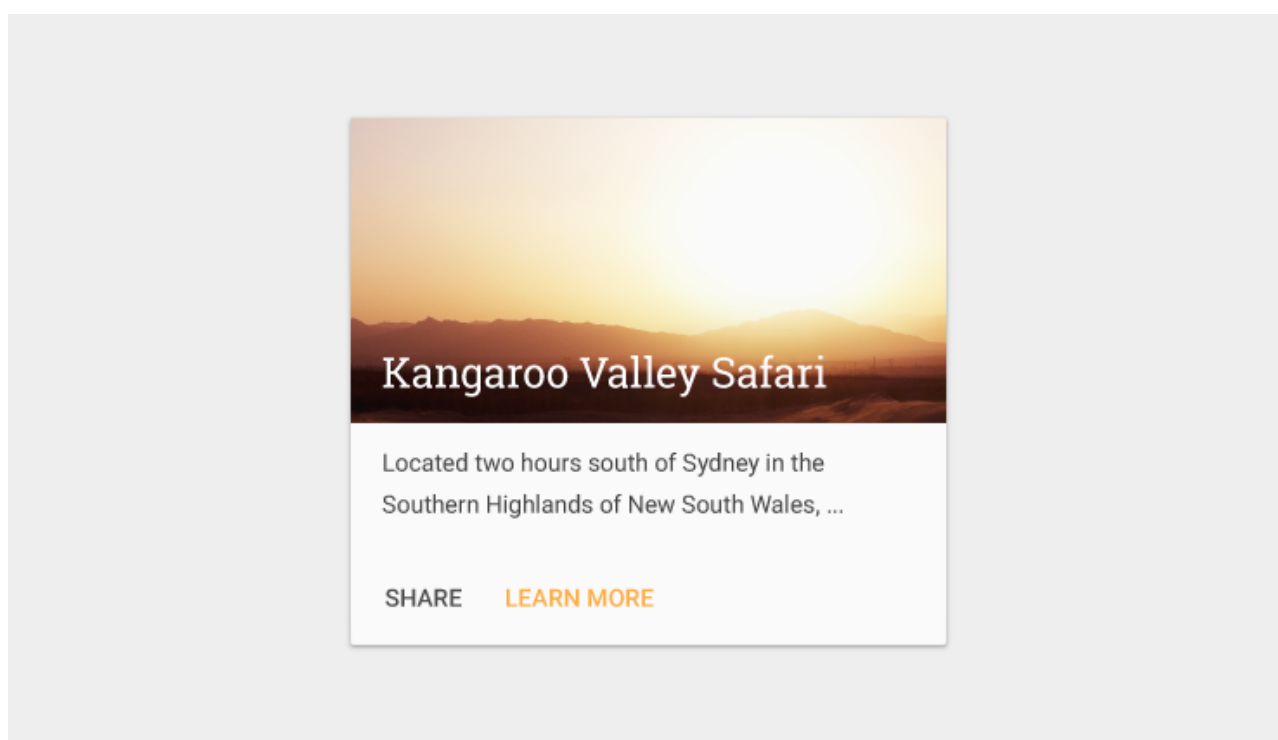
5.3 卡片

卡片是包含一组特定数据集的纸片，数据集含有各种相关信息，例如，关于单一主题的照片，文本，和链接。卡片通常是通往更详细复杂信息的入口。卡片有固定的宽度和可变的高度。最大高度限制于可适应平台上单一视图的内容，但如果需要它可以临时扩展（例如，显示评论栏）。卡片不会翻转以展示其背后的信息。

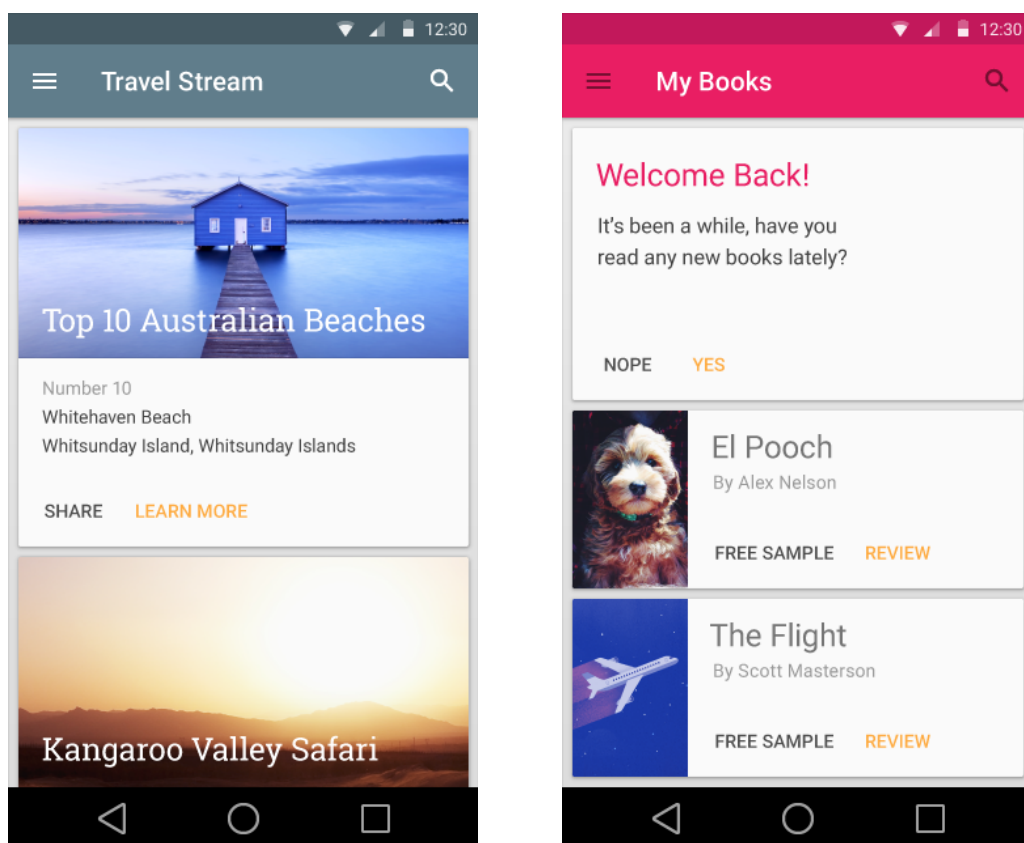
用途

卡片是用来显示由不同种类对象组成的内容的便捷途径。它们也适用于展示尺寸或操作相当不同的相似对象，像带有不同长度标题的照片。

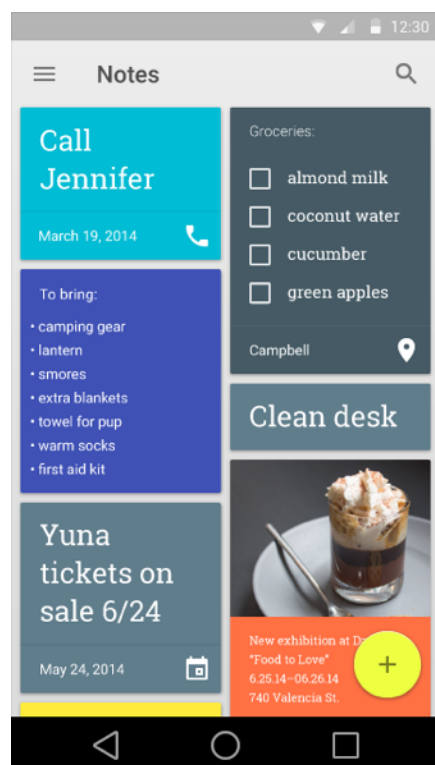
注意:即使外观相近，Now卡片是卡片的一个带有唯一表现和格式要求的独特子集。



卡片集是卡片的一个平面布局。



这些卡片每张包含一组特定数据集：带操作的确认表，带操作的笔记，带照片的笔记。



显示这些内容时使用卡片布局：

作为一个集合，由多种数据类型组成（例如，卡片集包含照片，电影，文本，图像）

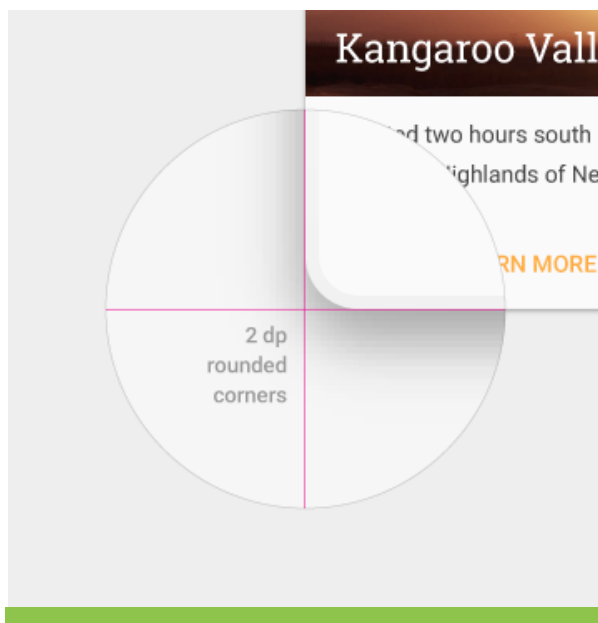
不要求直接比较（用户不直接与图像或字符串比较）

包含可变长度内容，例如评论

由富内容或互动操作组成，例如+1按钮，滑块，或评论

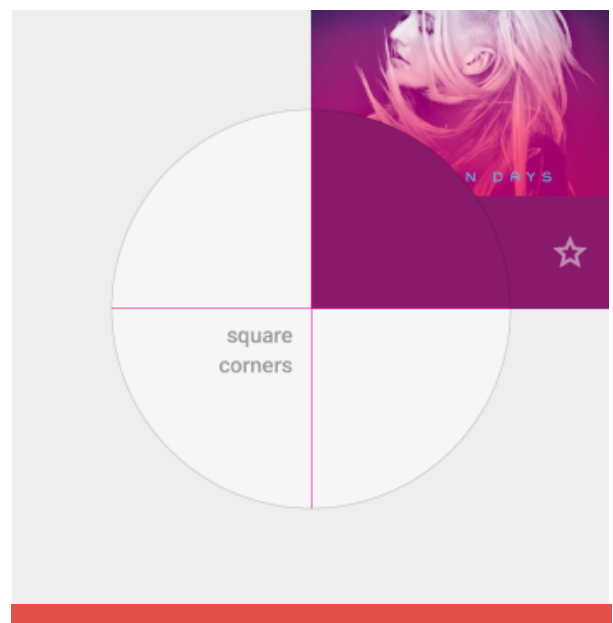
如果使用列表需要显示超过三行文本

如果使用网格列表需要显示更多文本来补充图像



正确

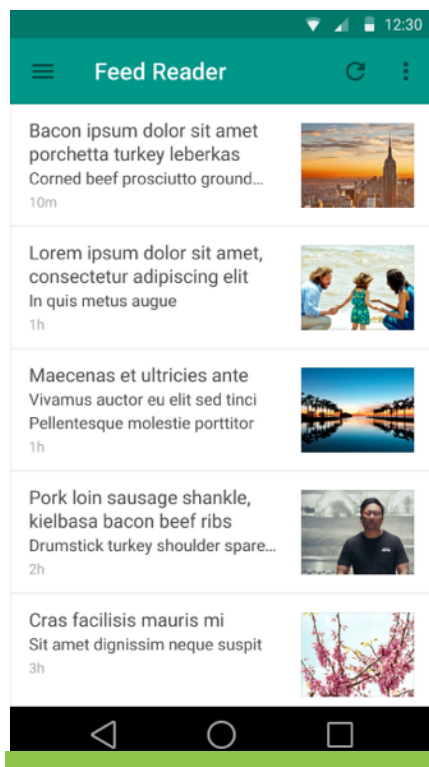
1. 卡片带圆角。
2. 卡片带多种操作。
3. 卡片可以忽略和重排



不正确

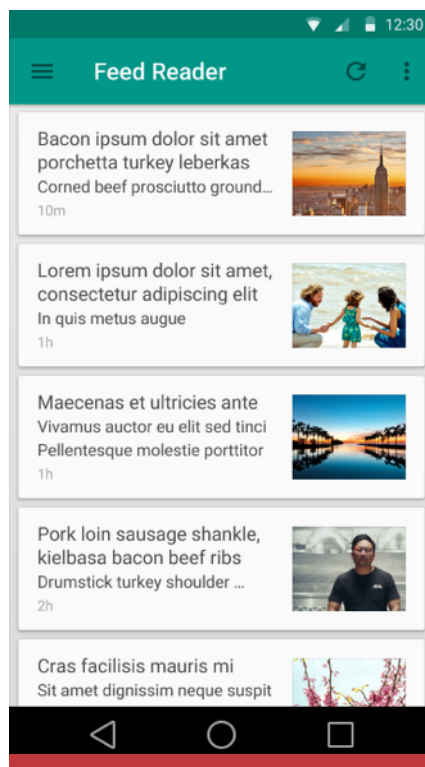
这是瓷砖，不是卡片。

1. 瓷砖带直角。
2. 瓷砖少于两种操作



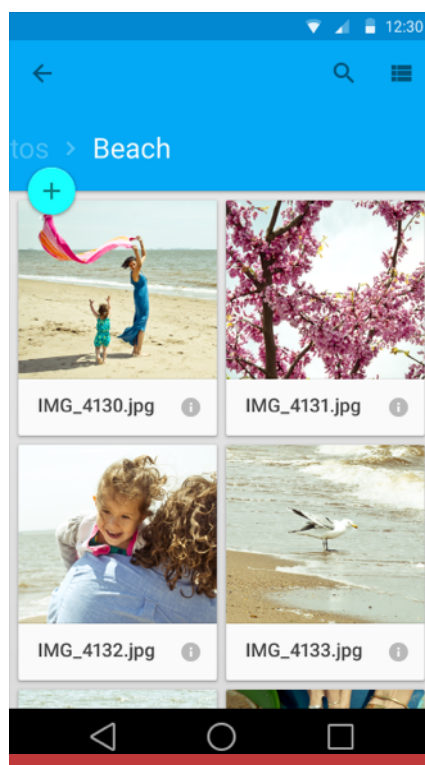
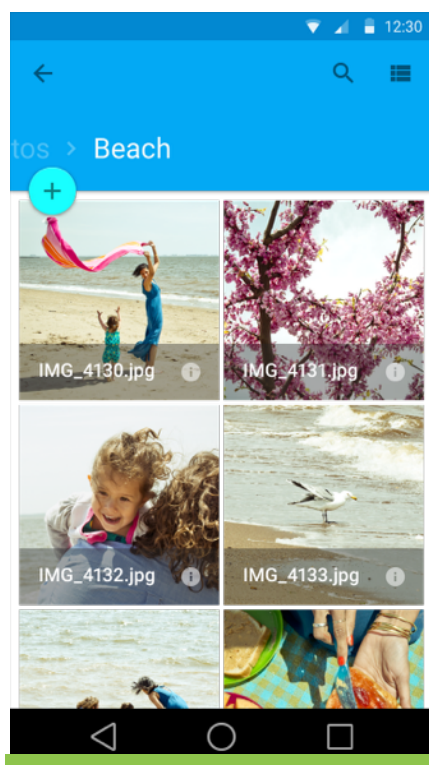
正确

可快速扫描的列表，用来代替卡片，是表现没有许多操作的同类内容的合适方法



不正确

这里卡片的使用分散了用户注意力，不能快速扫描。这些列表项也不能忽略，所以把它们放在不同的卡片上是难以理解的



正确

网格布局是表现图片库的干净轻量
的方法

不正确

卡片在图片库中没有必要

卡片布局准则

字体设计

正文：14 sp 或 16 sp

标题：24 sp 或更大

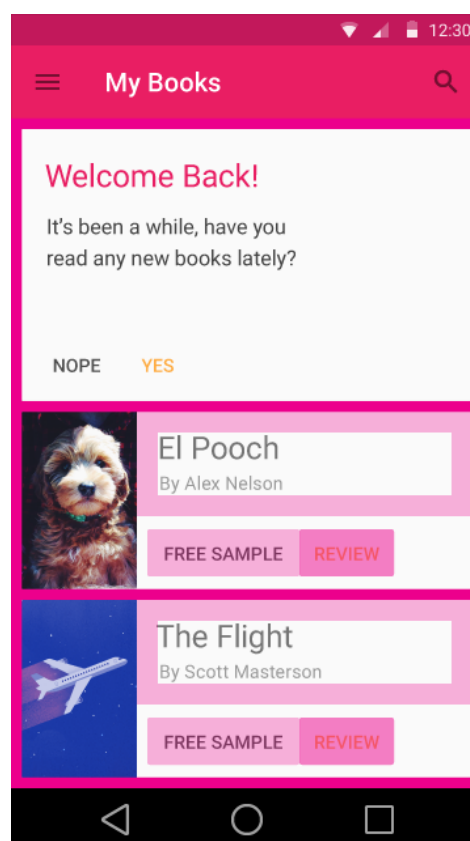
扁平按钮：Roboto Medium, 14 sp, 10 sp 字间距

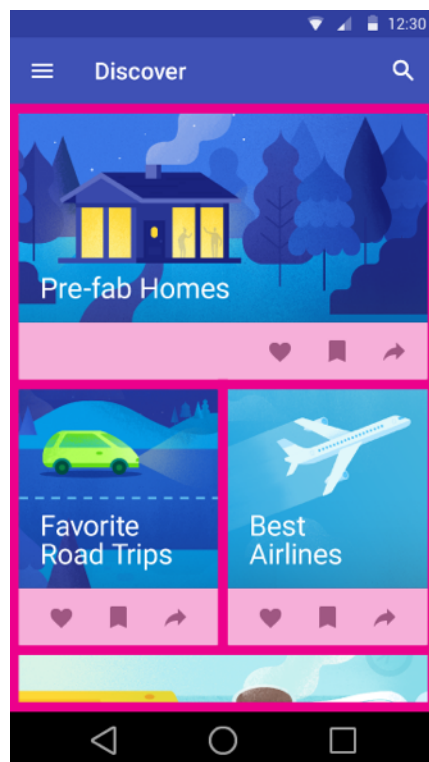
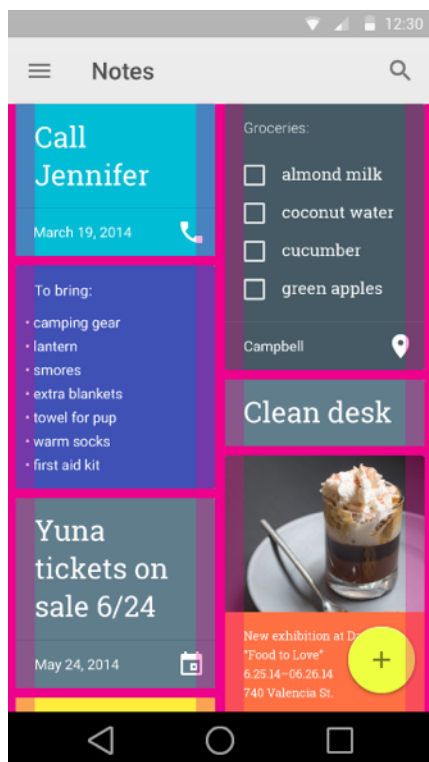
移动设备上的卡片间距

屏幕边界与卡片间留白：8 dp

卡片间留白：8 dp

内容留白16 dp

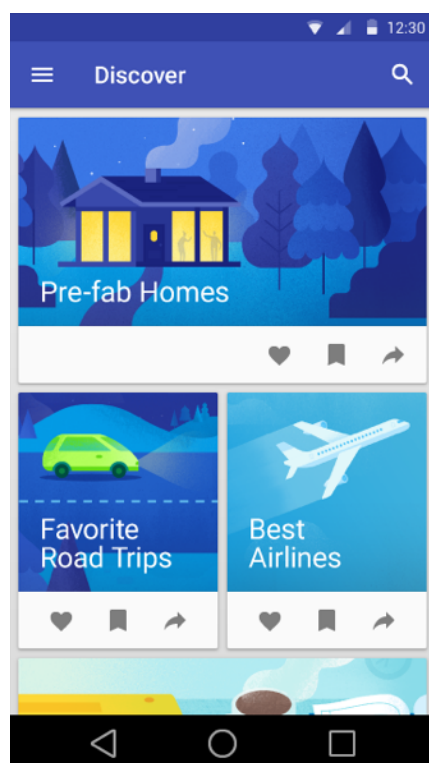
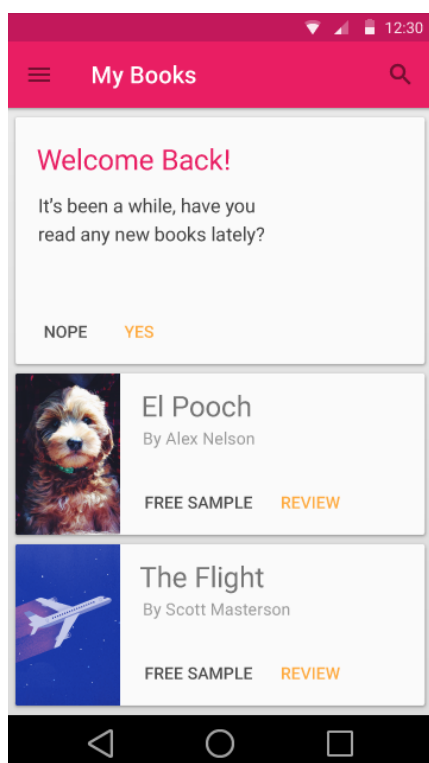


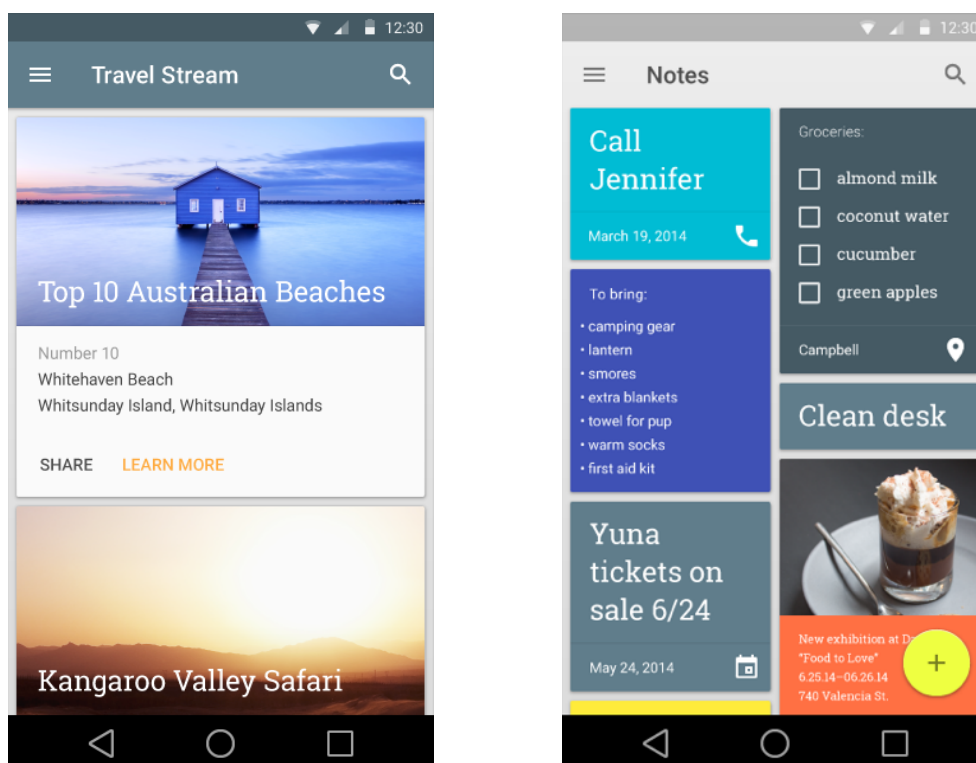


内容

卡片内容类型和数量可以很大程度上根据传递的内容变化。卡片提供上下文及通往更复杂信息与视图的入口；确保不要滥用带有无用信息或操作的卡片。

放置主要内容在卡片顶部。使用层级结构来引导用户注意到卡片上最重要的信息





操作

卡片中的主要操作一般就是点击卡片本身。

根据卡片中的内容类型和预期的结果有一些补充的操作，比如说在一组卡片中相同位置操作可以播放电影或打开书籍。

补充操作

在卡片中补充操作是使用的图标，文本，和UI控件来显示，通常放在牌的底部。

放置在主要内容中的行内UI控制可以调整主要内容的外观，例如，用滑块来选择日期，星级评分系统，或者分段的按钮来选择日期范围。

补充的操作限制在两个动作以内,除了一个弹出菜单。

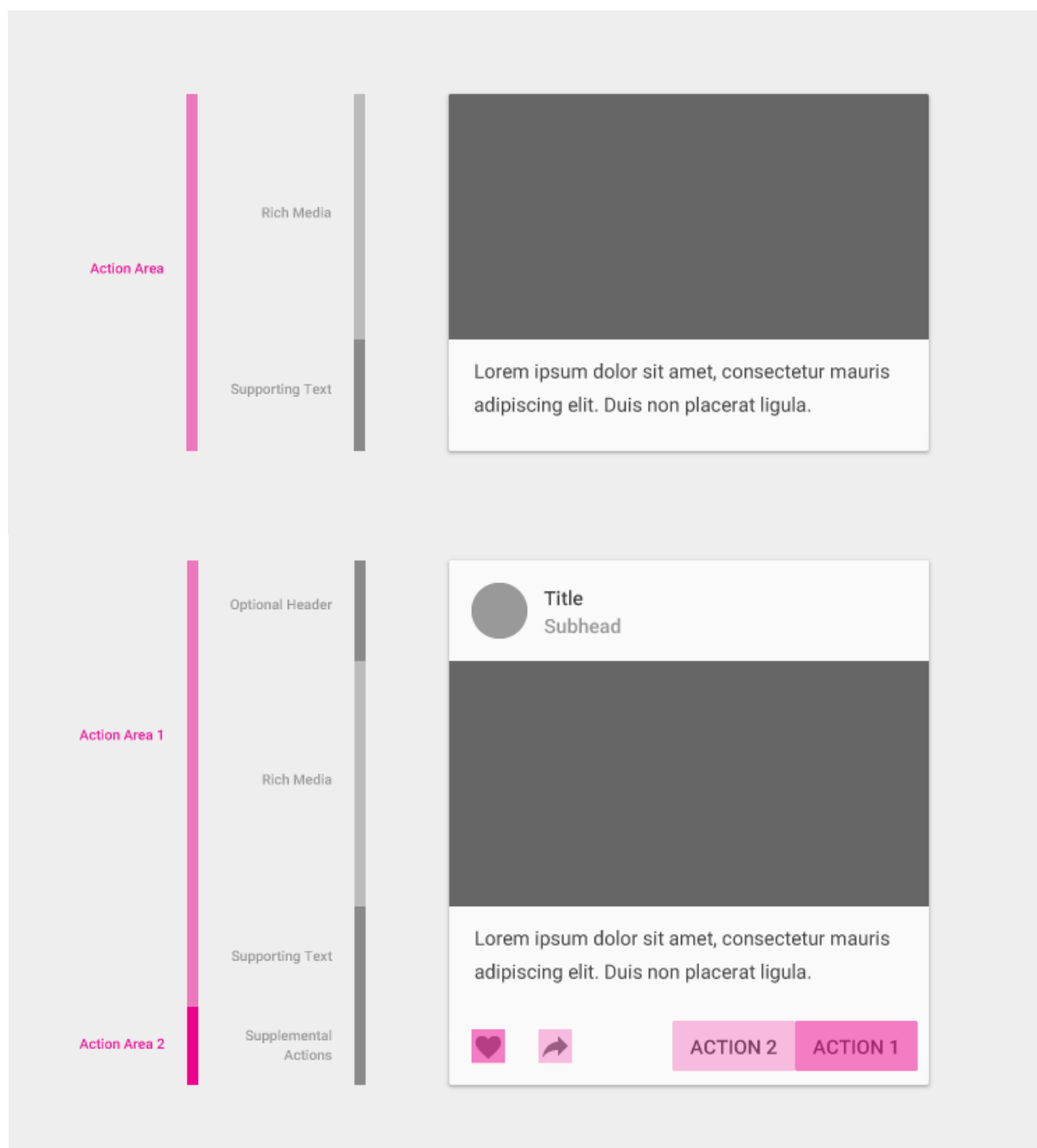
弹出菜单

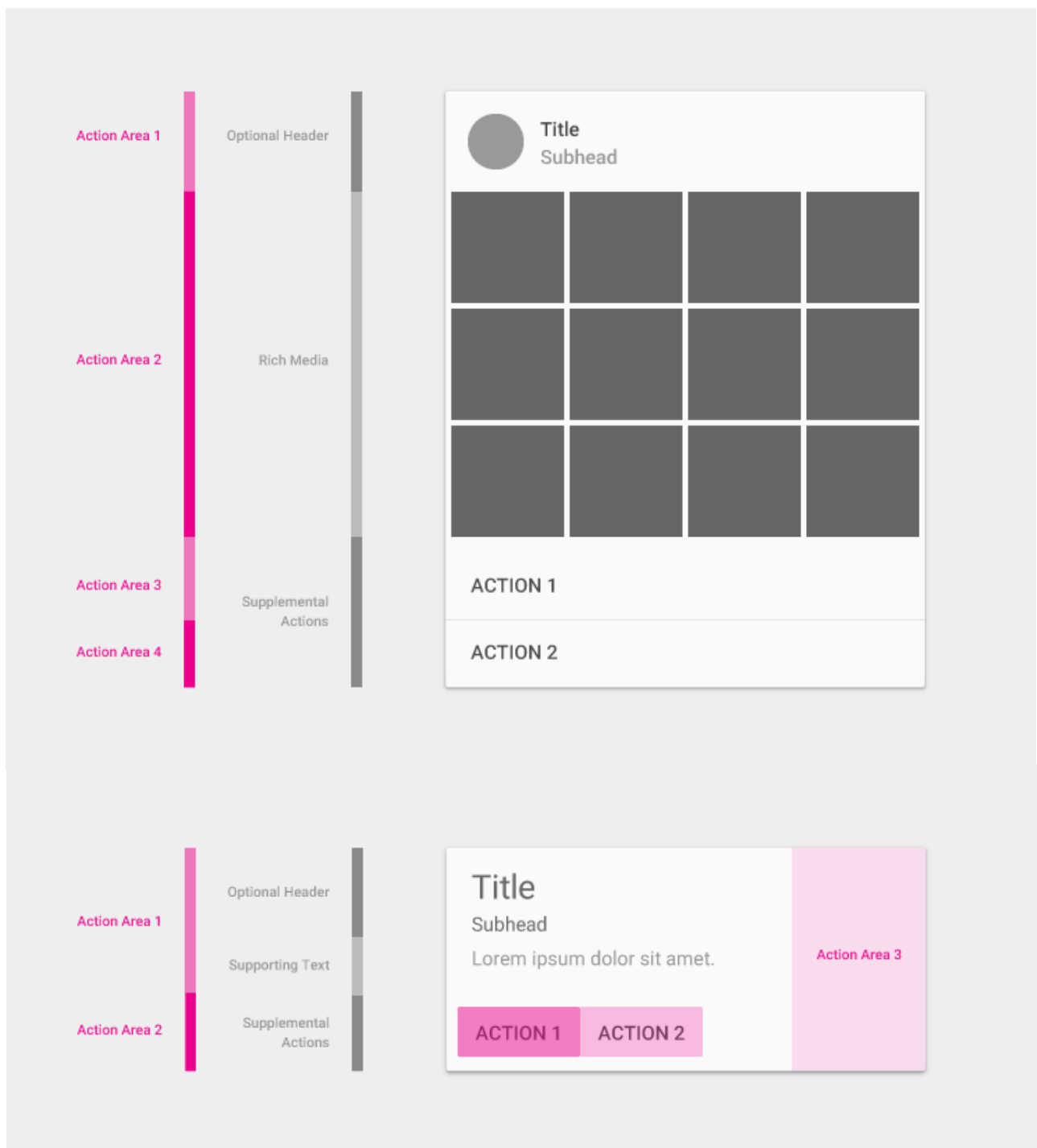
弹出菜单（可选）通常放置在卡片的右上角，但它也可以放置在右下方，如果这样安排改善内容布局和易读性。注意不要滥用带过多操作的弹出菜单。

注意事项

强烈不推荐文本内容的行内链接。

尽管卡片可以提供多种操作，UI控制，和弹出菜单，谨慎使用并且记得卡片是通往更复杂详细信息的入口。





表现

手势

支持单张卡片基准上的滑动手势。卡片手势表现应该始终在卡片组中实现。

按住并拖动手势可行。然而，考虑对用户能够在集合中排序卡片是否重要，或者如果按要求筛选/排序内容可以提供更好的体验。

卡片集筛选，排序，和重组

卡片集可以按要求排序或按日期，文件大小，字母表顺序，或其他参数筛选。集合中的第一项定位于集合的左上角，其余的从左至右从上至下延续。

滚动

卡片集只会竖直滚动。超过最大卡片高度的卡片内容将被截断且不可滚动。

带截断内容的卡片可以扩展，这样卡片高度可以超过视图的最大值。这种情况下，卡片将与卡片集一起滚动。

卡片焦点

对于取决于连续焦点遍历用于导航的界面（方向键，键盘），单张卡片应该仅有基本操作或打开一个带基本和可用追加操作的新视图。

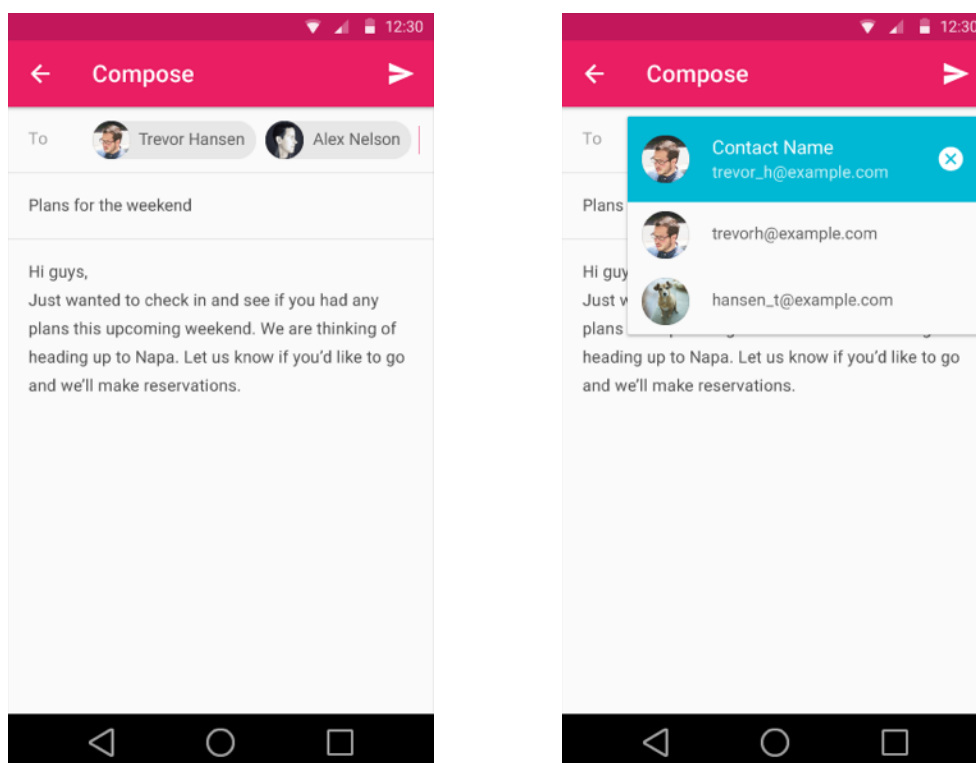
5.4 纸片(Chips)

纸片视图是一种小块的用来呈现复杂实体的块，比如说日历的事件或联系人。它可以包含一张图片，一个短字符串(必要时可能被截取的字符串)，或者是其它的一些与实体对象有关的简洁的信息。纸片可以非常方便的通过托拽来操作。通过按压动作可以触发悬浮卡片(或者是全屏视图)中的纸片对应实体的视图，或者是弹出与纸片实体相关的操作菜单。

联系人纸片

联系人的纸片视图用于呈现联系人的信息。当用户在输入框(收件人一栏)中输入一个联系人的名字时，联系人纸片视图就会被触发，用于展示联系人的地址以供用户进行选择。而且联系人的纸片可以被直接添加到收件人一栏中去。

联系人的纸片视图主要用于帮助用户高效的选择正确的收件人。



关闭状态的联系人纸片视图

联系人名字使用的字体为：Roboto，常规，14sp

当点击关闭状态下的联系人纸片视图，它就会展开并且显示出联系人的地址

打开状态的联系人纸片视图

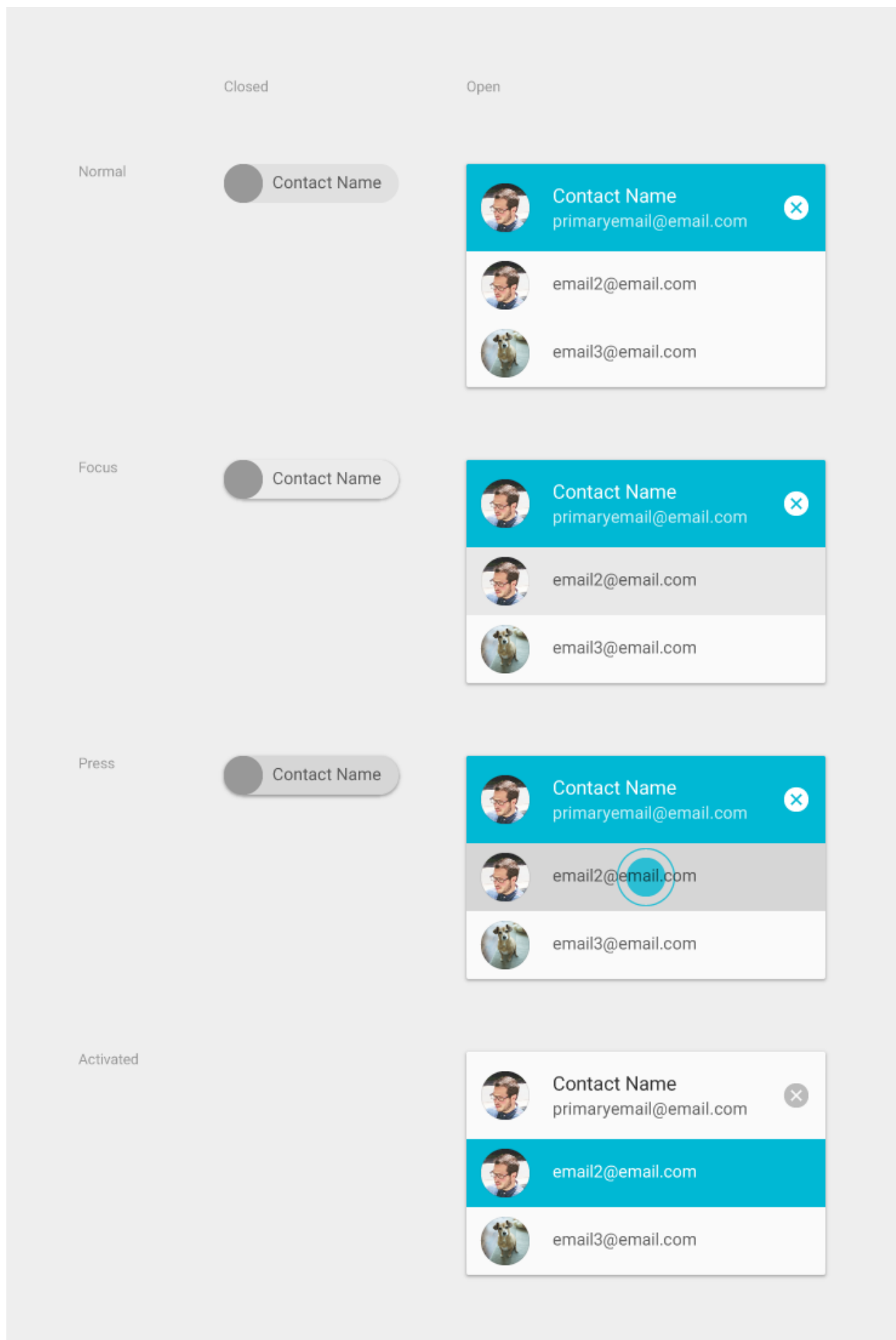
默认状态下，最顶的联系方式被激活并选中

联系人名字的字体为：Roboto，常规，16sp

地址文本的字体为：Roboto，常规，14sp

当用户选择后，纸片视图被关闭





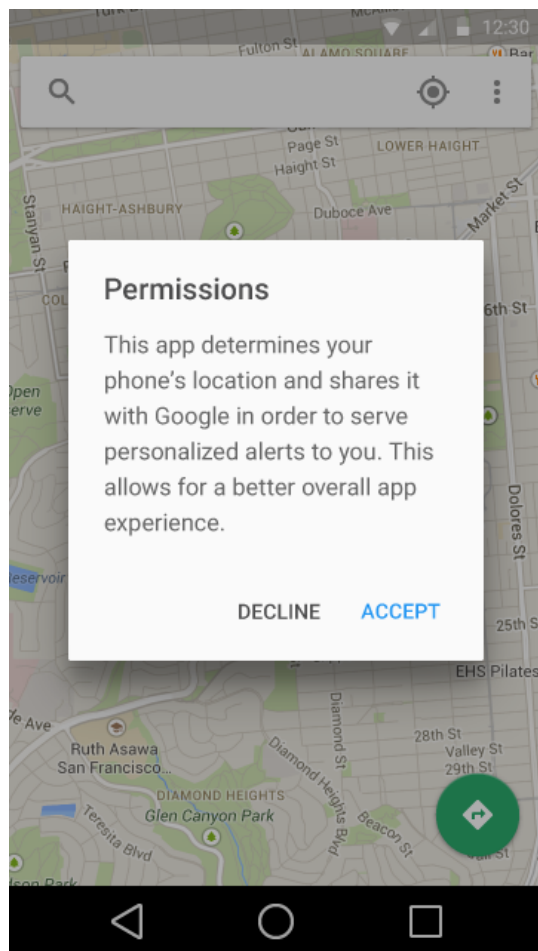
5.5 提示框(Dialogs)

提示框用于提示用户作一些决定，或者是完成某个任务时需要的一些其它额外的信息。提示框可以用一种 取消/确定 的简单应答模式，也可以是自定义布局的复杂模式，比如说一些文本设置或者是文本输入。

用途

提示框最典型的应用场景是提示用户去做一个些被安排好的决定，而这些决定可能是当前任务的一部分或者是前置条件。提示框可以用于告知用户具体的问题以便他们作用重要的决定，或者是用于解释。接下来的动作的重要性及后果。

一些复杂的操作，尤其是每个决策都需要相关解释说明的情况下是不适合使用提示框形式的

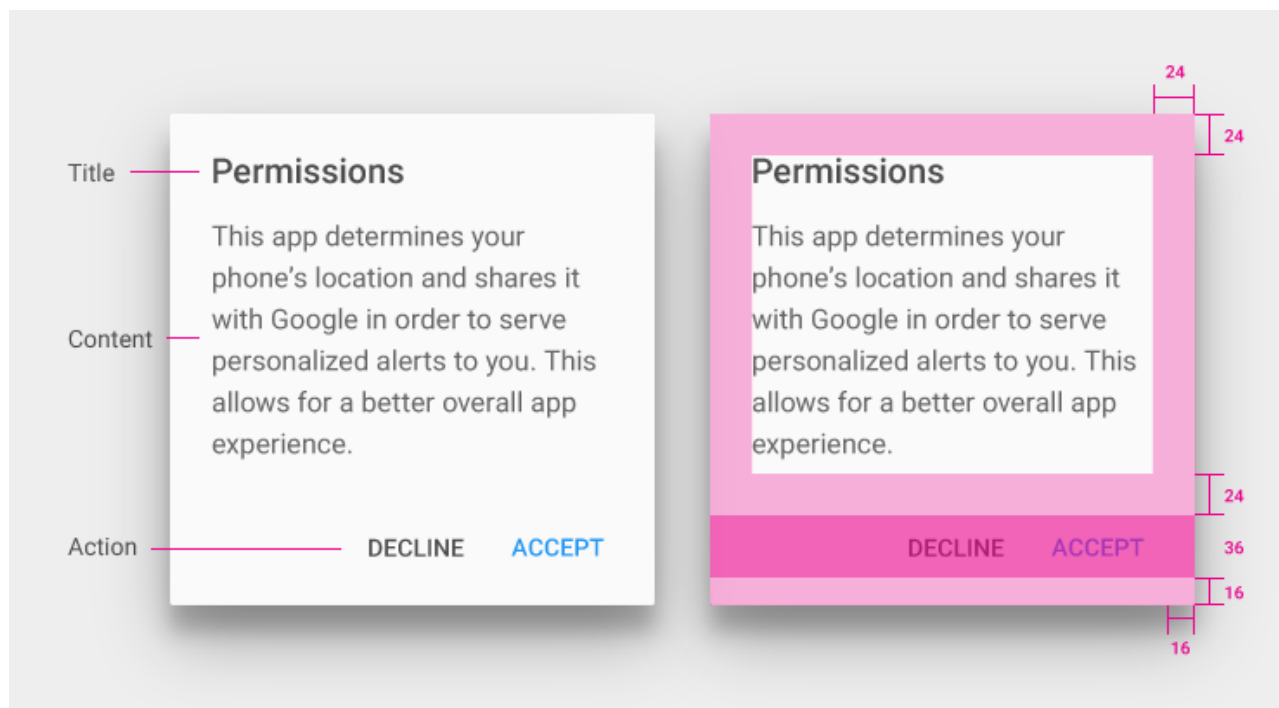


提示框包含了一个标题(可选)，内容，事件。

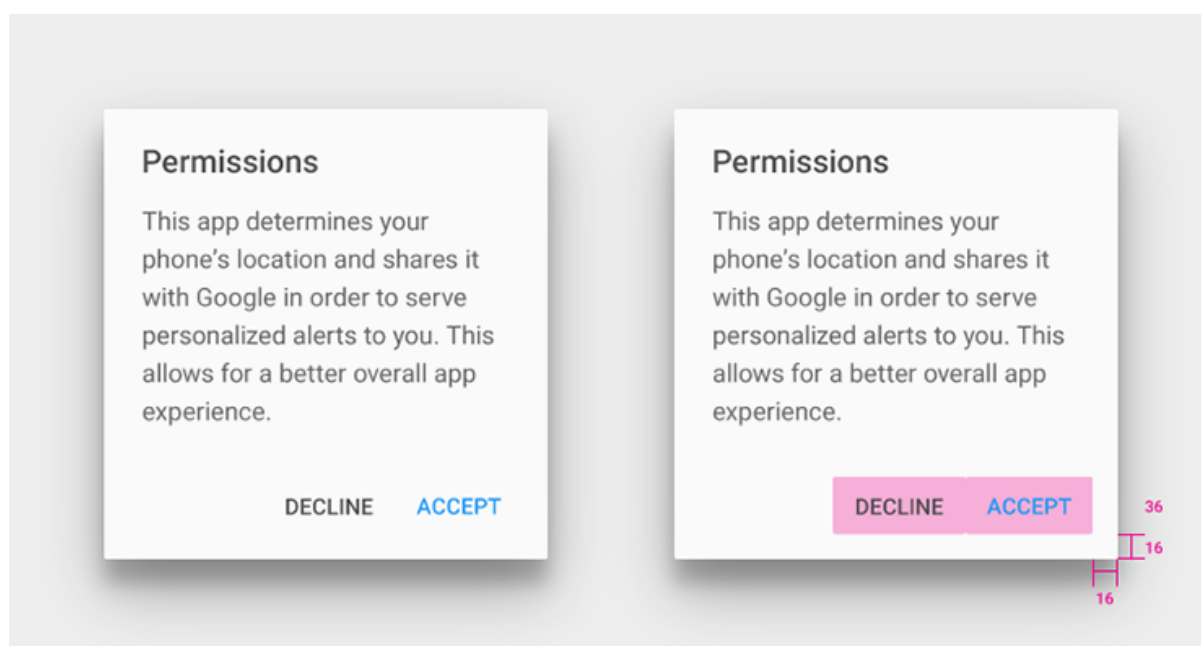
标题：主要是用于简单描述下选择类型。它是可选的，要需要的时候赋值即可。

内容：主要是描述要作出一个什么样的决定。

事件：主要是允许用户通过确认一个具体操作来继续下一步活动



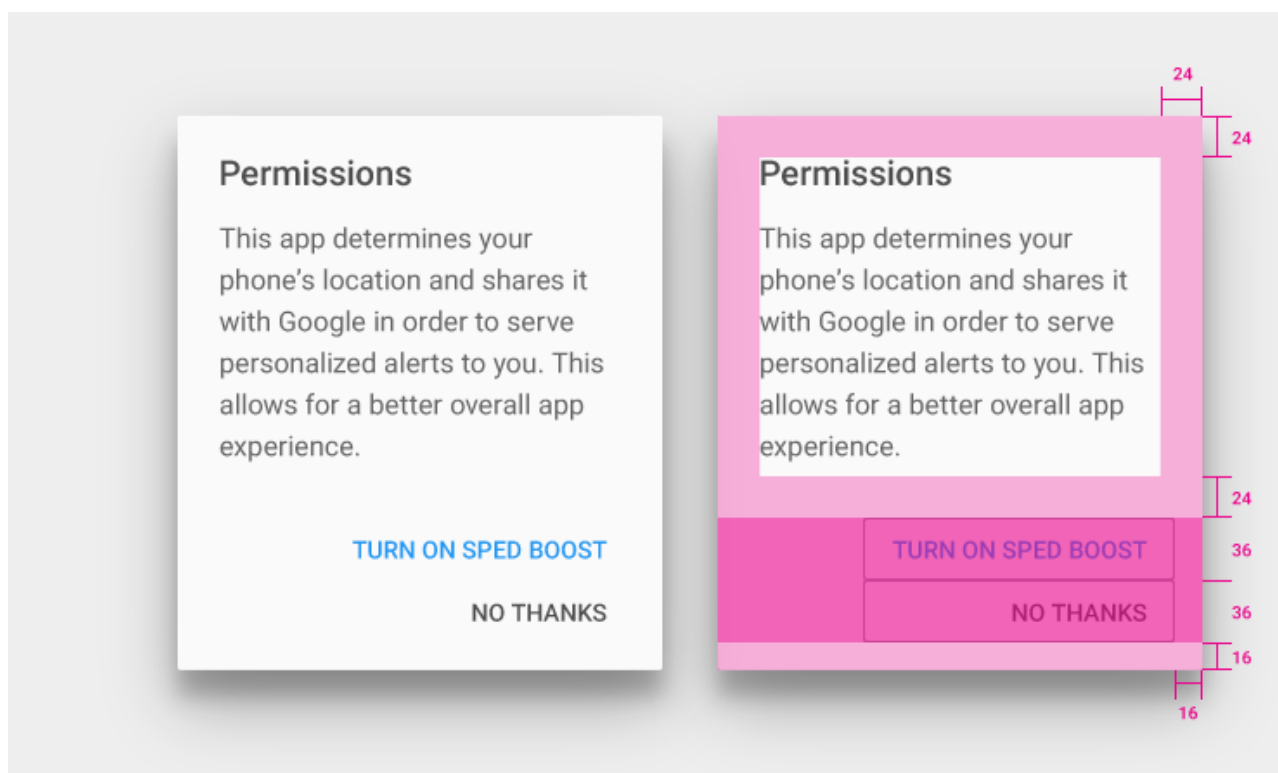
按钮的宽度及边框示例





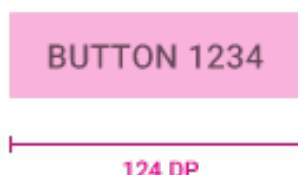
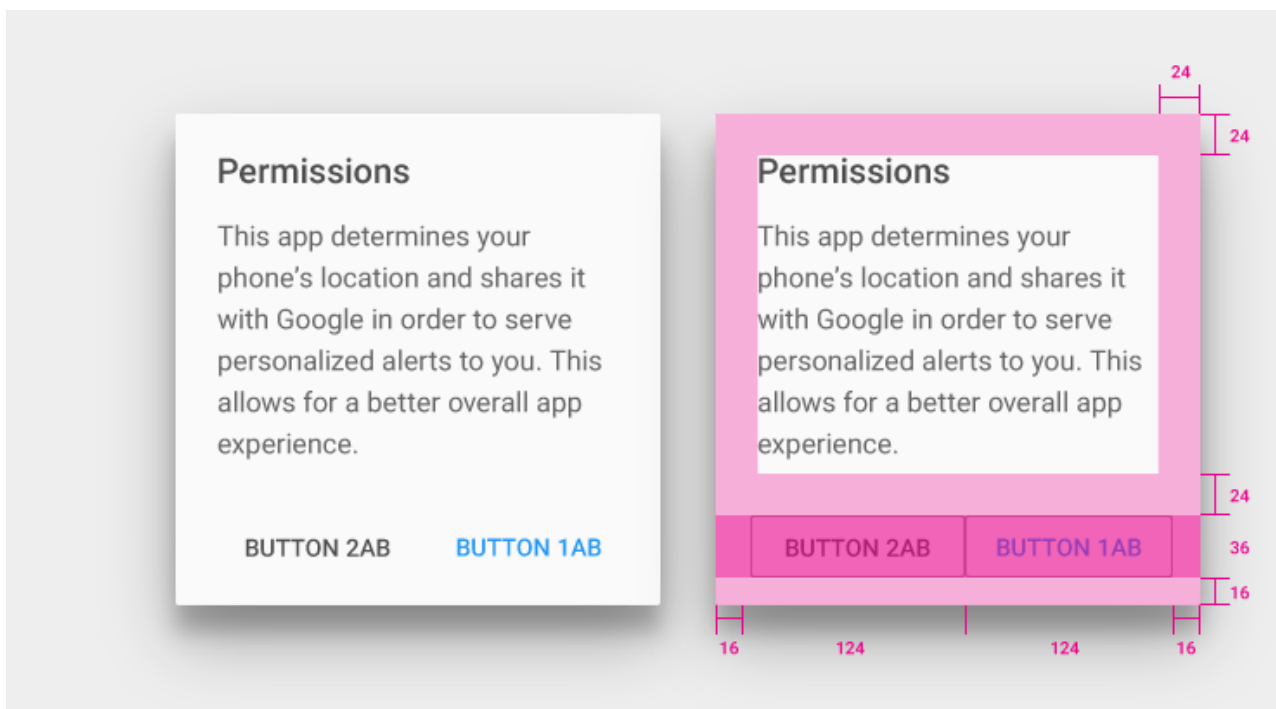
加宽型竖排按钮（**Stacked full-width buttons**）

当按钮的文本超过了通常的按钮宽度时，你就可以使用这种竖向叠模式来呈现我们的按钮文字信息。



并排按钮（**Side-by-side buttons**）

在每个按钮的文本信息都没有超过通常的按钮宽度时，推荐使用并排模式。比如说最常用的 确定/取消 按钮



内容

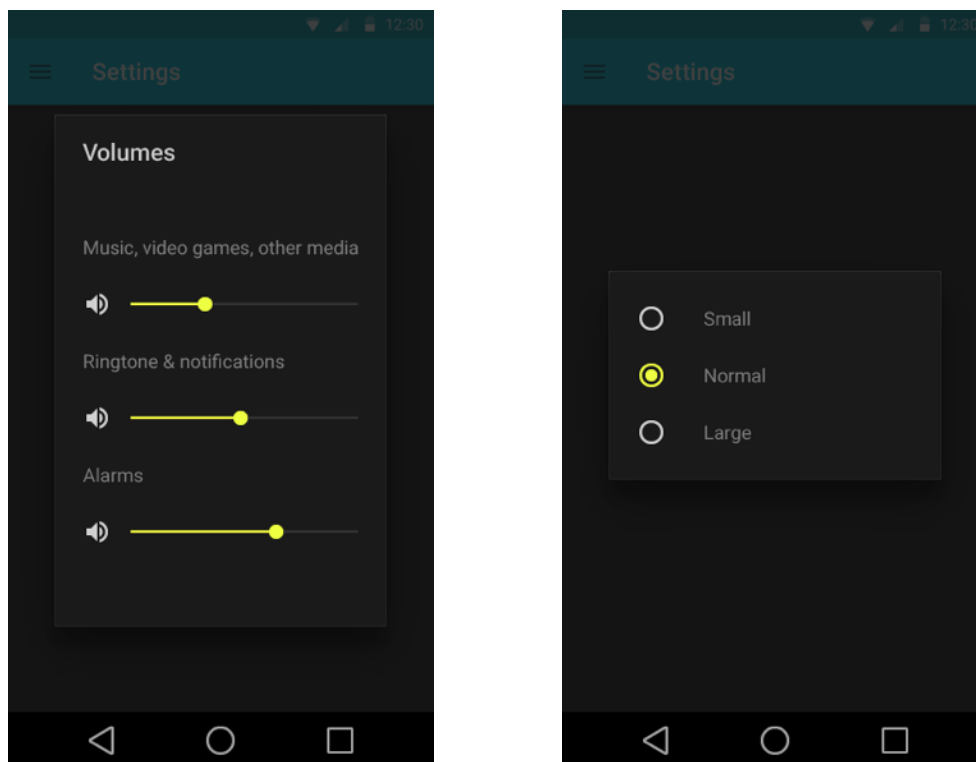
提示框标题

提示框的标题是可选的，用于说明提示的类型。可以是与之相关的程序名，或者是选择后会影响到的内容。例如：设置

提示框标题应该作为提示框的一部分被整体地显示出来。

提示框内容

提示框的内容是变化多样的。但是通常情况下由文本 和(或) 其它UI元素组成的，并且主要是用于聚焦于某个任务或者是某个步骤。比如说“确认”、“删除”或选择某个选项。



事件

提示框事件

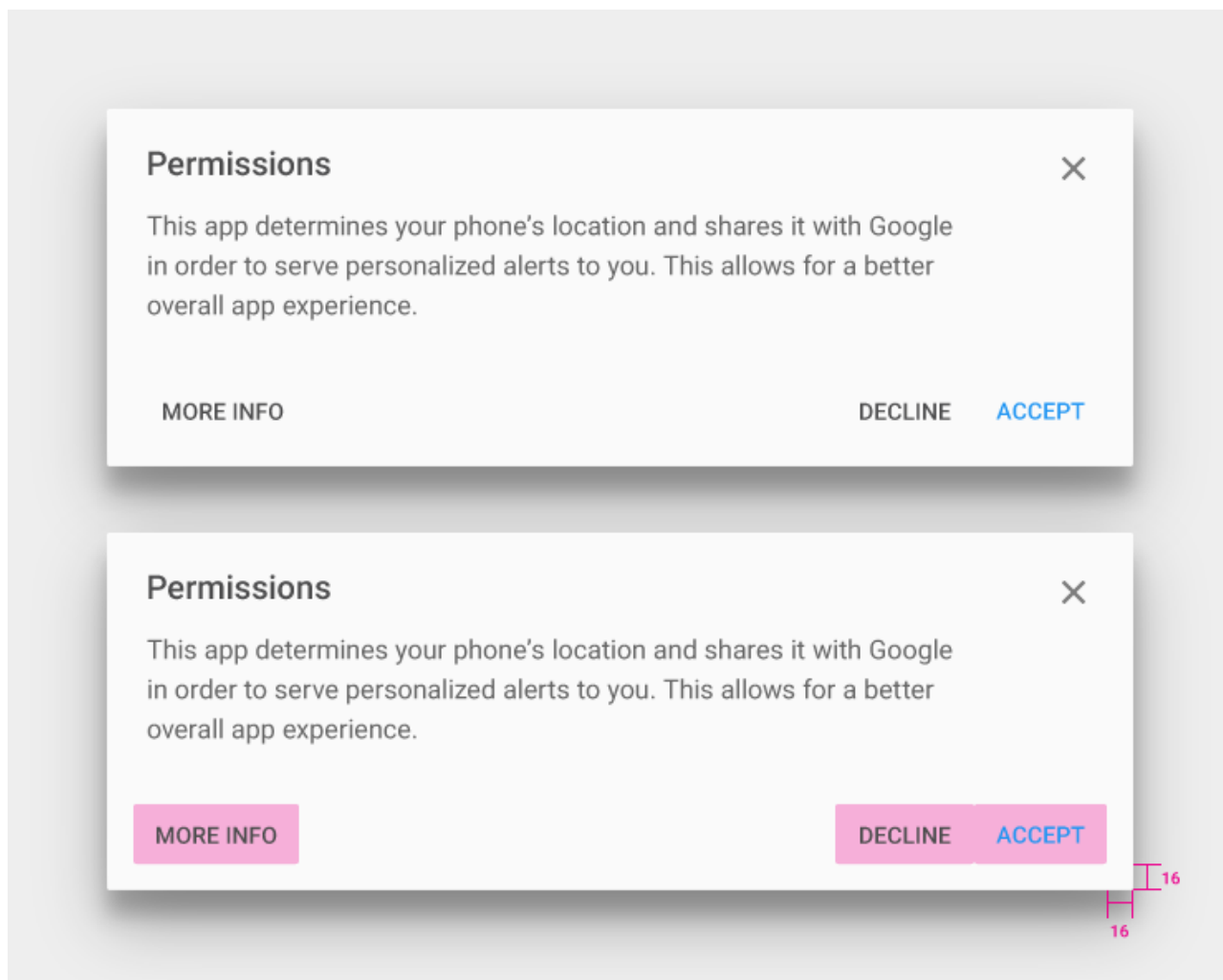
提示框呈现的是一组聚焦和有限的事件，通常是一个肯定的事件和否定(与肯定的事件对立)的事件组成。

肯定的事件是放于提示框的右边并且可以继续接下来的步骤。肯定的事件可以是据有破坏性的，比如：“删除”，“移除”。(话外音：肯定的事件主要是指产品期望用户的一个决策。与按钮文字呈现的语意无关)

否定的事件是放于提示框的左边。用于返回用户原始的屏幕或者是步骤。(话外音：一般就是关闭提示框作用)

事件的按钮排列类型可以是并列的，也可以是竖向叠加加宽型的。这取决于事件按钮里面的文字长短。

肯定事件和否定事件除了可以使用“确认”/“取消”外，也可以是其它一些动词或者是动词短语来表明决策后的结果。

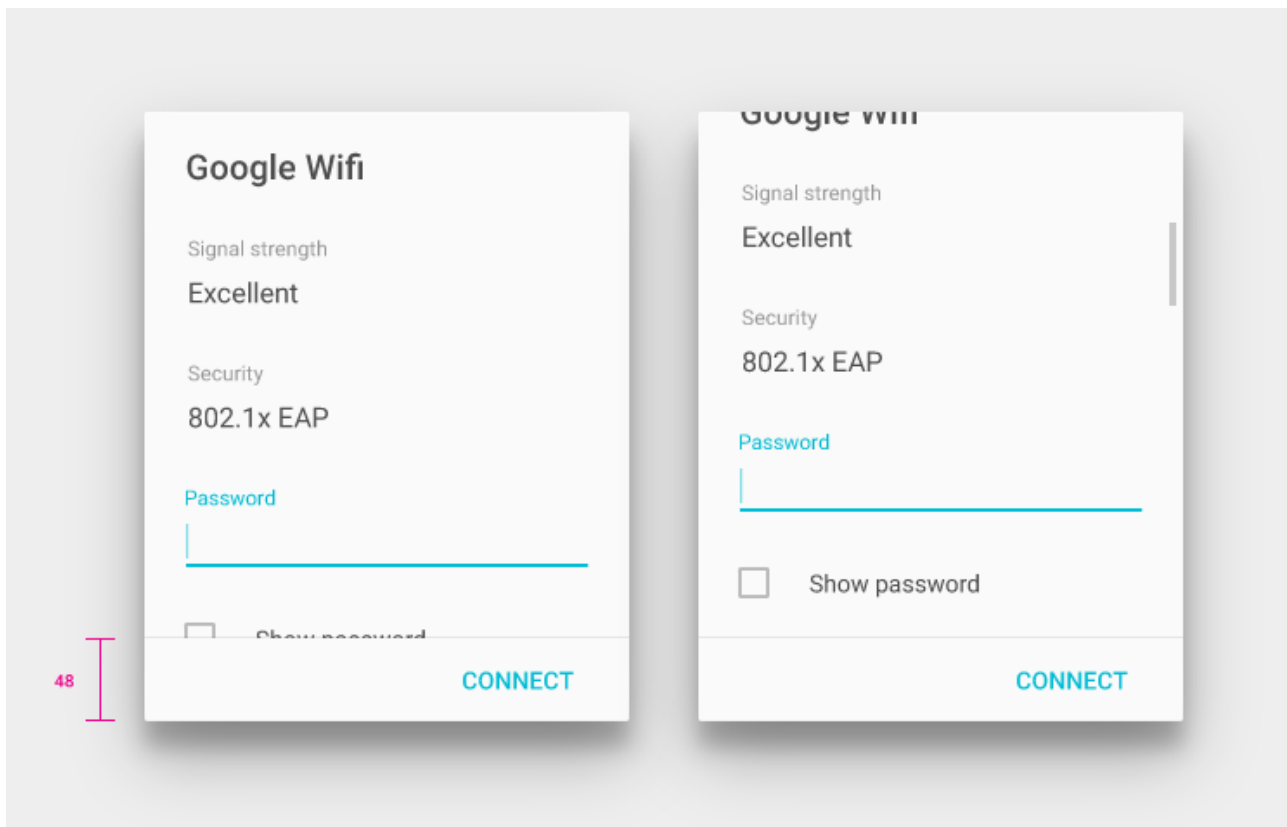


表现(Behavior)

滚动

提示框是与父视图是分隔开的。不会随着父视图滚动。

如果可以，请尽量保持提示框里面的内容不需要滚动。如果滚动的内容太多了，那么可以考虑使用其它的容器或者是呈现方式。然而，如果内容是滚动的，那么请使用较明显的方式来提示用户。比如说被让文字或者是控件露出一截出来。



手势

触摸提示框外面的区域可以关闭提示框

提示框焦点

提示框的焦点是整个屏幕。提示框在关闭前或者是用户选择了一个事件(比如说选择了一个选项)前都会持有焦点。

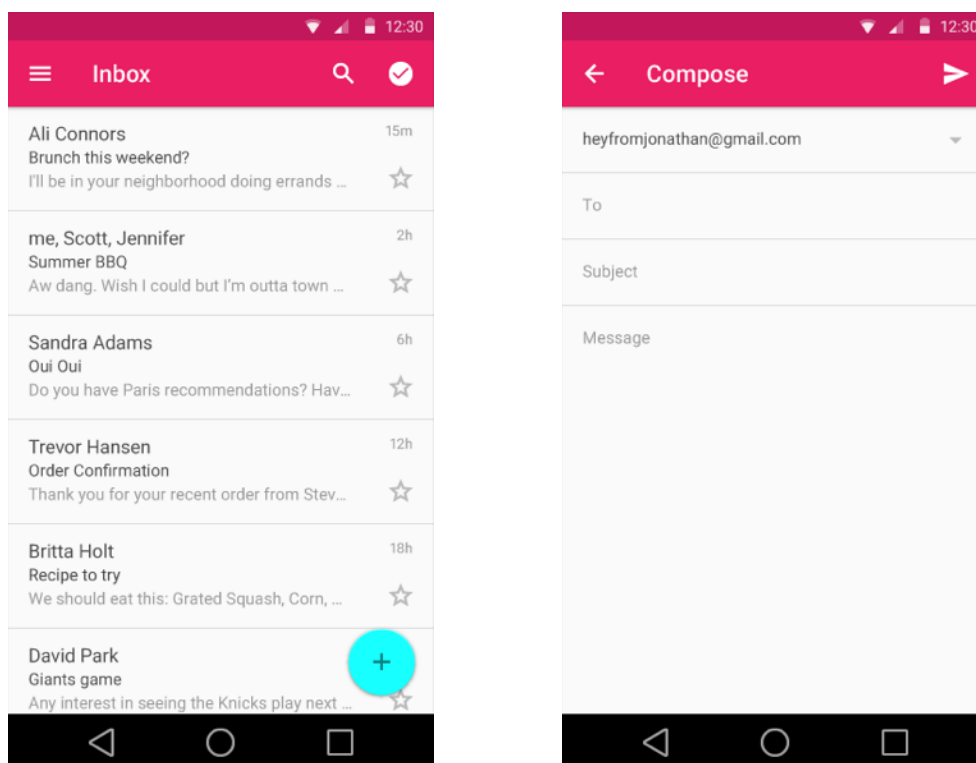
5.6 分隔线(Dividers)

分隔线主要用于管理和分隔列表和页面布局内的内容，以便让内容生成更好的视觉效果及空间感。示例中呈现的分隔线是一种弱规则，弱到不会去打搅到用户对内容的关注。

用途

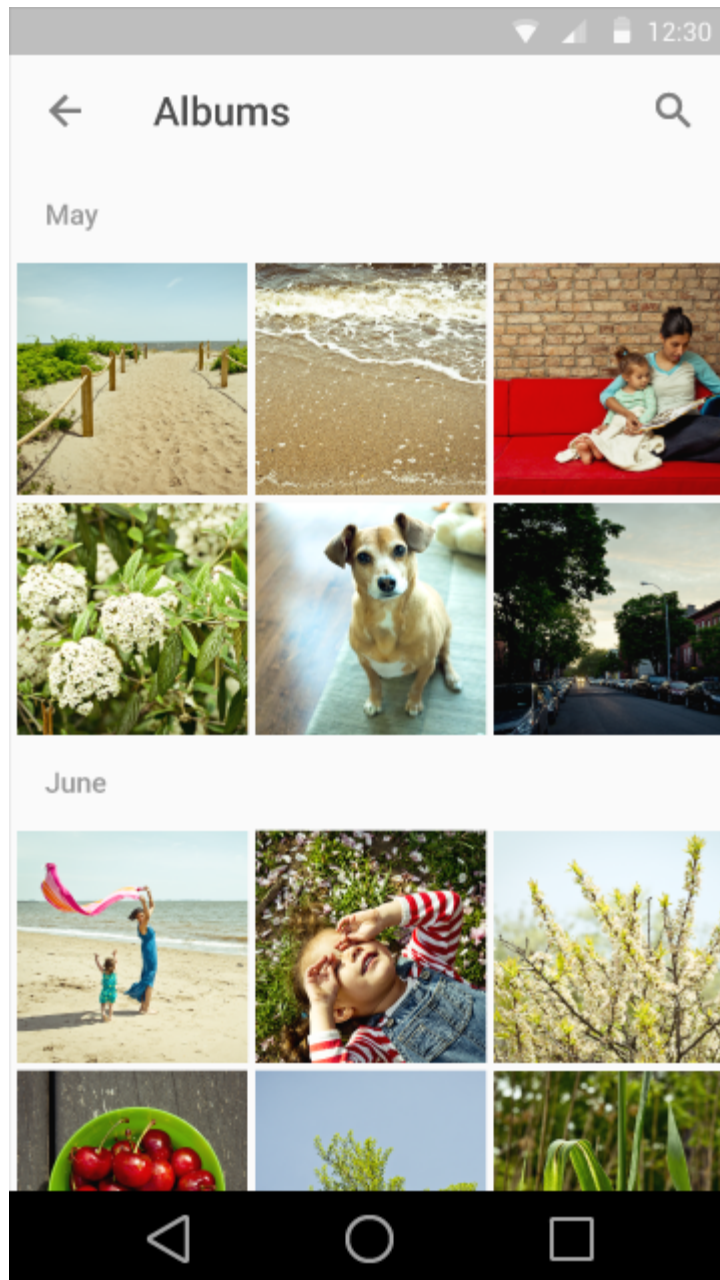
没有锚点的项

当在列表中没有像头像或者是图标之类的锚点元素时，单靠空格并不足以用于区分每个数据项(原文中使用的是“瓦片”)。这种情况下使用一个等屏宽(full-bleed)的分隔线就会帮助区别开每个数据项目。使其它看起来更独立和更有韵味。



基于图片的内容

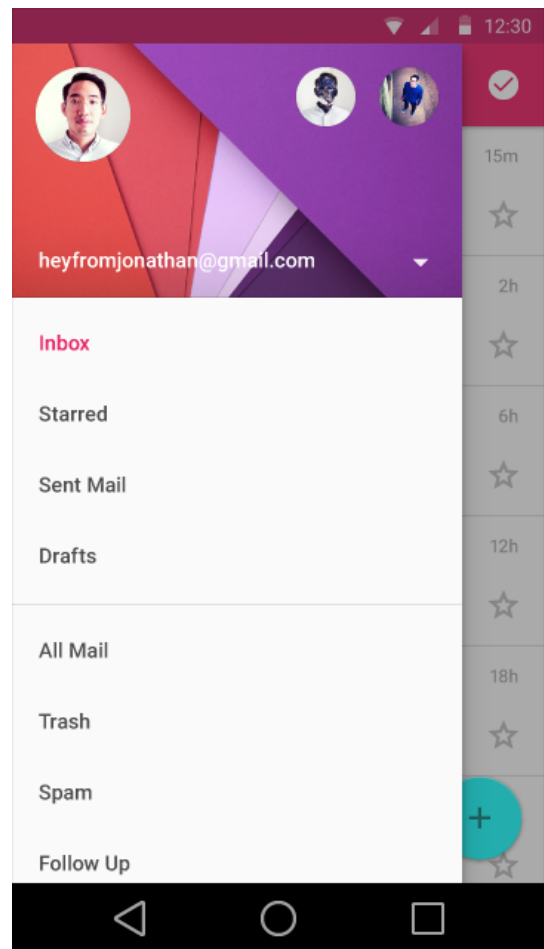
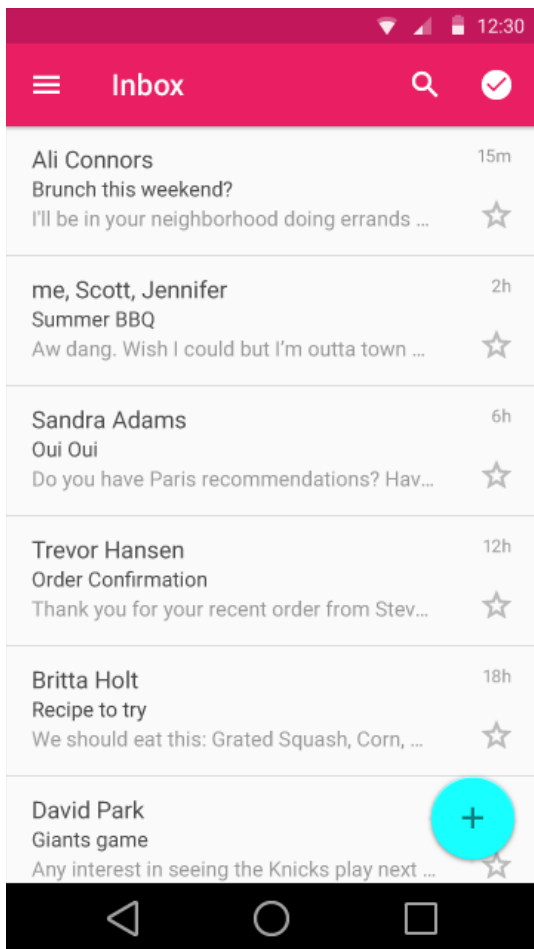
由于网格列表（grid）本身属性而造成的视觉效果，这就导致在网格列表中是不需要分隔线来区别子标题与内容的。在这种情况下，子标题与内容间的空白区域就可以分隔每块的内容了。



分隔线的类型

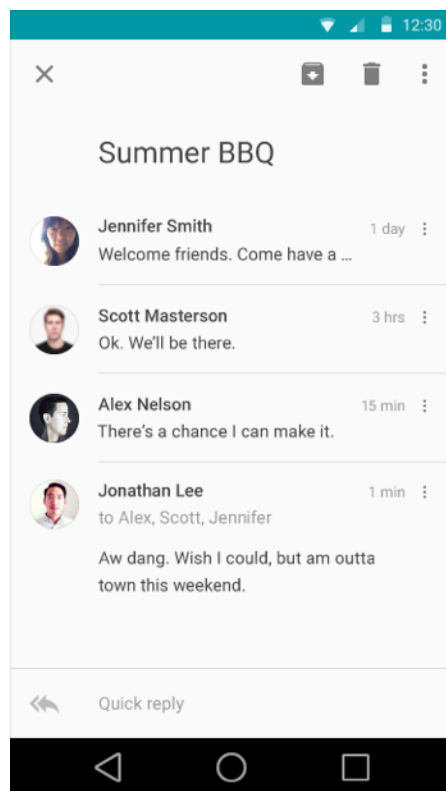
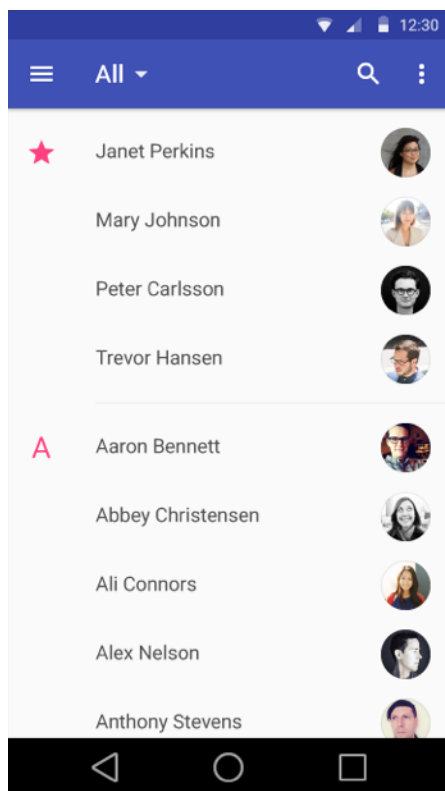
等屏宽分隔线 (Full-bleed dividers)

等屏宽分隔线或以用于分隔列表中的每个数据项或者是页面布局中的不同类型的内容。



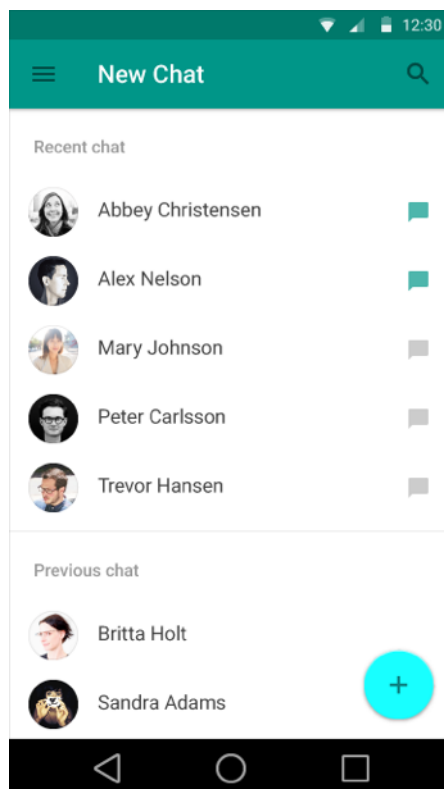
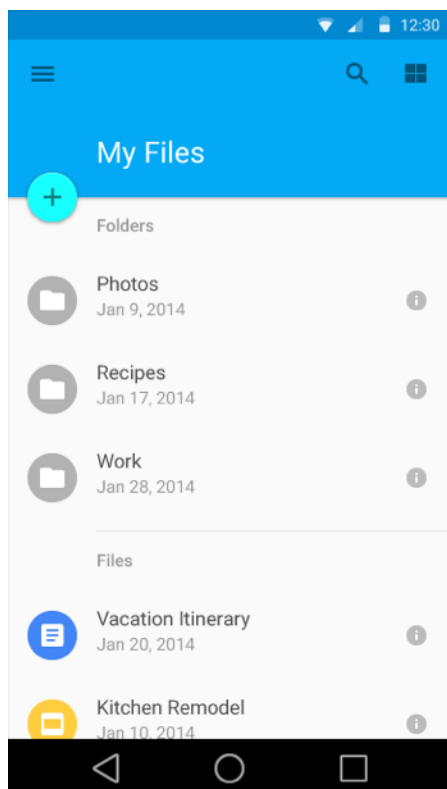
内凹分隔线 (Inset dividers)

在有锚点元素 (头像或者是图标) 并且有关键字的标题列中，我们可以使用内凹分隔线。



子标题和分隔线

在使用分隔的子标题时，可以将分隔线置于子标题之上，可以加强子标题与内容关联度。

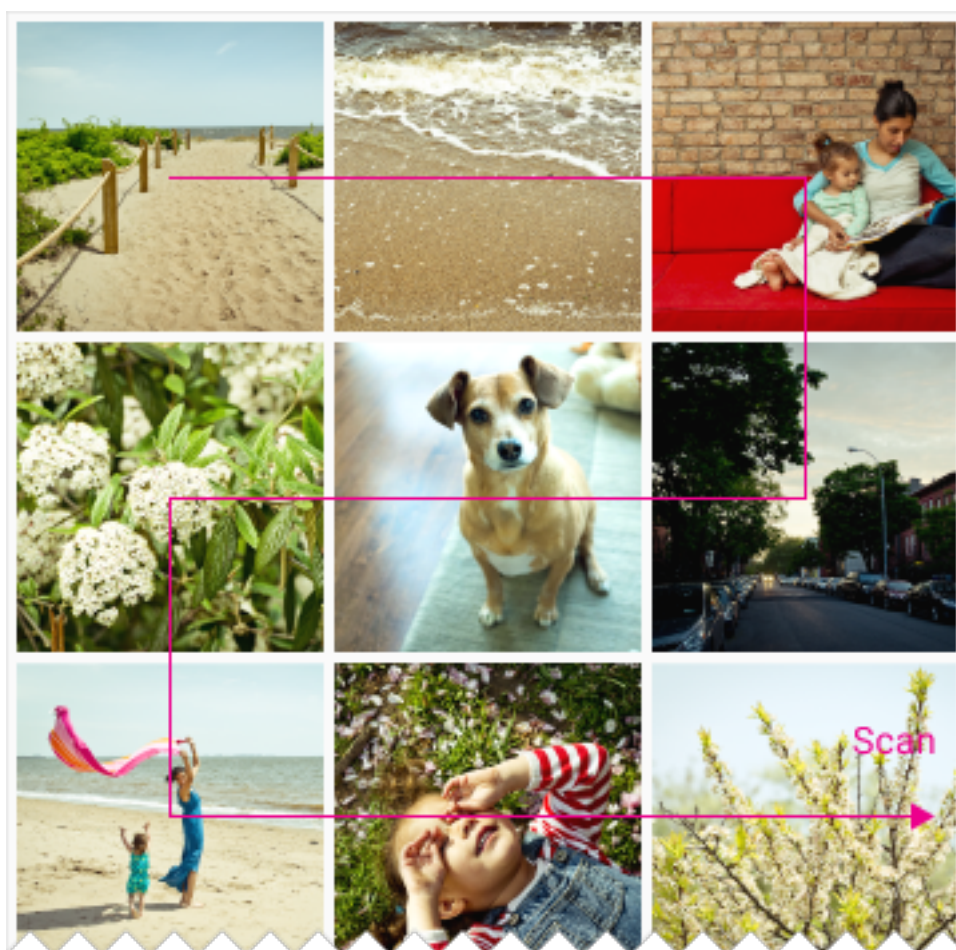


5.7 网格

网格列表是一种标准列表视图的可选组件。网格列表与应用于布局和其他可视视图中的网格有着明显的区别。

用法

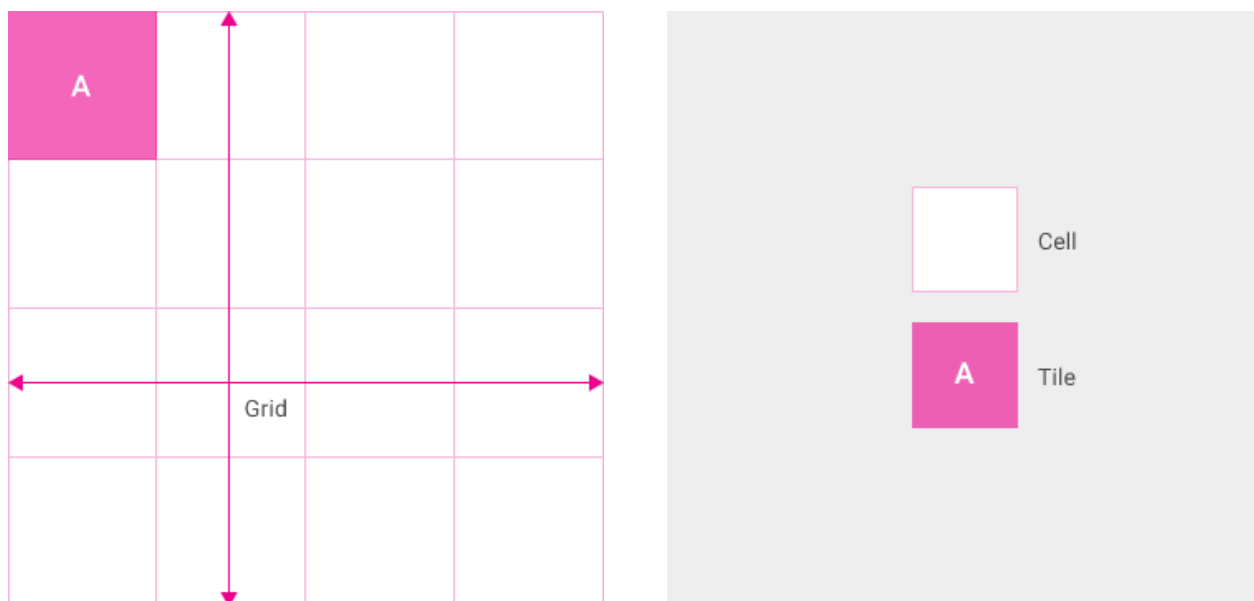
网格列表最适合用于同类的数据类型（homogeneous data type），典型的如图片，并且对，比如数据类型的可视化理解（visual comprehension）和区别进行了优化。



网格列表是一个连续元素（continuous element），该元素由棋盘式、规律性的小格子构成，通常称这些格子为单元格（cells）。单元格中包含有瓦片（tiles）。

单元格在网格中以垂直和水平的方式排列。

瓦片用以存放内容，并且可以跨越一个或者多个垂直或者水平的单元格。



如果瓦片中的文本需要与其他主要的内容有着足够显著的区别，可以考虑使用一个容器，比如列表（Lists）或者卡片（Cards）。这是显示文本、增强阅读理解的最佳方案。

Lists：是增强阅读理解的最佳方案，尤其是在比较一组具有多种数据类型的数据时。

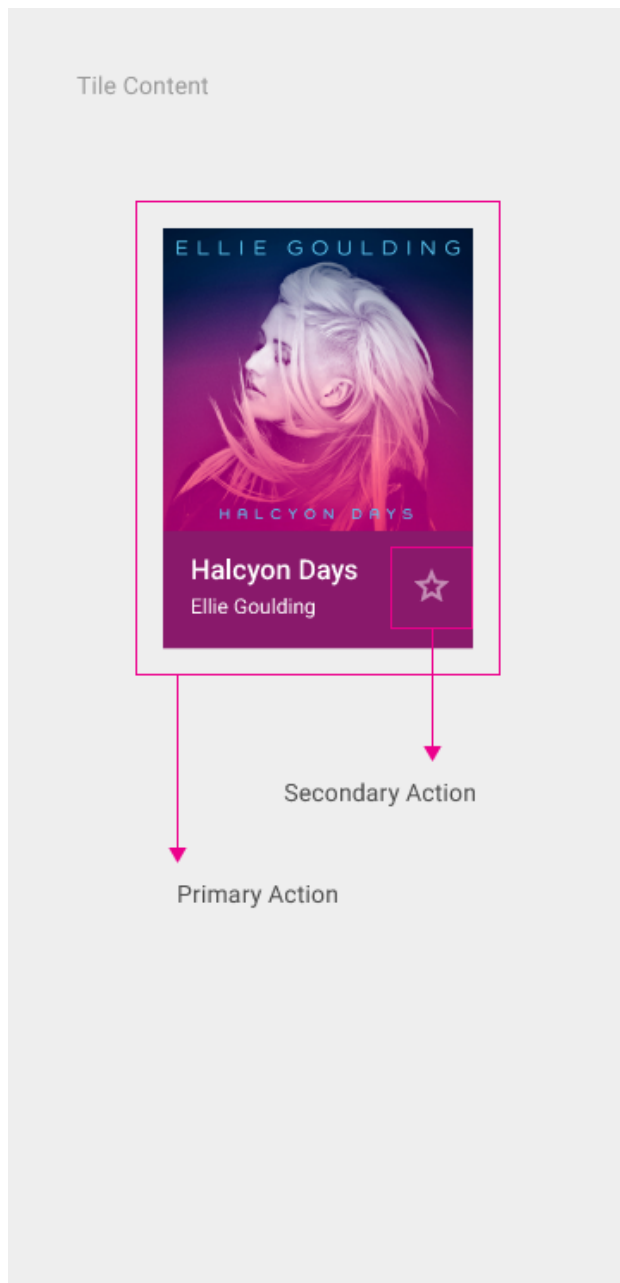
Cards：用于不同格式的内容，比如带有不同高度题注的图片，或者应用于不同类内容的数据集合中，比如具有图片、视频和图书的混合式数据集。

内容

瓦片中的内容

瓦片内容包括主要内容（primary content）和次要内容(secondary content)。主要内容是有着重要区别的内容，典型的如图片。次要内容可以是一个动作按钮或者文本。

为瓦片提供一个不带有主要内容的默认图片。



瓦片中的动作

主要内容和次要内容中的动作—比如播放、放大、删除或者选择—是一种瞬时性动作，通常不会在网格列表中弹出选项子菜单（动作溢出列表，action overflow）。

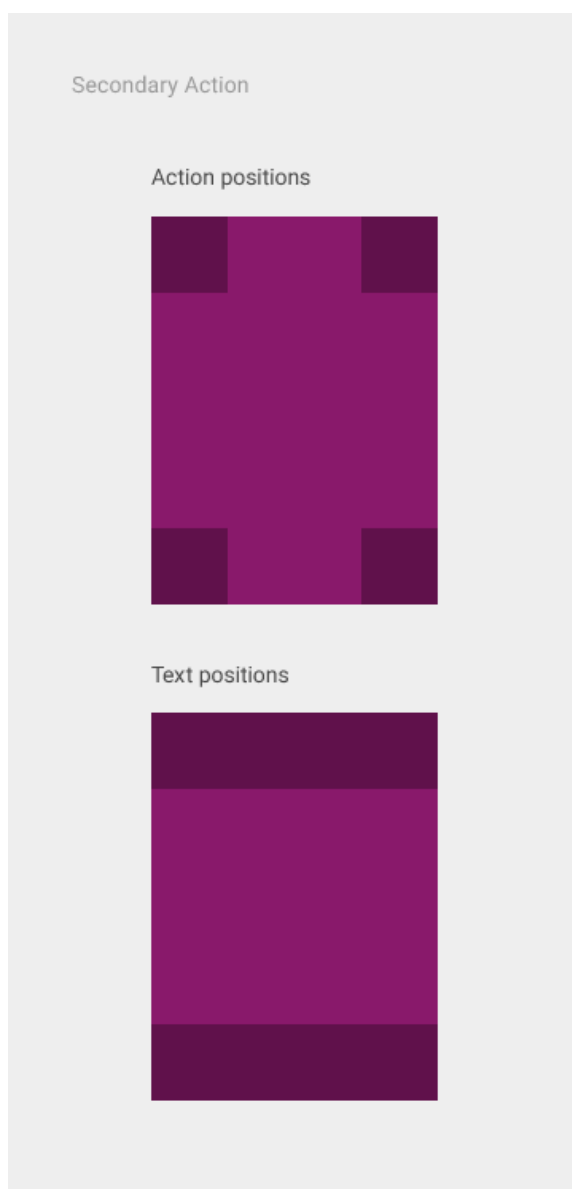
动作可以打开一个随后的视图，比如卡片或者悬浮卡片（hovercard）。

主要动作

- 充满整个瓦片，因此不会通过图标或者文本呈现。
- 在指定的网格中，所有瓦片的动作是一致的。比如，在单个网格中，对于所有瓦片的主要动作可以用于查看图片的详细信息。

次要动作和内容

- 通过图标或者文本呈现出来。
- 在指定的网格中，所有瓦片的动作是一致的。
- 在指定网格的瓦片中放置的位置是一致的，但是位置可能会在不同的网格（边角或者边界，corners or edges）间有变化。比如，网格中的所有瓦片可能放置在了左下角。



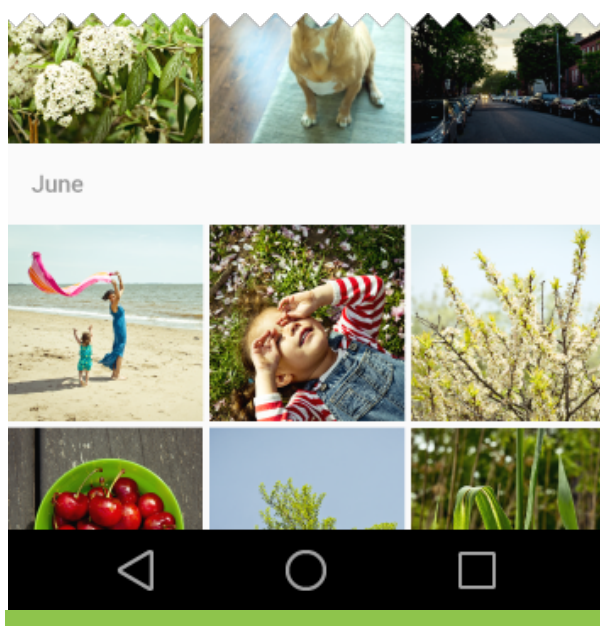
行为

滚动

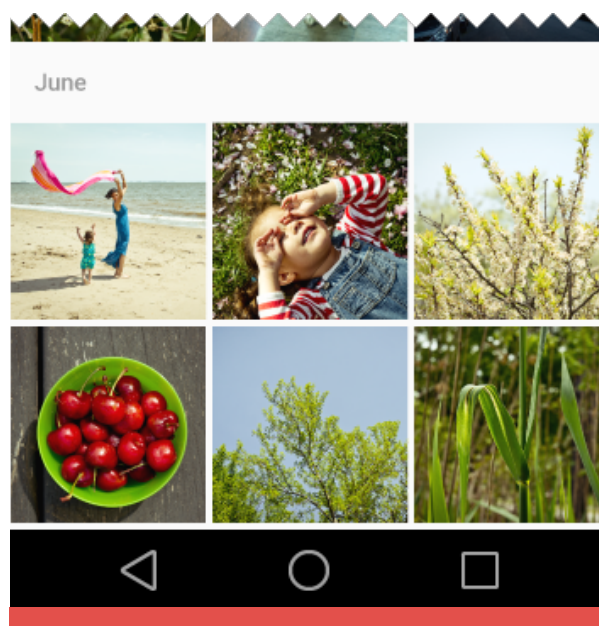
网格通常只滚动垂直。

水平滚动的网格是不鼓励使用的，这通常与用户的阅读习惯有关，影响阅读上的理解。

截去网格瓦片并支持滚动方向，告诉用户这里还有更多的内容。



正确：说明这里还有更多的内容



手势

不允许使用猛击（swipe）手势。选中并移动（pick-up-and-move）动作不鼓励使用。

瓦片过滤与排序

网格列表中的内容可以编程实现其过滤和排序，比如通过数据类型、文件大小、字母顺序或者其他参数等。

网格中的第一个条目置于网格的左上角，并且其顺序为从左到右，自上而下。

维度与重置尺寸

重置网格列表的尺寸会导致瓦片在有水平空间可用时重新排序。但是瓦片并不会缩放以填充可用的水平空间。

当水平空间受限时，网格列表不会发生改变。网格列表与列表在强调不同类型时时不同的结构：图片优于文本与文本优于图片的区别。

边框

网格列表表头/表尾 (header/footers)

单行表头/表尾高：48dp

文本内边距：16dp

默认字体大小：16dp

次要动作与尾右对齐

两行表头/表尾高：68dp

文本内边距：16dp

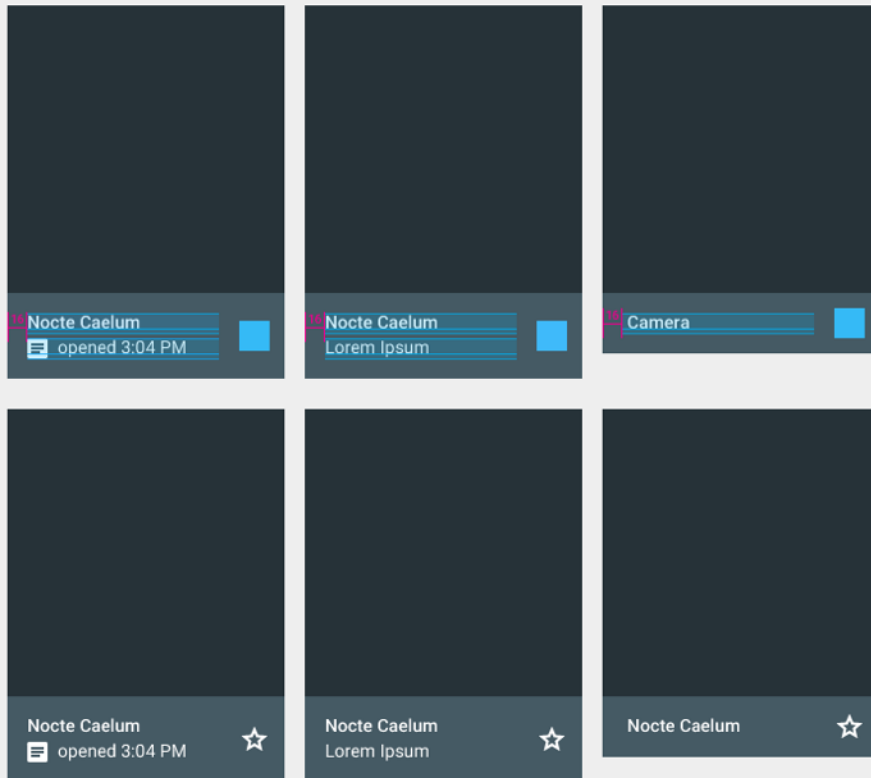
每行的默认字体大小：16sp/12sp或者14sp/14sp

Type 14 / 14

Two Line with Icon — 68dp Bar

Two Line — 68 dp Bar

Single Line — 48dp Bar

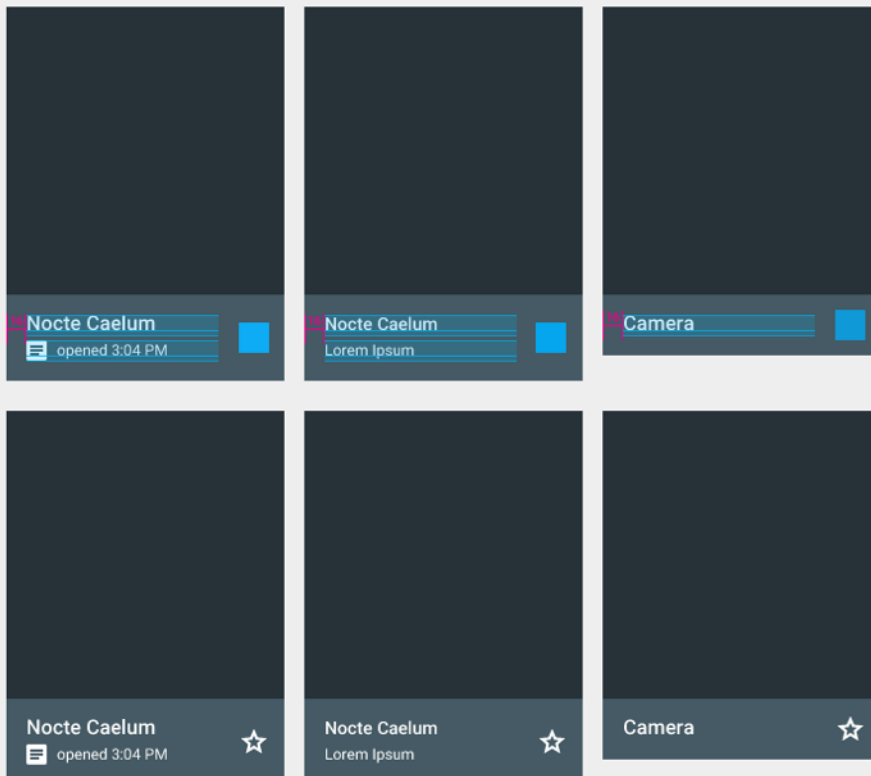


Type 16 / 12

Two Line with Icon — 68dp Bar

Two Line — 68 dp Bar

Single Line — 48dp Bar

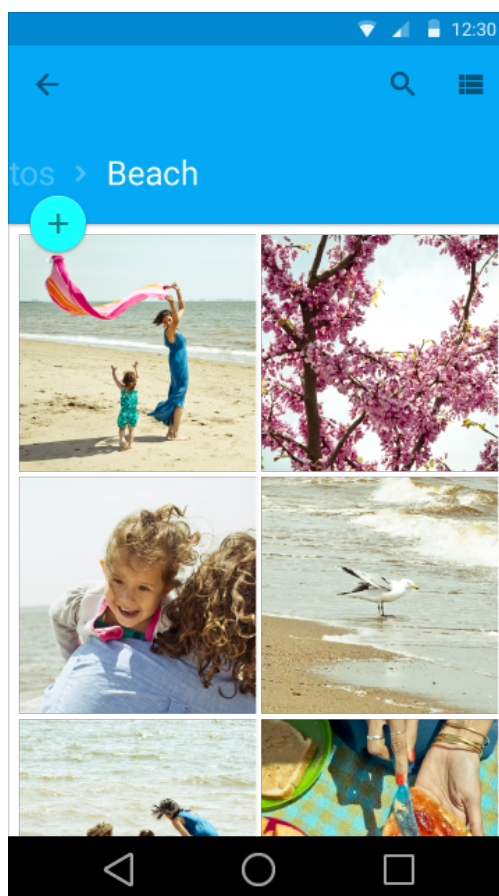


仅有图片的网格列表

网格内边距：4dp

网格列表中的瓦片可以跨多列。

仔细考虑网格列表中的次要文本是否需要使用多列瓦片，因为大的瓦片可能会造成很大的空间浪费

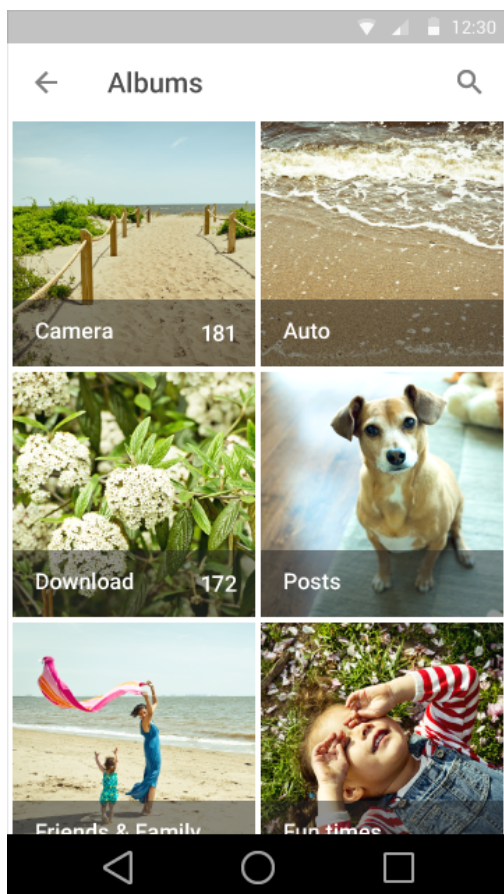


单行网格列表

仅有文本高: 48dp, 文本内边距: 16dp, 默认字体大小: 16sp, 网格内边距: 4dp



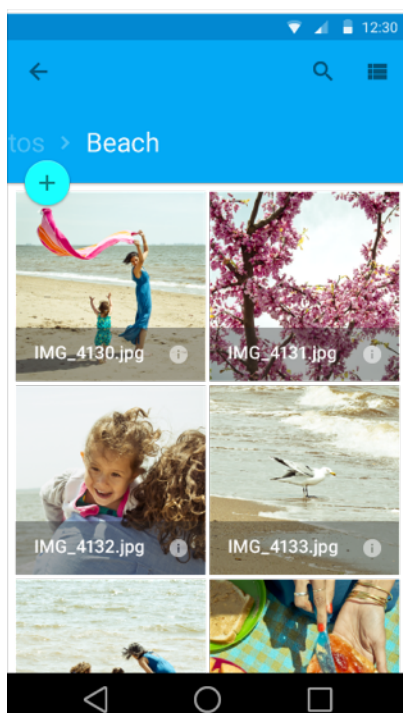
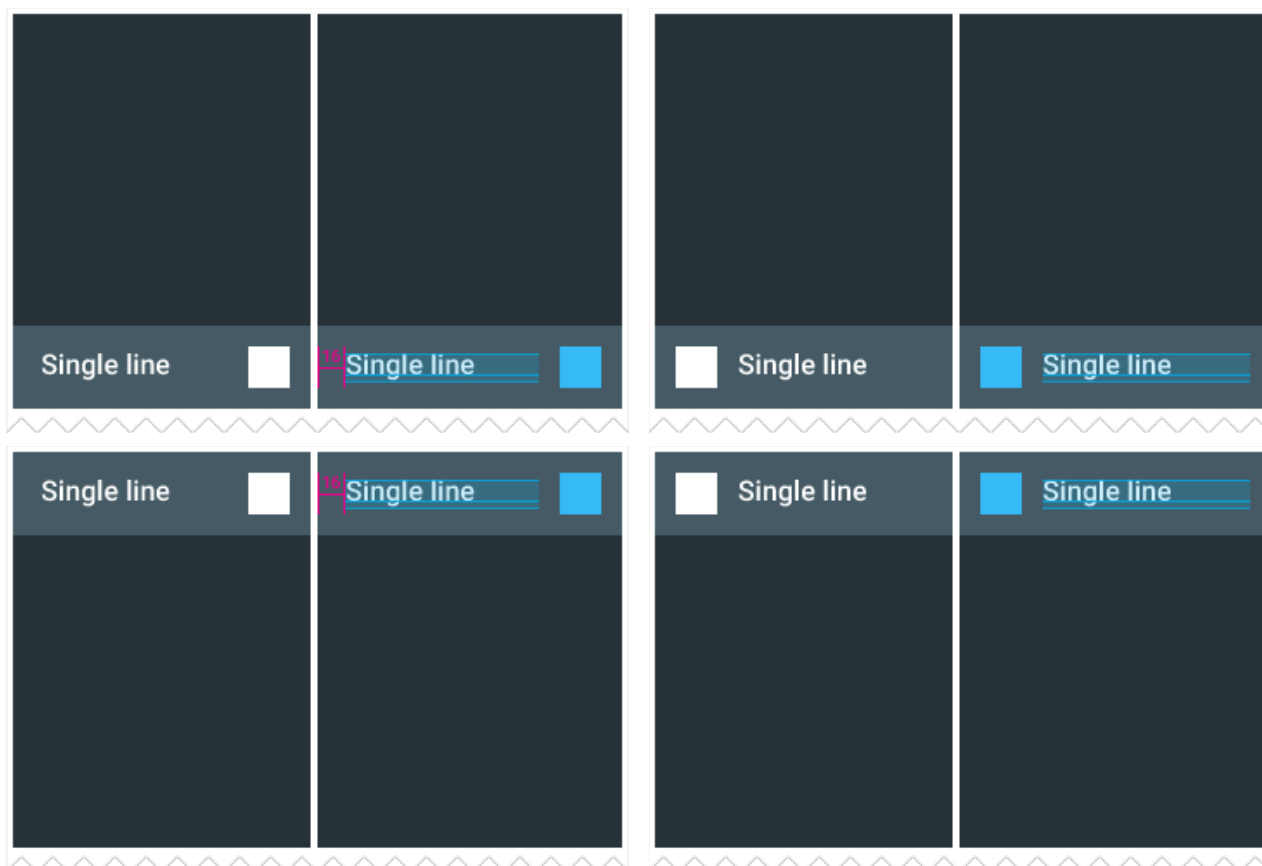
元素



带图标的文本网络

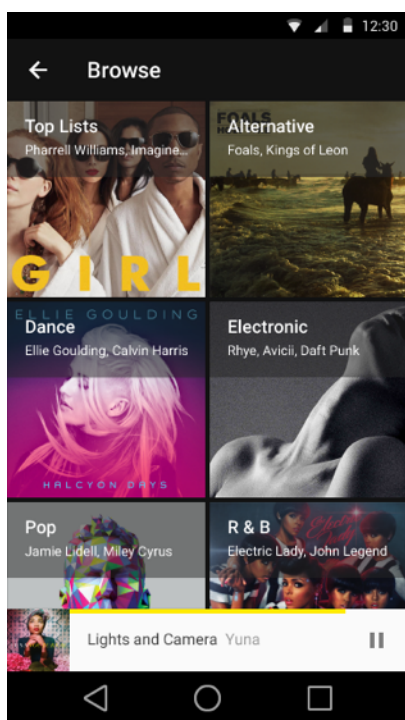
高：48dp，文本内边距：16dp，默认字体大小：16sp，网格内边距：4dp

网格列表表尾或者表头的中的次要文本可以右对齐或左对齐



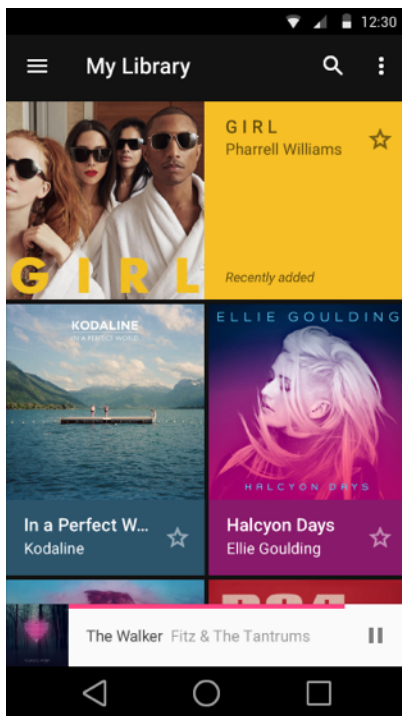
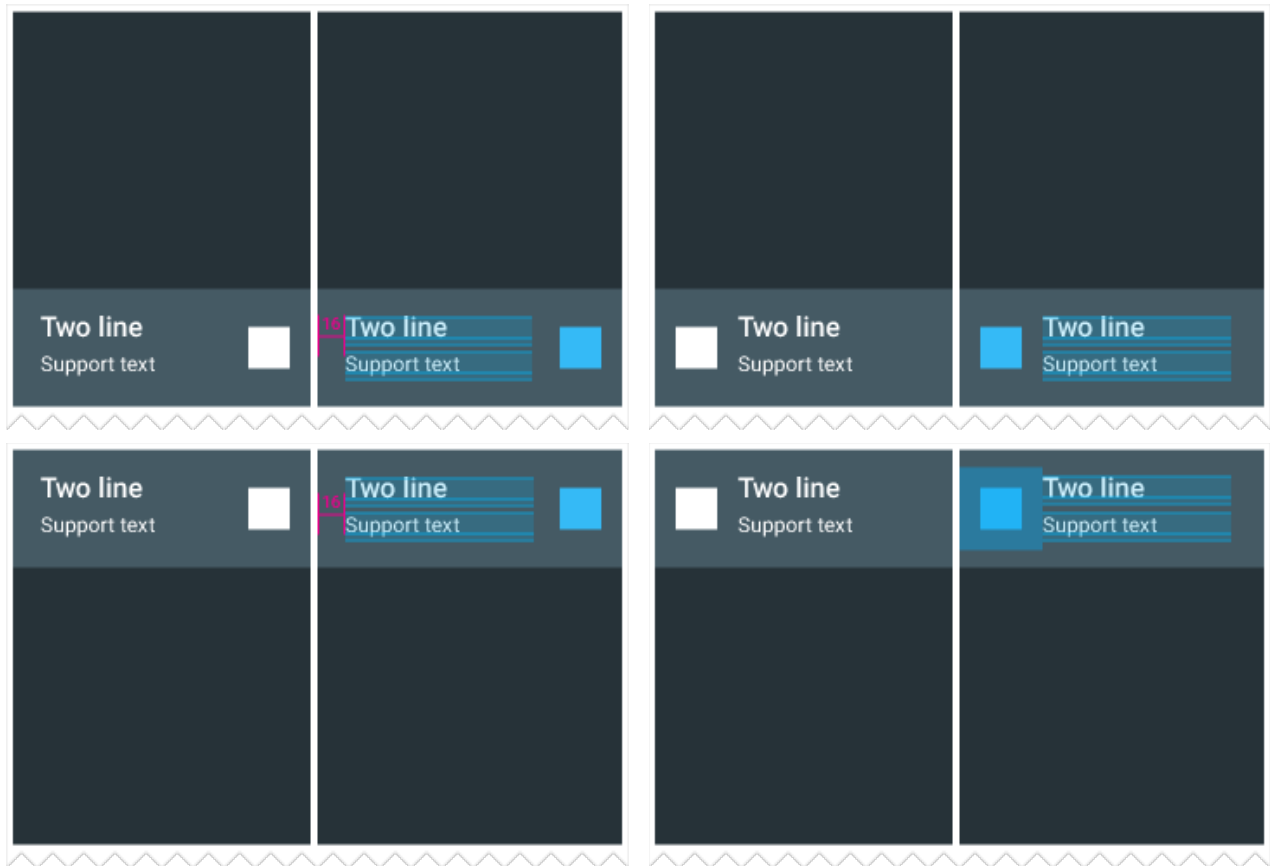
两行网格列表

仅有文本高：68dp，文本内边距：16dp，每行的默认字体大小：16sp/12sp或14sp/14sp，网格内边距：4dp



带有图标的文本

高：68dp，文本内边距：16sp，每行的默认字体大小：16sp/12sp或14sp/14sp，网格列表表尾或者表头中的次要文本可以右对齐或左对齐。，网格内边距是4dp



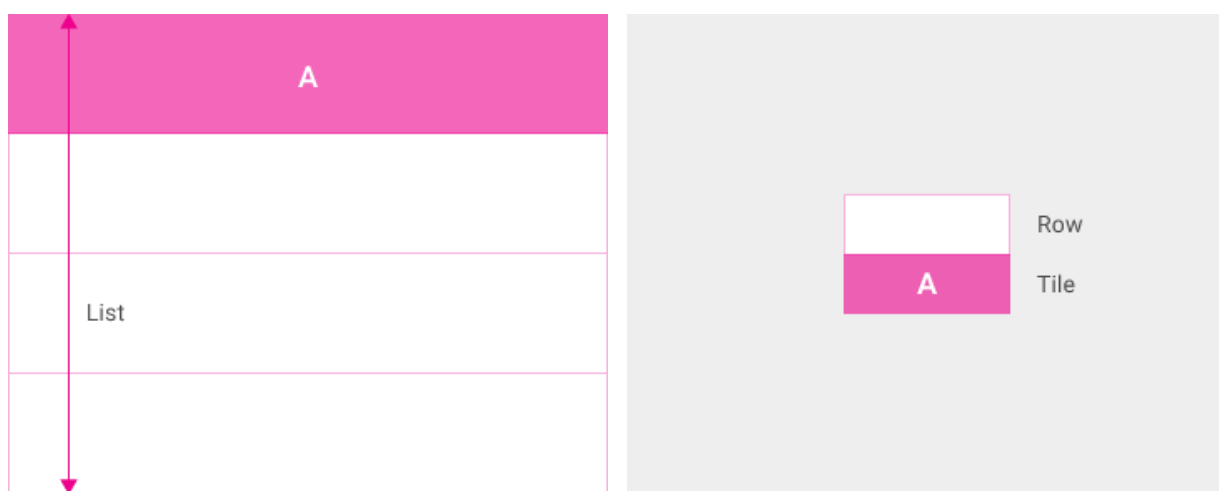
5.8 列表

列表作为一个单一的连续元素来以垂直排列的方式显示多行条目。

用法

列表由单一连续的列构成，该列又等分成相同宽度称为行(rows) 的子部分。行是瓦片(tiles) 的容器。

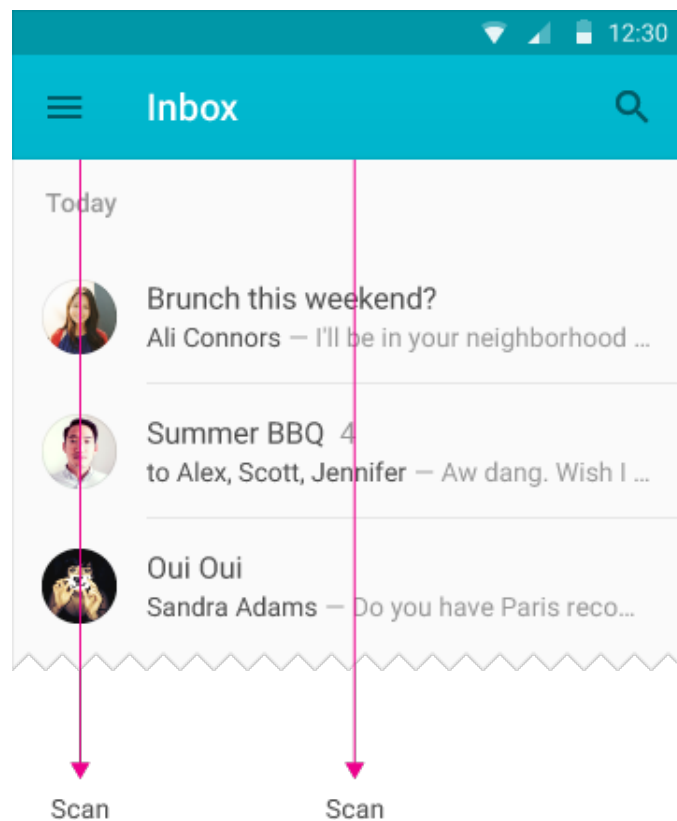
瓦片中存放内容，并且在列表中可以改变高度。



列表最适合应用于显示同类的数据类型或者数据类型组 (homogeneous data type or sets of data types)，比如图片和文本，目标是区分多个数据类型数据或单一类型的数据特性，使得理解起来更加简单。

如果有超过三行的文本需要在列表中显示，换用卡片 (cards) 代替。

如果内容的主要区别来源于图片，换用网格列表 (grid list)。



内容

文本内容

列表瓦片以一致的格式来显示一组相关的内容，采取层次感优于一致性类型或者一组内容的方式来增强阅读感。比如，在时间戳上强调头像和文本片段。这有助于使用者可以在一组内容中更容易地区分出他们要找的信息。

列表瓦片可以包含三行的文本，并且文本的字数可以在同一列表的不同瓦片间改变。要显示多于三行的文本，使用卡片。

与瓦片的左侧和多行内容中的瓦片内的第一行相比，更偏向于瓦片中的有明显区别的内容。



文本动作

列表瓦片中空间大部分应用专用于主要的动作。

因为动作是列表瓦片中不能识别的元素，所以将额外的动作放置在瓦片的右边。

主要动作和额外动作，比如播放、放大、删除和选择，是瞬时性的，并且通常不会在列表中弹出选项子菜单（动作溢出列表，action overflow）。

动作可以打开一个随后的视图，如卡片或者悬浮卡片（hovercard）。

主要动作

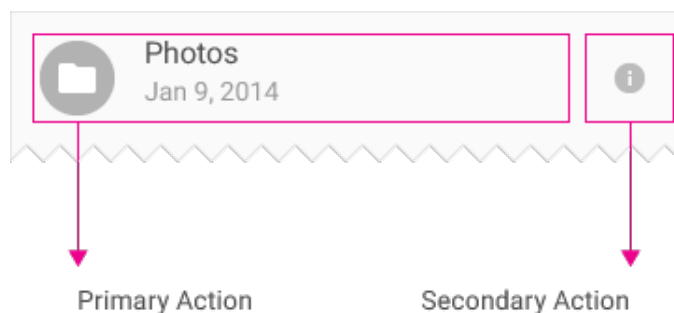
- 充满整个瓦片，因此不能通过图标、文本等元素呈现。
- 在特定的列表中所有瓦片的动作是一致的。比如在指定的音乐列表中，瓦片的主要内容是播放一首歌曲，或者在邮件列表中是打开一封邮件来阅读。

额外动作

- 在瓦片中通过图标、次要文本等来呈现出来。
- 在指定的列表中所有瓦片的动作是功能一致的，比如图片标识某人某人是否在线等。
- 在指定列表的瓦片中，动作放置的位置是一致的。

避免不断在瓦片中使用额外动作来制造“噪声”，比如在每个瓦片中显示分享动作按钮。

触发按钮，比如星标（starts）或者置顶（pins）是一个特例，因为它们通过显示状态来呈现出有效的信息。



行为

滚动

列表只支持垂直滚动。

手势

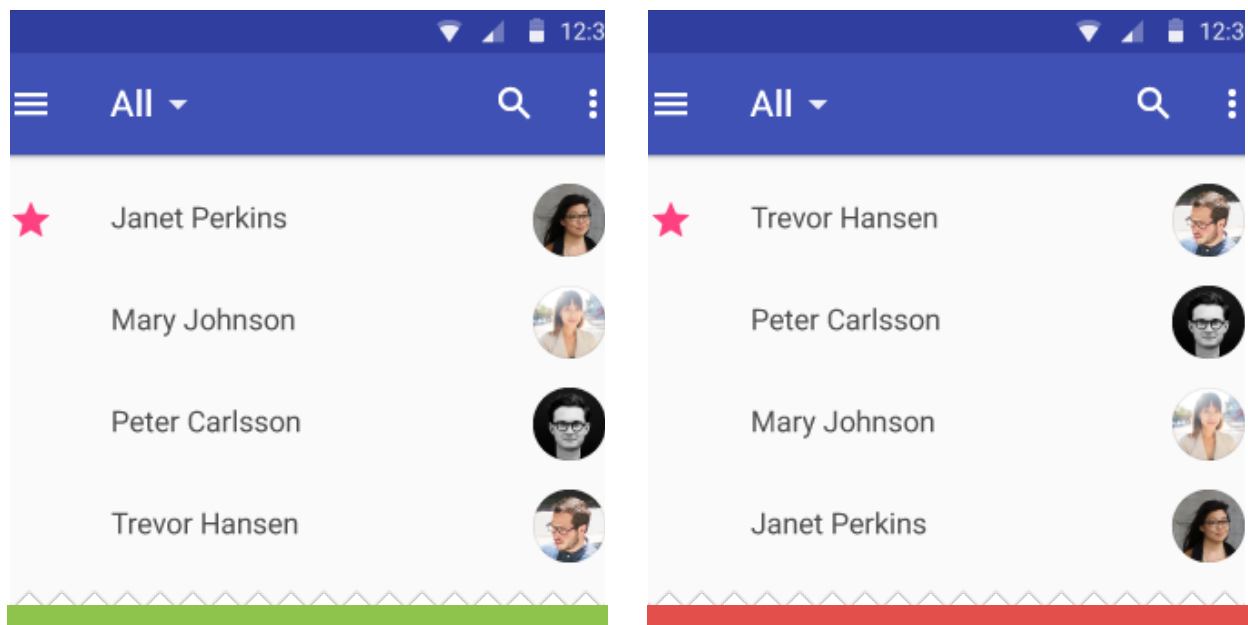
在列表中，每个瓦片的猛击（swipe）动作应当是一致的。

在操作正确时，瓦片可以在列表和下拉目标间移动（比如，移动一个文件到文件夹）。

在操作正确时，瓦片可以被选中并且在列表中可以手动改变顺序。

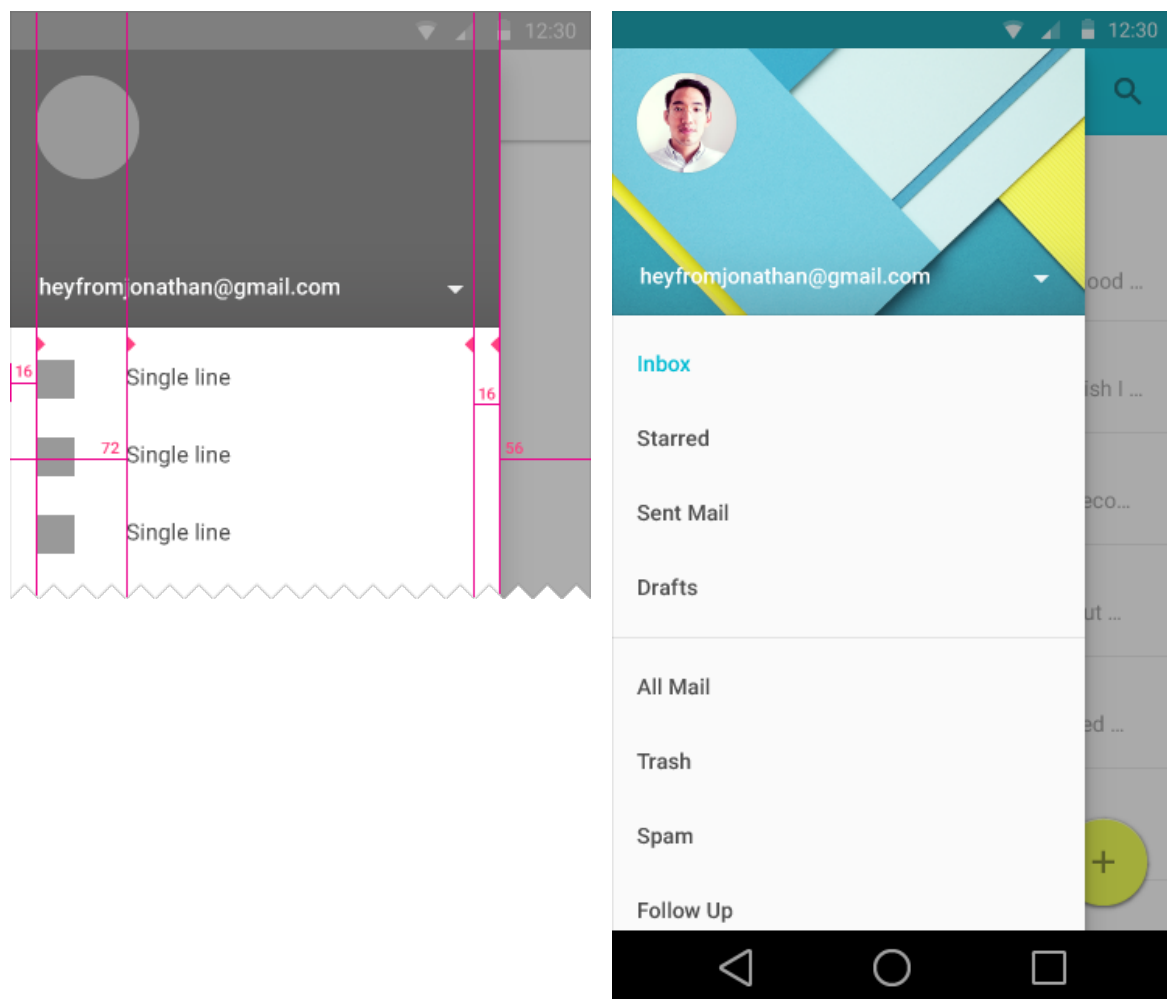
文本过滤与排序

列表瓦片可以通过数据、文件大小、字母顺序或者其它参数来编程改变其顺序或者实现过滤。

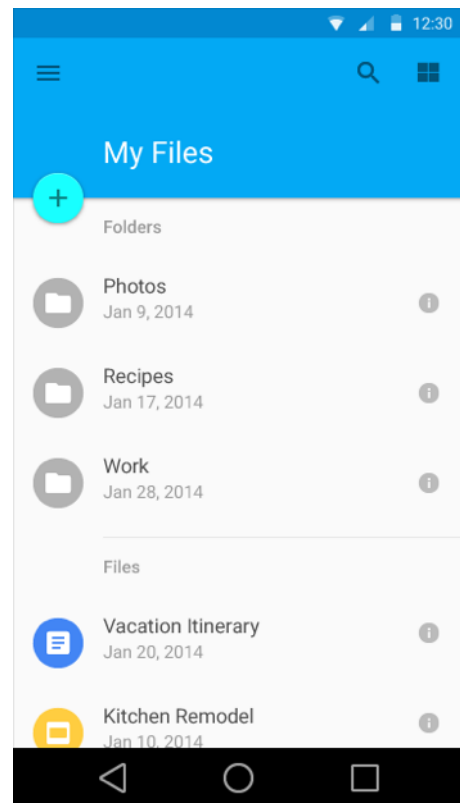
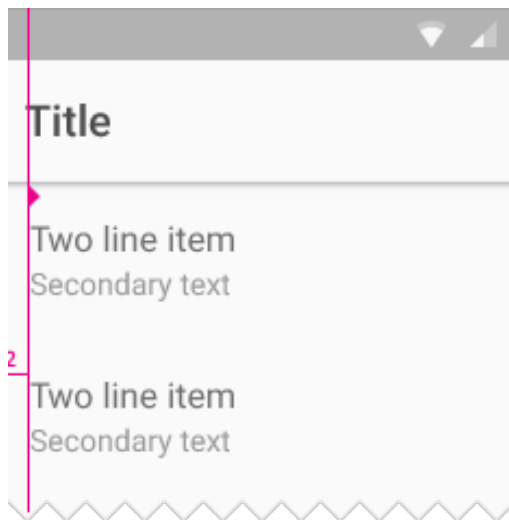


边框

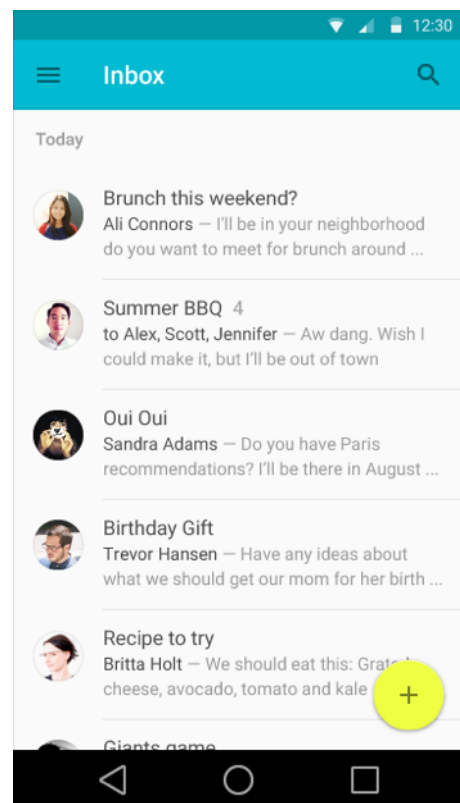
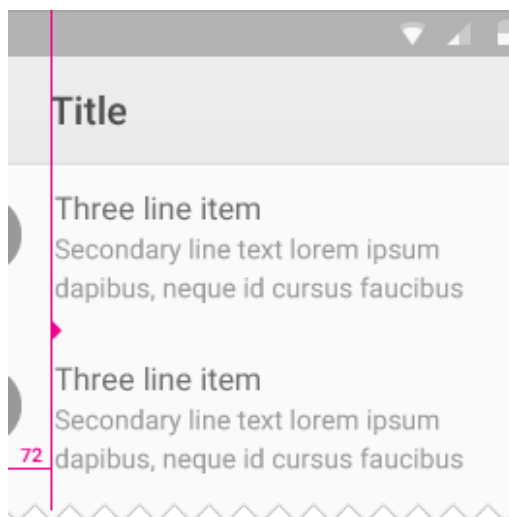
在单行列表中，每个瓦片包含了单行的文本。文本字数可以在同一列表的不同瓦片间改变



在两行列表中，每个瓦片最多包含两行的文本。文本字数可以在同一列表的不同瓦片间改变



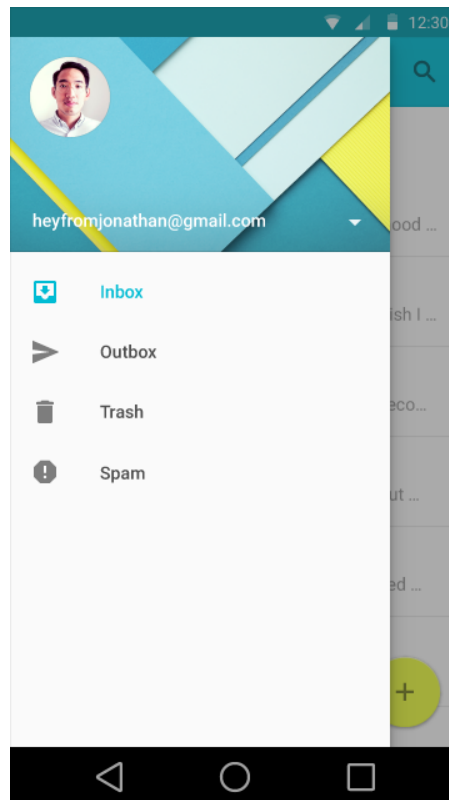
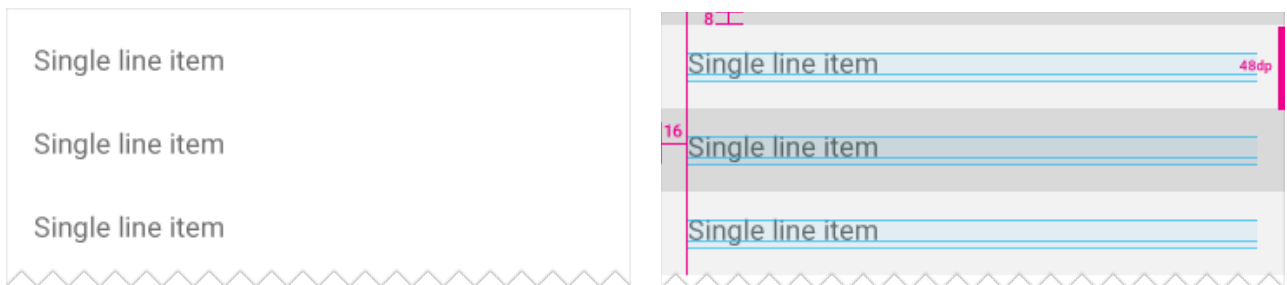
在三行列表中，每个瓦片最多包含三行文本。文本的字数可以在同一列表的不同瓦片间改变。



单行列表

仅有文本，字体：Roboto Regular 16sp，瓦片高：48dp，文本内边距：16dp

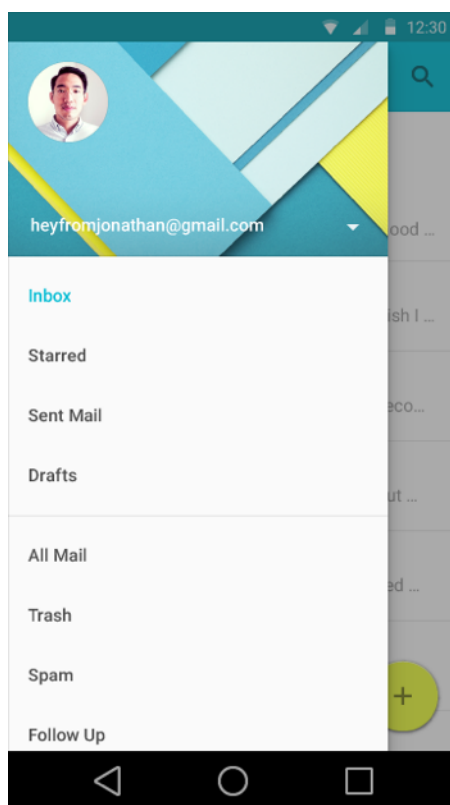
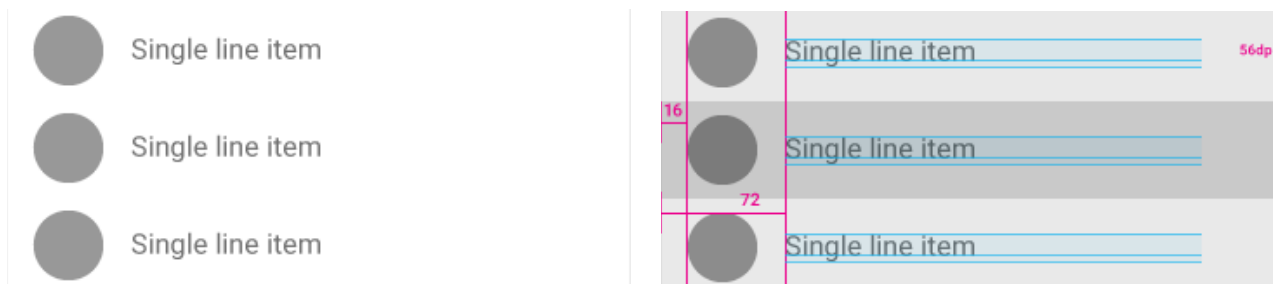
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



带有文本的图标

字体：Roboto Regular 16sp，瓦片高：48dp，左边图标内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：16dp

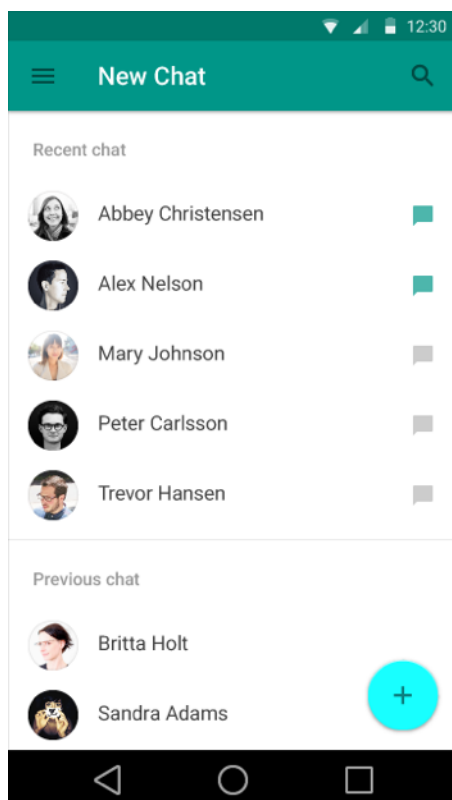
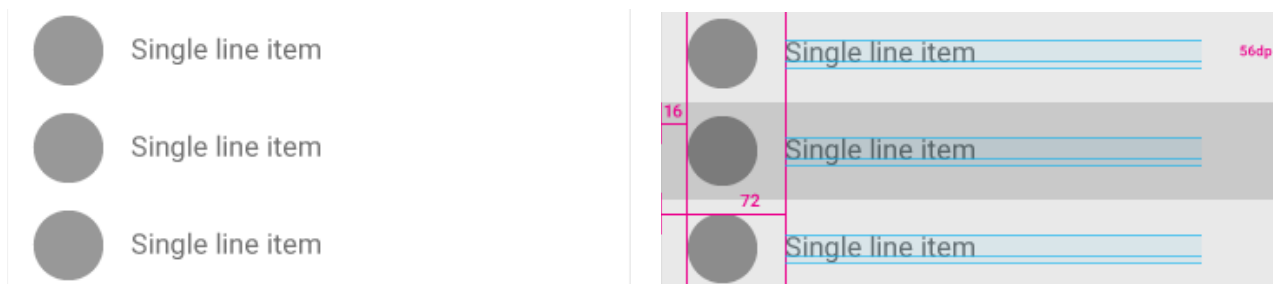
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距



带有文本的头像

字体：Roboto Regular 16sp，瓦片高：56dp，左边头像内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：20dp

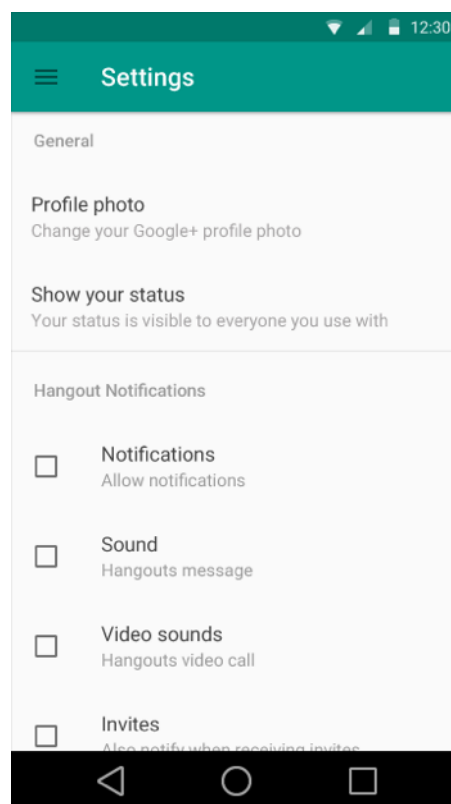
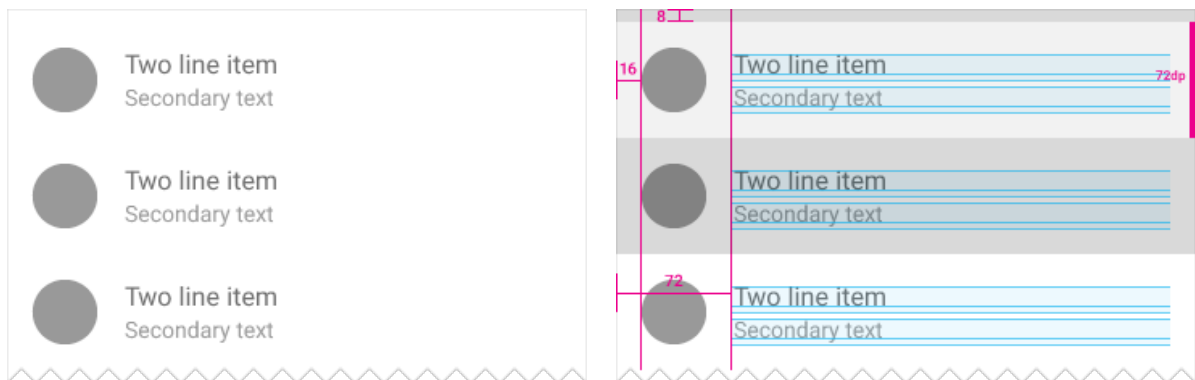
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



两行列表

仅有文本，主要文本的字体：Roboto Regular 16sp，次要文本的字体：Roboto Regular 14sp，瓦片高：72dp，文本内边距：16dp

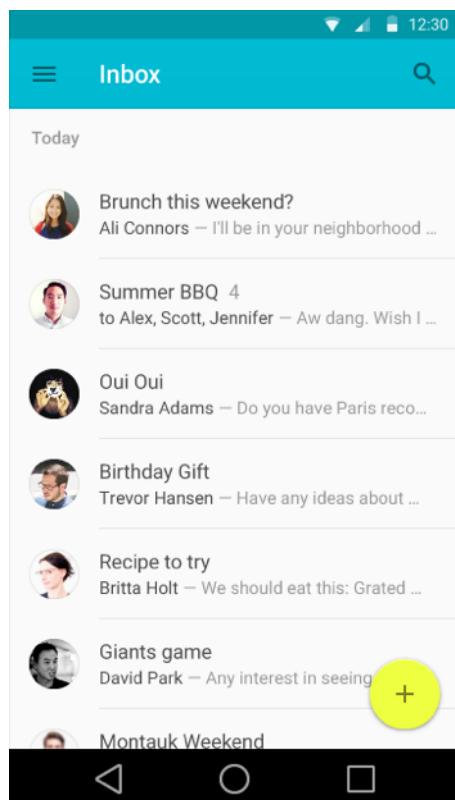
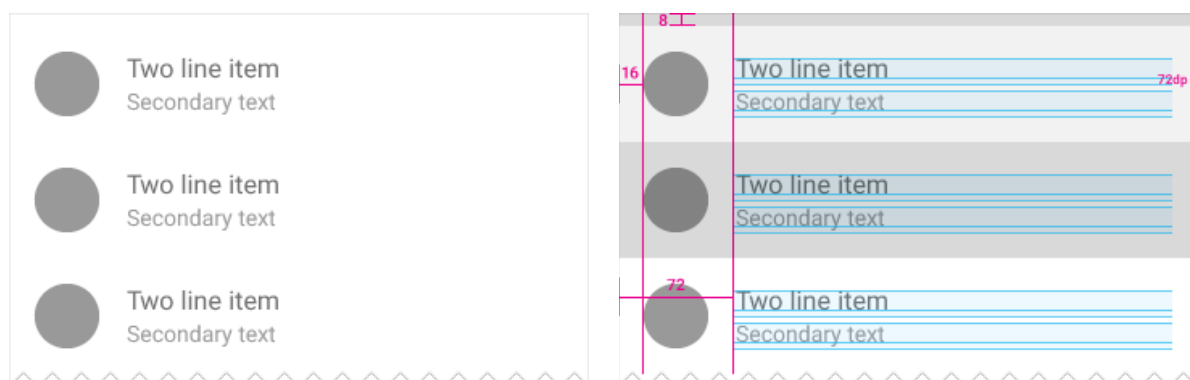
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



带有文本的头像

主要文本的字体：Roboto Regular 16sp，次要文本的字体：Roboto Regular 14sp，瓦片高：72dp，左边头像内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：16dp，图标和文本区域中心对齐。

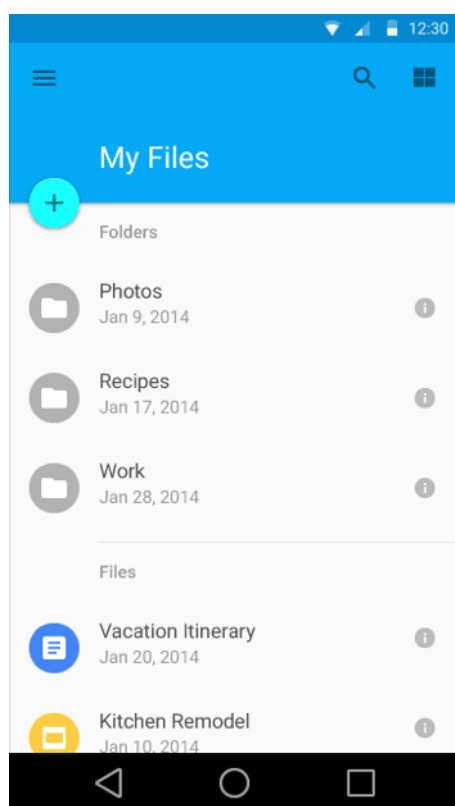
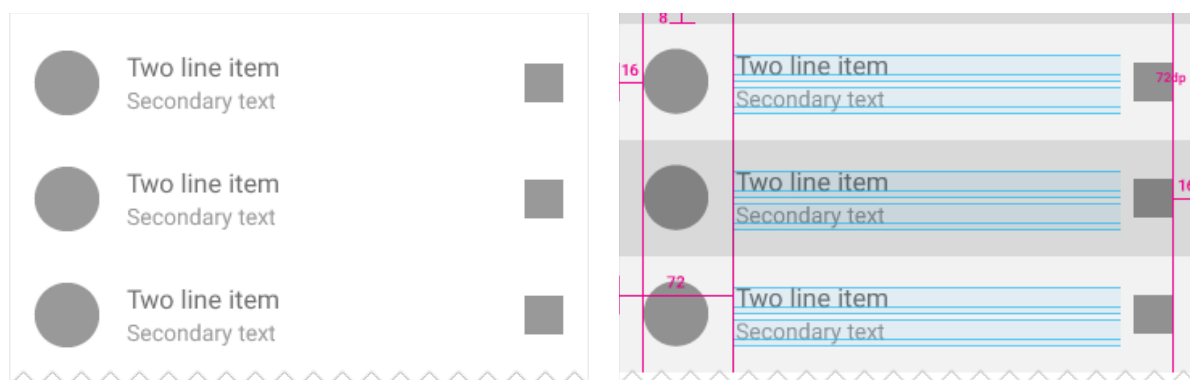
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



带有文本和图标的头像

主要文本的字体：Roboto Regular 16sp,次要文本的字体：Roboto Regular 14sp，瓦片高：72dp，左边头像内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：16dp，右边图标内边距：16dp

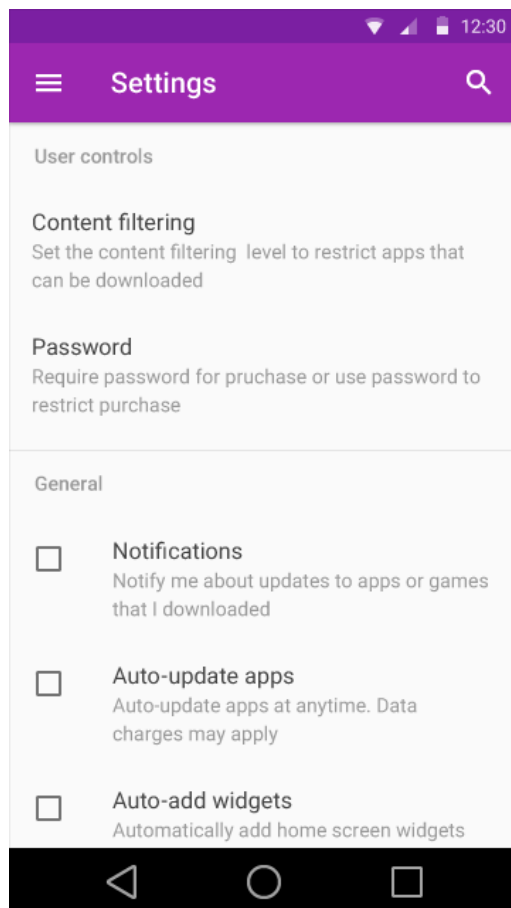
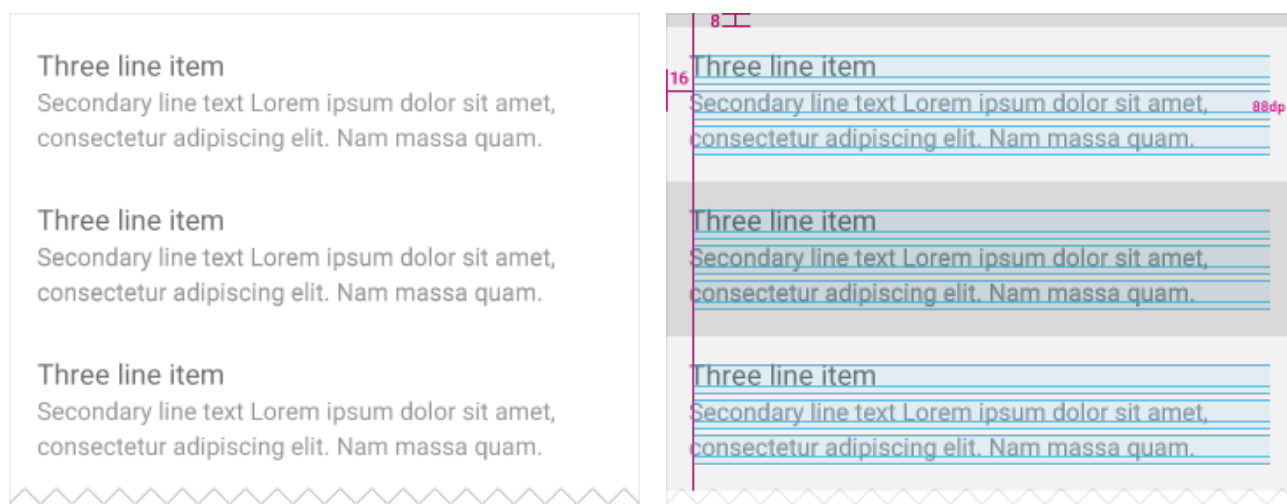
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



三行列表

仅有文本，主要文本的字体：Roboto Regular 16sp，次要文本的字体：Roboto Regular 14sp，瓦片高：88dp，文本内边距：16dp

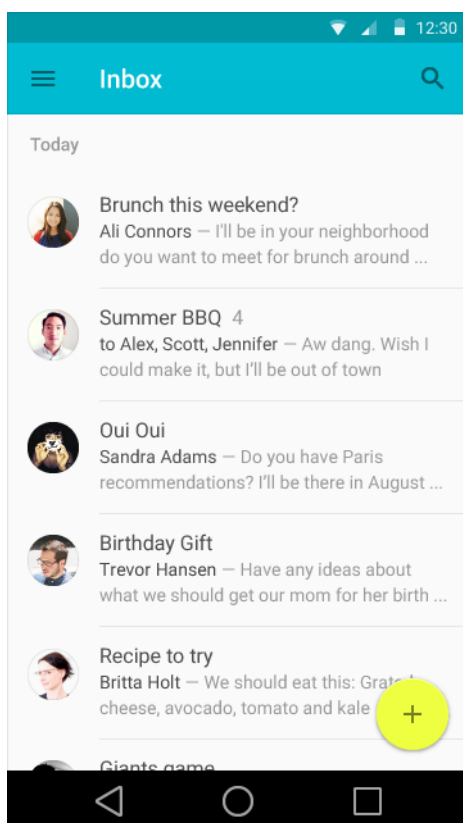
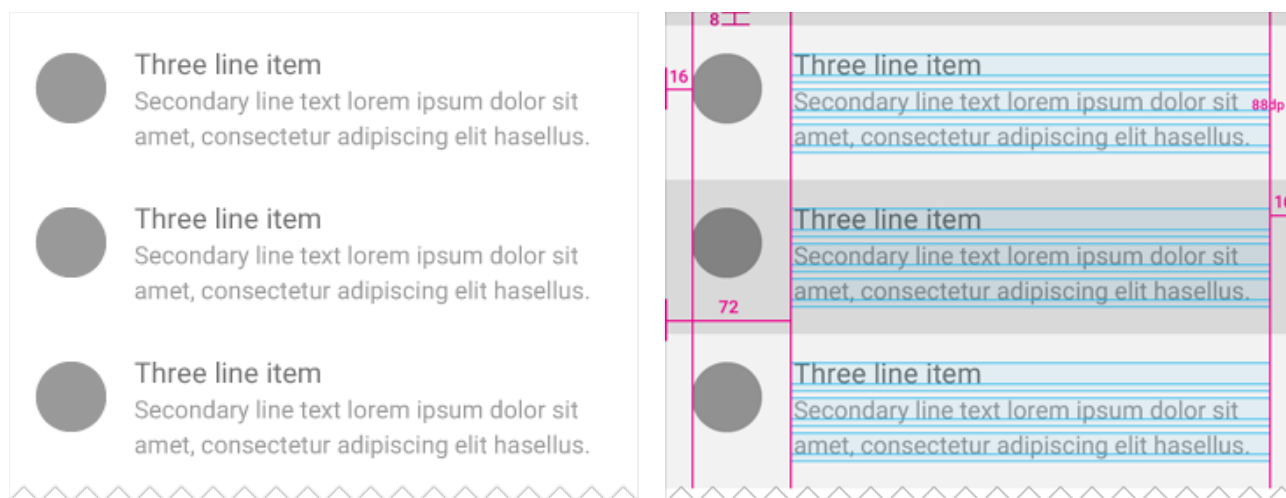
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



带有文本的头像

主要文本的字体：Roboto Regular 16sp，次要文本的字体：Roboto Regular 14sp，瓦片高：88dp，左边头像内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：16dp，头像和主要文本上部对齐。

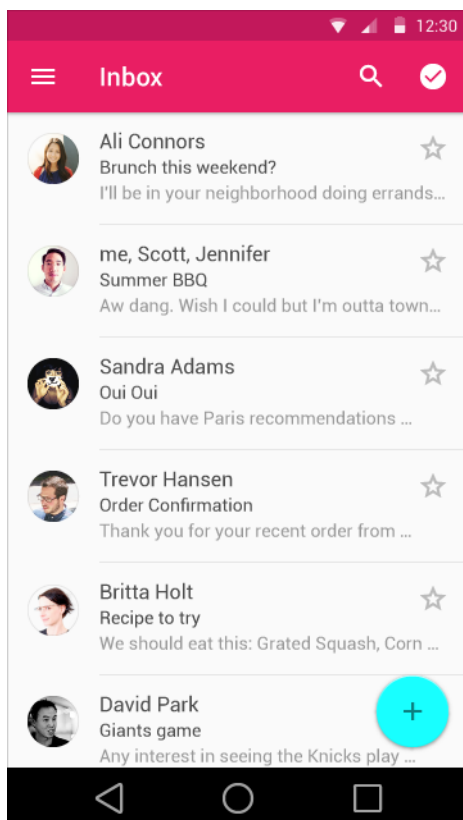
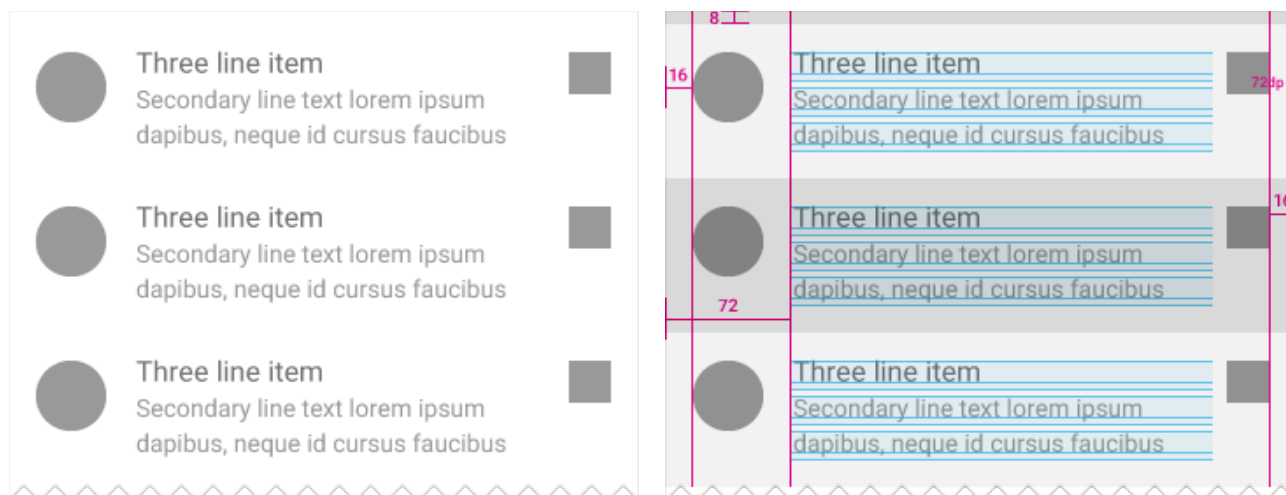
在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



带有文本和图标的头像

主要文本的字体：Roboto Regular 16sp，次要文本的字体：Roboto Regular 14sp，瓦片高：88dp，左边头像内边距：16dp，文本左内边距：72dp，文本上部和下部内边距：16dp，右边图标内边距：16dp，头像和图标与主要文本上部对齐。

在列表的上部和下部添加8dp的内边距。一个特例是在带有表头条目（subheader）的列表上部，因为表头条目包含了它们自己的内边距。



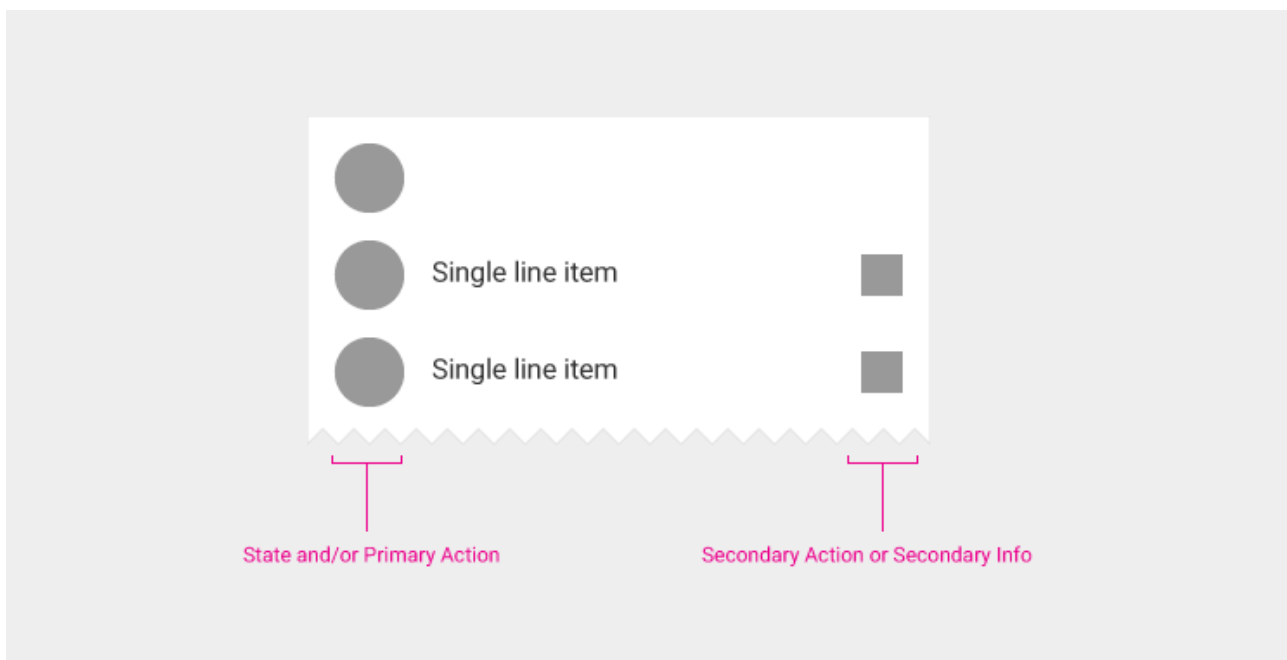
5.9 列表控制

用法

列表控制被归类为下面四种:

- 状态
- 主操作(包括文本字符)
- 次要操作
- 次要信息

分辨列表标题的元素，我们首先关注的是阅读顺序，所以，状态和主动作放在标题列表的



左边。列表里面的文本内容在这里也被认为是主要操作的一部分。

不要把两个展示图标和操作图标放在一起，比如复选框和头像

如果列表的主动作是做引导作用的，那么就不用icon。列表它本身以及它的上下文就已经可以让用户明白这个列表的用处是什么。

次要操作以及信息应该放在标题的右边，次要动作通常都是和主要操作分开的，因为越来越多的用户希望每个图标都能触发一个动作。

列表控制的类型

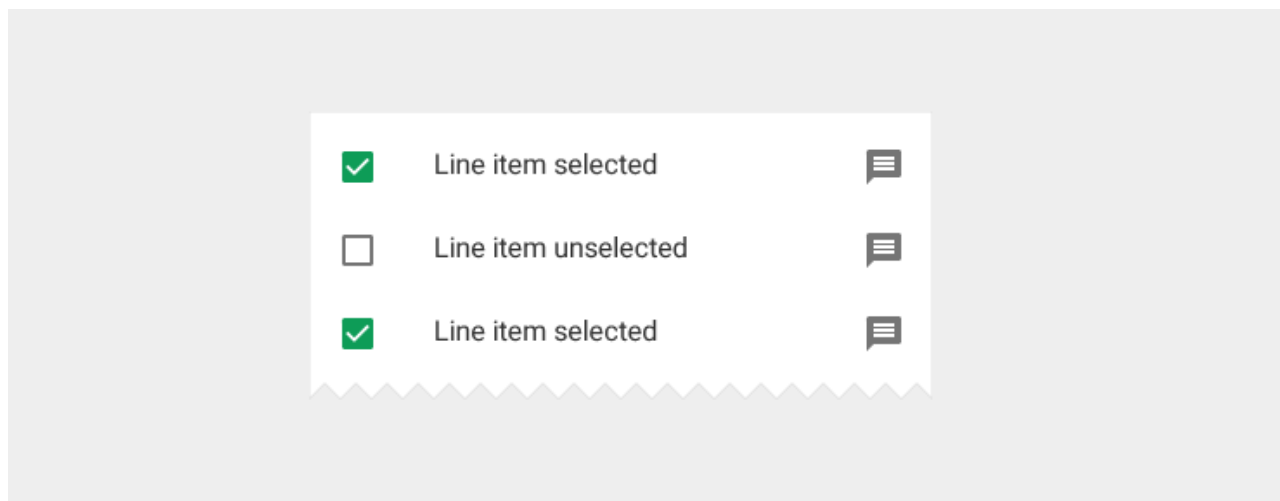
复选框(Checkbox)

复选框既可以被定义成是主动作也可以是附属动作。

类型: 主动作/状态

分割号 (与次要操作分离)

桌面上只悬停

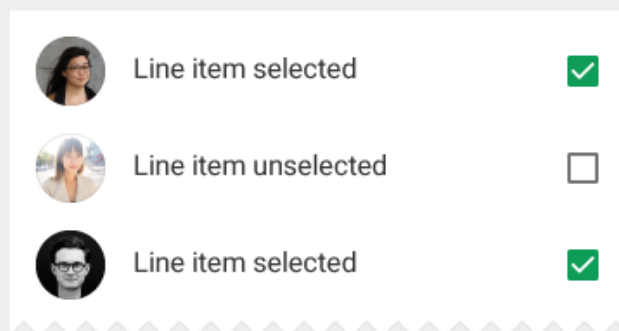


在这个列表中，复选框是主要操作和状态标志符

类型: 次要操作

分割号 (与主要操作分离)

当控制一系列的变量而不仅仅只是控制一个变量的情况下，考虑使用开关控件来替换。



在这个列表中，复选框是次要操作

开关

类型: 次要操作

分割号（与主要操作分离）

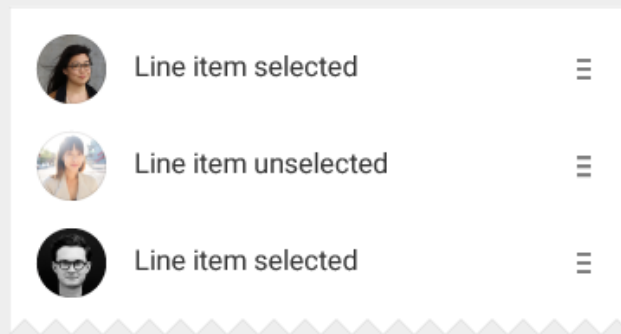


重新排序

类型: 附属动作

通常它都是独立的，依赖于当前列表所处的模式。

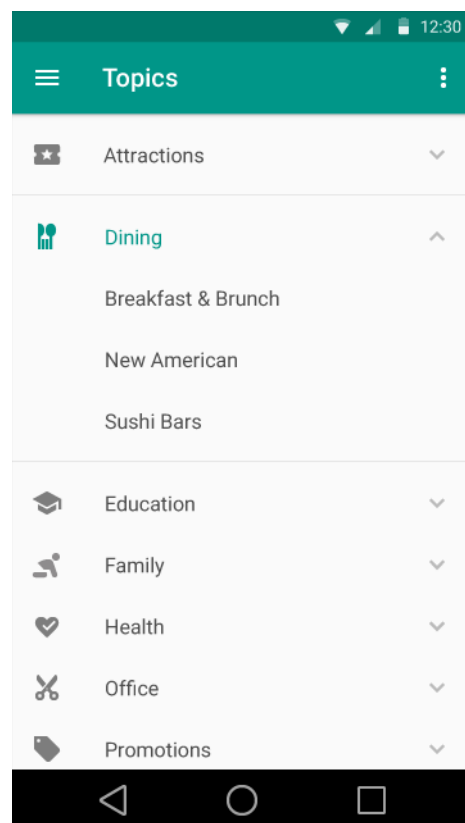
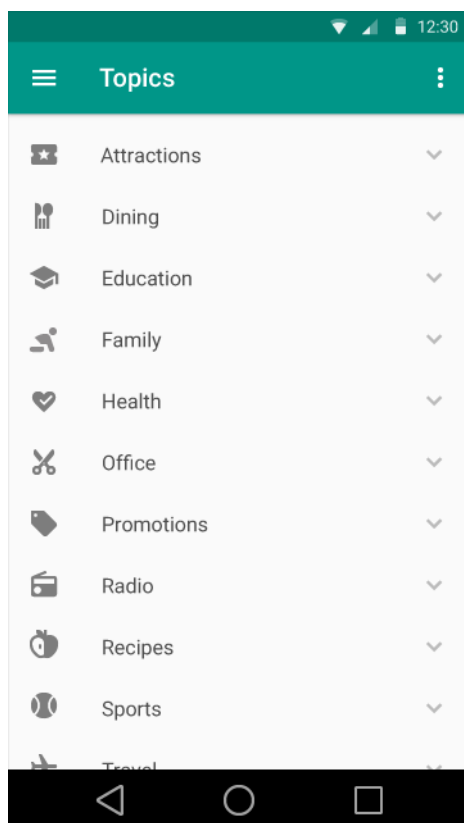
这个动作允许用户把一个列表项拖到列表的其他位置。通常，这个按钮是在编辑的模式下会出现。

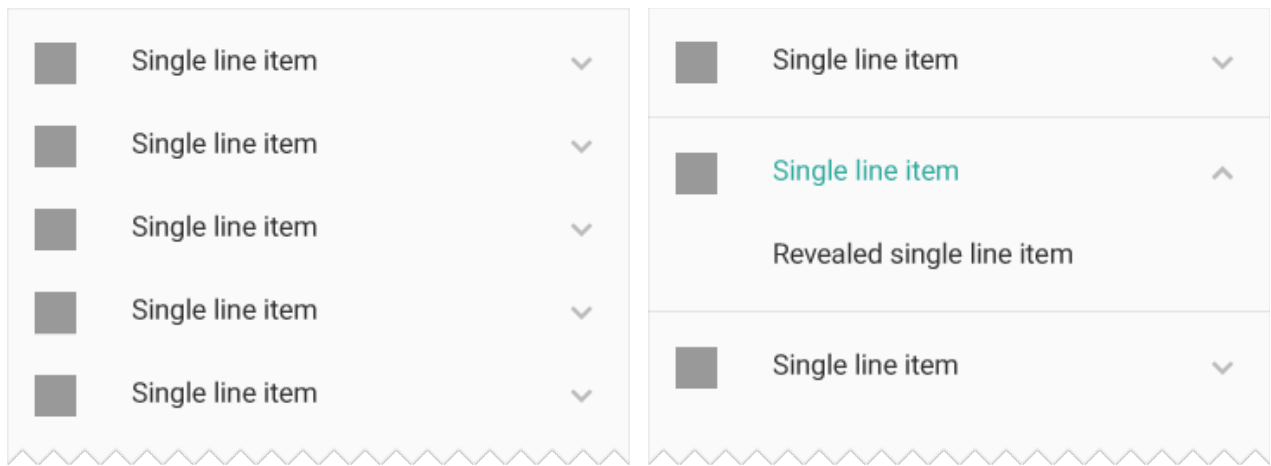


展开/折叠

类型: 附属动作

垂直展开或者折叠列表来显示或者隐藏当前列表



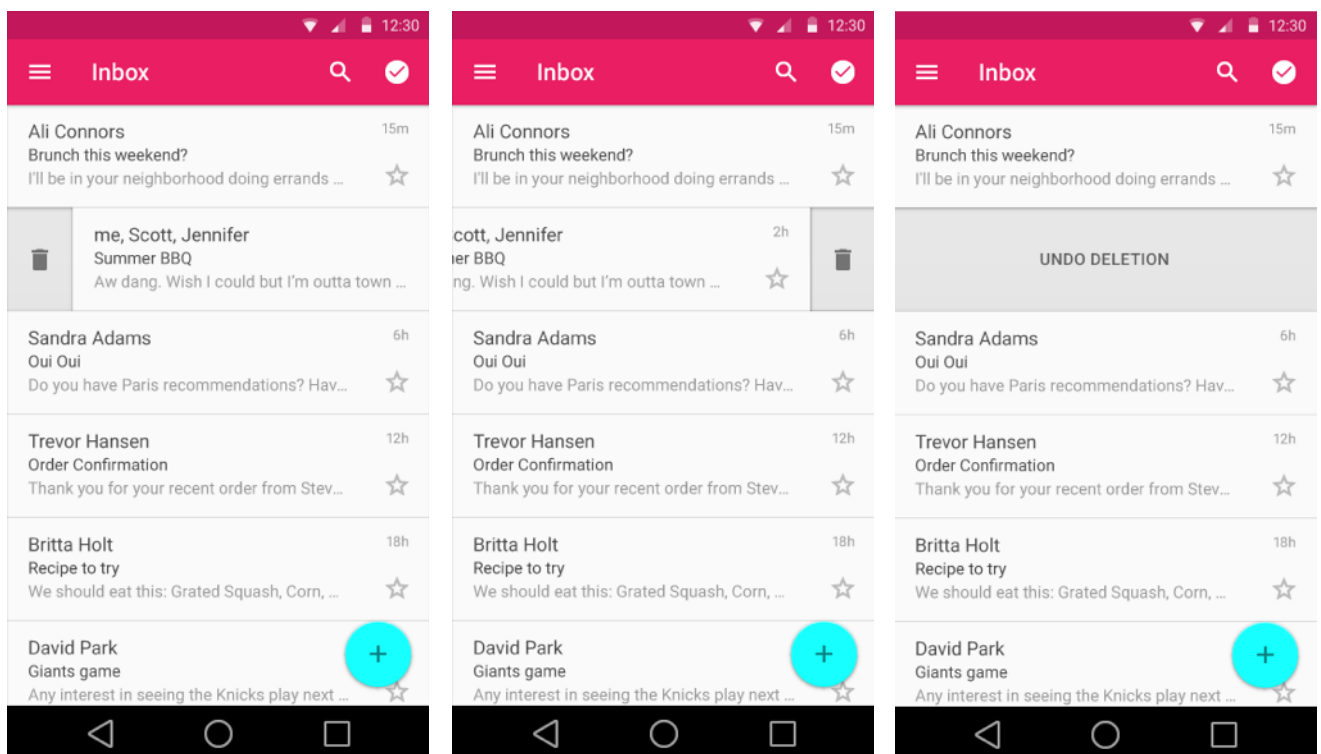


底层

类型: 其他

底层是在当某一项列表被滑开之后的操作提示。Leave-behind可以被转换成一项操作。

无论从哪个方向滑动列表，都会出现操作图标。滑动了之后，操作图标就会出现列表空白处的中间。



不推荐的做法:导航列表控制

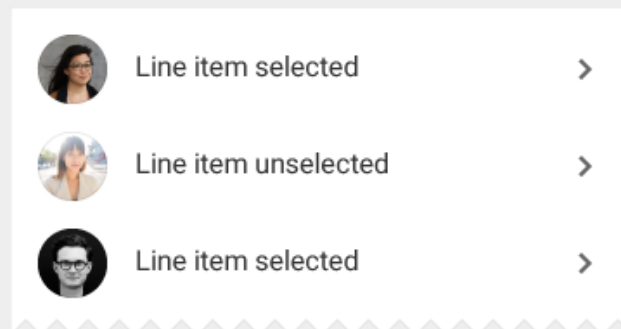
通常情况下，列表本身的内容就已经隐含了导航信息，因此，列表里面就不需要额外的图标。

查看更多

类型: 主要操作（连同其他行）

不需要跟其他次要操作分离

点击之后跳转到与当前列表相关详细信息的页面，通常这都是一个新的页面或者面板。



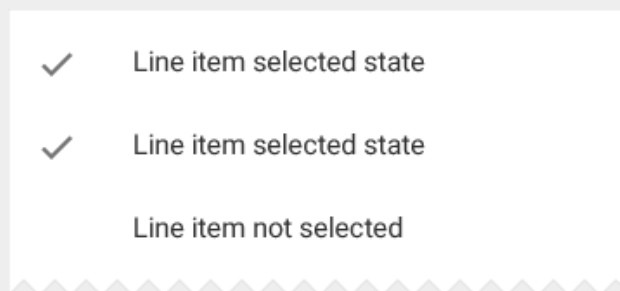
菜单控制的类型

选中

类型: 状态

不需要跟其他状态分离

仅适用于菜单。用来表示当前列表是否通过不同的操作之后被选中。

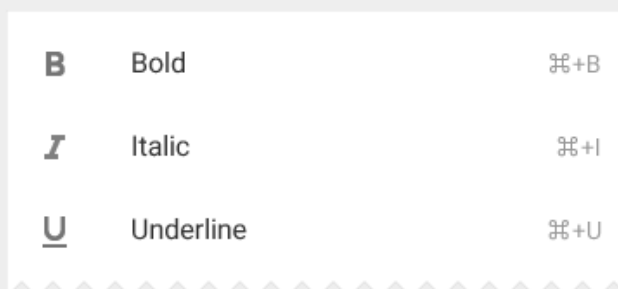


内联信息

类型: 次要信息

不需要跟其他信息分离

仅适用于菜单。内联信息是列表中一小段文字用来提供当前标题相关的信息或者提示，比如快捷键。它是不能被删除。



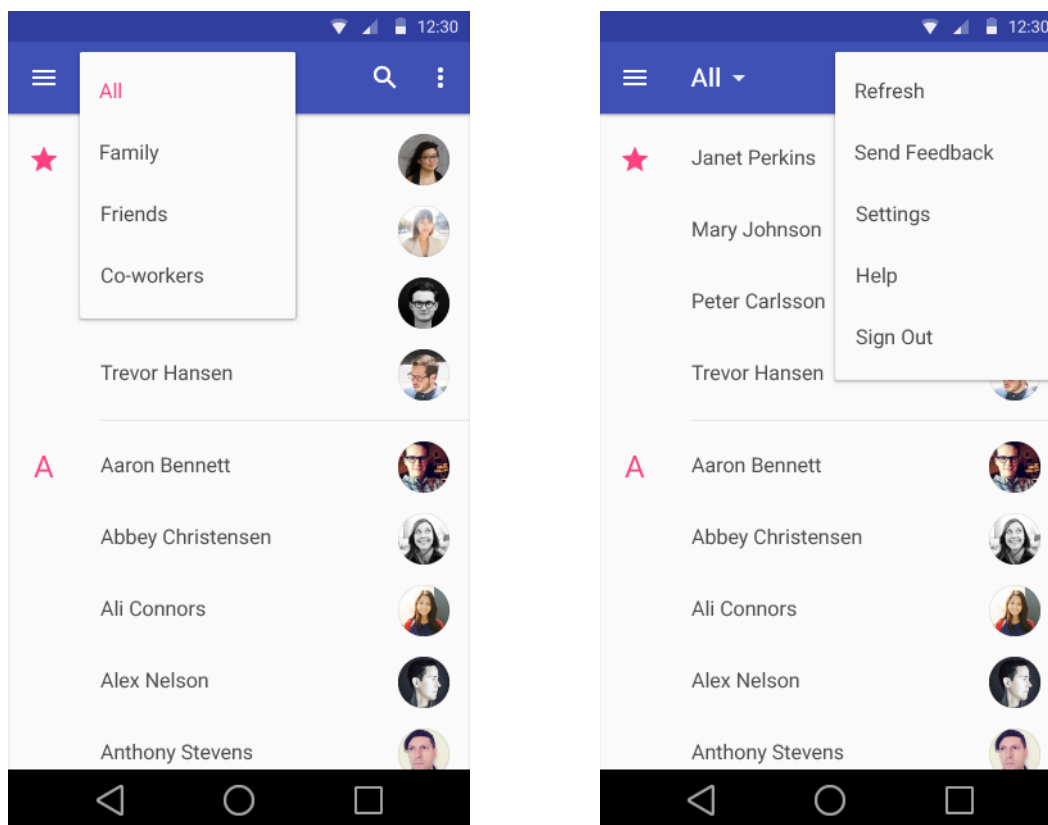
5.10 菜单

用法

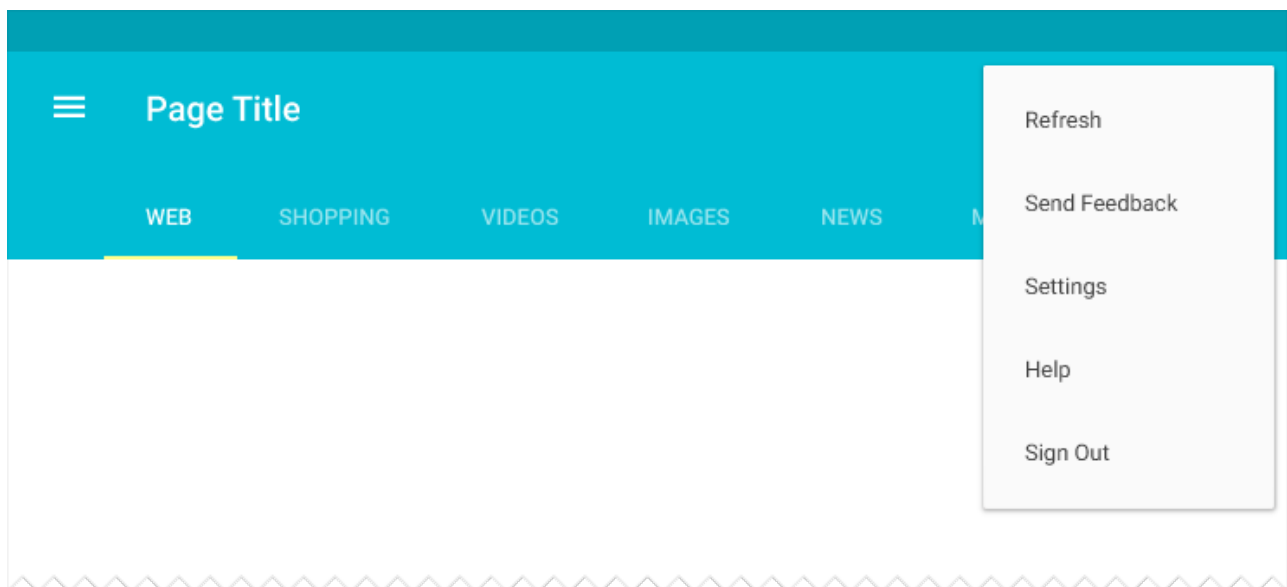
菜单是临时的一张纸（paper），由按钮（button）、动作（action）、点（pointer）或者包含至少两个菜单项的其他控件触发。

每一个菜单项是一个离散的选项或者动作，并且能够影响到应用、视图或者视图中选中的按钮。

菜单不应该用作应用中主要的导航方法。



触发按钮或者控件的标签（label）可以简明准确的反映出菜单中包含的菜单项。菜单栏通常使用一个单词作为标签，像“文件”，“格式”，“编辑”和“视图”，然后其他内容或许有更冗长的标签。

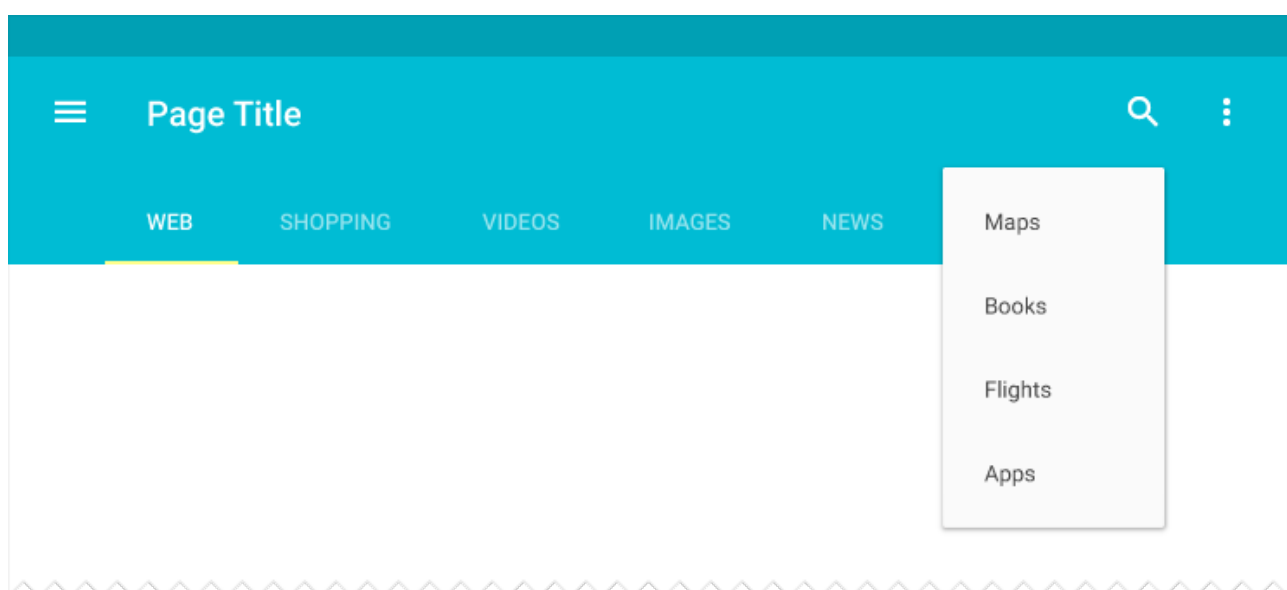


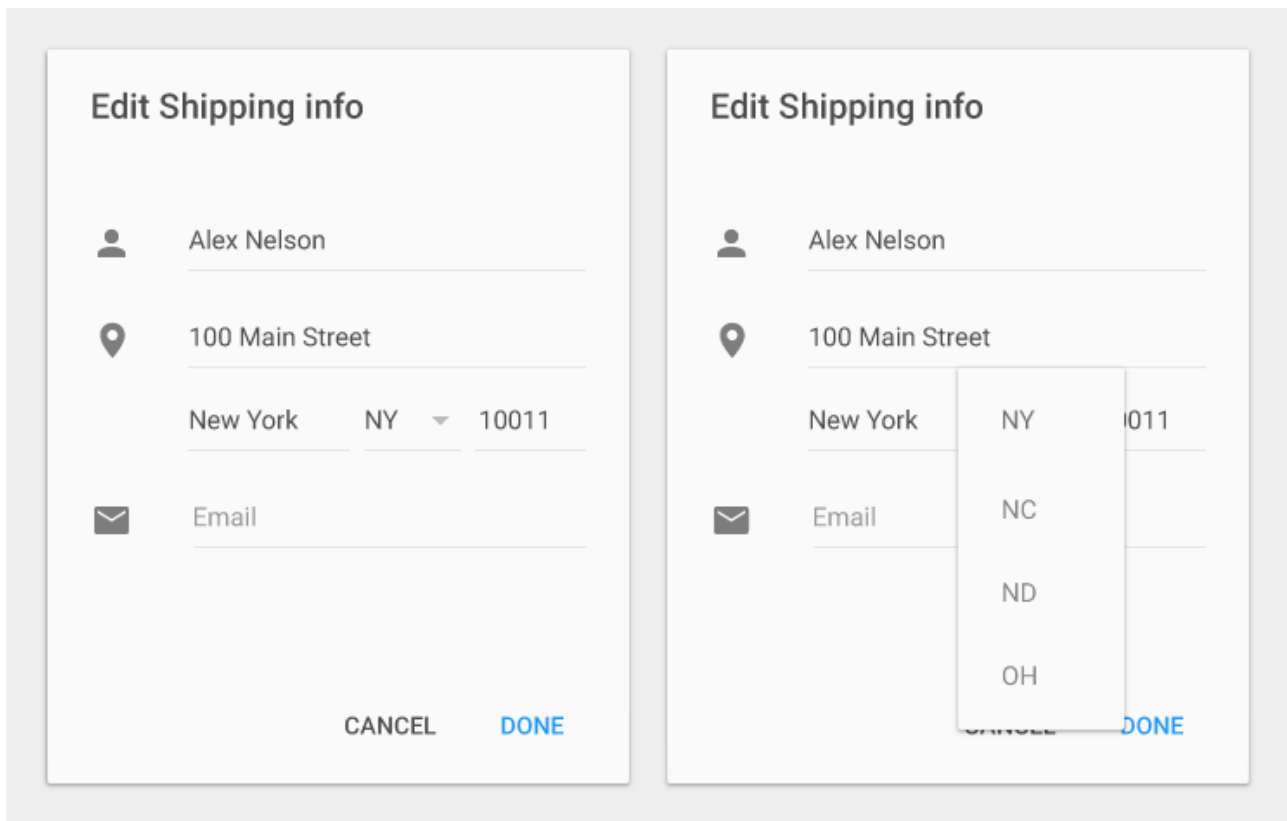
菜单显示一组一致的菜单项，每个菜单项可以基于应用的当前状态来使能。

上下文菜单（Contextual menus）能够基于应用的当前状态动态的改变菜单项的可用性和让菜单项使能。

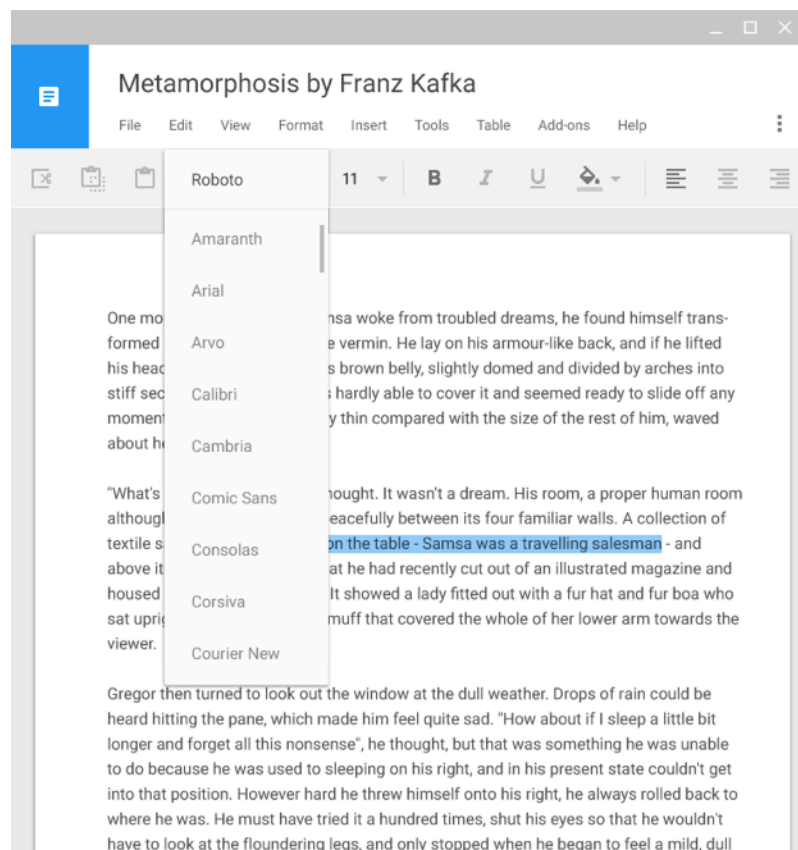
通常，移除与当前上下文不相干的菜单项，禁用那些需要满足特定条件才能使用的相关的菜单项（比如，当文本选中后“复制”变得可用）。

特定的应用状态可能使得上下文菜单只包含一个菜单项。比如，当在网页上高亮文本时，Android只显示“复制”，所以用户不能使用“剪切”和“粘贴”文本。



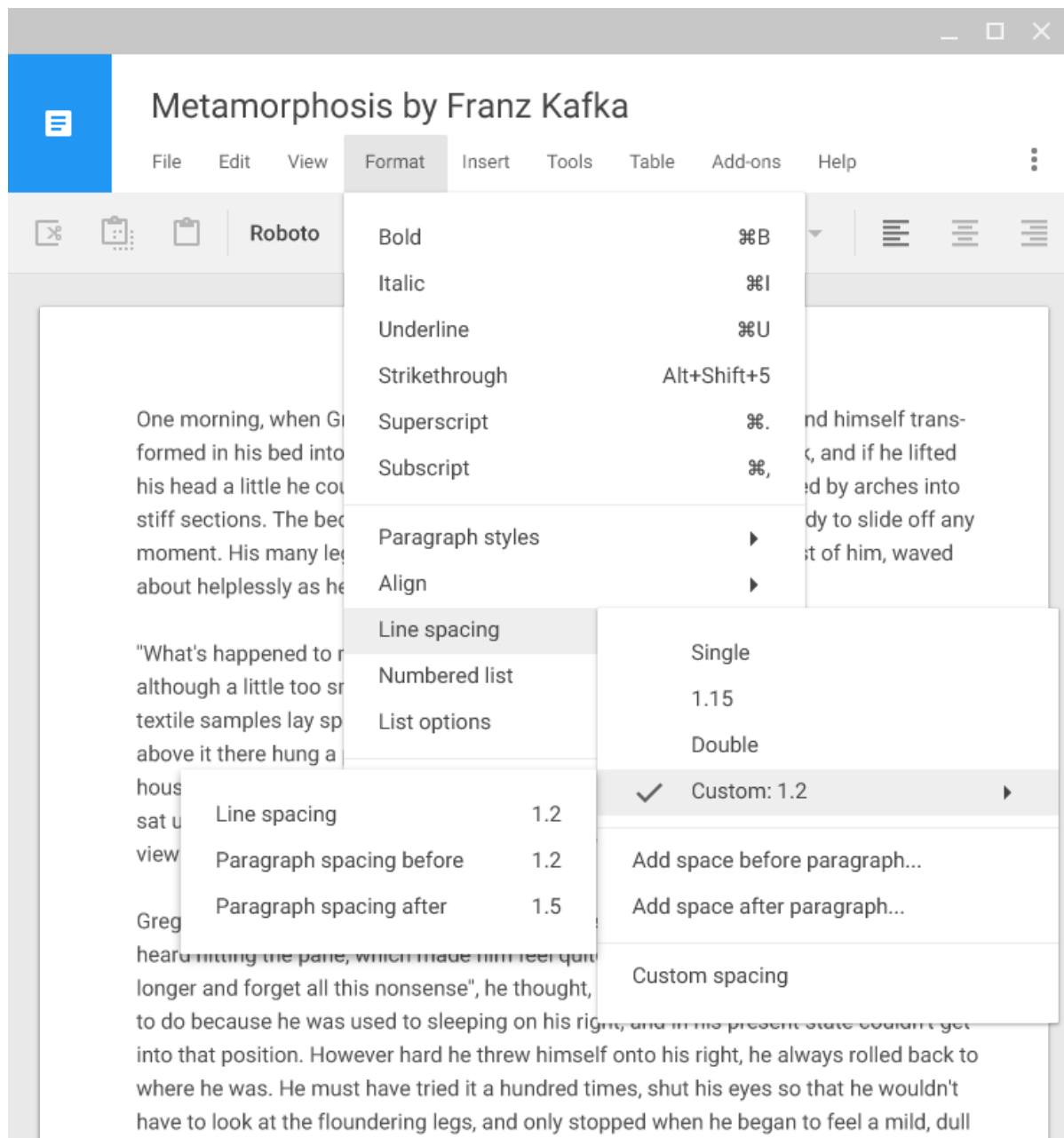


菜单依赖于它们距屏幕边的距离垂直和水平的调整位置

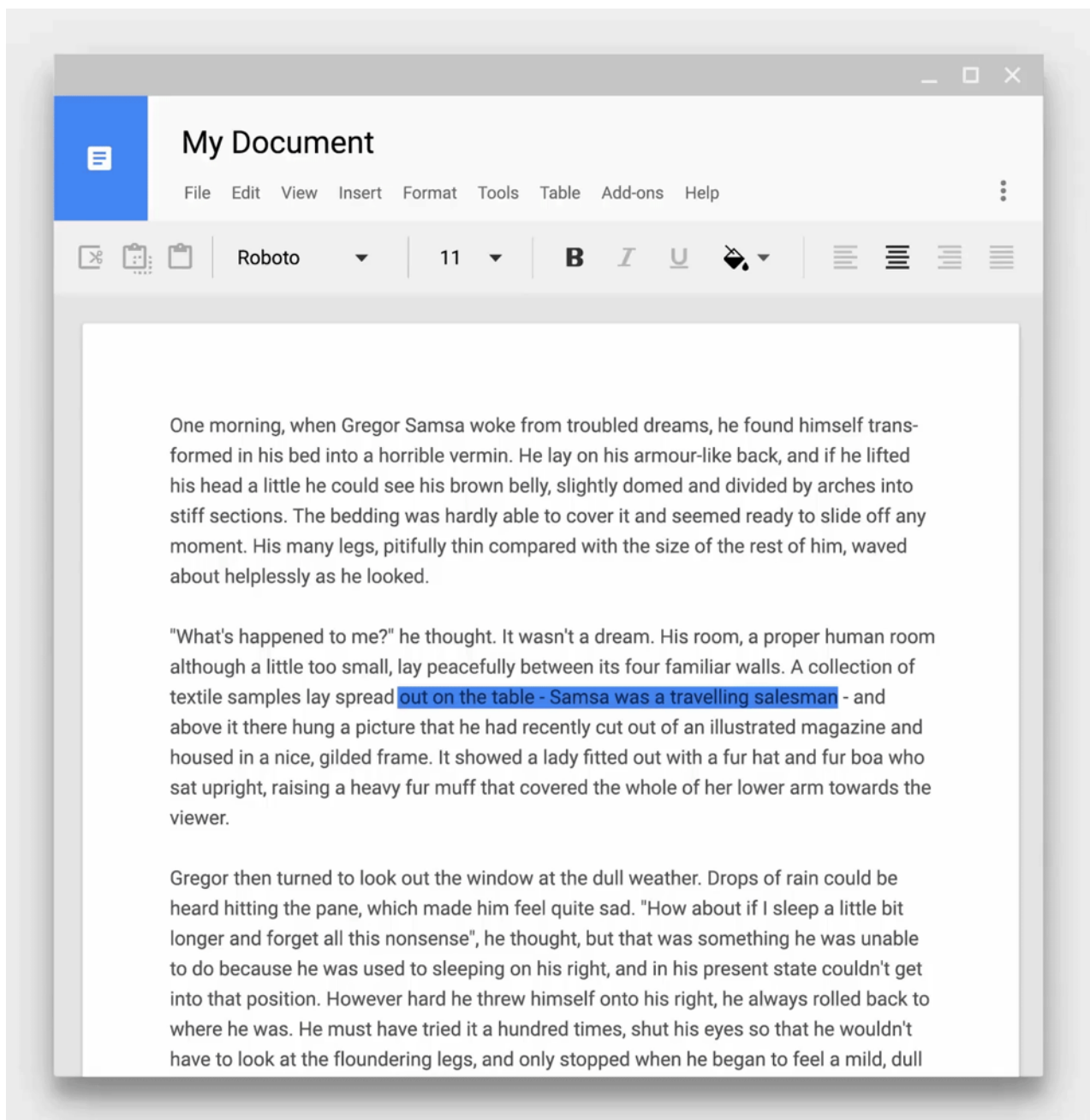


如果菜单的高度使得菜单项不能完全显示，那么菜单会支持内部滚动。一个典型的例子是在手机横屏状态下查看菜单

菜单可以是级联的





下面这些动画演示了菜单的滚动和级联。




下拉

Edit Shipping info

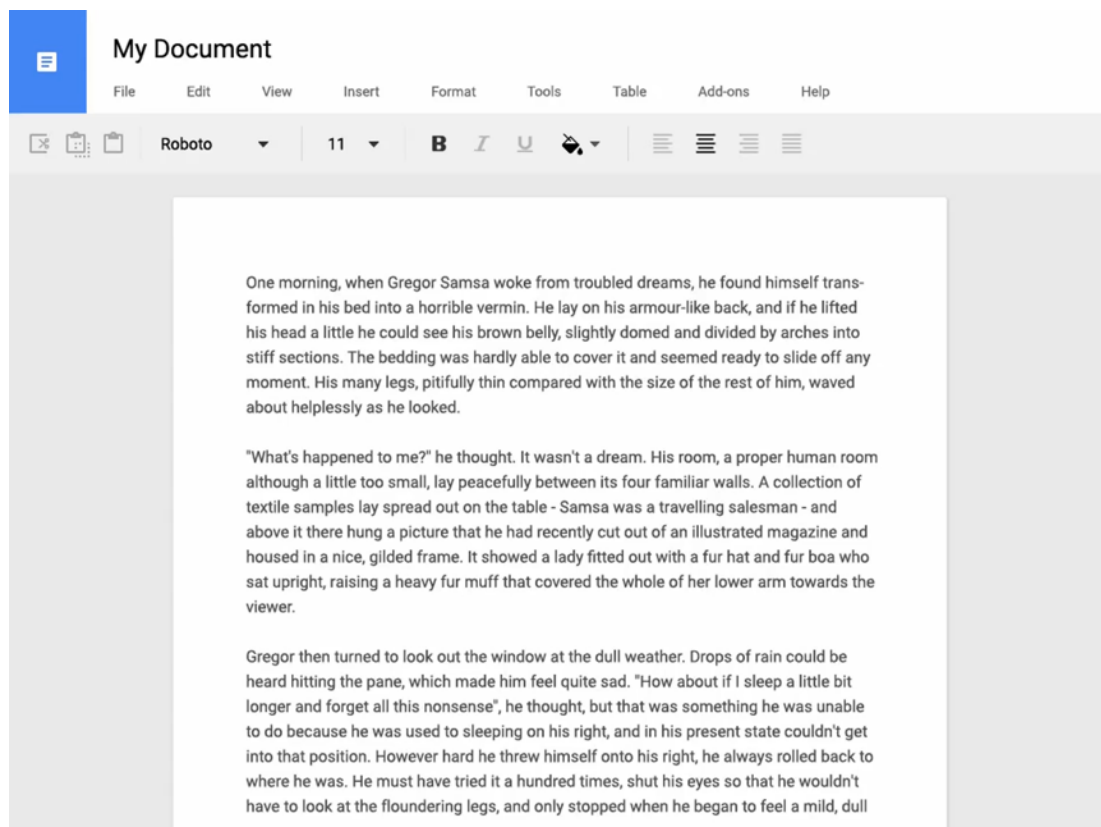
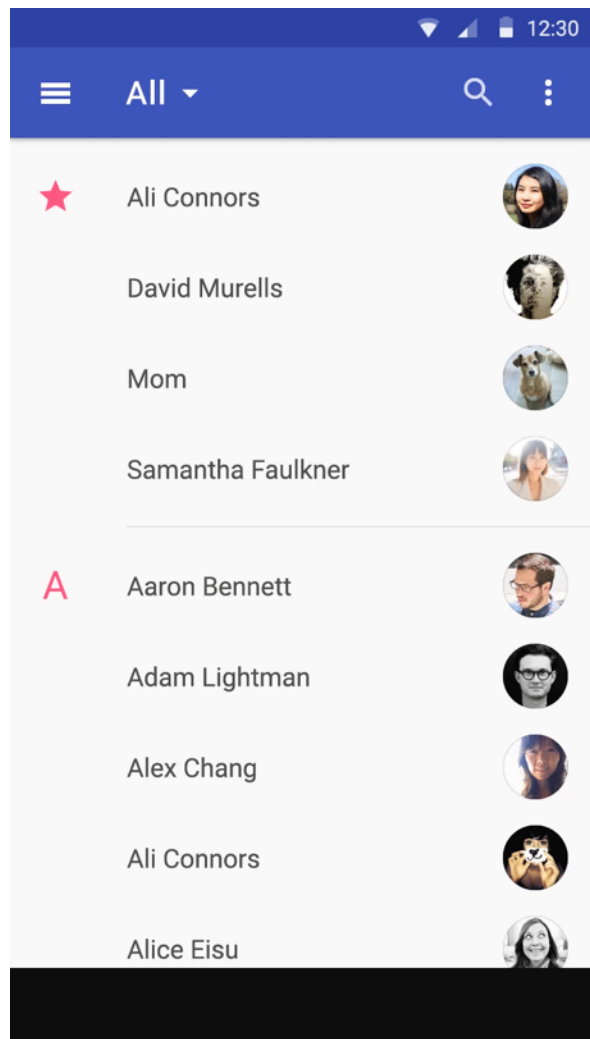

Ali Connors


2000 Main Street

Hoboken
FL
94122


Email

CANCEL
DONE



菜单项

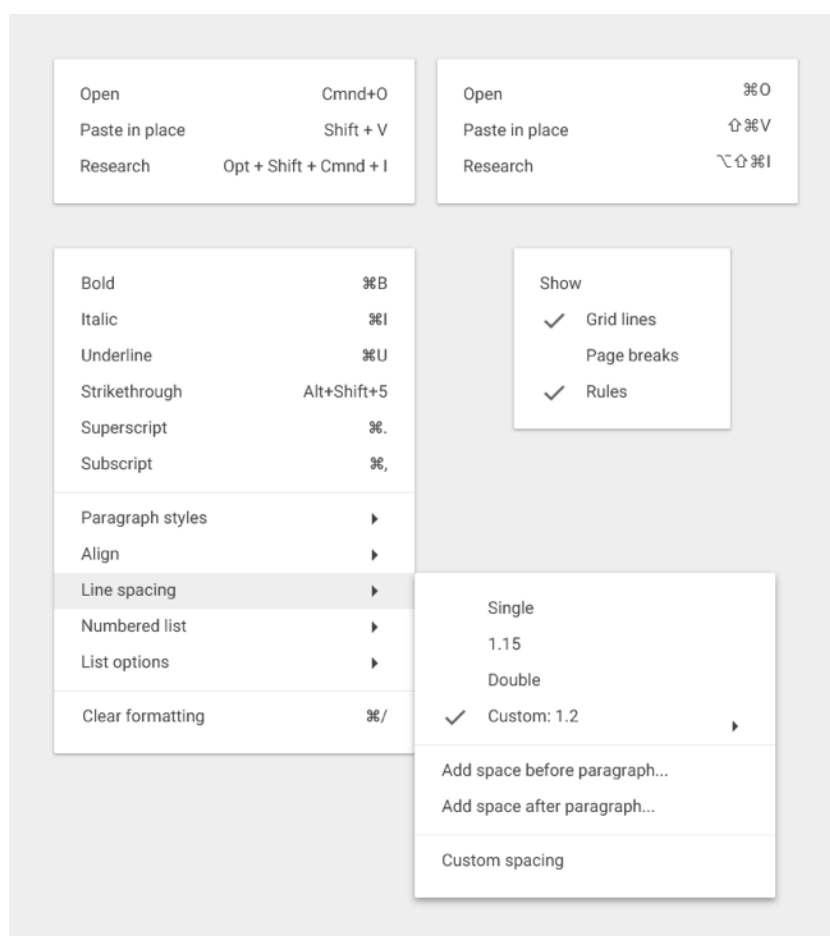
每一个菜单项限制为单行文本，并且能够说明在菜单项选中时所发生的动作。

菜单项的文本一般是单个单词或者短语，但是也可能包含了图标和帮助文本，比如快捷方式，同时也可包含像复选标记之类的控件来标识多选条目或状态。可以参考列表控件。

带有静态文本的菜单应当在菜单的上部放置最常使用的菜单项。

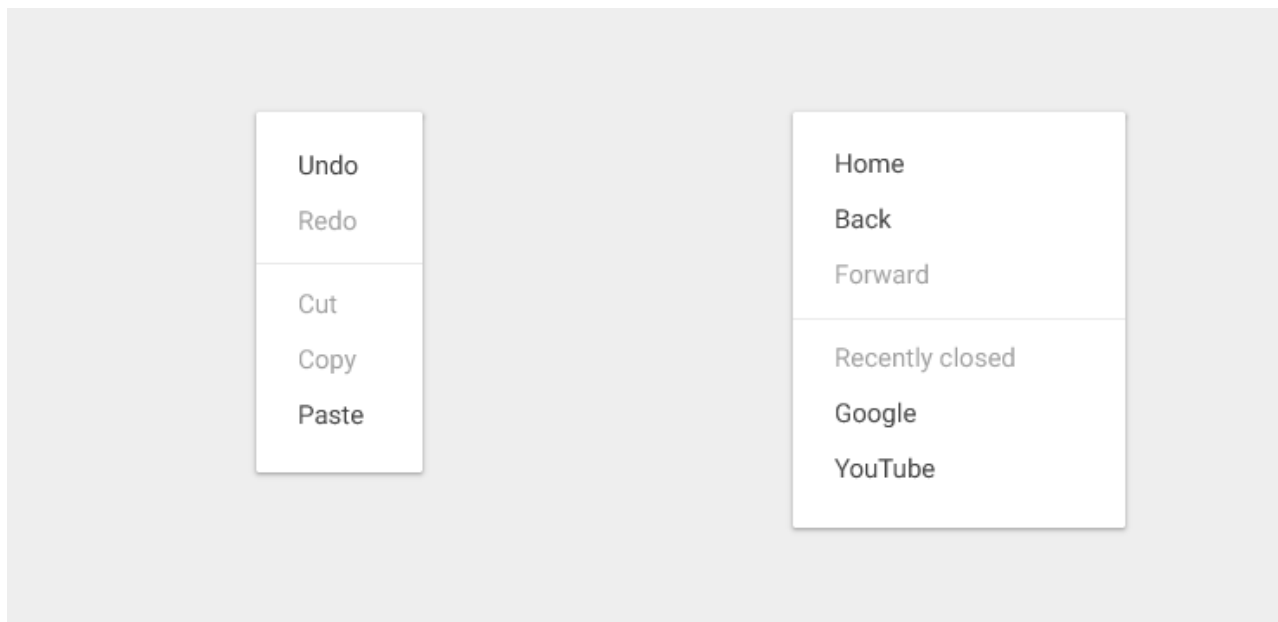
带有动态文本的菜单可能具有其它行为，比如在菜单上部放置预先使用的字体。顺序可以根据用户的动作而改变。

菜单项可以内嵌自己的子菜单。尝试着将菜单层级限制在一级，因为导航多级内嵌子菜单是困难的



将动作菜单项显示为禁用状态，而不是移除它们，这样可以让用户知道在正确条件下它们是存在的。

比如，当没有重做任务时禁用重做（Redo）动作。当内容被选中后，剪切（Cut）和复制（Copy）动作可用

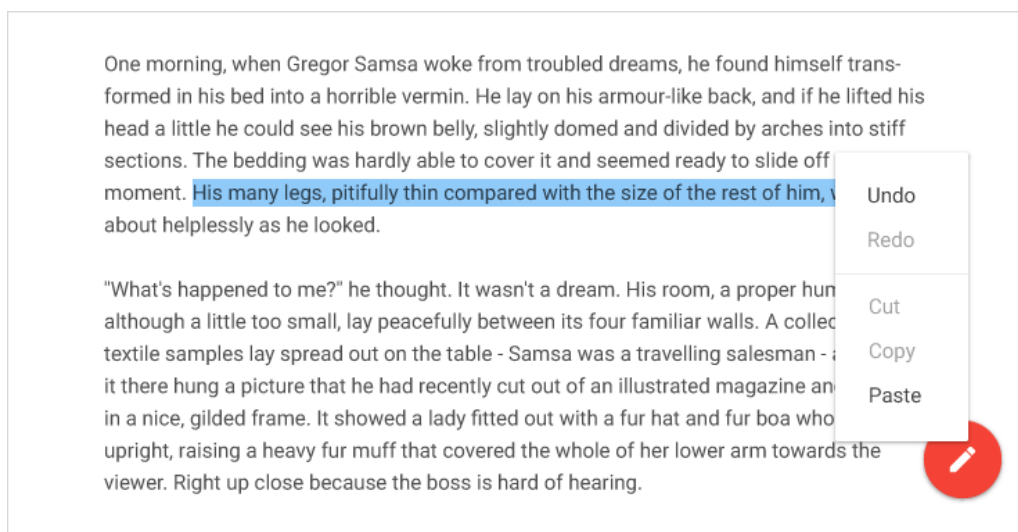


行为

菜单出现在所有的应用内部的UI元素之上。

通过点击菜单以外的部分可以让菜单消失，或者点击触发按钮（如果按钮可见）。

通常，选中一个菜单项后也会让菜单消失。一个特例是当菜单允许多选时，比如使用复选标记。

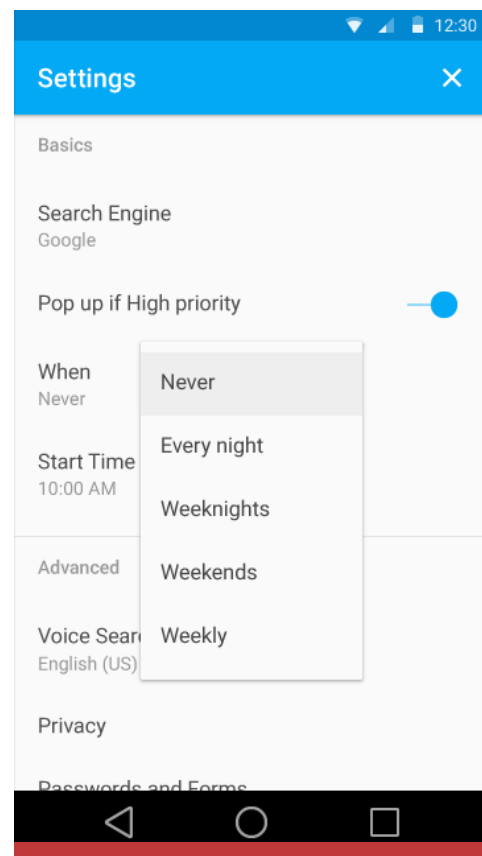
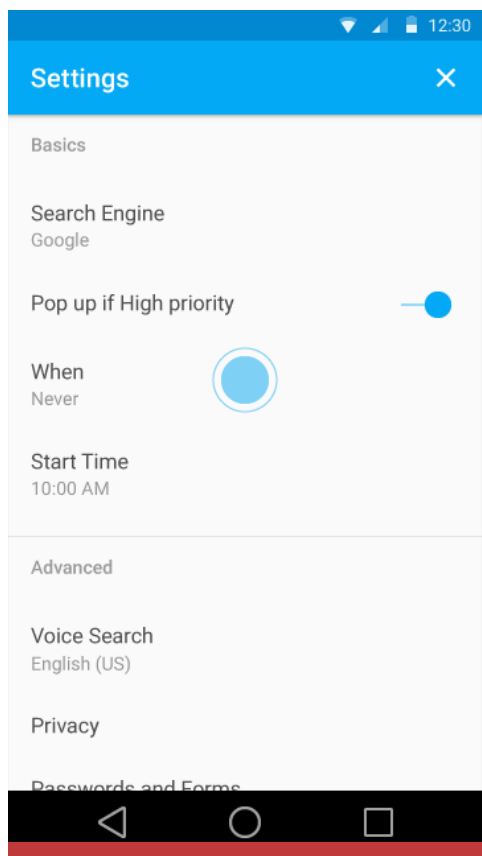
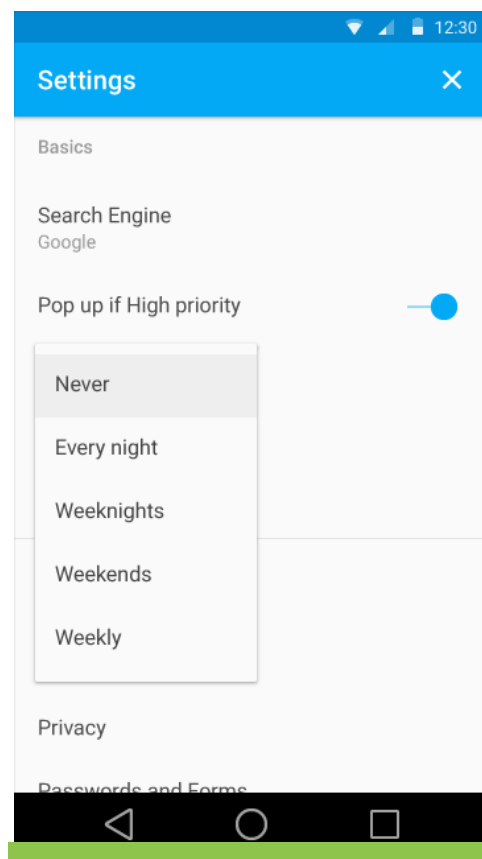
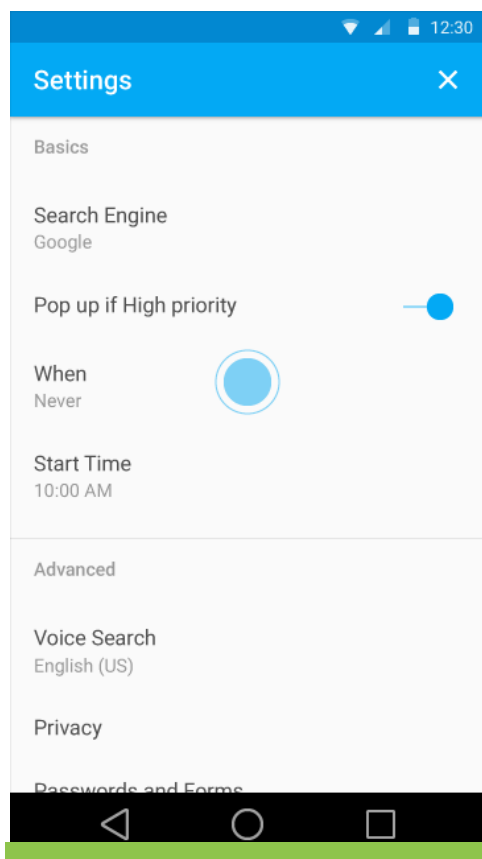


菜单显示在触发它的元素处，比如当前选中的菜单项现在在触发元素的顶部。
不要显示选中的菜单项的副本。

The image shows two side-by-side 'Edit Shipping info' forms. Both forms have a title 'Edit Shipping info' and a user profile icon with the name 'Alex Nelson'. Below the name is a location pin icon with the address '100 Main Street'. Underneath the address is a state dropdown menu with 'New York' selected, followed by 'NY' and a small downward arrow. To the right of the state dropdown is a zip code field with '10011'. At the bottom of the form is an email field with an envelope icon and the label 'Email'. At the very bottom are two buttons: 'CANCEL' and 'DONE'. In the left form, the state dropdown menu is open, showing a list of states: 'NY', 'NC', 'ND', and 'OH'. In the right form, the state dropdown menu is also open, showing the same list of states, but the selected item 'NY' is not visible in the menu list.

The image shows two side-by-side 'Edit Shipping info' forms. Both forms have a title 'Edit Shipping info' and a user profile icon with the name 'Alex Nelson'. Below the name is a location pin icon with the address '100 Main Street'. Underneath the address is a state dropdown menu with 'New York' selected, followed by 'NY' and a small downward arrow. To the right of the state dropdown is a zip code field with '10011'. At the bottom of the form is an email field with an envelope icon and the label 'Email'. At the very bottom are two buttons: 'CANCEL' and 'DONE'. In the left form, the state dropdown menu is open, showing a list of states: 'NY', 'NC', and 'ND'. In the right form, the state dropdown menu is also open, showing the same list of states, but the selected item 'NY' is not visible in the menu list.

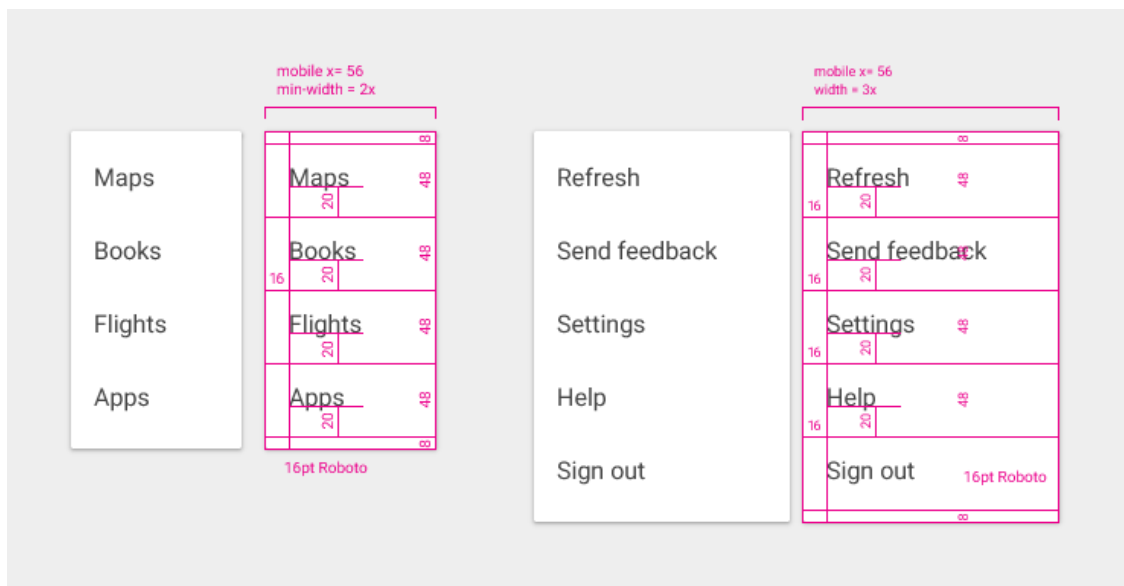
菜单不要与触摸的位置水平对齐



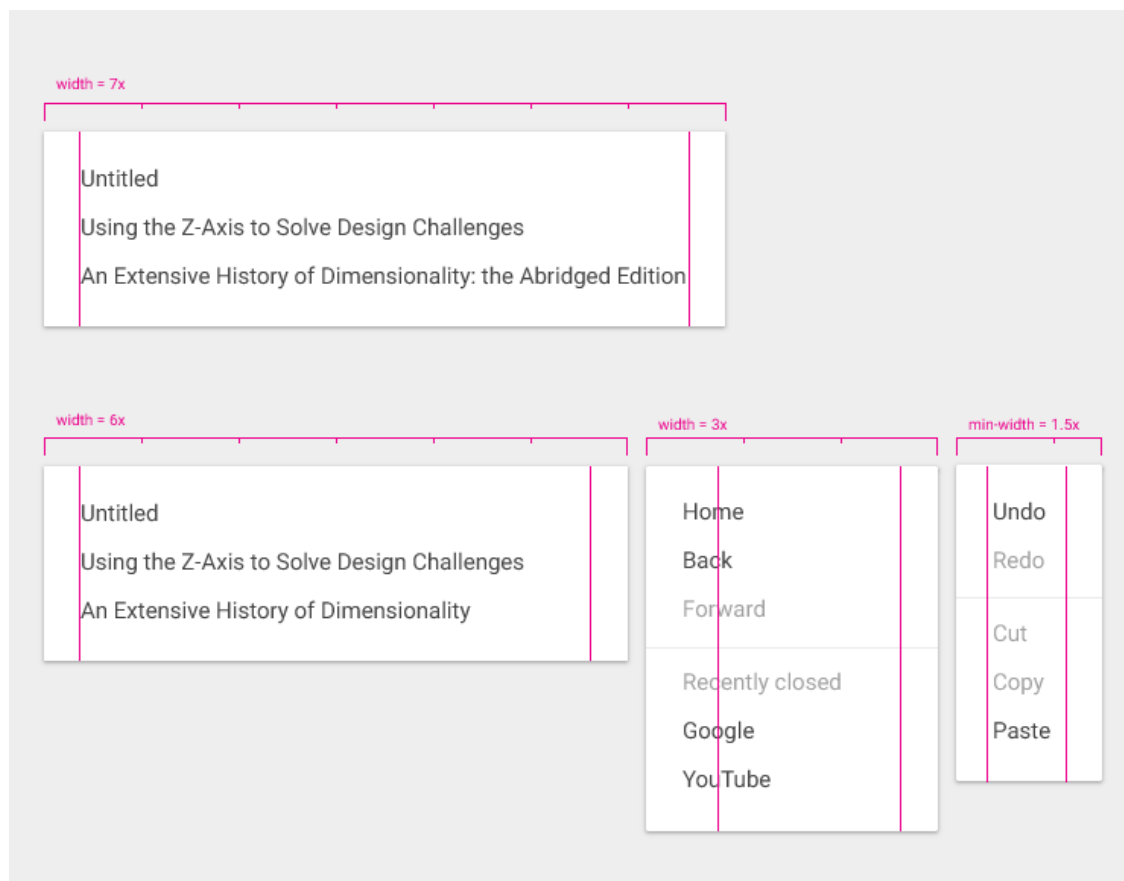
度量

度量提供了多样的尺寸和类型菜单以及不同的平台。

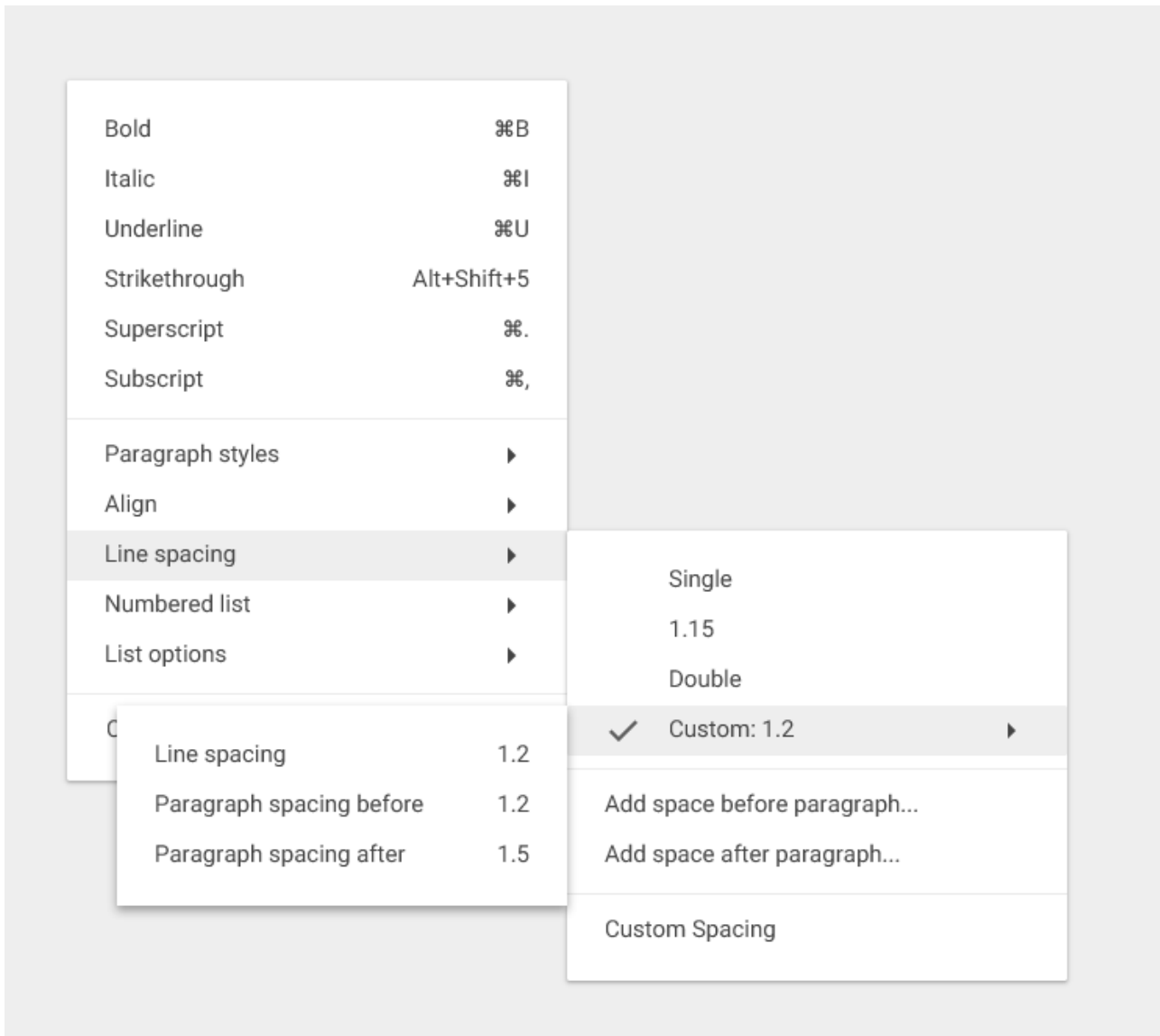
移动设备



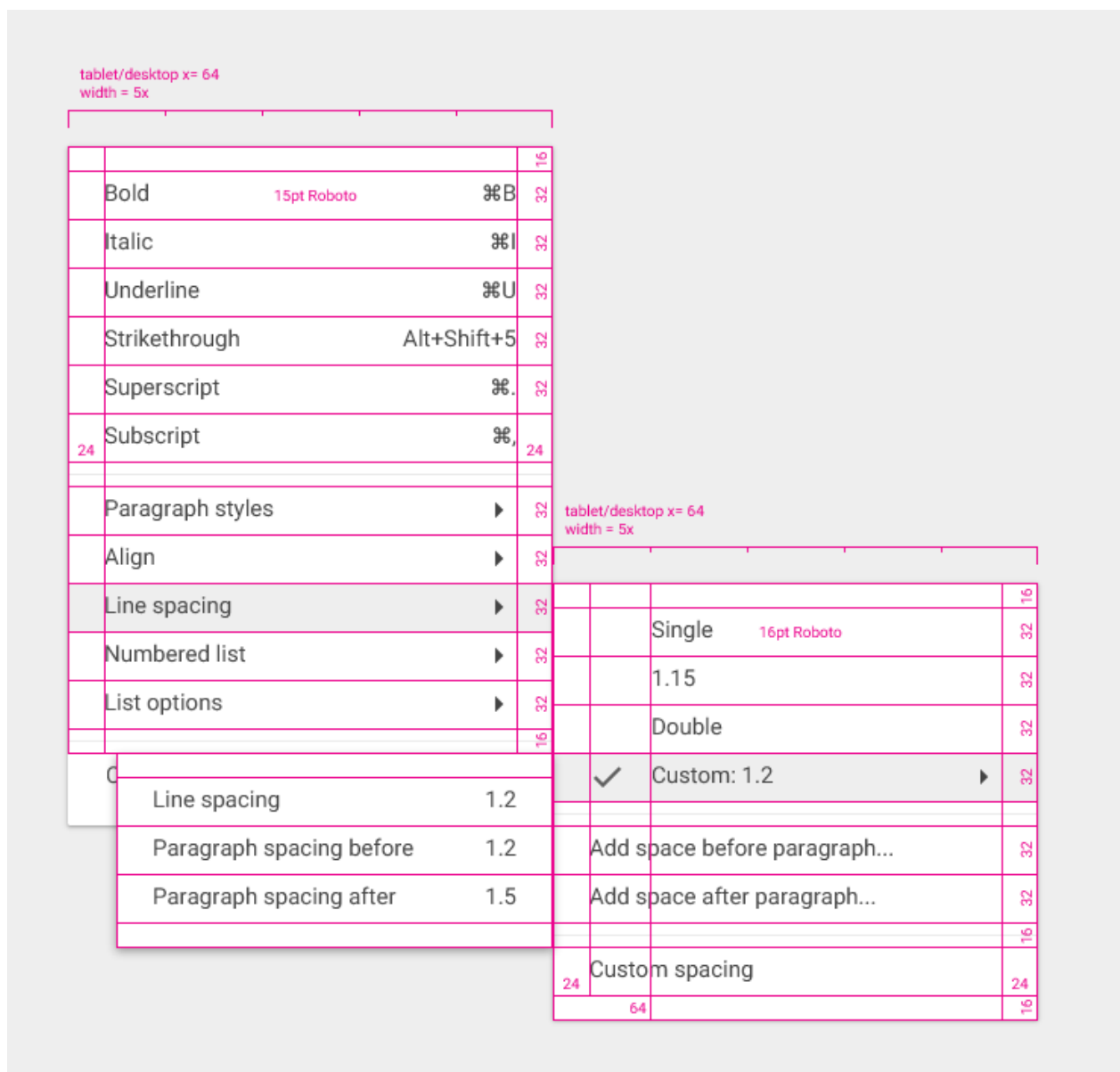
不同的宽度



级联菜单



级联标尺



5.11 进度和动态

在用户可以查看并与内容进行交互之前，我们通过减少用户视觉上的变化，尽量使应用加载过程令人愉快。每次操作只能由一个活动指示器呈现，例如，一次刷新操作不应该同时显示刷新条和动态圆圈。

指示器类型

在操作中，对于完成部分可以确定的情况下，使用一个确定的指示器，他们会告知用户操作所需的时间。

在操作中，对于完成部分不确定，用户需要等待，无需告知后台的情况以及所需时间，这时可以使用一个不确定的指示器。

有两种类型的指示器：线形进度指示器和圈形进度指示器。您可以使用确定的和不确定的操作中的任何一个。

线形进度指示器

线形进度指示器应始终从0%开始标到100%，绝不会低于100%。如果有多个操作发生在序列中，使用进度指示器来表示整体的延迟，这样，当进度达到100%时，它不会返回到0%。

Determinate

Indeterminate

Buffer

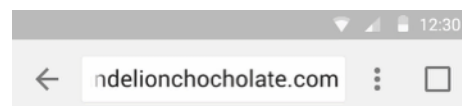
Query Indeterminate and Determinate



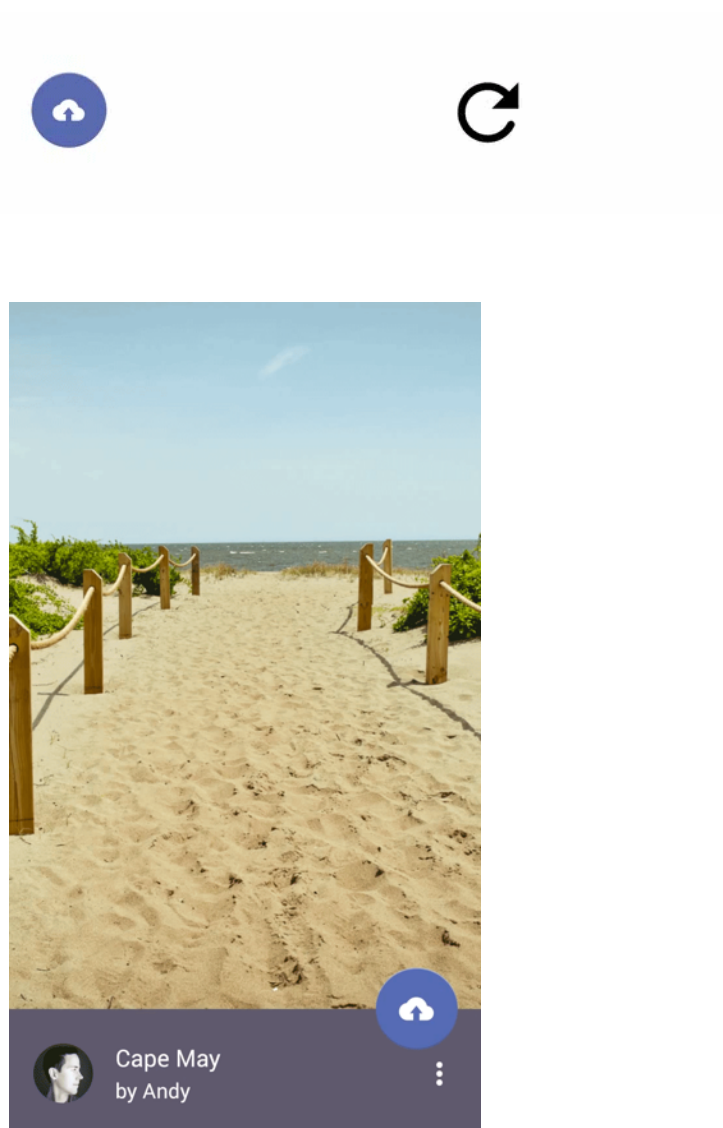
Google Glass Explorer
Story: Roy Choi

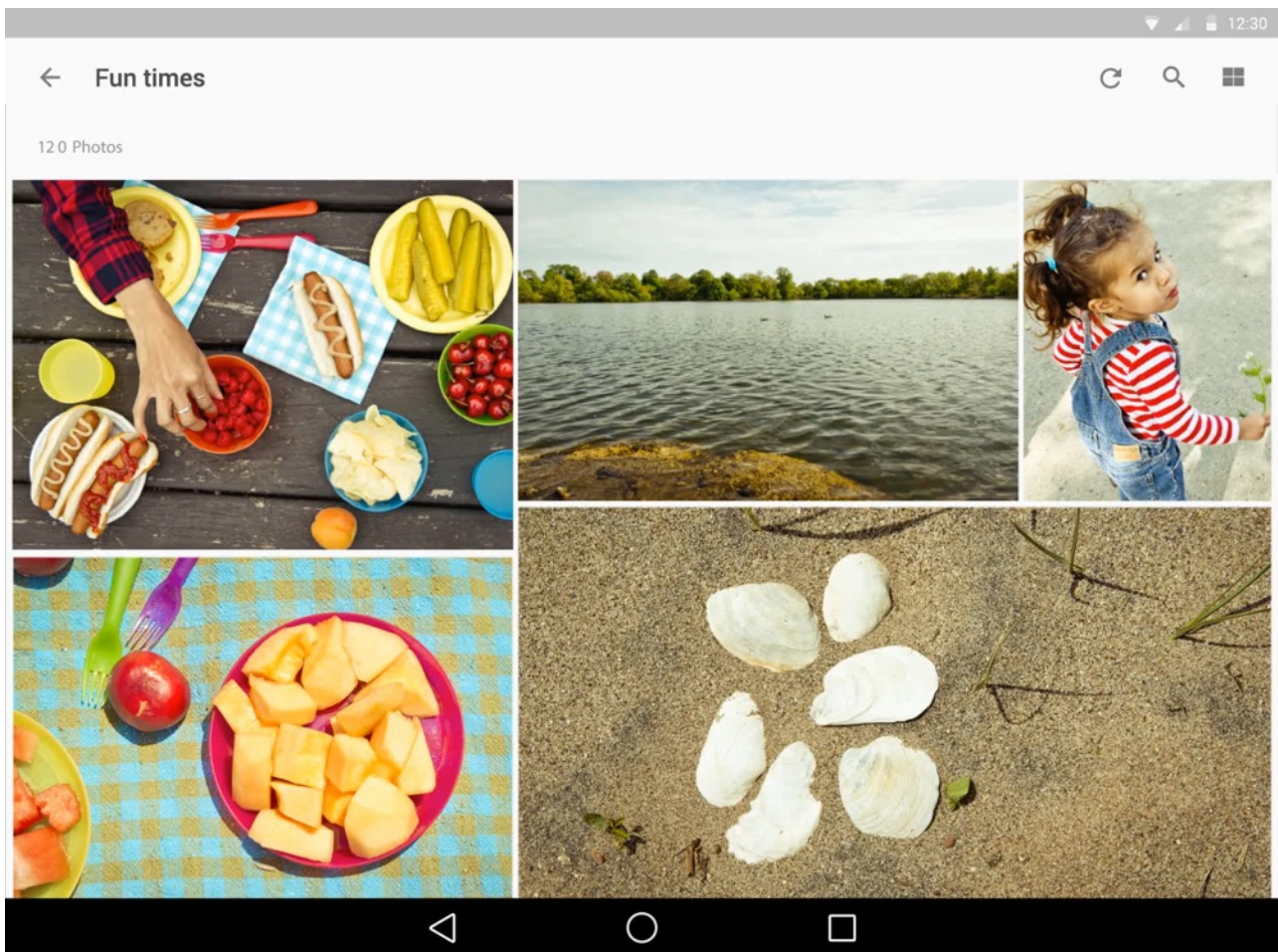
1,809 60 1,667,817 views

Google Glass
68,227 subscribers



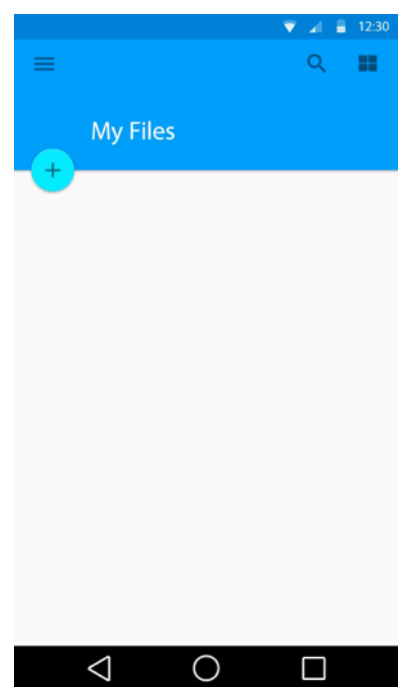
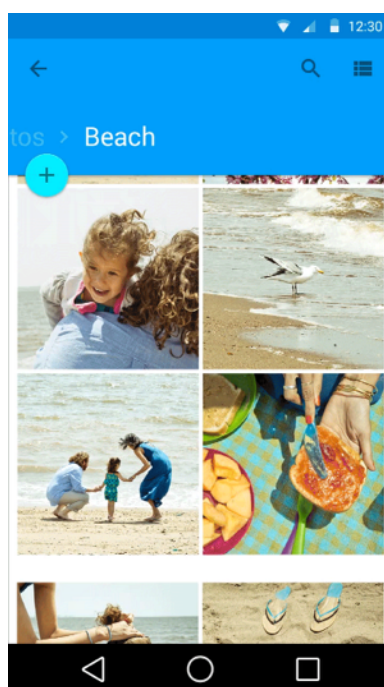
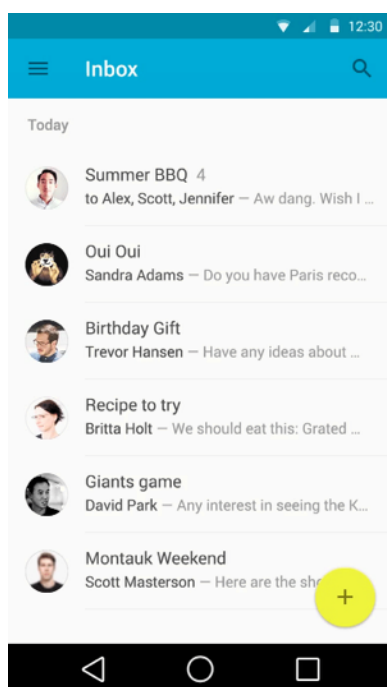
线形进度条应该放置在页眉或表格的边缘。





圆形进度指示器

有集成功能的圈形进度指示器



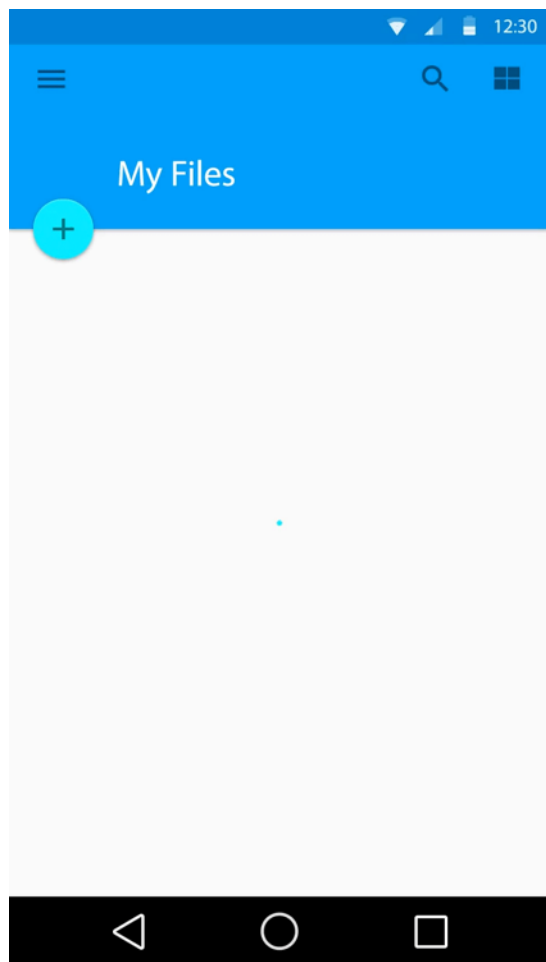
单个定向加载

在现有的，不变的容器
油墨（副本和图像）的
加载。

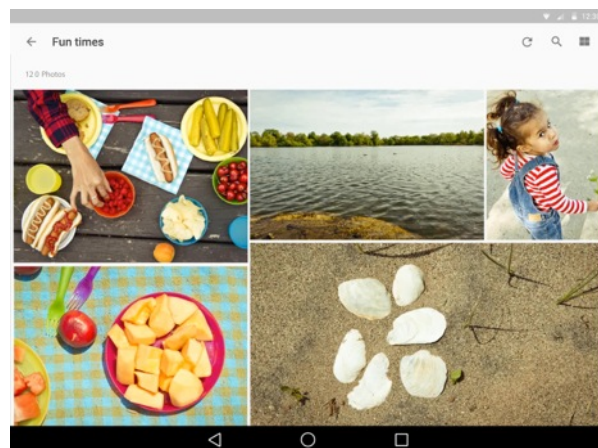
双向加载

纸制容器中产生，然后
再将油墨（副本和图像）
加进去。

第一次加载内容



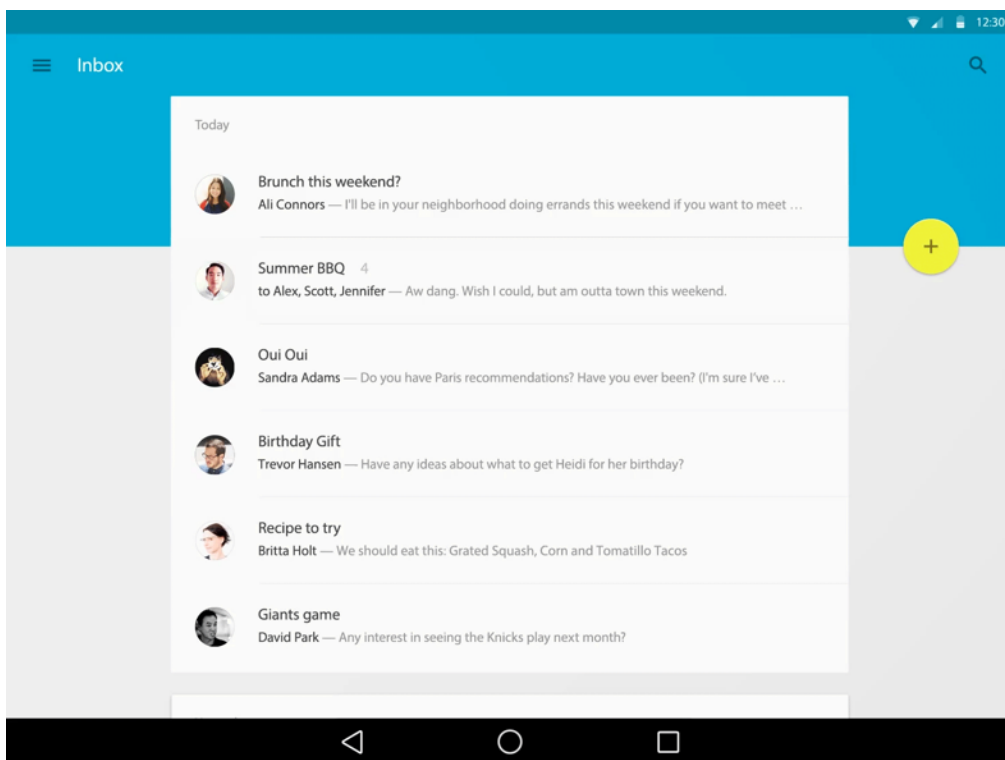
同时加载和显示所有内容



分两个阶段加载和显示

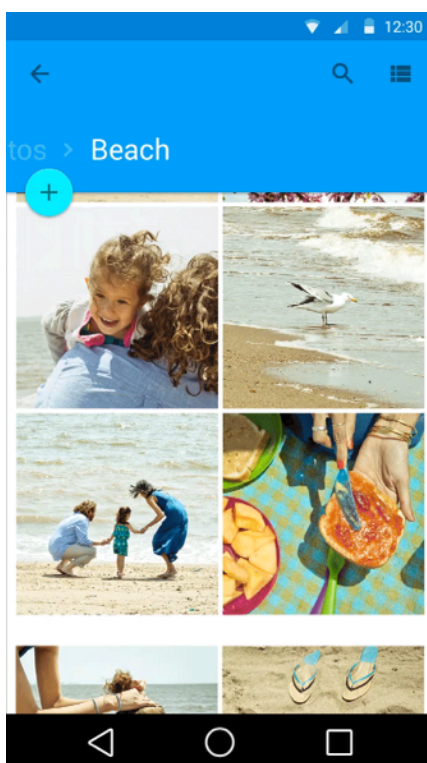
行为

加载阶段



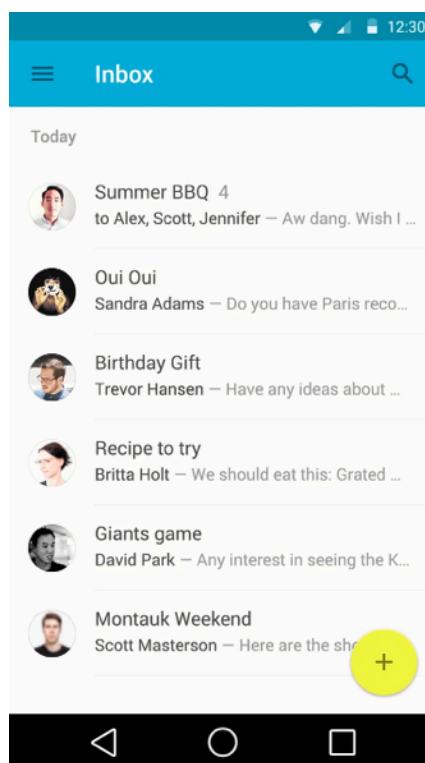
例1：卡扩展

当卡扩展上表面较大时，比如桌面程序，推荐一个不确定的线形进度指示器。



例2：向上滚动刷新

当从下刷新列表时，推荐一个不确定的带有油墨喷发的圆形指示器



例3：向下滑动刷新

当从上刷新列表时，推荐一个不确定的带有油墨喷发的圆形指示器。

5.12 滑块

滑块控件(Sliders)

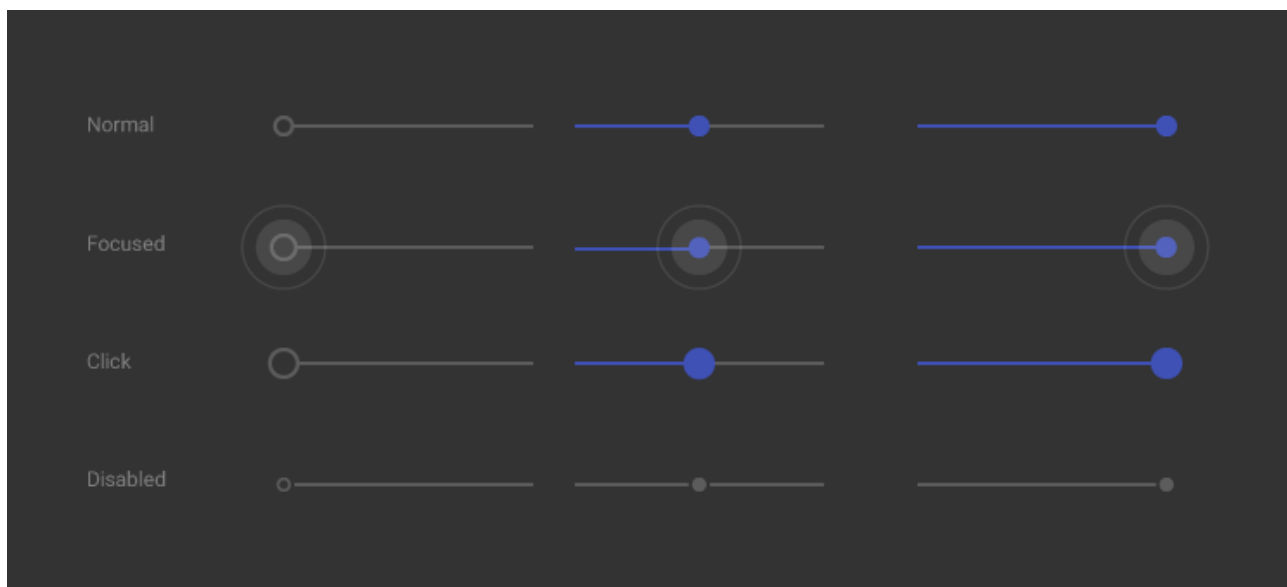
滑块控件可以让我们通过在连续或间断的区间内滑动锚点来选择一个合适的数值。区间最小值放在左边，对应的，最大值放在右边。滑块(Sliders)可以在滑动条的左右两端设定图标来反映数值的强度。这种交互特性使得它在设置诸如音量、亮度、色彩饱和度等需要反映强度等级的选项时成为一种极好的选择。

连续滑块(Continuous Slider)

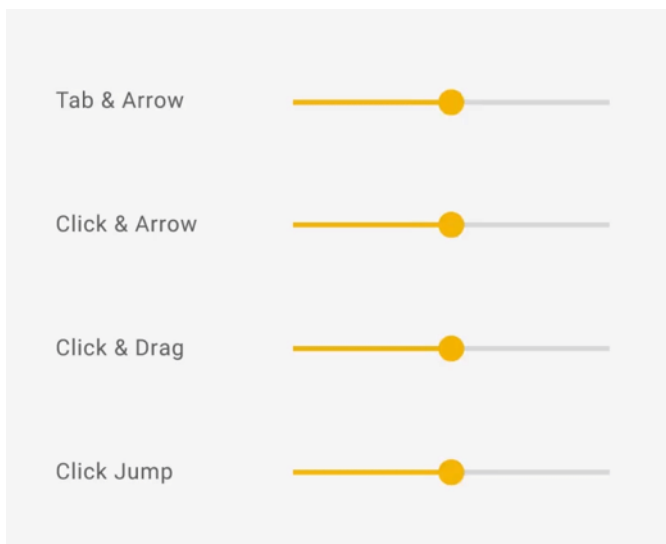
在不要求精准、以主观感觉为主的设置中使用连续滑块，让使用者做出更有意义的调整



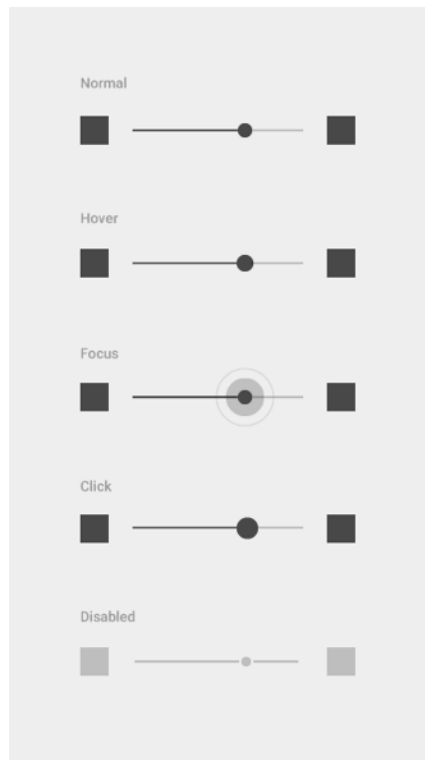
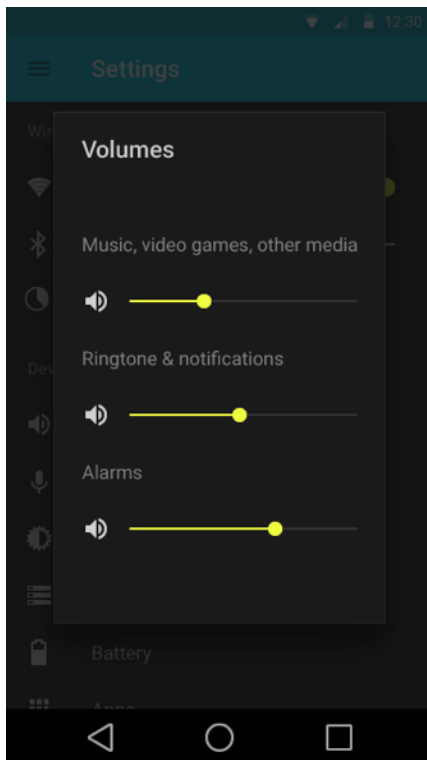
亮色主题



暗色主题

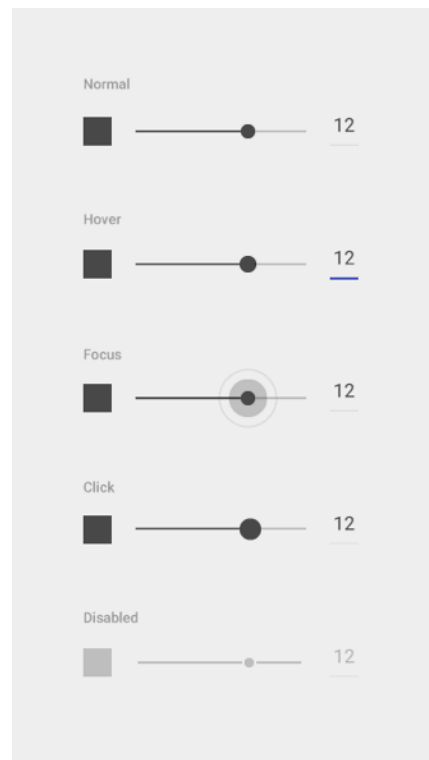
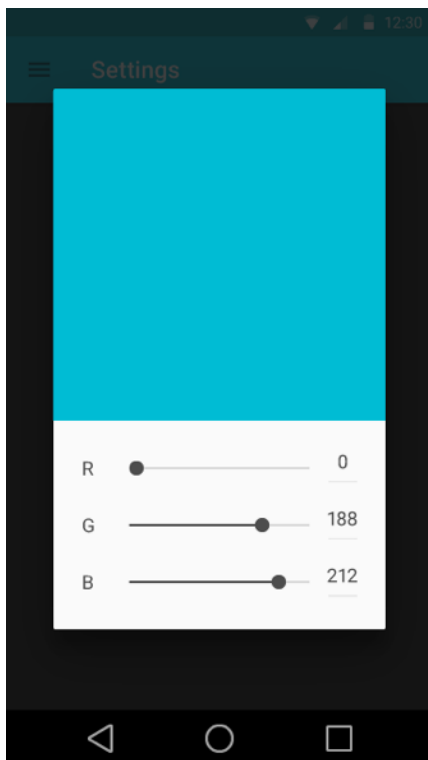


在滑动条(Bar)上图标相对于左边和/或右边的间距的呈现



带有可编辑数值的滑块

用于使用者需要设定精确数值的设置项，可以通过点触缩略图、文本框来进行编辑。

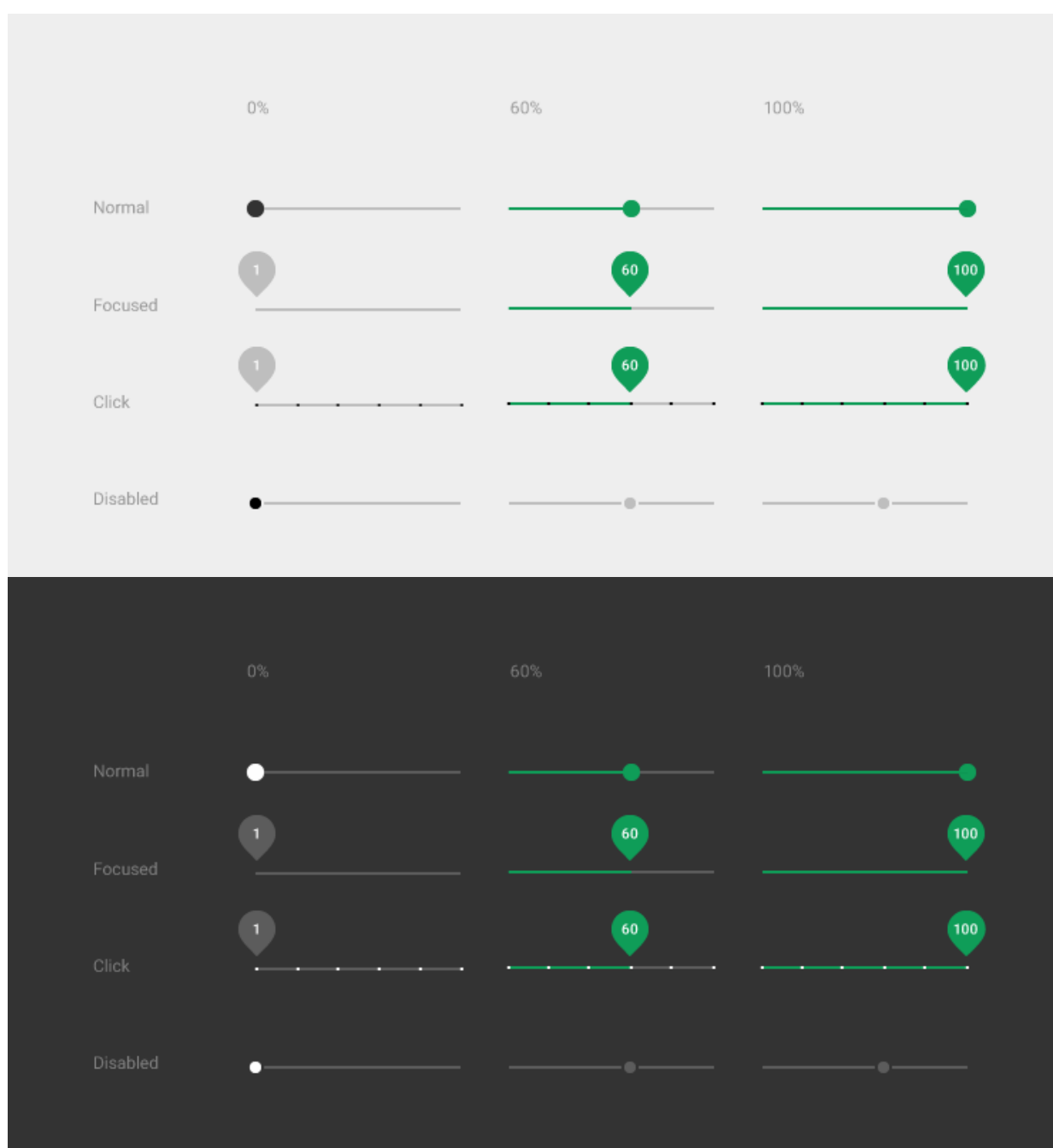


间断滑块(Discrete Slider)

间断滑块会恰好咬合到在滑动条上平均分布的间断标记(tick mark)上。在要求精准、以客观设定为主的设置项中使用间断滑块，让使用者做出更有意义的调整。间断标记(tick mark)应当设定一定的等级区间进行分割，使得其调整效果对于使用者来说显而易见。这些生成区间的值应当是预先设定好的，使用者不可对其进行编辑。

附带数值标签的滑块

用于使用者需要知晓精确数值的设置项。



Tab & Arrow



Click & Arrow



Click & Drag



Click Jump



5.13 消息栏和弹出消息

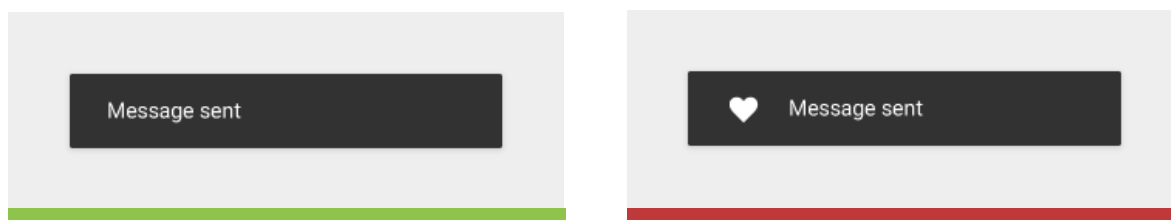
消息栏是一种针对操作的轻量级反馈机制，通常出现在手机屏幕或者桌面端左下方，以浮动弹出框的形式存在。它们会在屏幕所有层的最上方，包括浮动操作按钮。

它们会在设定时间内或者用户在屏幕其他地方触摸之后自动消失。消息栏可以在屏幕上滑动关闭。当它们出现时，不会阻碍用户在屏幕上的输入，在屏幕上同时只能显示一个提示消息。

Android也提供了一种胶囊状的消息栏，主要用于提示系统消息。弹出消息和消息栏非常相似，但是弹出消息并不能操作也不能滑动关闭。

用法

很短的文本流 (Very short text strings)



通常消息栏会调整高度来适应所有的文本，而文本应该与执行的操作相关联。消息栏中不能包含图标，只能以纯文本的形式存在。

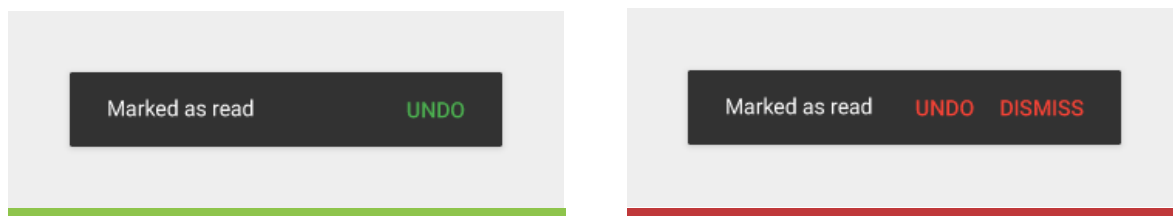
短暂

为了保证可用性，消息栏不应该成为通往核心模块的唯一方式。作为被置于顶层的元素，消息栏不应该长时间存在并且相互堆叠。

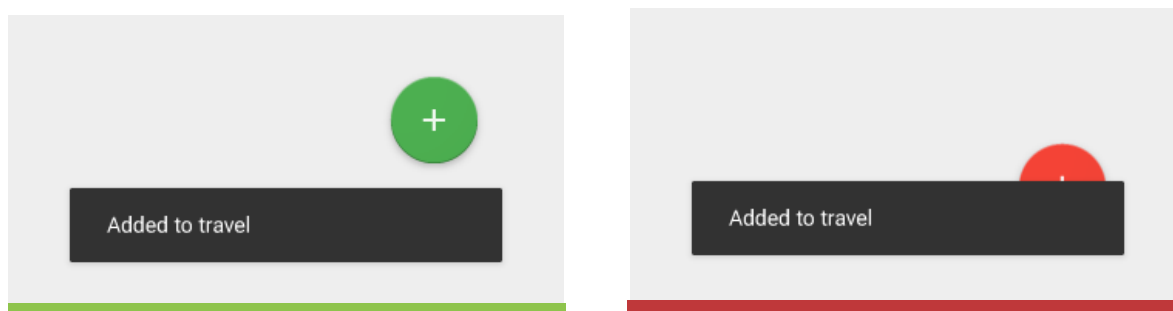


最多1个动作，不要有取消操作

当设计一个操作的时候，应当遵循消息栏规格与预设的规则。当有2个或者更多的操作出现时，应该使用提示框而不是消息栏，即使其中的一个提示内容是取消某个操作。如果消息栏中提示的操作重要到需要打断屏幕上正在进行的操作，那么理当使用提示框而非消息栏。



不要遮挡浮动操作按钮

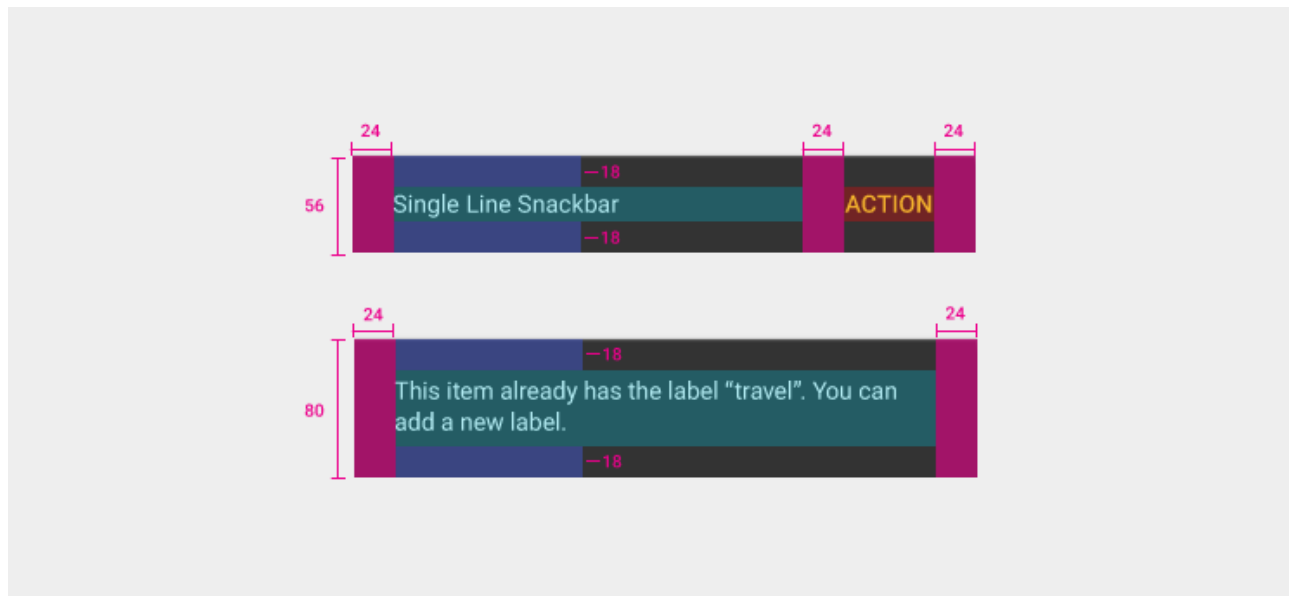
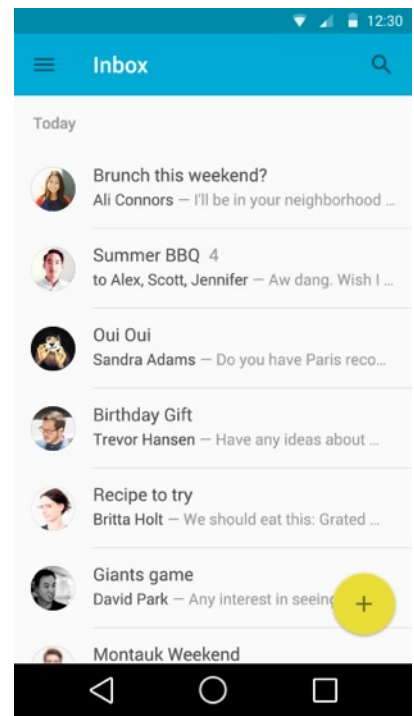
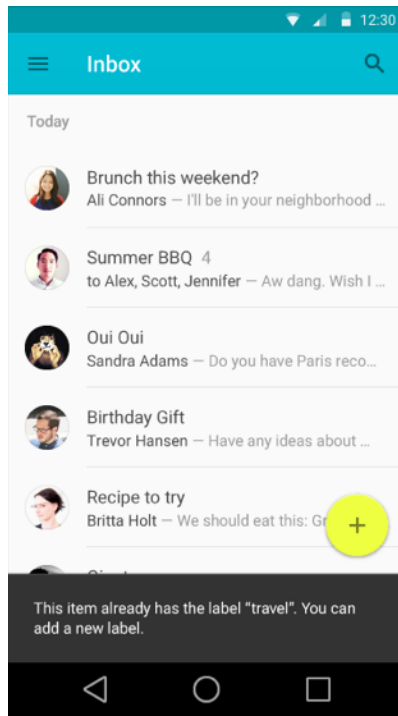
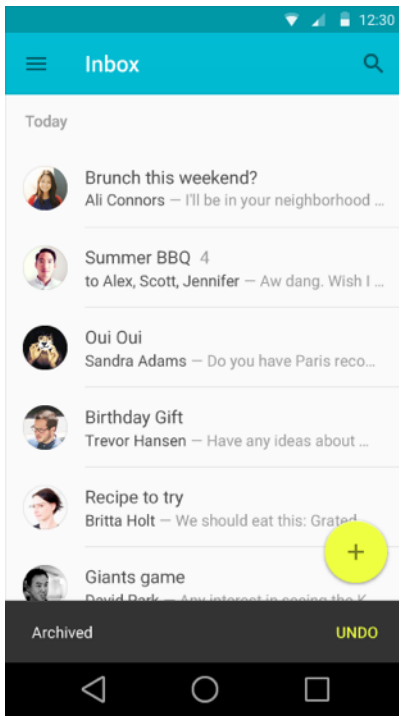


向上移动你的浮动按钮，直到容纳下消息栏的高度。

规格

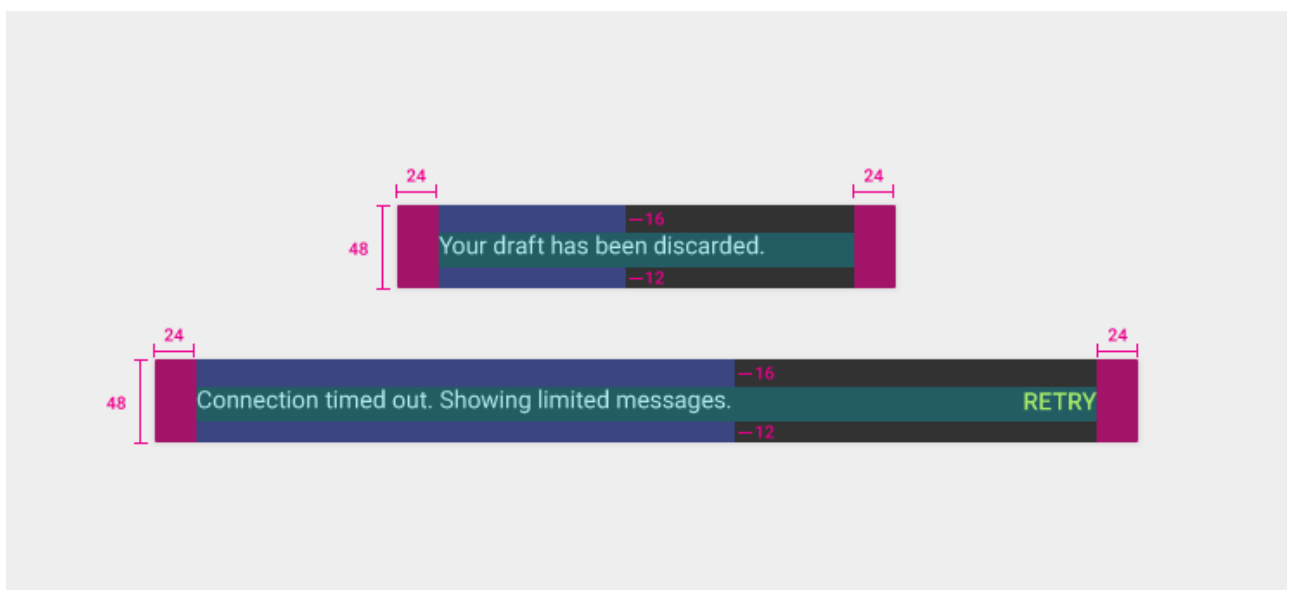
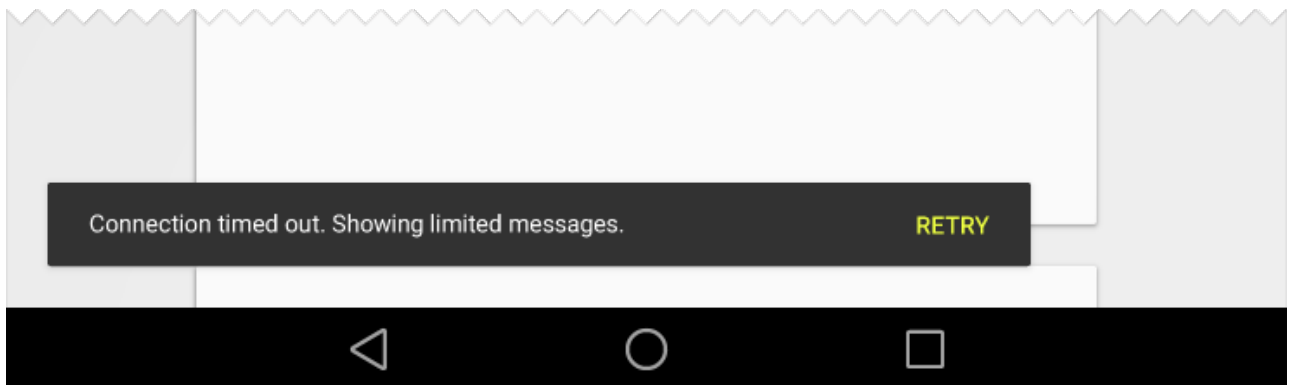
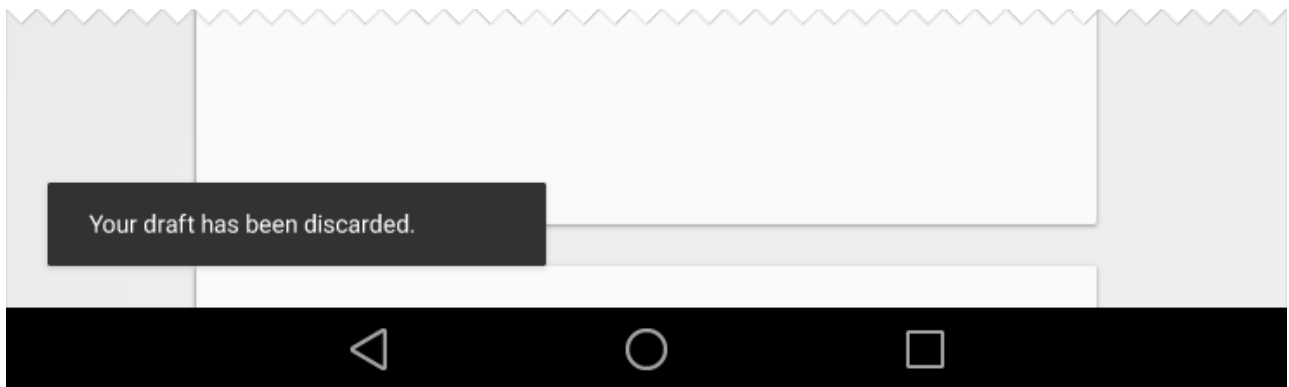
手机端消息栏

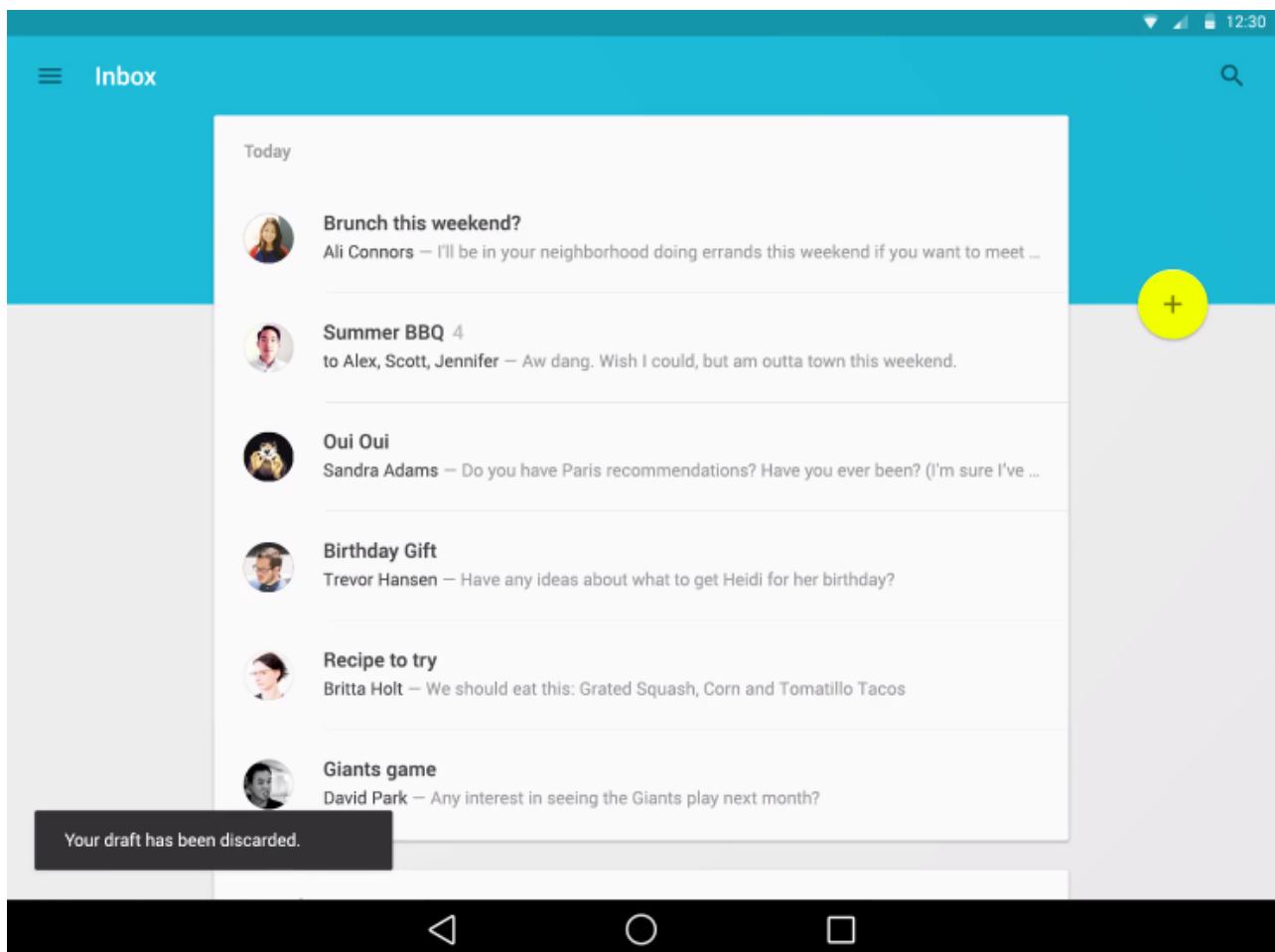
- 单行消息栏 高度: 48 dp
- 多行消息栏 高度: 80 dp
- 文本 : Roboto Regular 14 sp
- 操作按钮: Roboto Medium 14 sp, 大写文本
- 默认背景填充色: #323232 100%



平板/桌面端消息栏

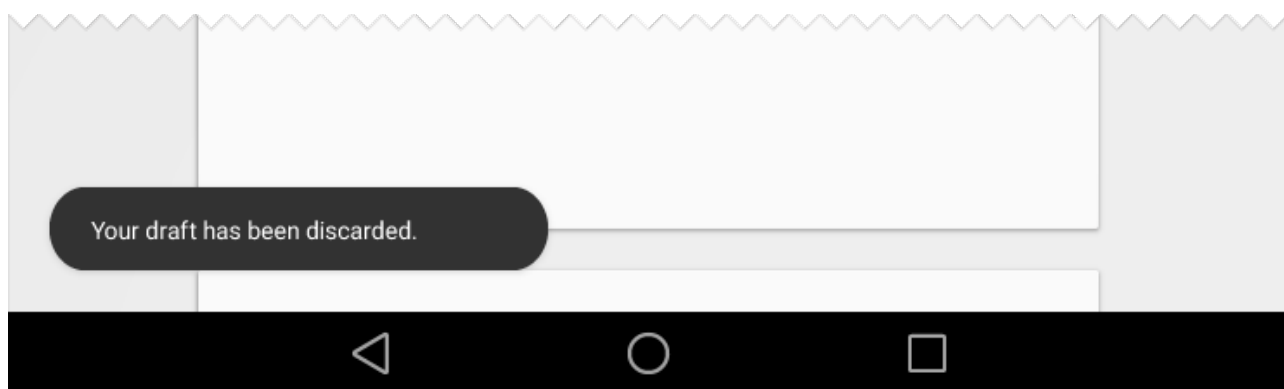
- 单行 snackbar 高度: 48 dp tall
- 最小宽度: 288 dp
- 最大宽度: 568 dp
- 2 dp 圆角
- 文本: Roboto Regular 14 sp
- 操作按钮: Roboto Medium 14 sp, 大写文本
- 默认背景填充色: #323232 100%

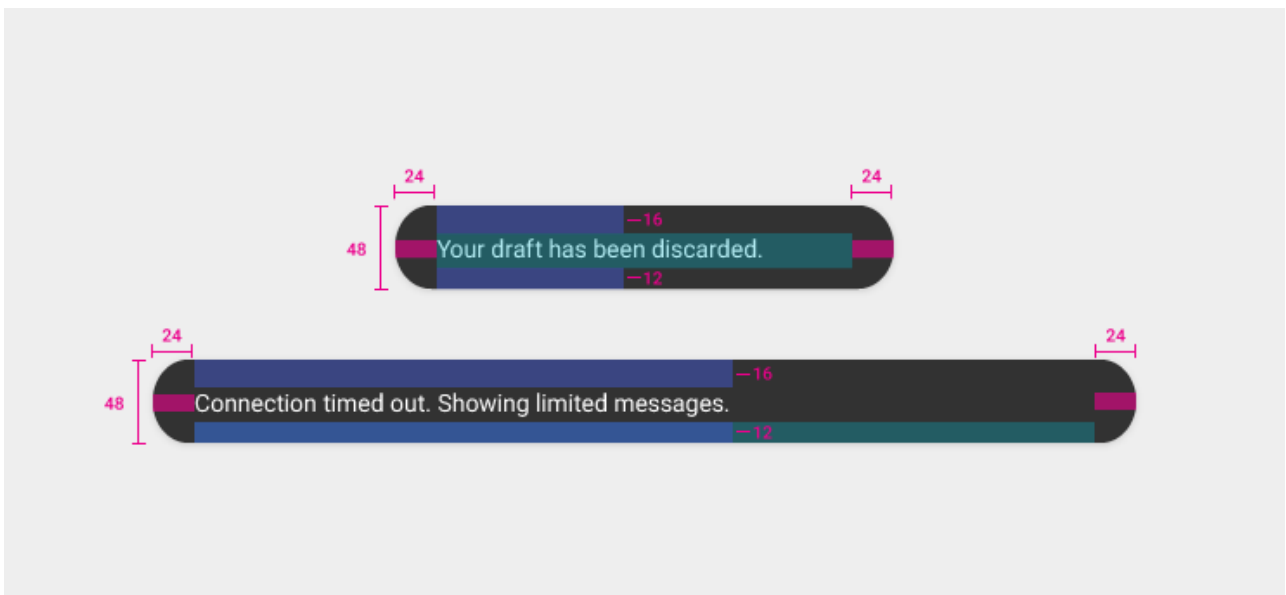
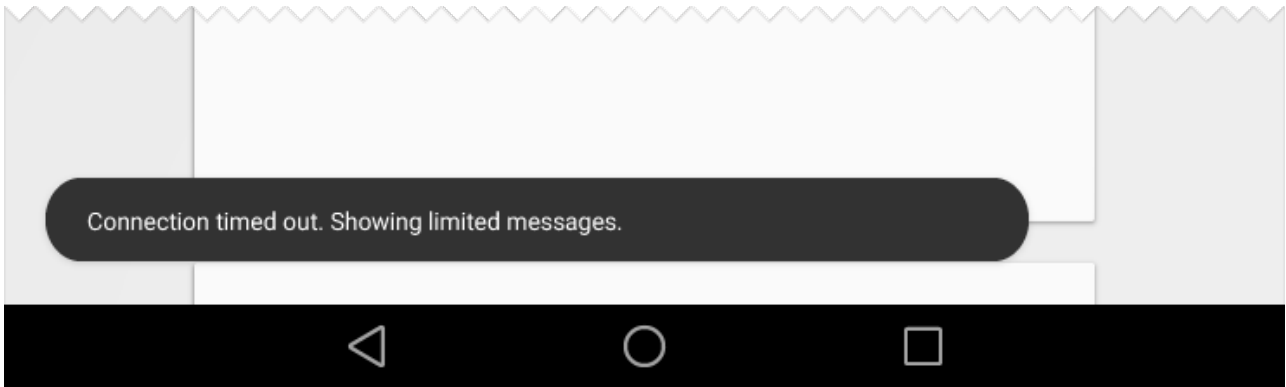




Android 弹出消息

开发者可以自定义的弹出消息和它在屏幕上的位置。如果你打算自定义弹出消息的话，强烈建议按照上述消息栏的样式来设计





5.14 副标题

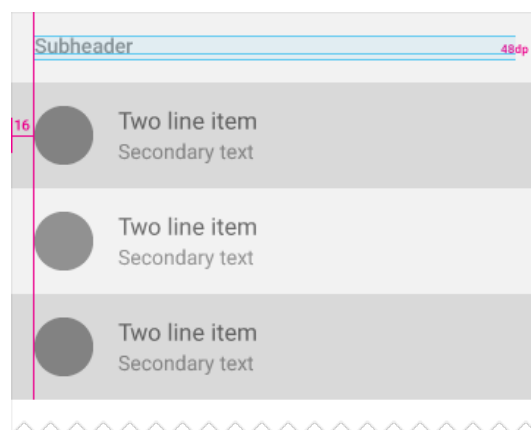
副标题是特殊的列表区块，它描绘出一个列表或是网格的不同部分，通常与当前的筛选条件或排序条件相关。副标题可以内联展示在区块里，也可以关联到内容里，例如，关联在相邻的分组列表里在滚动的过程中，副标题一直固定在屏幕的顶部，除非屏幕切换或被其他副标题替换。为了提高分组内容的视觉效果，可以用系统颜色来显示副标题。

列表副标题

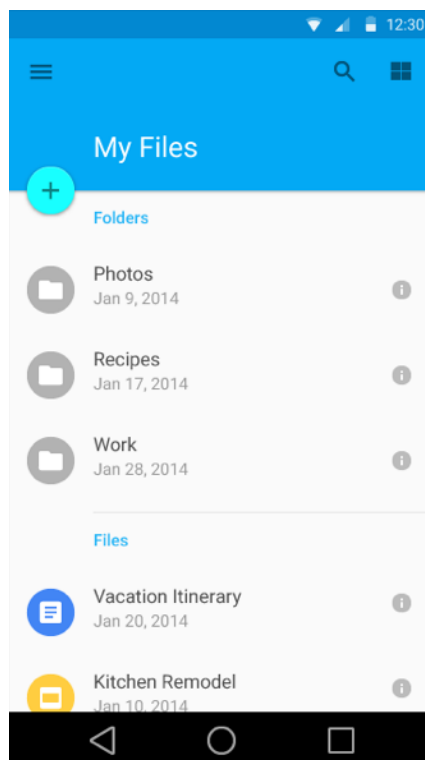
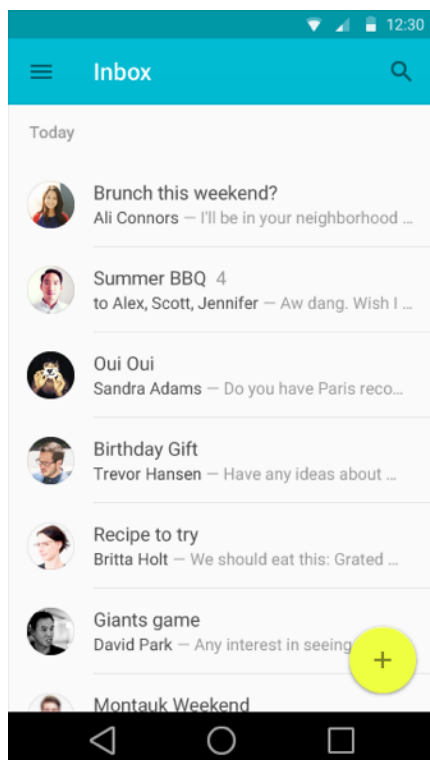
区块高度是48dp。

副标题字体为 Roboto Medium 14sp。

副标题应该跟列表中头像或是图标左对齐，除非那个地方有突显操作。



如果有突显操作，它需要跟列表中的头像或图标左对齐。副标题则需要和列表中的文字左对齐

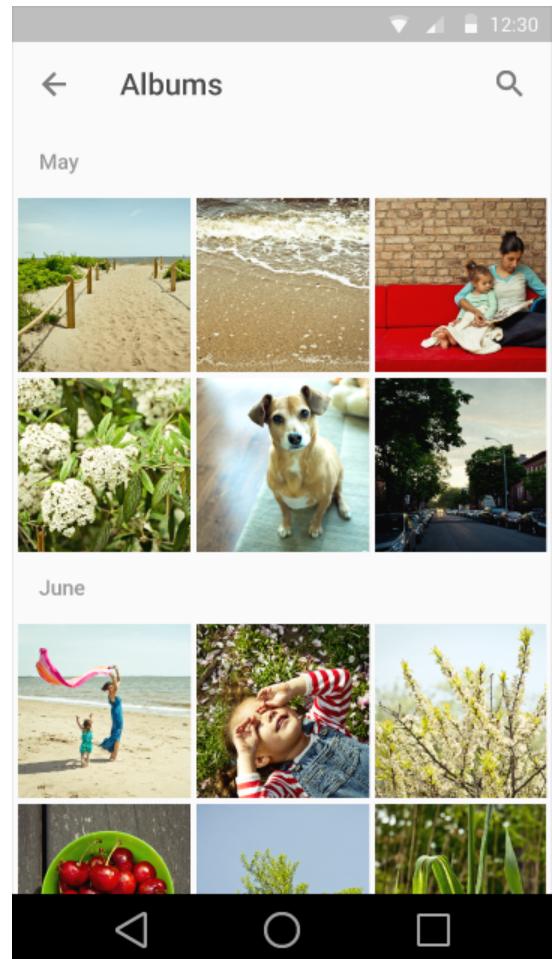
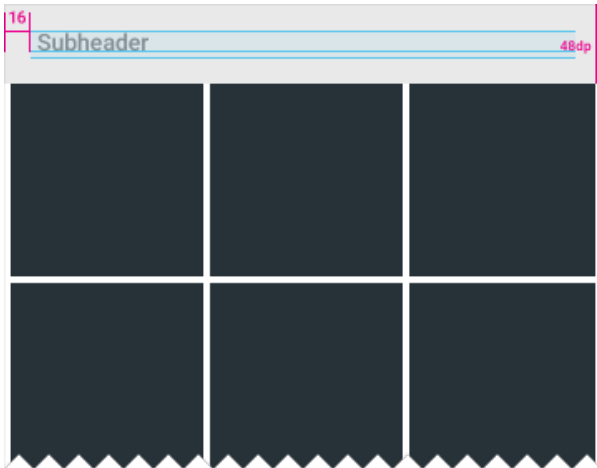


网格副标题

区块高度是48dp。

副标题字体为 Roboto Medium 14sp。

副标题跟左边缘距离为16dp。



5.15 开关

开关允许用户选择选择项。一共有三种类型的开关：复选框、单选按钮和 on/off 开关。

注释：下面示例中所示的图形环代表一个动画，并不是实际按钮的外观。

复选框

复选框允许用户从一组中选择多个选项。

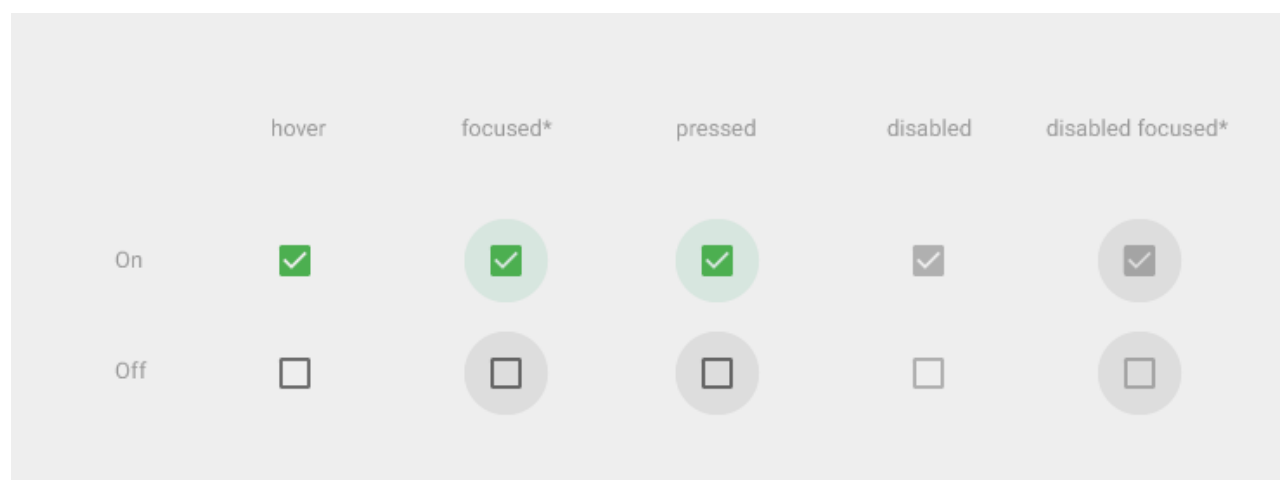
如果在一个列表中出现多个 on/off 选择，复选框是一种很好节省空间的方式。

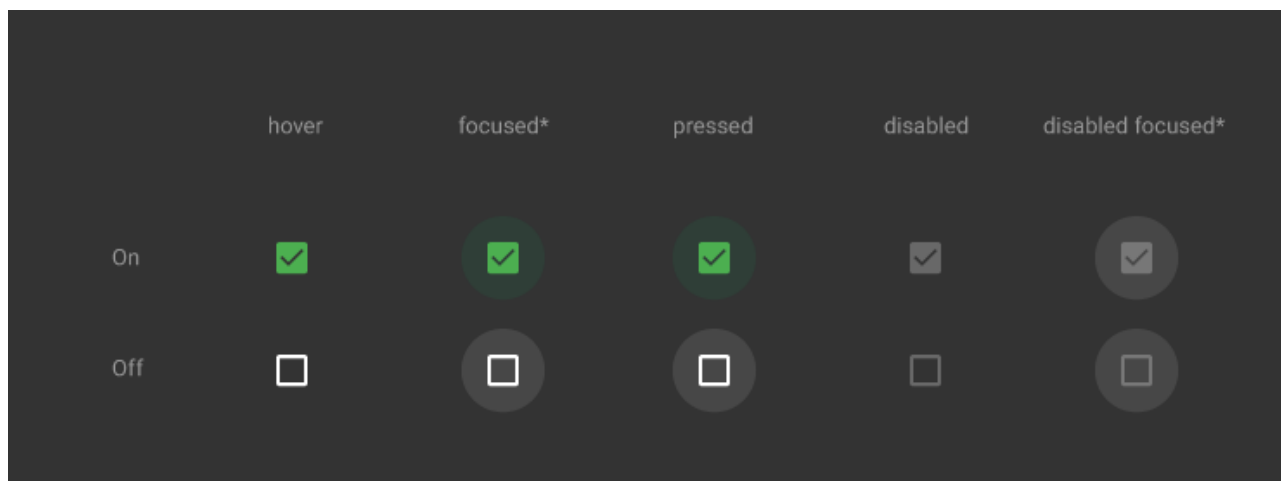
如果只有一个 on/off 选择，不要使用复选框，而应该替换成 on/off 开关。

通过积极地将复选框换成勾选标记，进而使去掉勾选操作变得更加令人满意和更加直接。

复选框通过动画来传达被聚焦和被按下的状态。

- ☐ Milk
- ☐ Eggs
- ☒ Cereal
- ☒ Pony



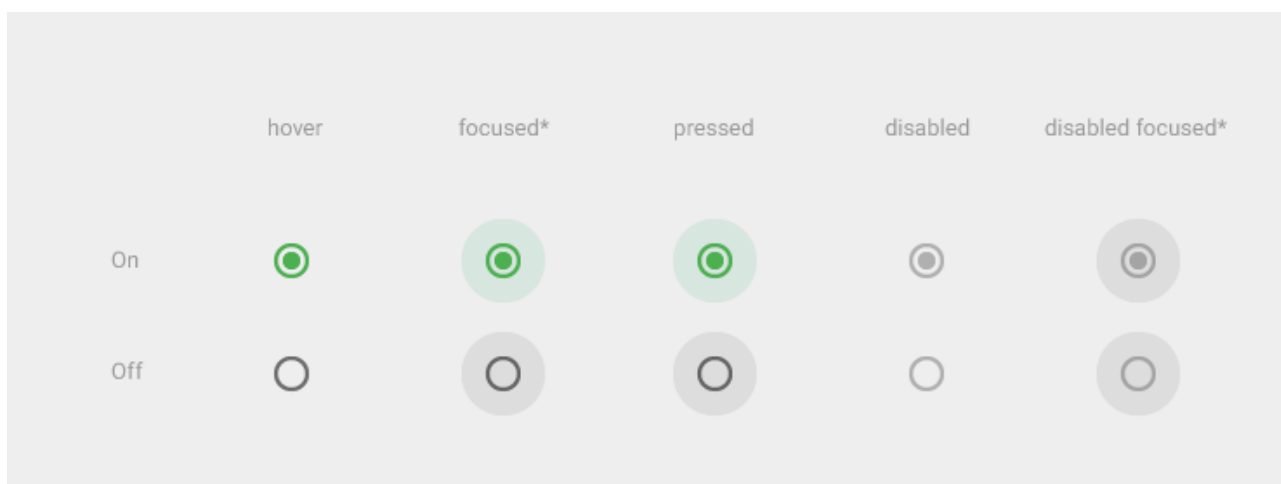
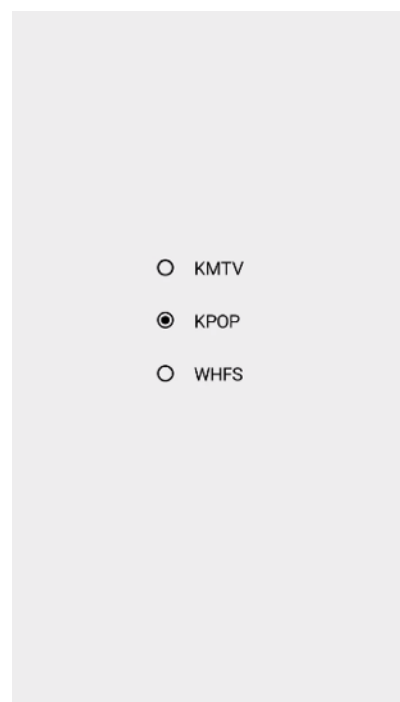


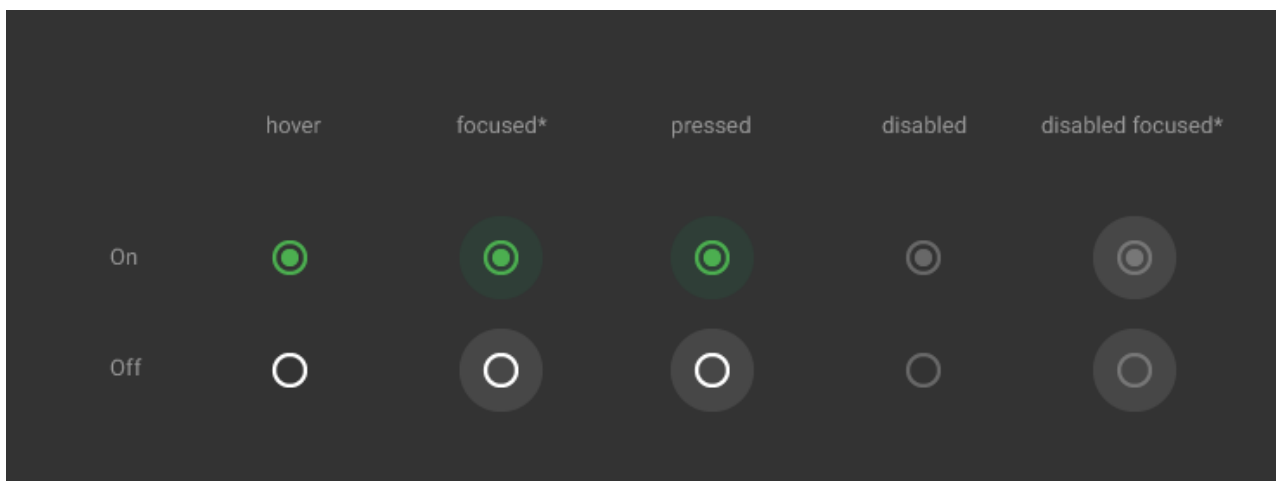
单选按钮

单选按钮只允许用户从一组中选择一个选项。如果你认为用户需要看到所有可用的选项并排显示，那么请为排他选择使用单选按钮。

否则，考虑相比显示全部选择更节省空间的下拉。

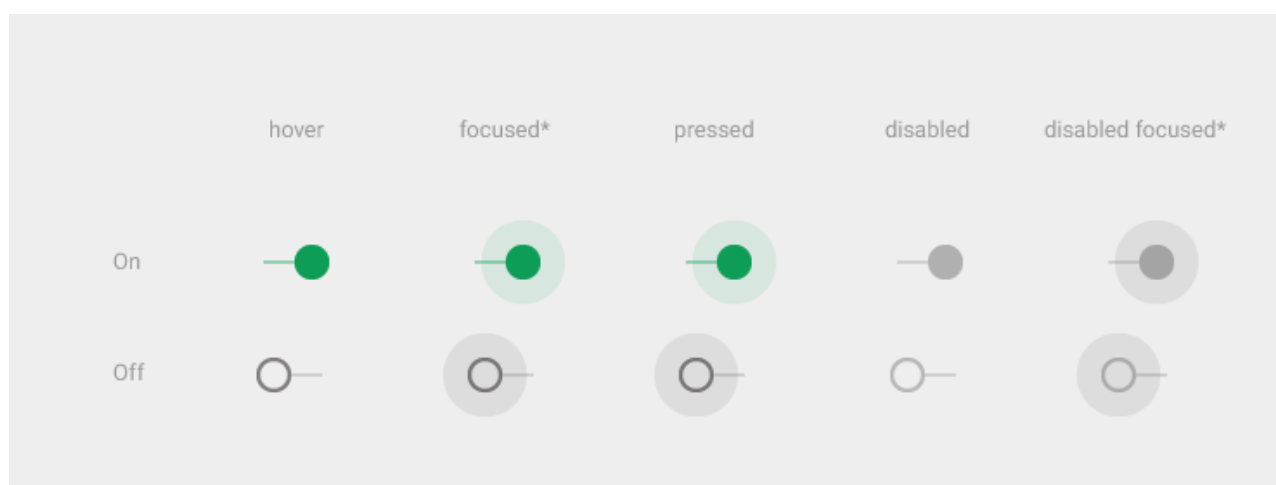
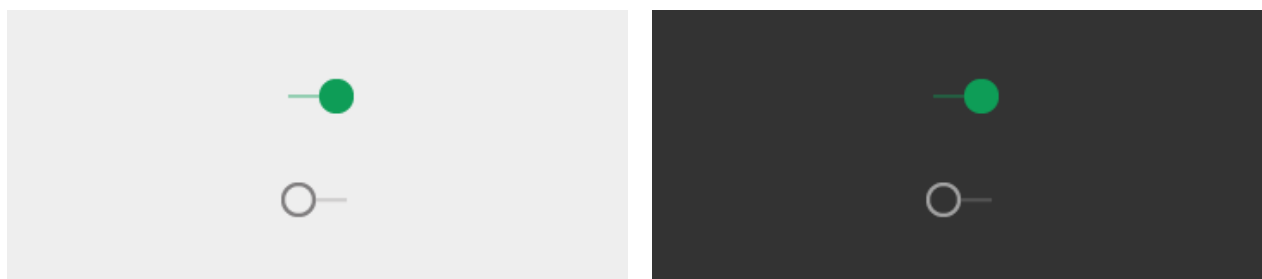
单选按钮通过动画来传达被聚焦和被按下的状态。



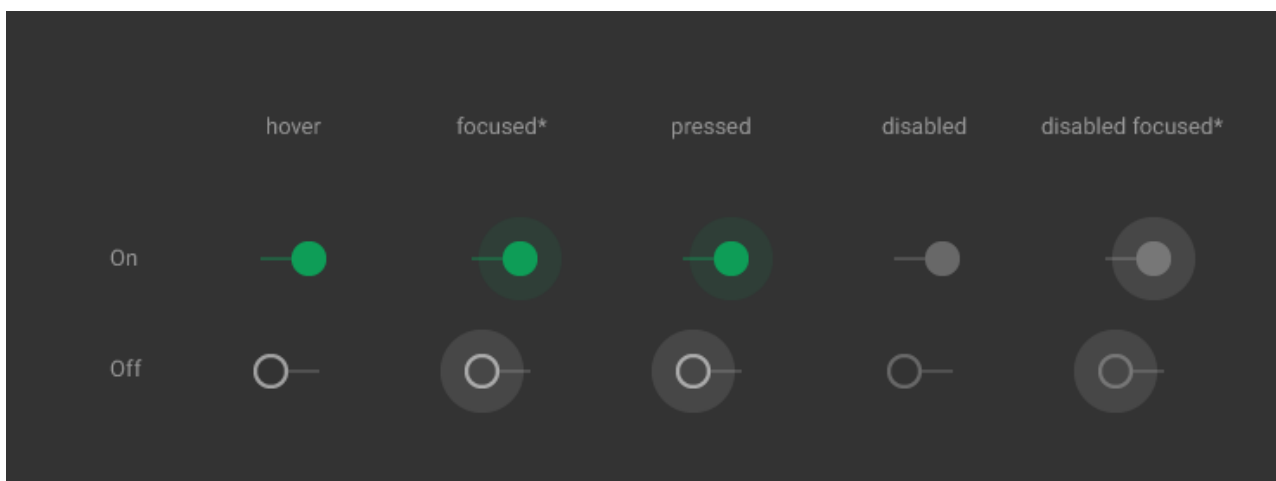


开关

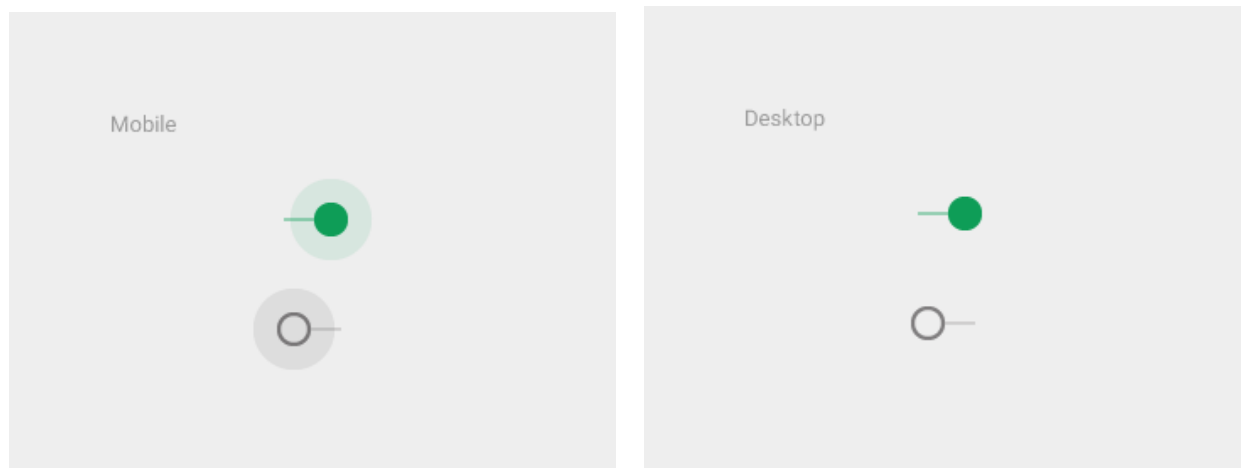
On/off 开关切换单一设置选择的状态。开关控制的选项以及它所在的区域，应该明确标示出来与内部的标签相一致。开关将和单选按钮起到相同的视觉特性。



开关通过动画来传达被聚焦和被按下的状态。



开关滑块上标明 “on” 和 “off” 的做法被弃用，取而代之的是下图所示的开关



仅在有利于手指触摸的形式因素下，对在交互中被完全遮挡的元素使用外部径向扩充。桌面使用的是鼠标，你不需要这个额外的指示

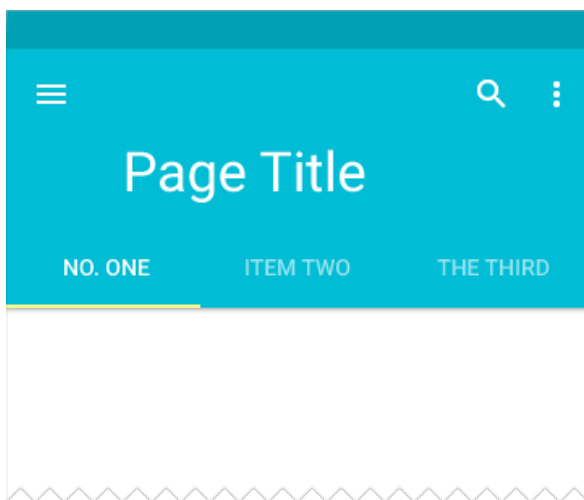
5.16 切换标签

在一个软件中，切换标签让不同的视图，功能或者浏览不同类别的数据集合切换起来变得简单。

用法

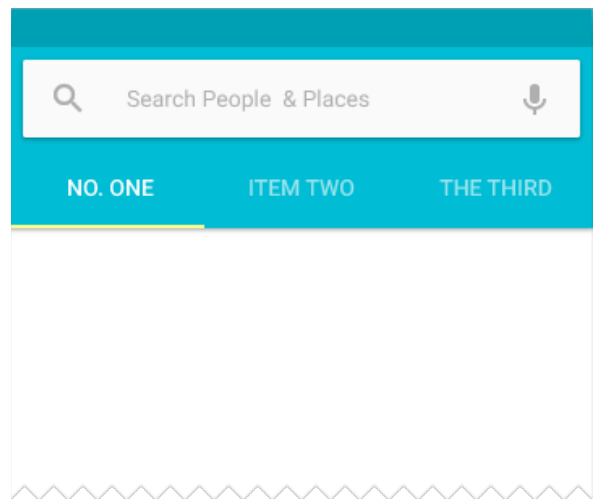
一个 切换标签 为显示内容相关的分组提供启示。一个 切换标签 标签简要地描述了相关分组的内容

移动端的切换标签



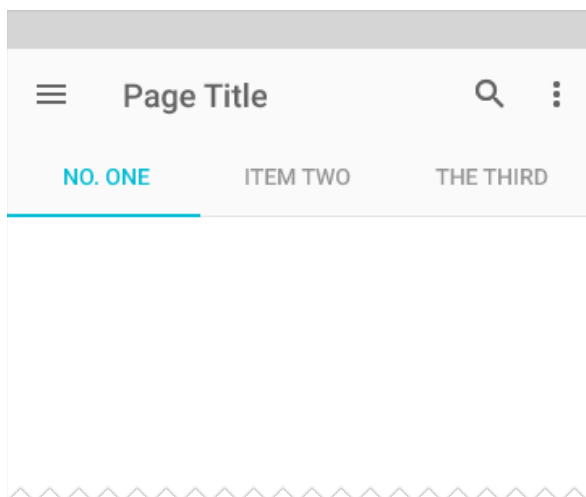
扩展的软件标题栏

默认的切换标签

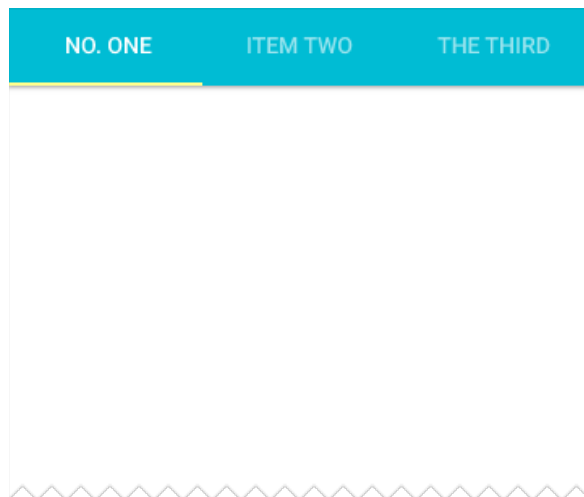


带有搜索的切换标签

支持横向滚动的

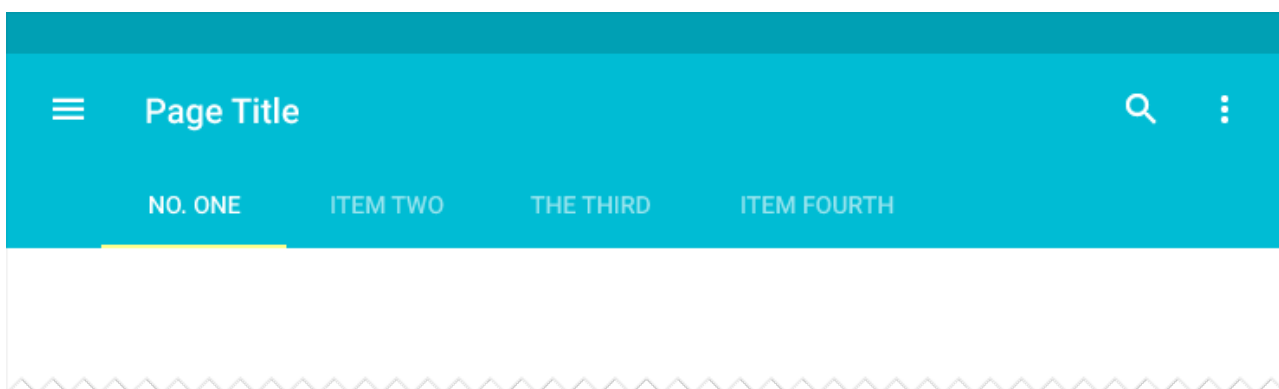


和指示器一样字体颜色的

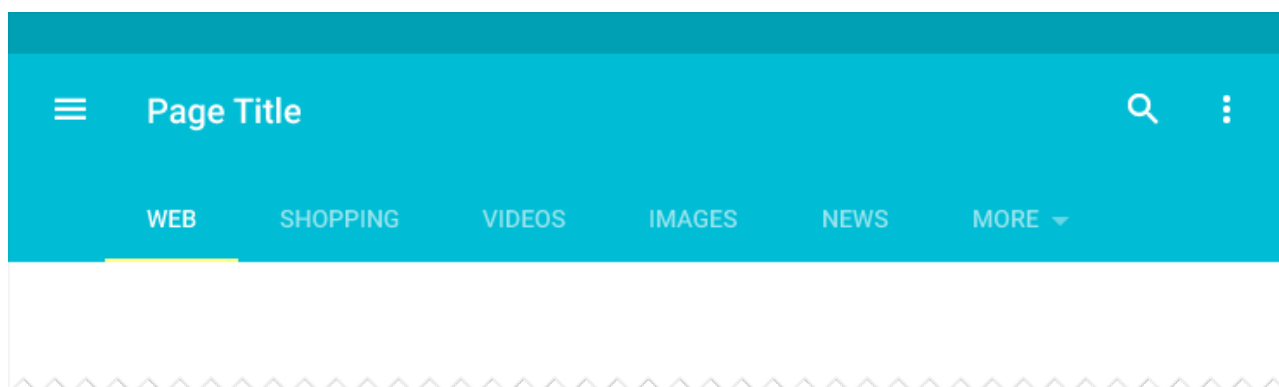


悬浮的切换标签

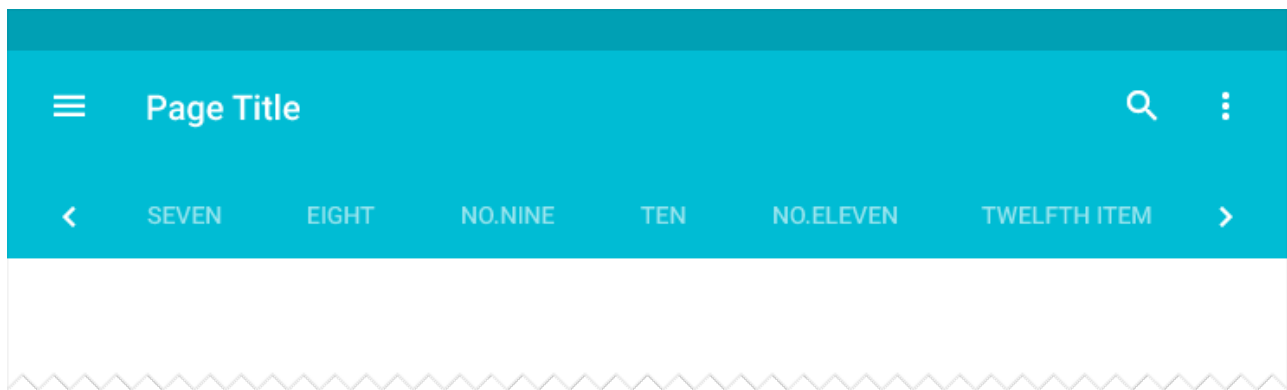
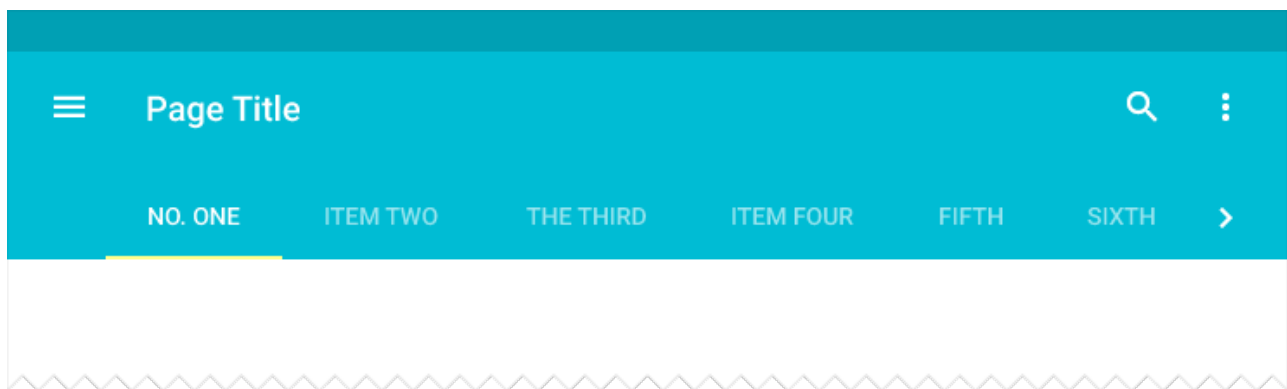
桌面端的切换标签



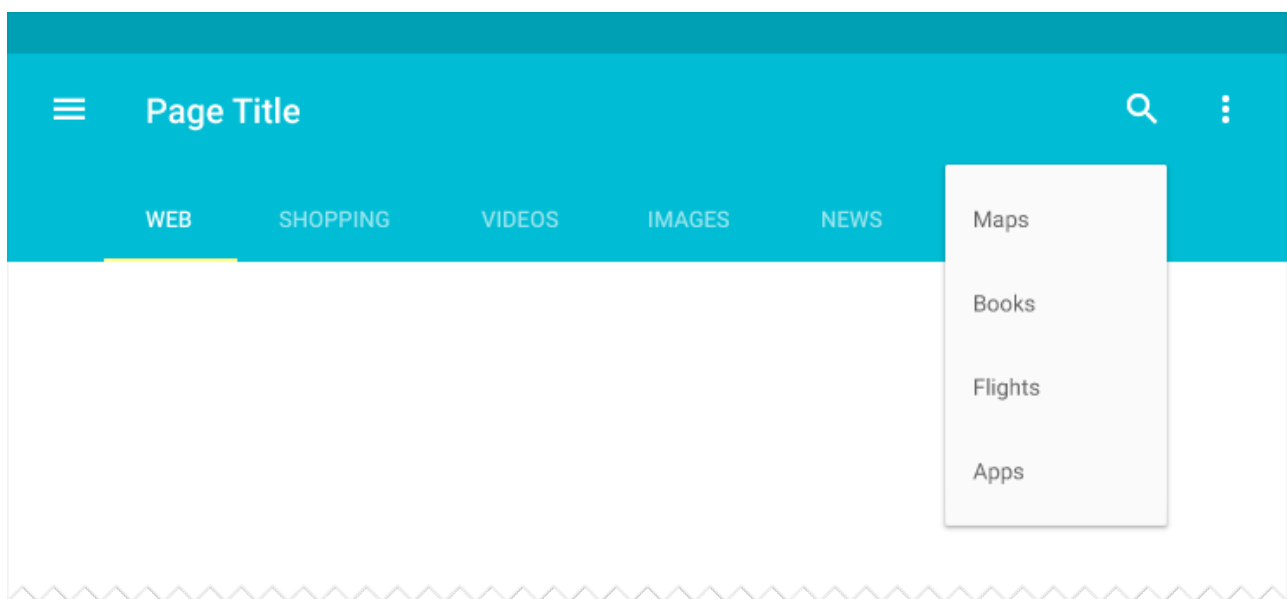
默认的标题和切换标签



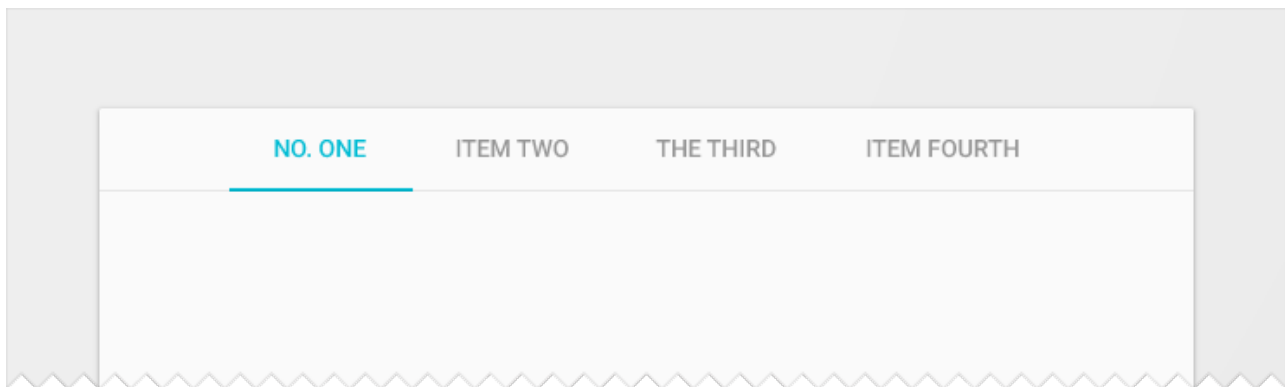
附加“更多”下拉菜单的



开始的标签，带有更多方向指示按钮的



展开的菜单



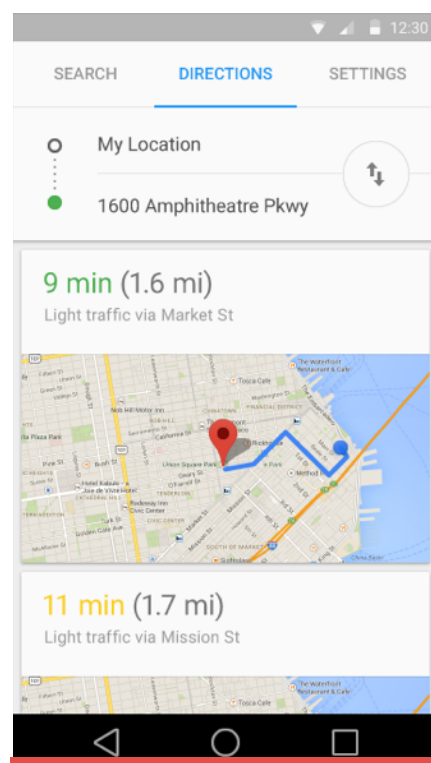
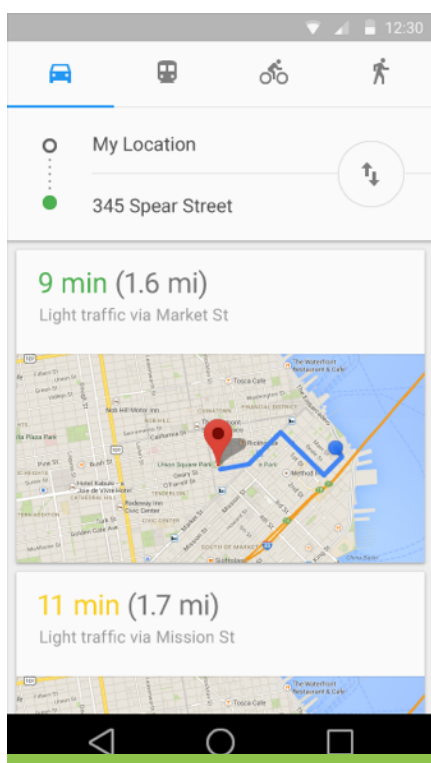
居中的切换标签

何时使用

使用 切换标签将大量的关联数据或选择划分地更容易理解，专注在有助于内容导航或内容结构为目标的组，而不需要导航出当前的上下文。

虽然被指定的内容可能是自然的导航(例如，道路选项可以改变地图的视图，检索结果引导到其他网站)，但切换标签本身没有被用于导航。

切换标签s 也不被用于旋转或者是分页（比如：页面之间的滑动）



切换标签特性

切换标签只有一行。

切换标签不应该被嵌套。也就是说，一个切换标签里的内容不能有另外一组被切换标签化的内容。

一组切换标签至少包含 2 个切换标签并且不多于 6 个 切换标签。

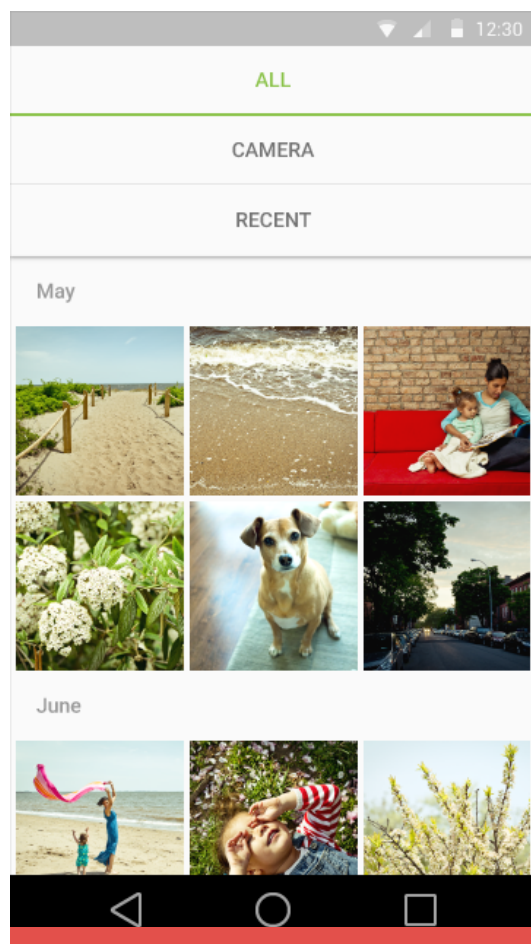
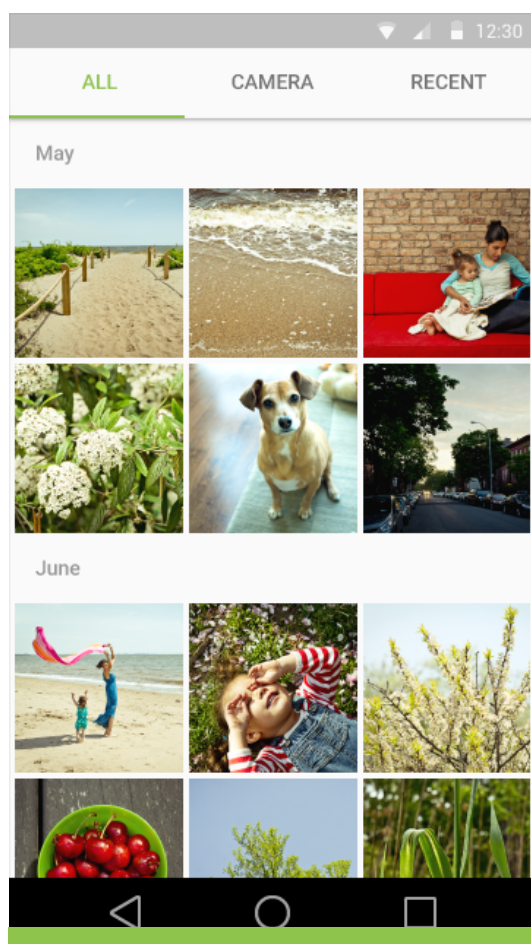
切换标签控制的显示内容的定位要一致。

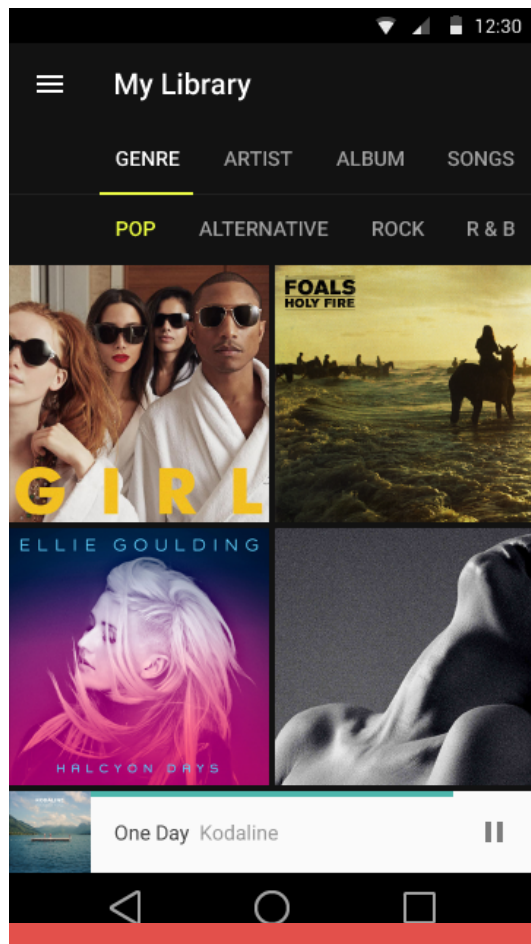
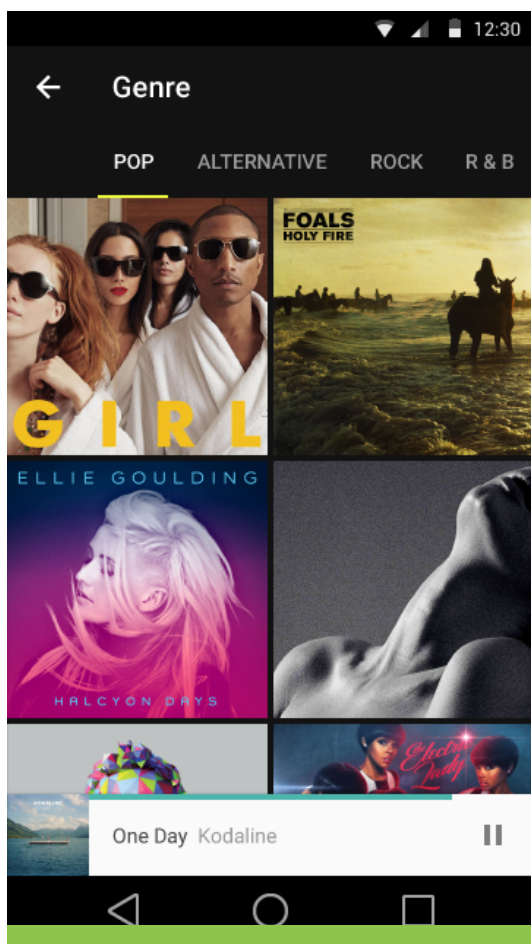
切换标签相对应可见内容要高亮显示。

切换标签归类并且每组切换标签是依次与他们的内容连接起来。

保持切换标签与内容相近，可以帮助维护两者之间的关系，过大的分离会引起歧义。

向下箭头表示内容已经被收起，向上箭头表示内容已经被展开。





内容

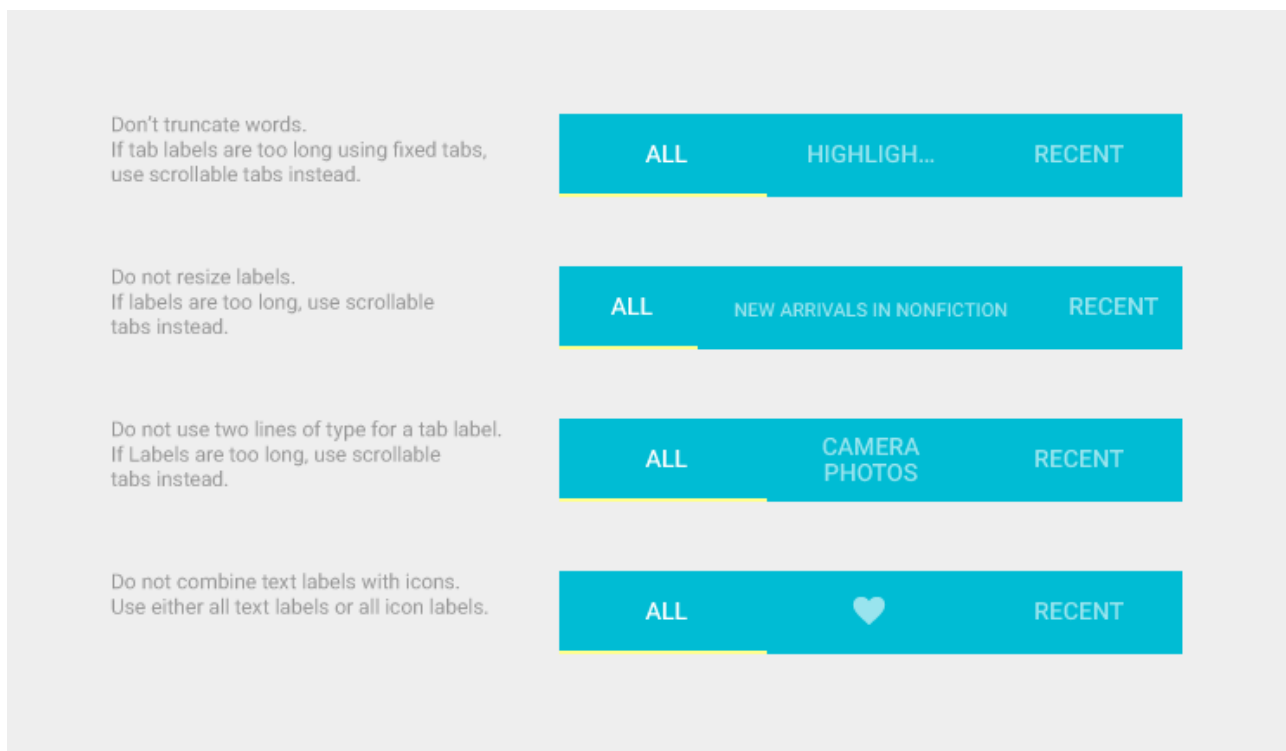
切换标签的内容

即使是两个标签之间，标签中呈现的内容可以有很大的差别。比如，不同年份的标签组合或者是不同类型的设置。

一组标签中的所有内容应该是相关联的，但是每个标签又是相排斥的（例如，设置和指南）。

标签应该在逻辑组织相关的内容上，提供有意义的区分。

标签可能是图标或者文字并且不能被省略。



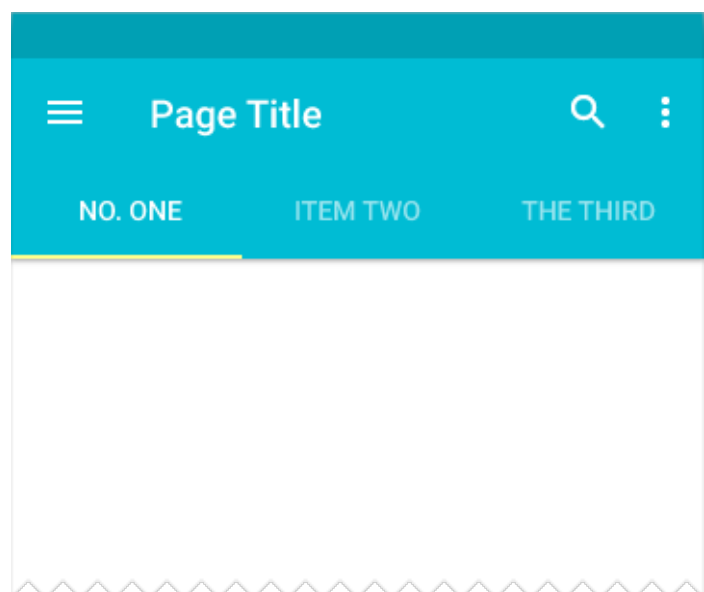
避免对比内容的需求；显著对比内容可能暗示内容将受益于不同的组织或表现。

切换标签的种类

根据平台和使用的上下文，标签内容可以表现为固定的标签或者是滚动（滑动）的标签。

固定的标签

固定的标签最适合用于同时显示所有标签，优势是利于标签之前快速切换（例如，在地图中切换线路的交通方式）。

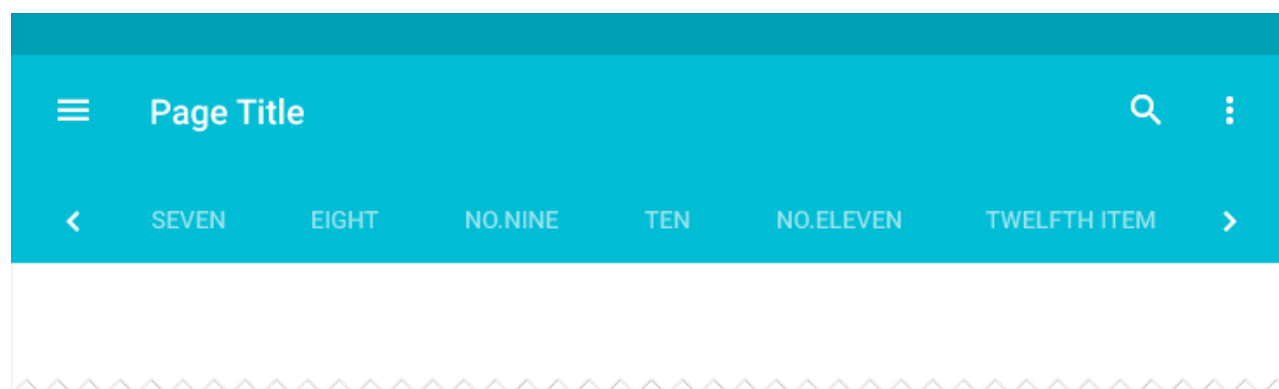
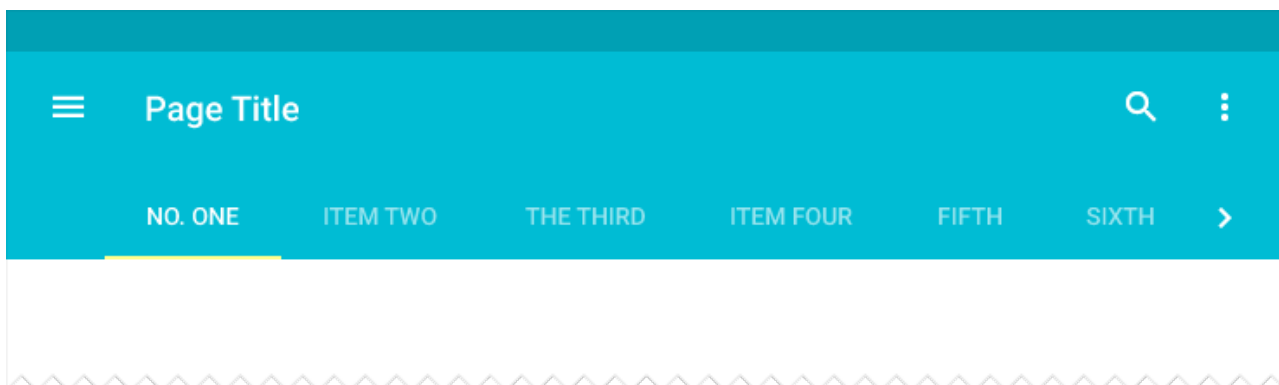


标签的最大数量是要以视图的宽度为限制。固定的标签以最宽的标签的宽度为基础设置每个标签的宽度。点击标签或者是左右滑动内容区域都可以切换固定的标签。

可滚动的标签

滚动标签是显示标签的一个子集，在一些特殊的情况下它可以包含较长的标签或更多数量的标签。他们最好是用于浏览上下文联系时，用户不需要直接触发标签就可以切换。

可滚动的标签之间通过点击选项，左右滑动标签或者左右滑动内容区域来导航



规格

固定的宽度

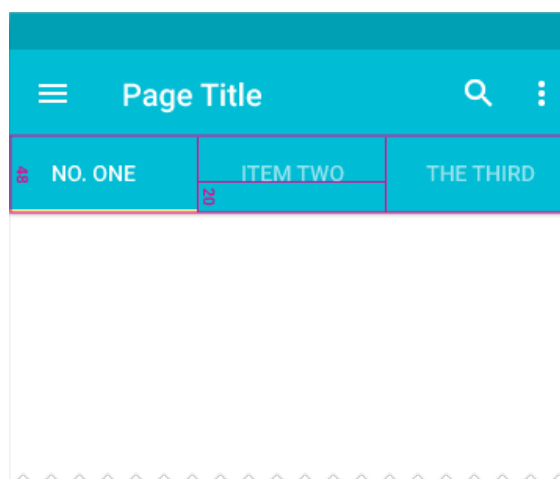
标签宽度: 屏幕的 1/3

标签指示器: 2 dp 高度

文本: 14 sp Roboto 中号，文本在标签单元中居中

当前文字颜色: #fff 或 间色(secondary color)

未选中文字颜色: #fff 60%



可滚动的

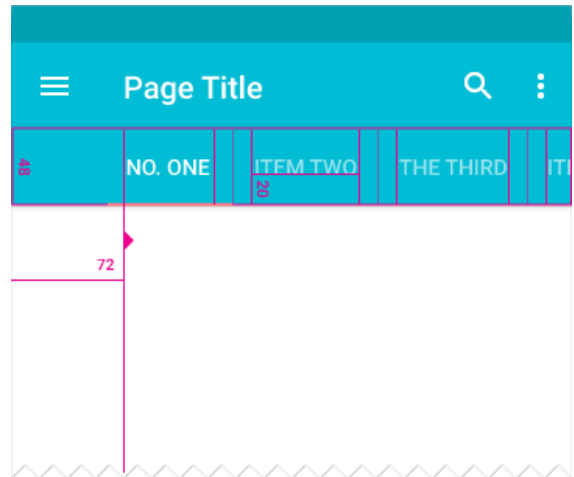
标签宽度: 12 dp + 文本宽度 + 12 dp

标签指示器: 2 dp 高度

文本: 14 sp Roboto 中号

当前文字颜色: #fff 或 间色(secondary color)

未选中文字颜色: #fff 60%



桌面/平板

标签宽度: 24 dp + Word length + 24 dp

标签指示器: 2 dp height

文本: 14 sp 平板, 13 sp 桌面 Roboto 中号

当前文字颜色: #fff 或 间色(secondary color)

未选中文字颜色: #fff 60%



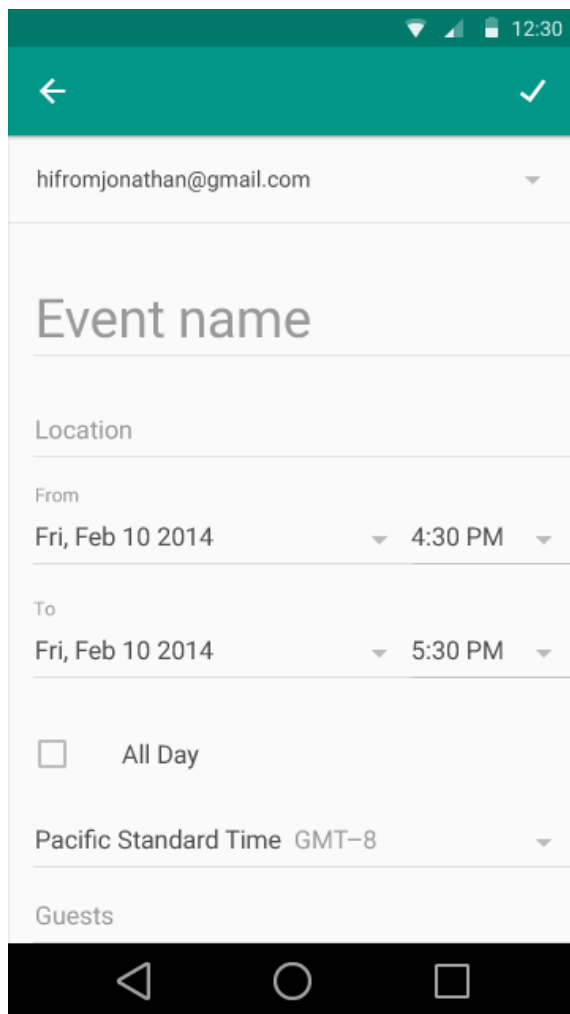
5.17 文本框

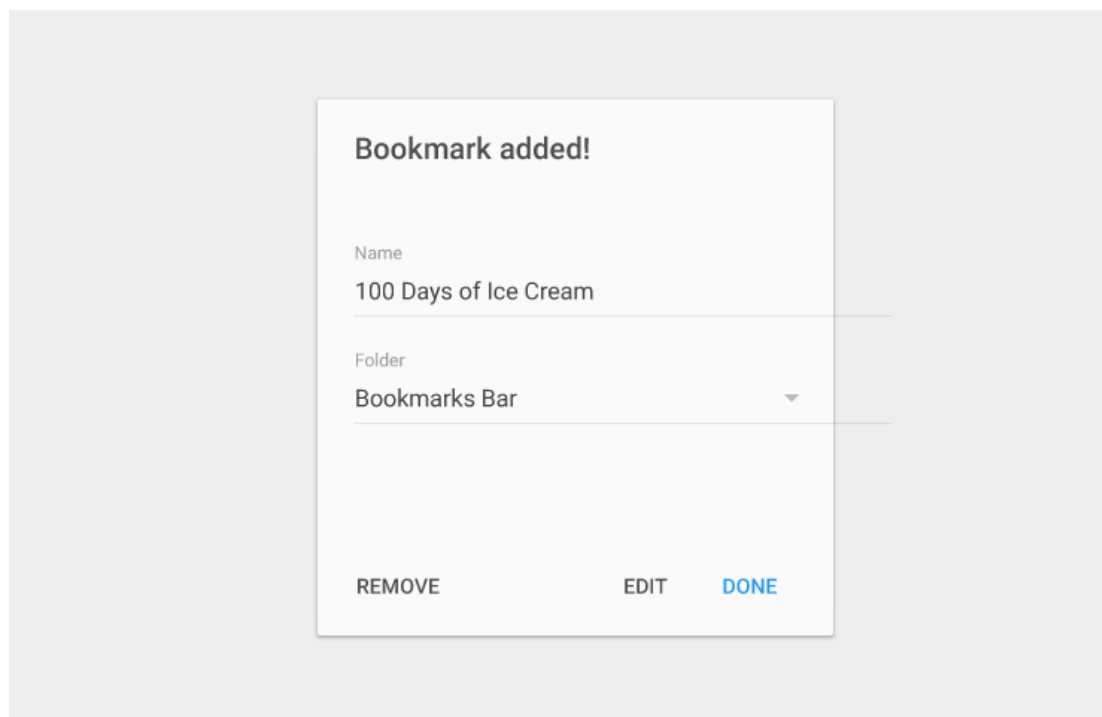
文本框可以让用户输入文本。它们可以是单行的，带或不带滚动条，也可以是多行的，并且带有一个图标。点击文本框后显示光标，并自动显示键盘。除了输入，文本框可以进行其他任务操作，如文本选择（剪切，复制，粘贴）以及数据的自动查找功能。

文本字段可以有不同的输入类型。类型决定文本框内允许输入什么样的字符，可能会提示虚拟键盘调整其布局来显示最常用的字符。常见的类型包括数字，文本，电子邮件地址，电话号码，个人姓名，用户名，地址，街道地址，信用卡号码，PIN码，以及搜索查询。

单行文本框

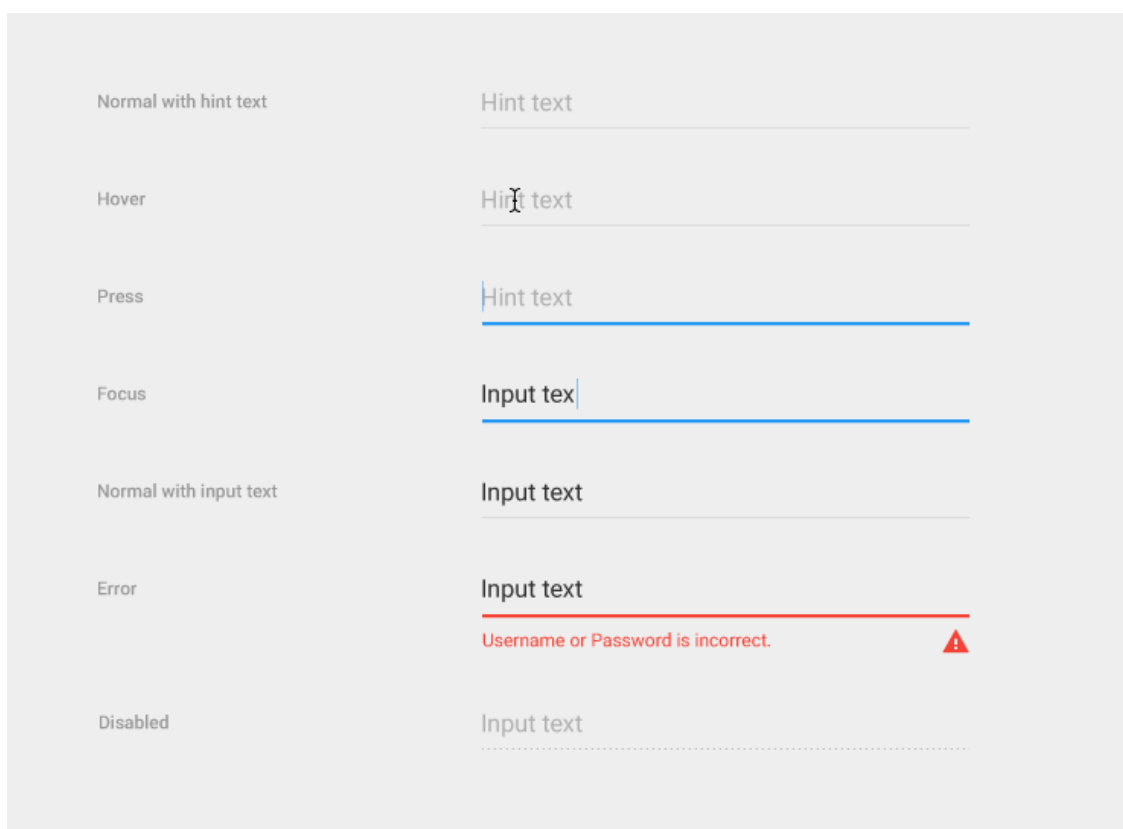
当文本输入光标到达输入区域的最右边，单行文本框自动滚动到内容的左边。

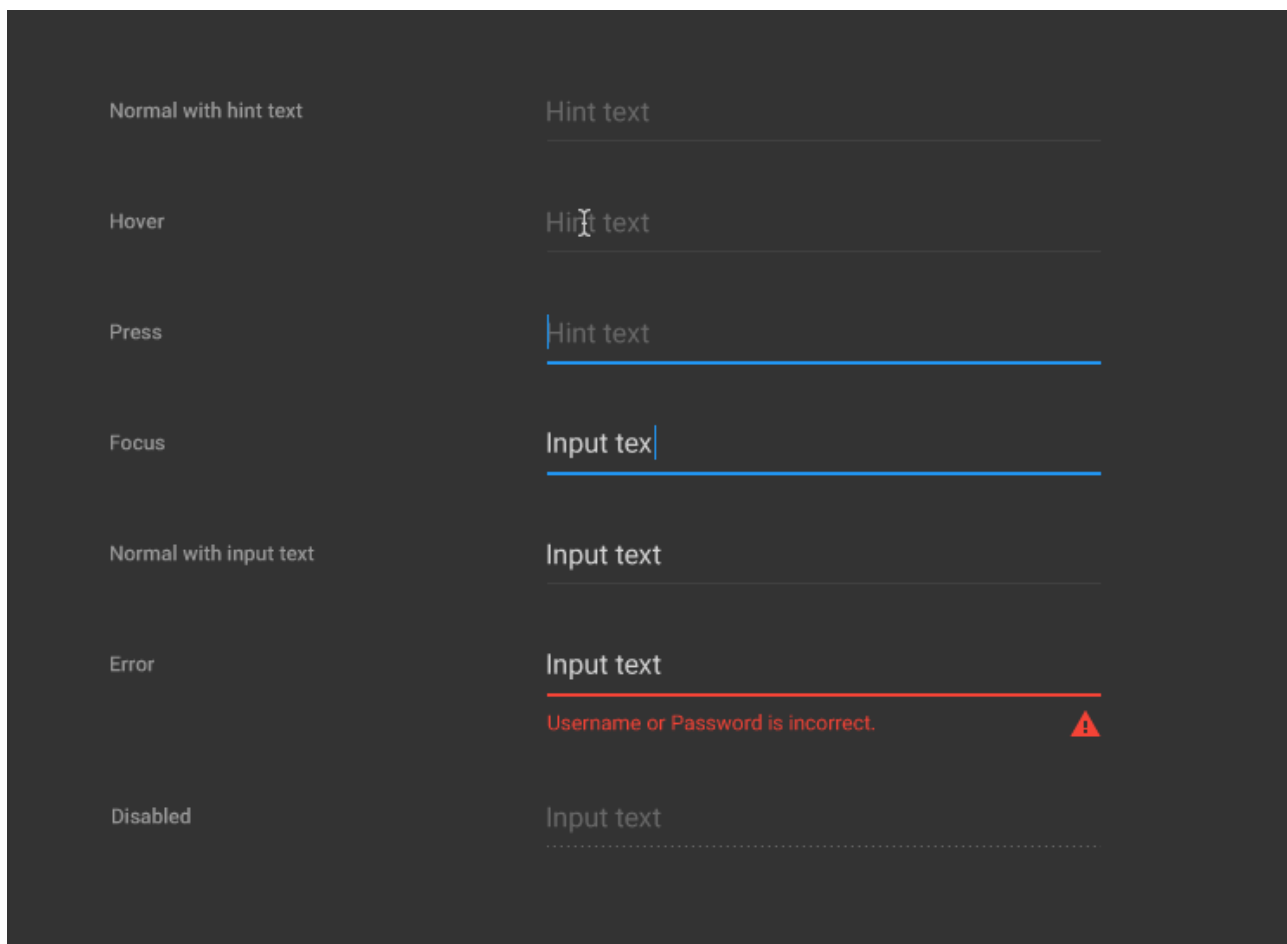
The image shows a mobile application interface for creating an event. At the top is a teal header bar with a back arrow on the left and a checkmark on the right. Below the header is a white card with a light gray border. The first field is an email address, "hifromjonathan@gmail.com", with a dropdown arrow on the right. Below this is a large text input field labeled "Event name" in a light gray font. Underneath the "Event name" field is a "Location" label. Further down are two date and time pickers: "From" (Fri, Feb 10 2014, 4:30 PM) and "To" (Fri, Feb 10 2014, 5:30 PM), each with a dropdown arrow. Below these is a checkbox labeled "All Day". Underneath is a time zone selector showing "Pacific Standard Time GMT-8" with a dropdown arrow. At the bottom of the card is a "Guests" label. The entire interface is set against a dark background, and the bottom of the screen shows the standard Android navigation bar.



浅色主题

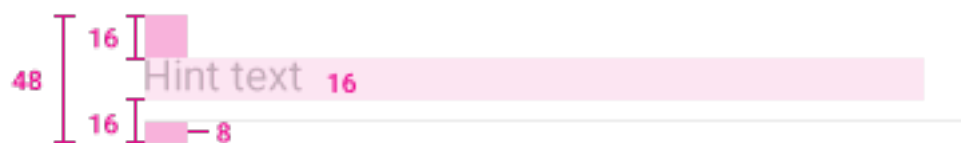
- 提示和输入字体：roboto regular 16 sp
- 输入框高度：48 dp
- 文本顶部和底部填充：16 dp
- 文本字段分隔填充：8 dp





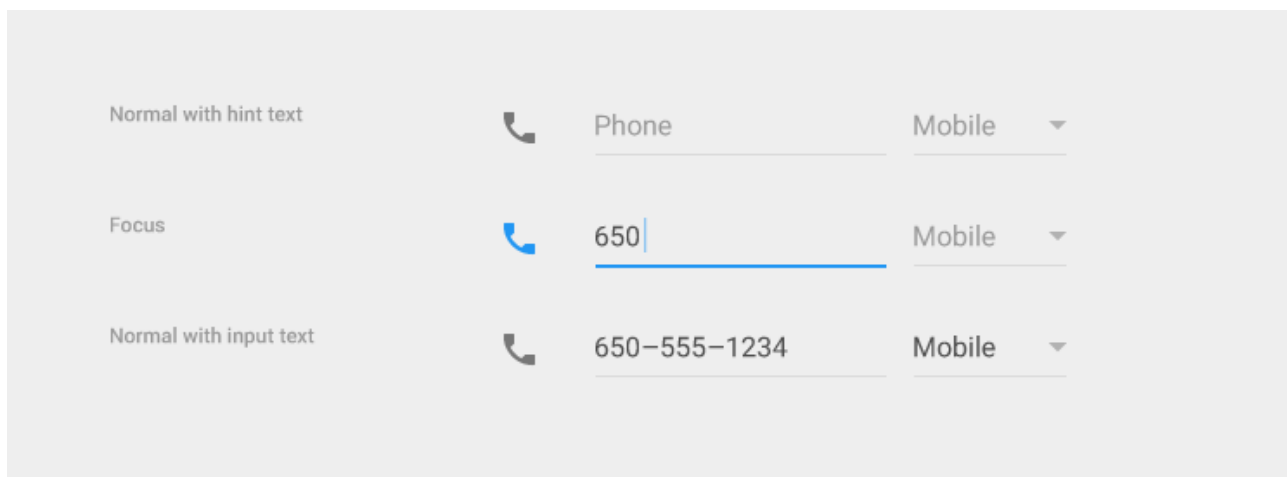
暗色主题

红色标注

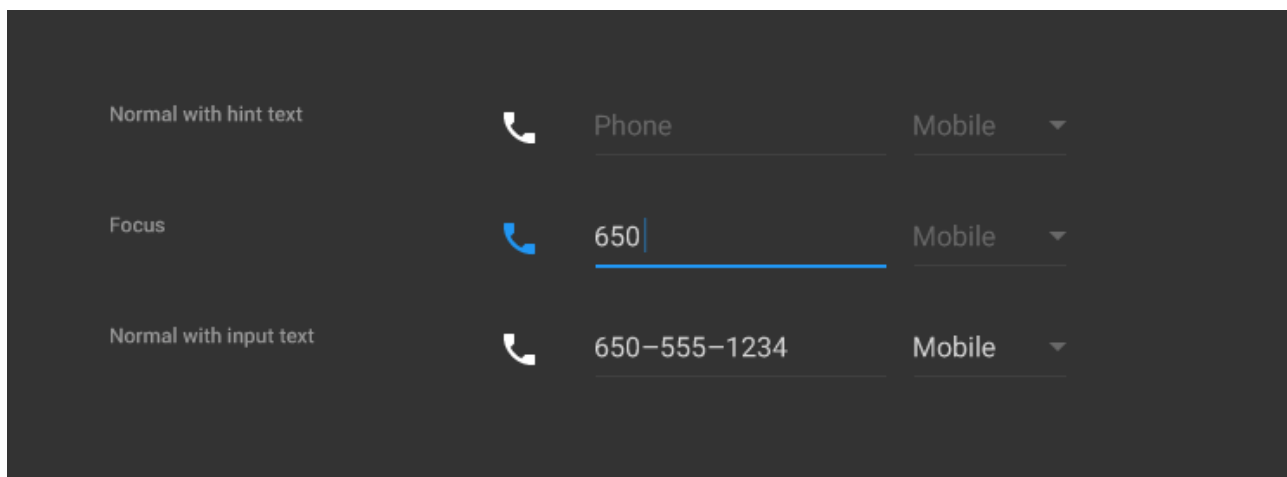


带有图标的浅色主题

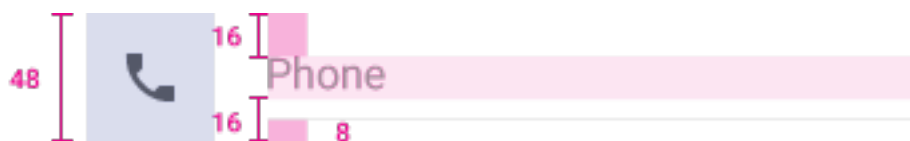
- 提示和输入字体：roboto regular 16 sp
- 输入框高度：48 dp
- 文本顶部和底部填充：16 dp
- 文本字段分隔填充：8 dp



带有图标的暗色主题



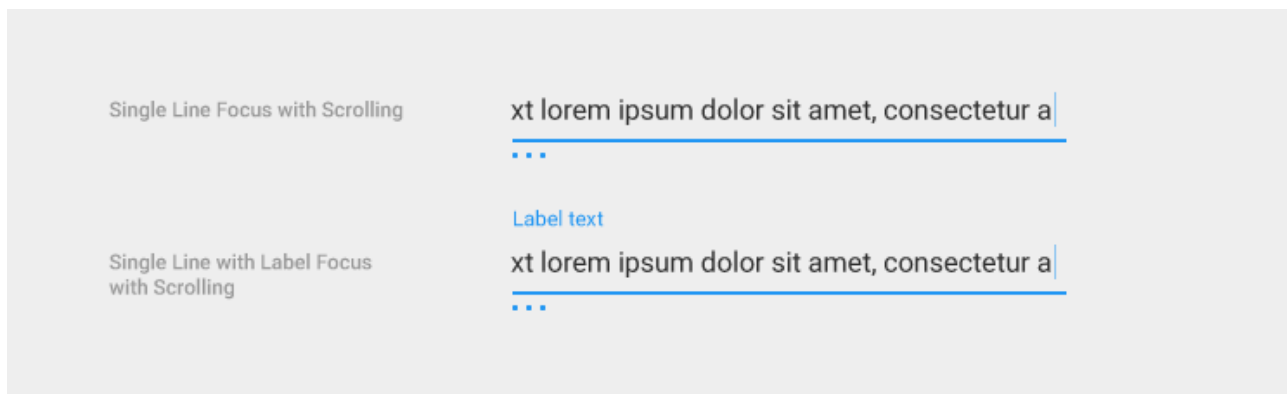
红色标注



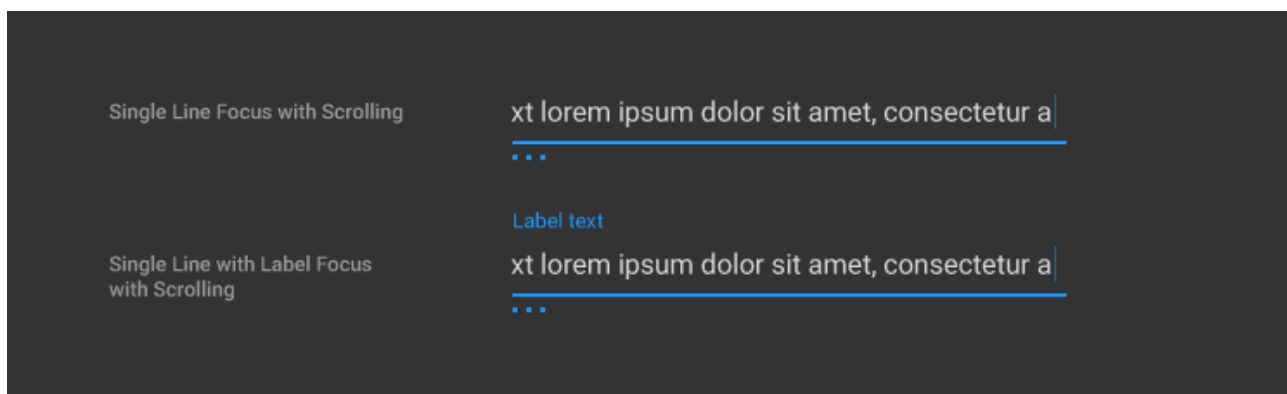
单行带有滚动条

当在单行文本字段输入过长，需跨越多行，则文本字段应该以滚动形式容纳文本。

在滚动文本字段，一个图形化的指示灯出现在线的下面。点击省略号，光标返回到字符的开头。



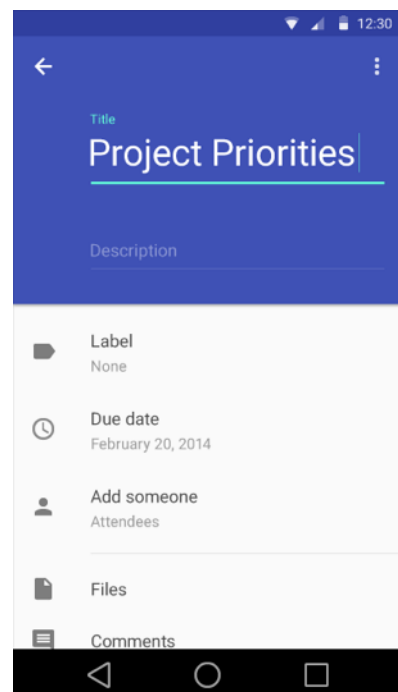
浅色主题



深色主题

浮动标签

浮动内嵌标签，当用户进行文本输入，标签漂浮在上方



浅色主题

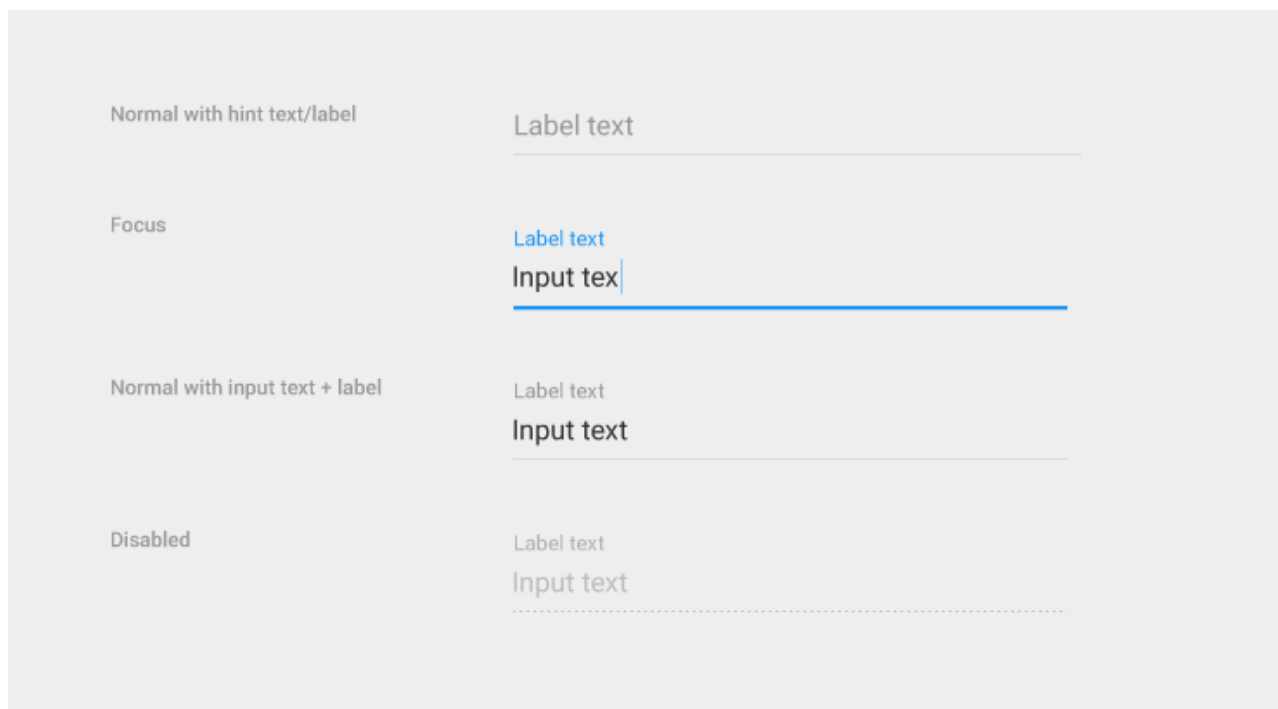
提示和输入字体：roboto regular 16 sp

标签字体：roboto regular 12 sp

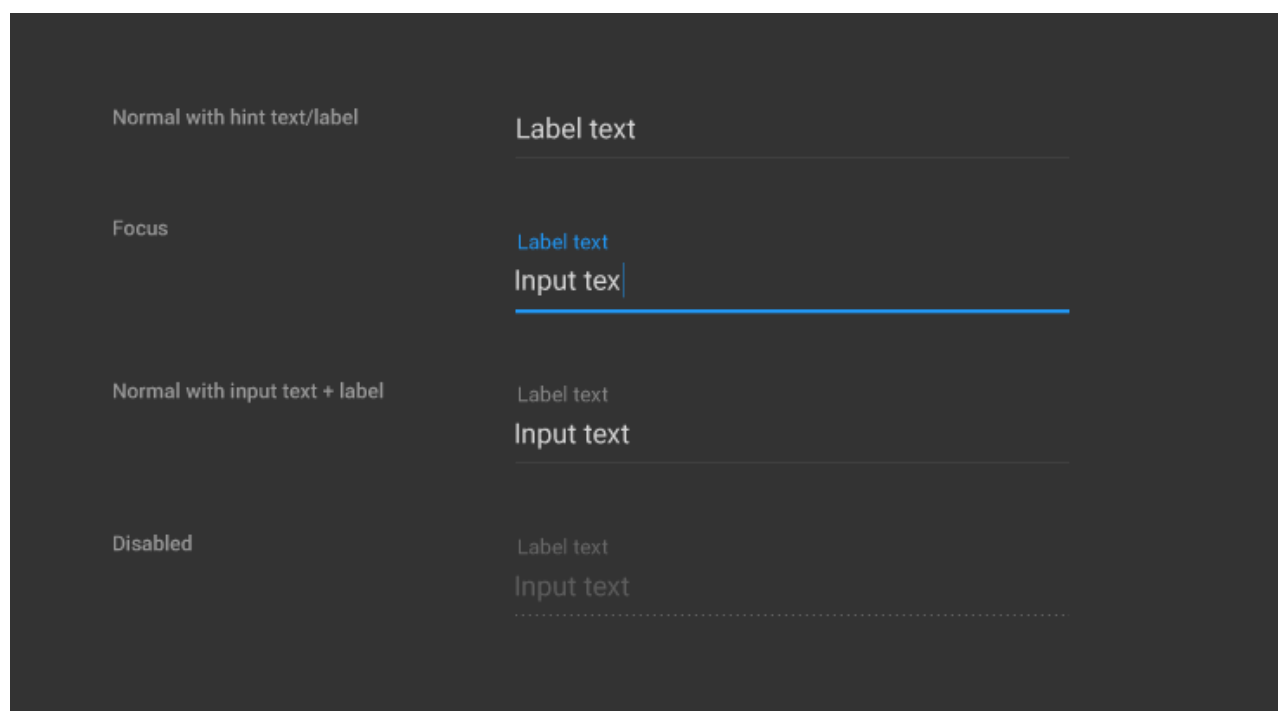
输入框高度：72 dp

文本顶部和底部填充：16 dp

文本字段分隔填充：8 dp



深色主题

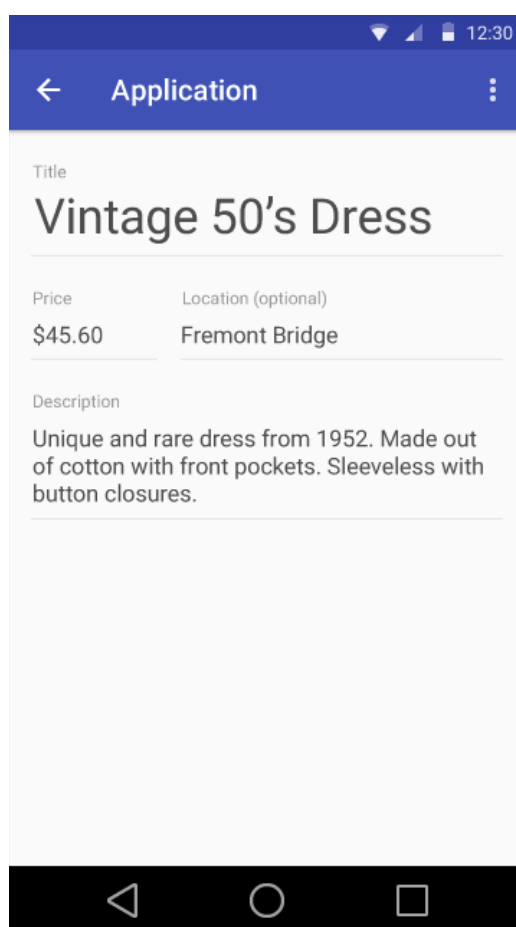
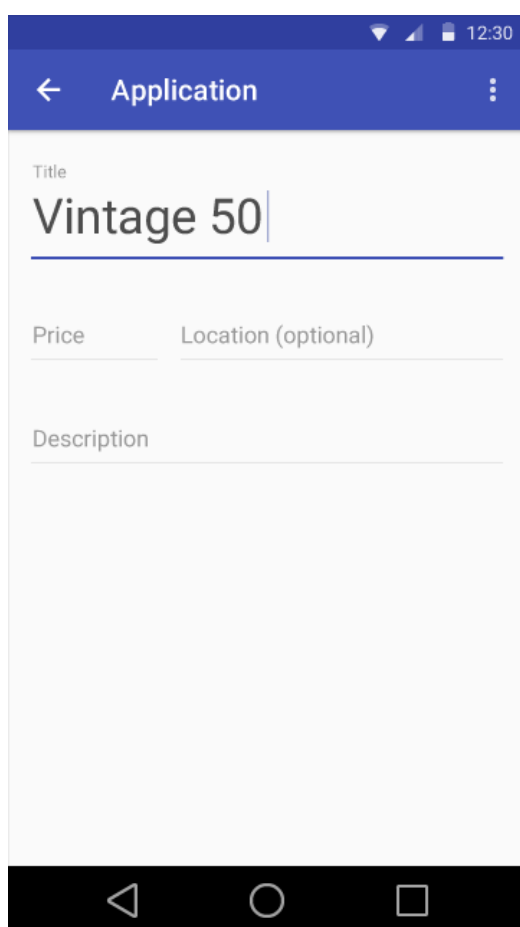


红色标注



多行文本框

当光标到达最下缘，多行文本自动断开，使文本可以换行和垂直滚动



浅色主题

提示和输入字体：roboto regular 16 sp

标签字体：roboto regular 12 sp

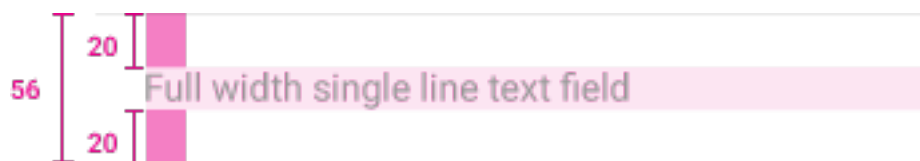
文本顶部和底部填充：16 dp

文本字段分隔填充：8 dp

Normal with hint text	Hint text
Focus	Input tex
Normal with multi-line input text	Lorem ipsum dolor sit amet, consectetur sit adipiscing elit. Vestibulum porta in diam vel ullamcorper tempus.
Normal with multi-line input text	Label text Lorem ipsum dolor sit amet, consectetur sit adipiscing elit. Vestibulum porta in diam vel ullamcorper tempus.

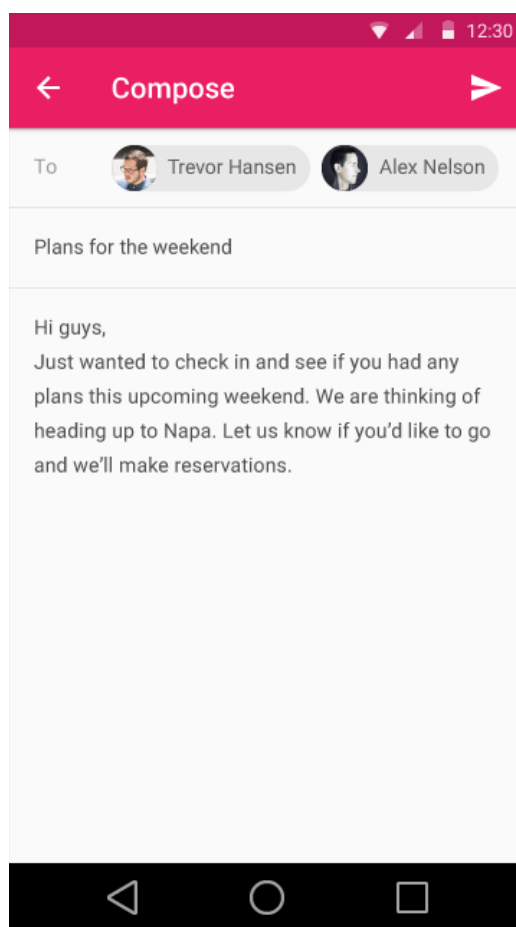
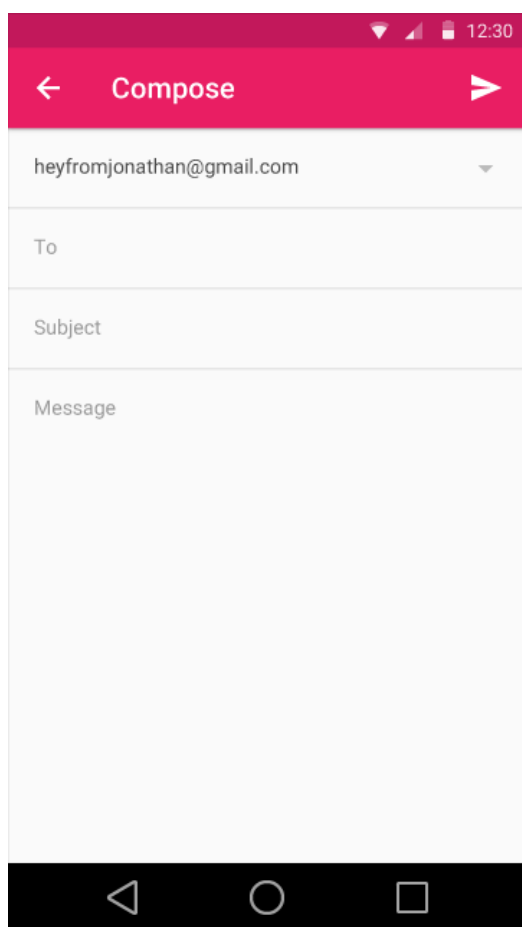
Normal with hint text	Hint text
Focus	Input tex
Normal with multi-line input text	Lorem ipsum dolor sit amet, consectetur sit adipiscing elit. Vestibulum porta in diam vel ullamcorper tempus.
Normal with multi-line input text	Label text Lorem ipsum dolor sit amet, consectetur sit adipiscing elit. Vestibulum porta in diam vel ullamcorper tempus.

红色标注



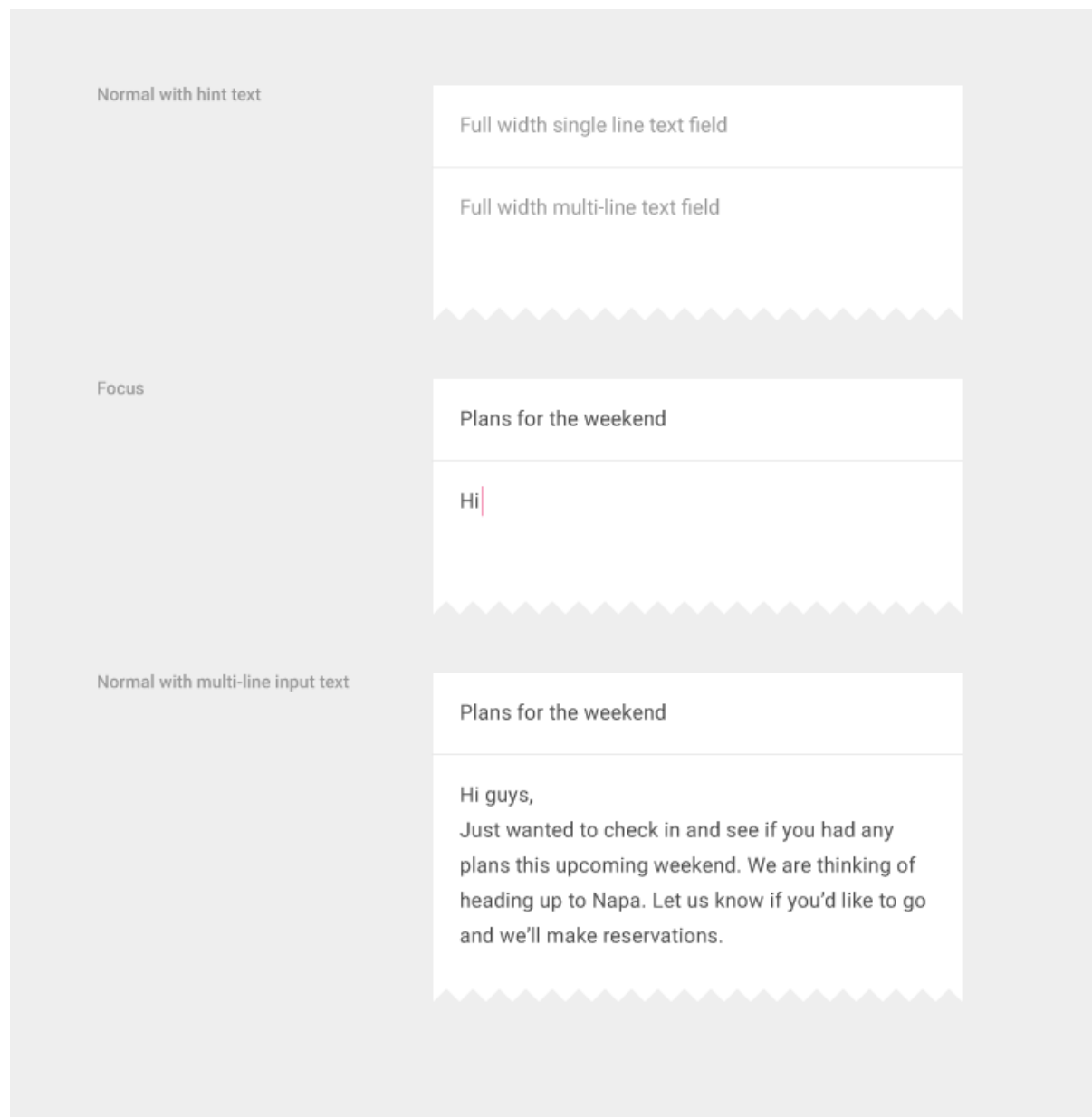
全宽度的文本字段

全宽度的文本字段适用于更深入的工作



单行和多行文本

- 提示和输入字体：roboto regular 16 sp
- 顶部和底部填充文本：20 dp



红色标注

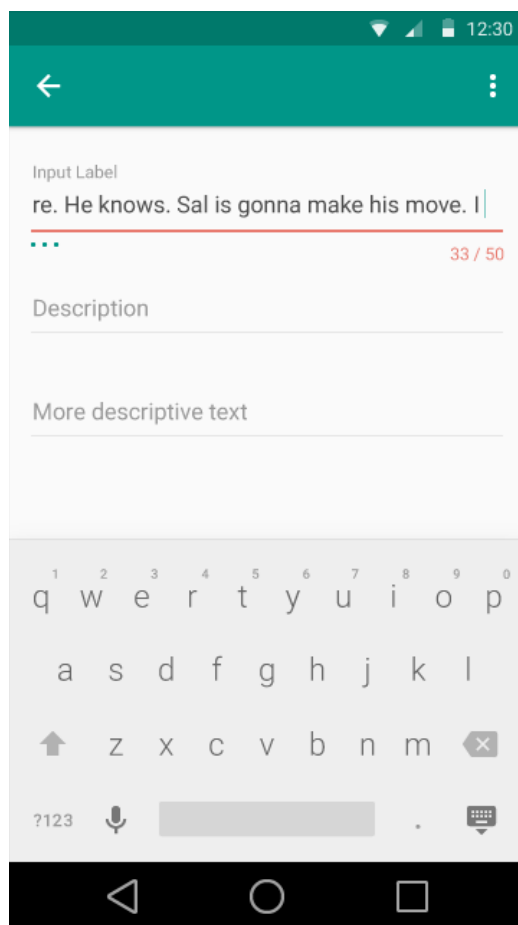
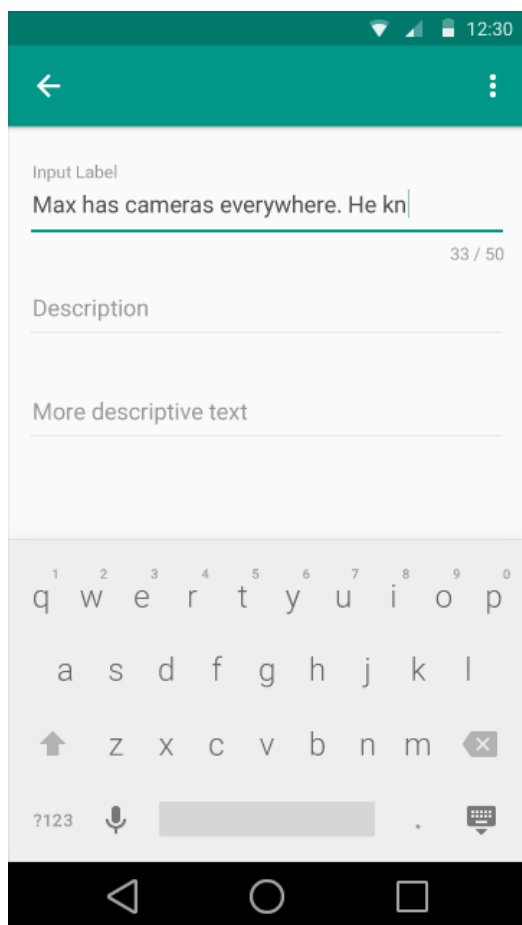


字符计数器

使用字段的字符计数器，能精准限制其中的字符。

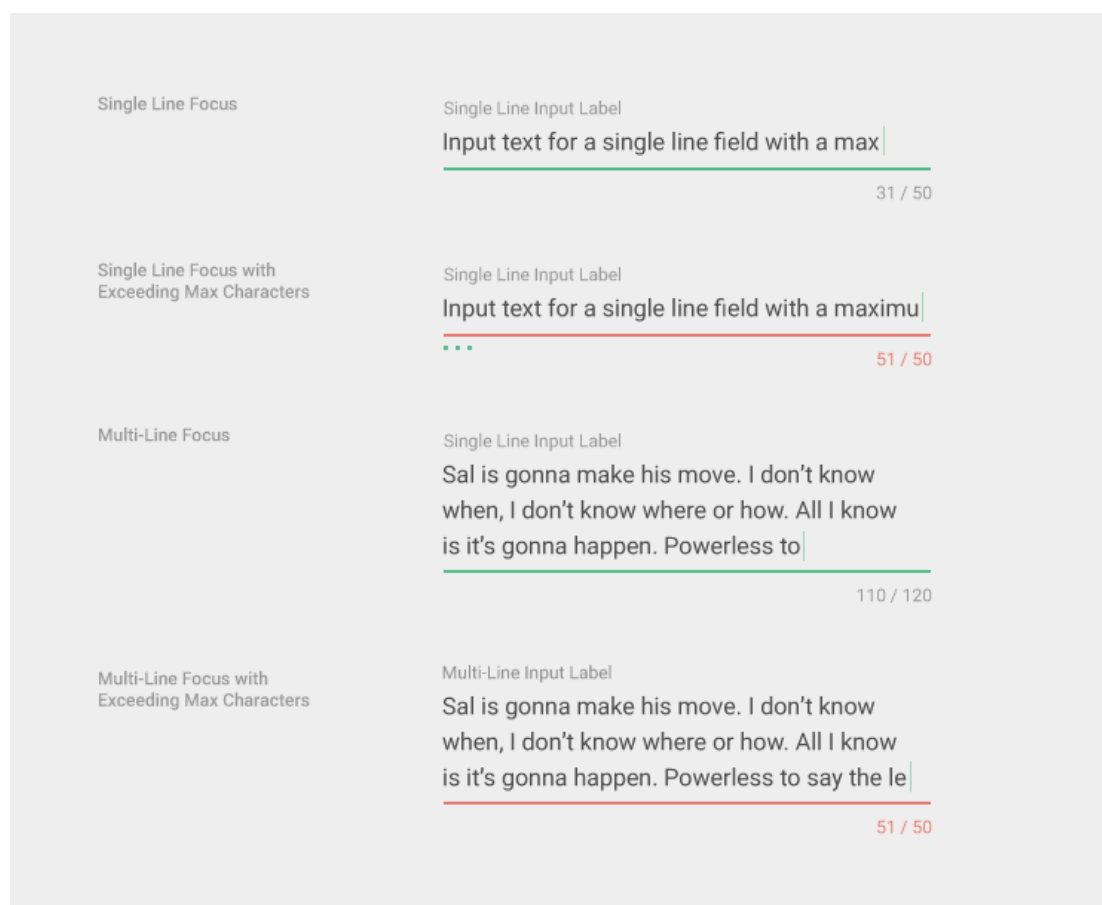
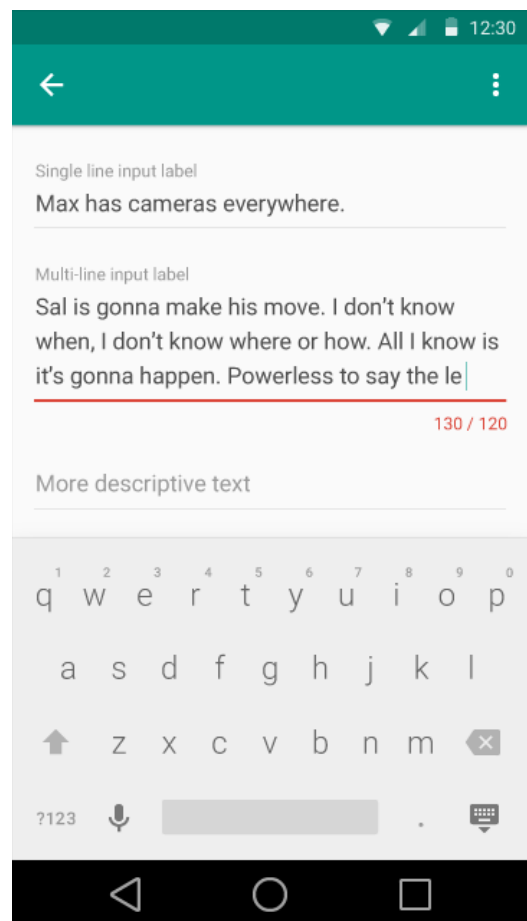
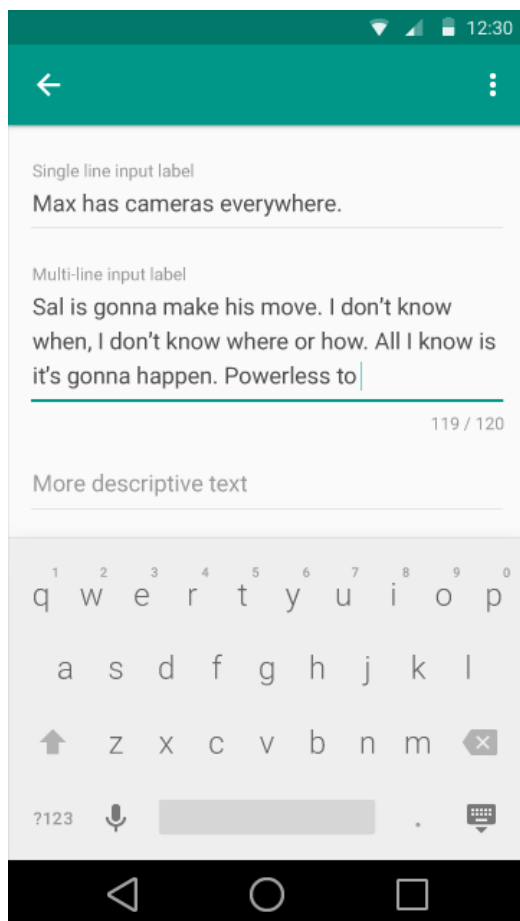
单行字符计数器

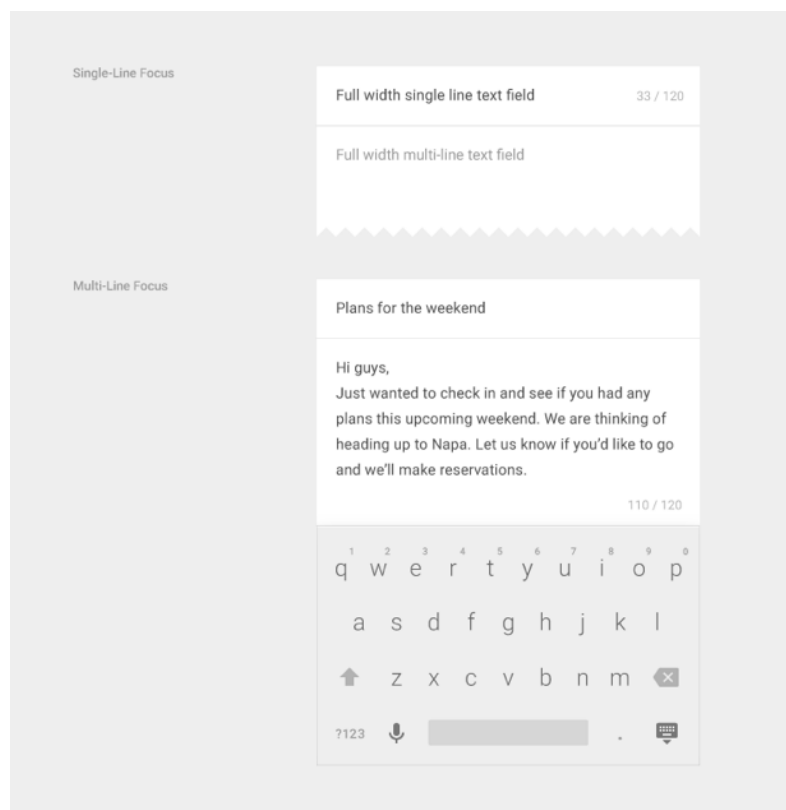
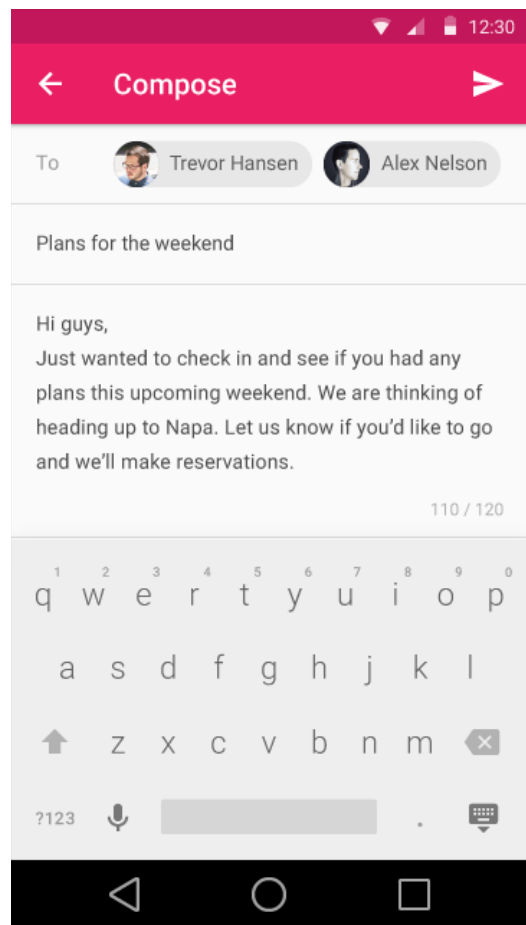
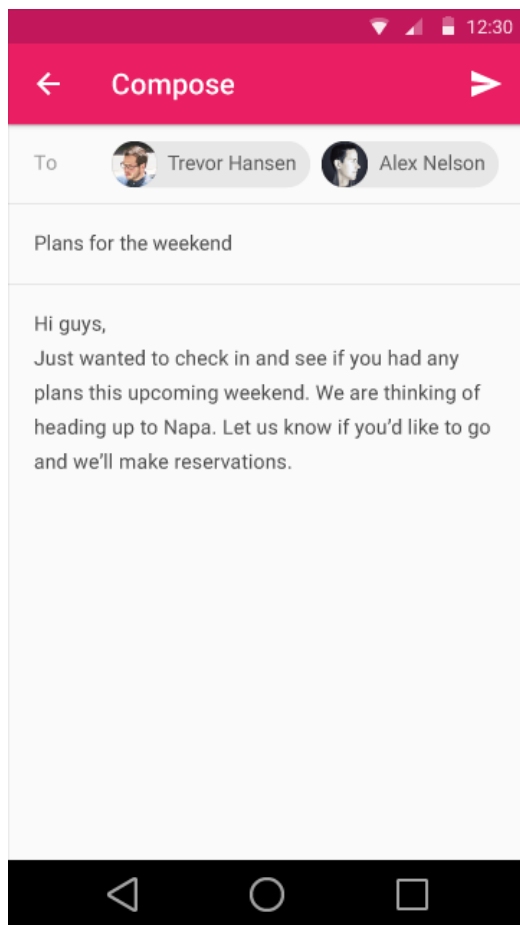
计数器是roboto regular 12 sp



多行字符计数器

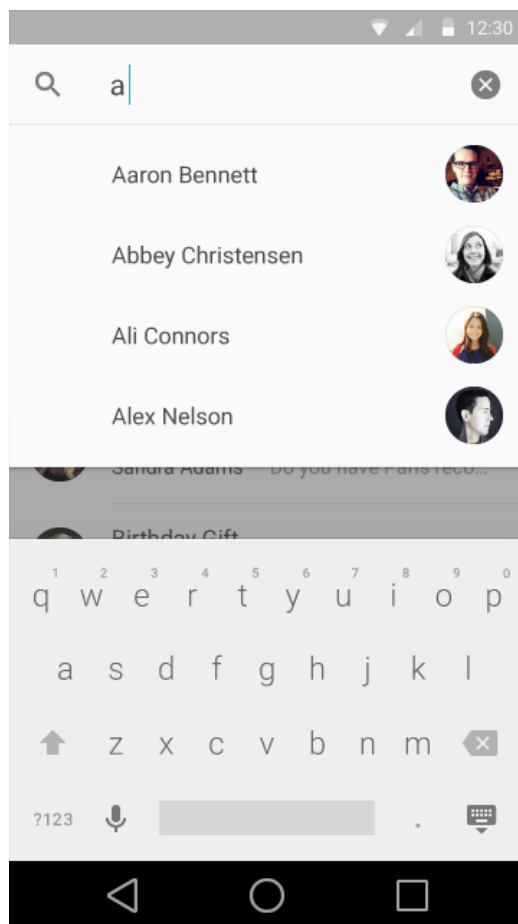
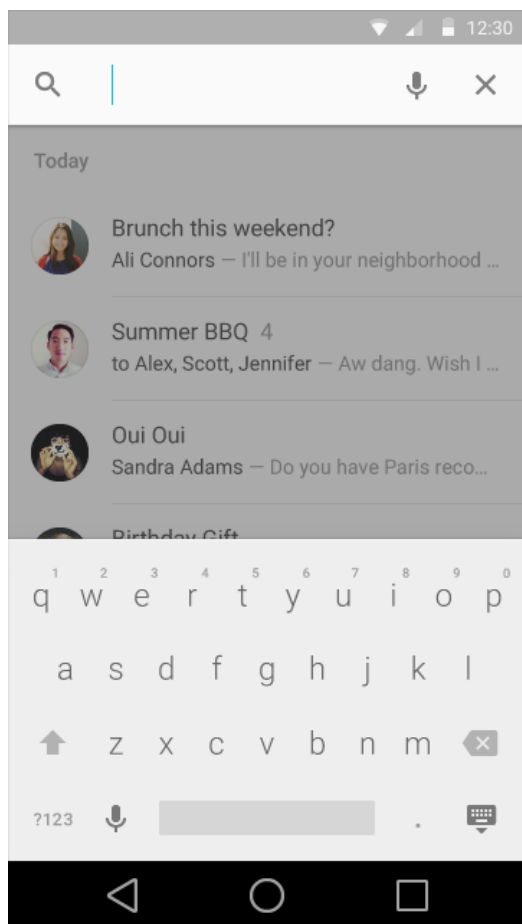
计数器文本 : roboto regular 12 sp



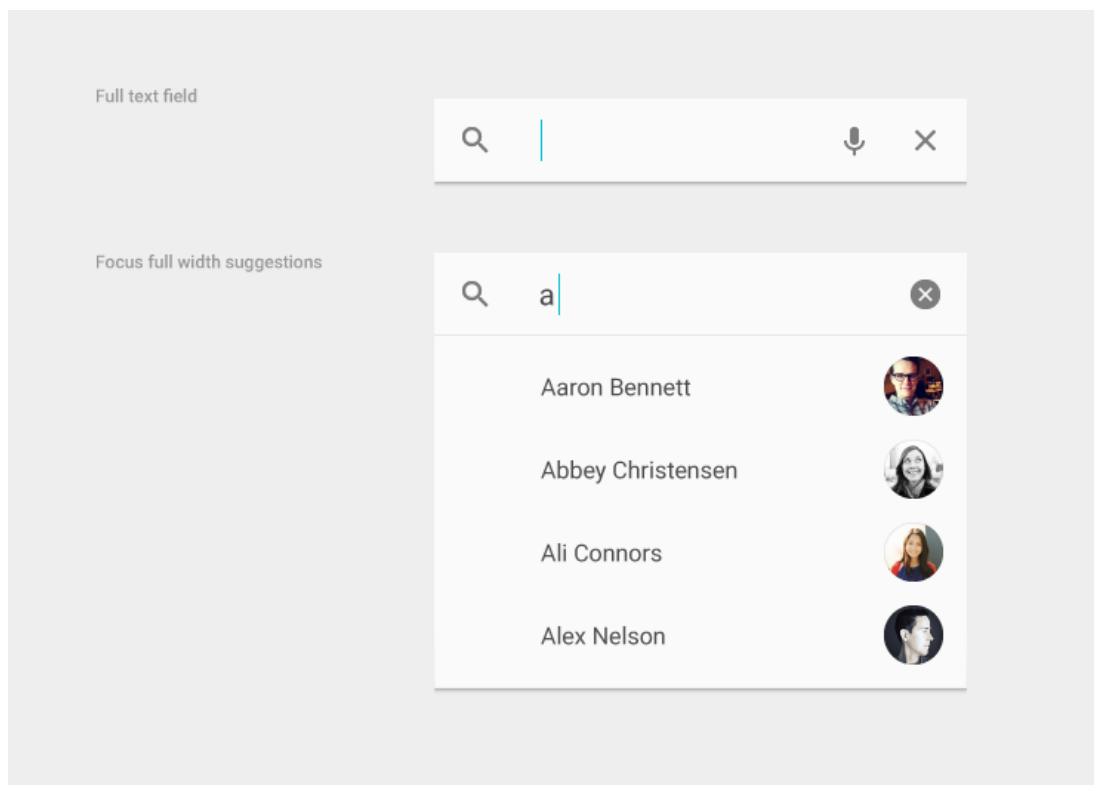


自动完成文本框

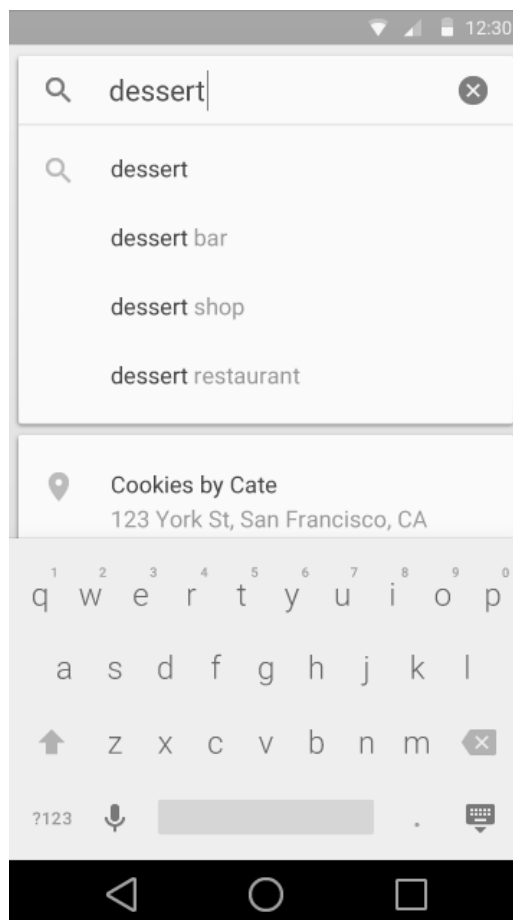
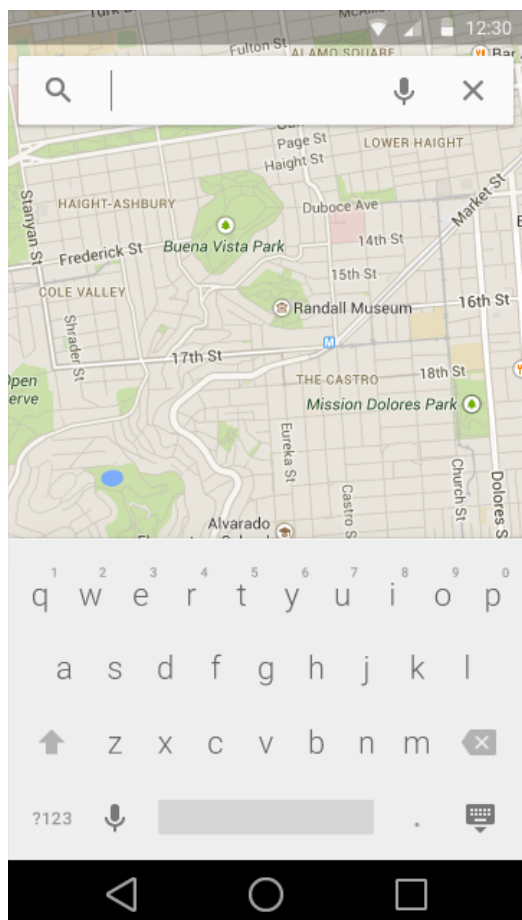
当有窗口弹出实时的建议或自动填充时使用自动完成文本字段，这样用户就可以更准确，更高效地输入信息。

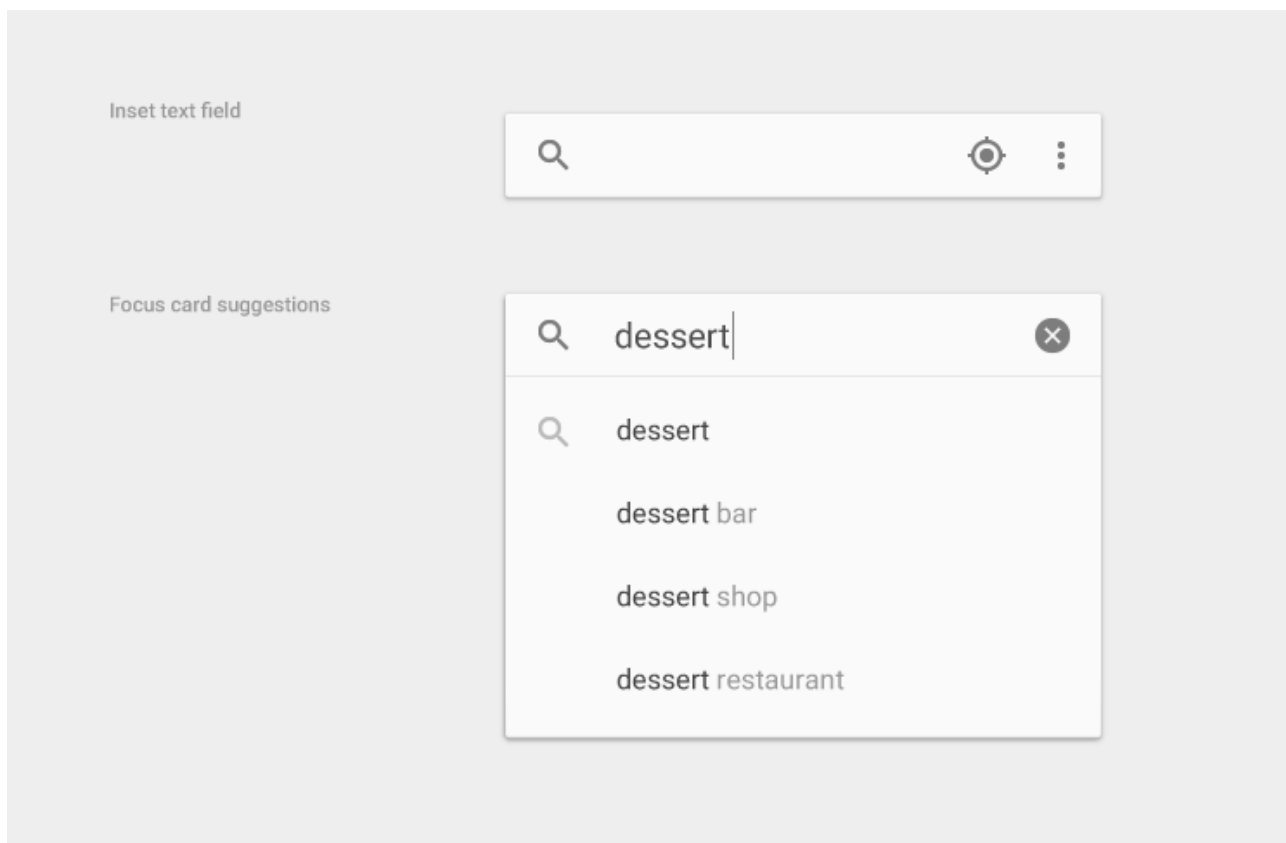


全宽度自动完成

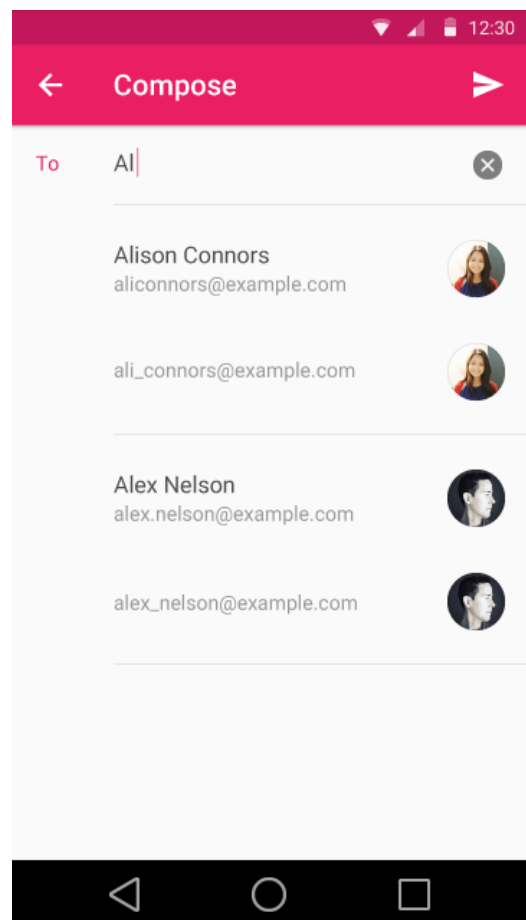
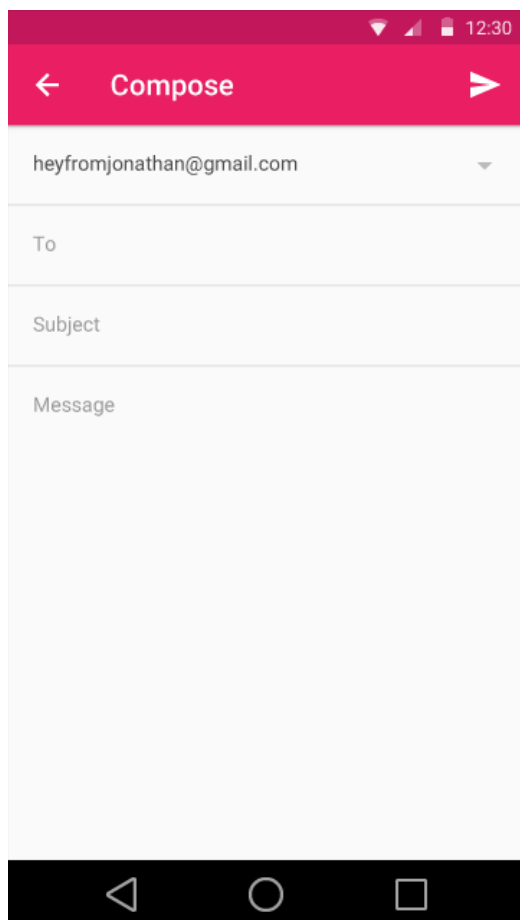


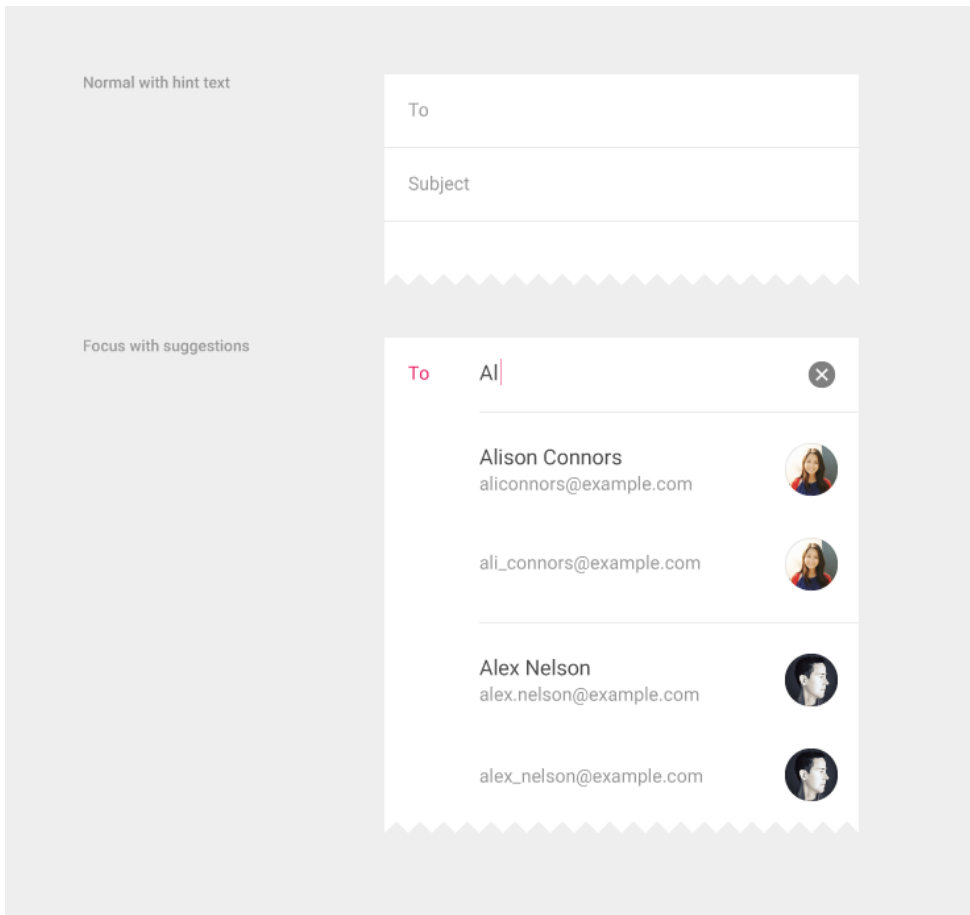
插入自动完成



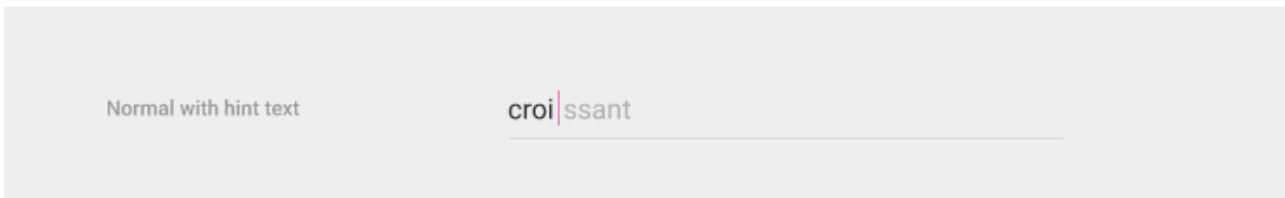


全宽行内自动完成



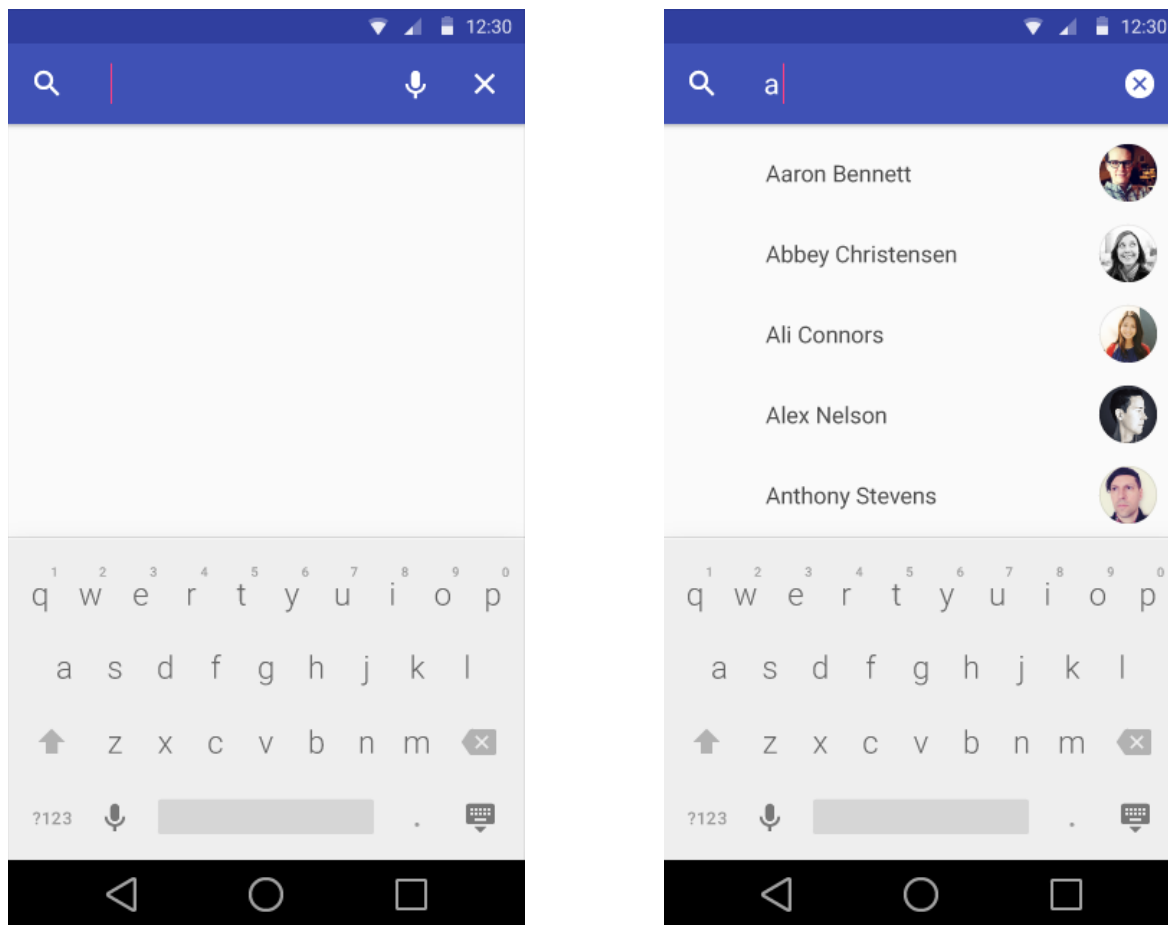


在线自动完成

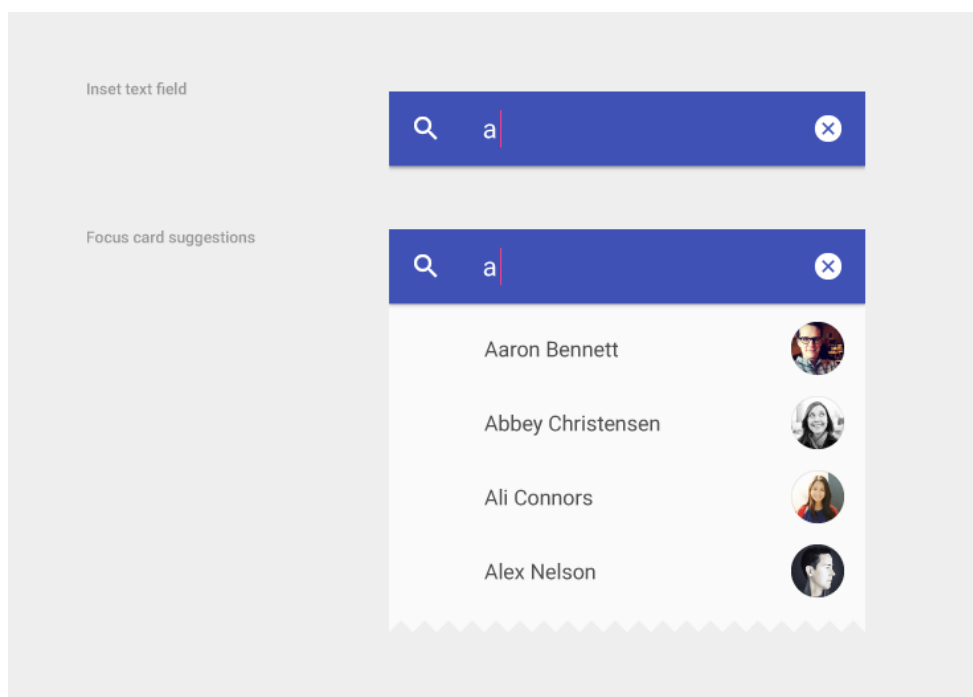


搜索过滤器

应用栏可以作为一个文本输入字段。当用户输入时，对下面的内容进行过滤和排序。



在应用程序栏中的全宽度的文本字段

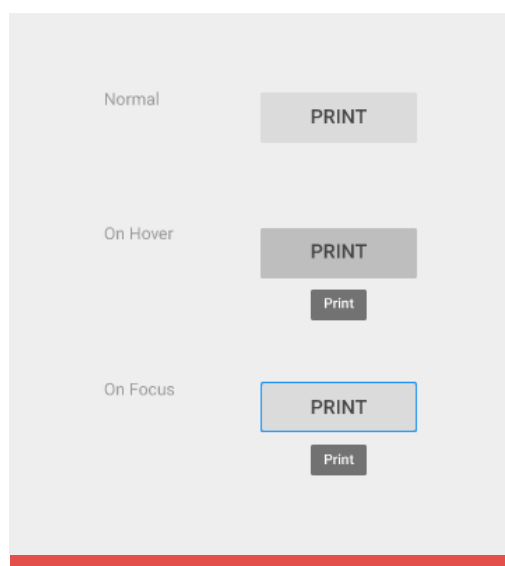
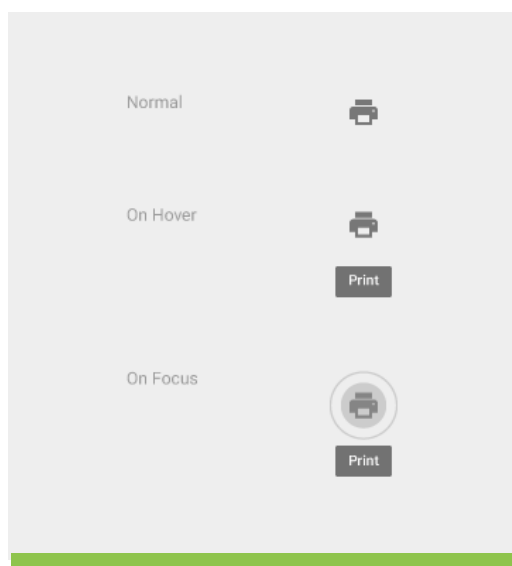


5.18 工具提示

用法

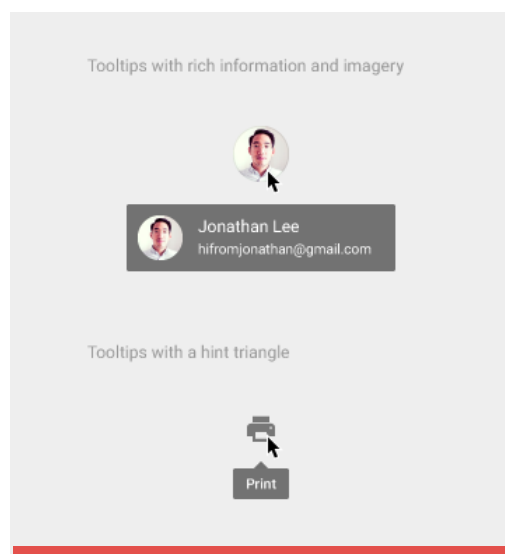
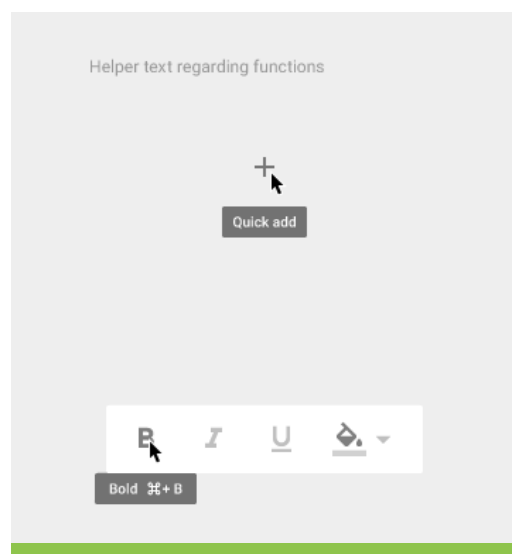
对同时满足以下条件的元素使用工具提示：

- 1.具有交互性
- 2.主要是图形而非文本



工具提示不同于悬浮卡片，后者用来显示图片和格式化的文本等更为丰富的信息。

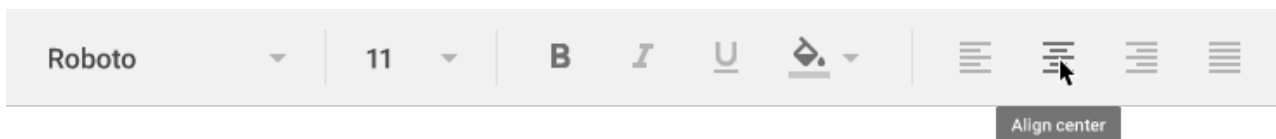
工具提示也不同于注释属性，后者用来提示静态图片的主旨



光标和键盘的工具提示

文本：Roboto Medium 10 sp

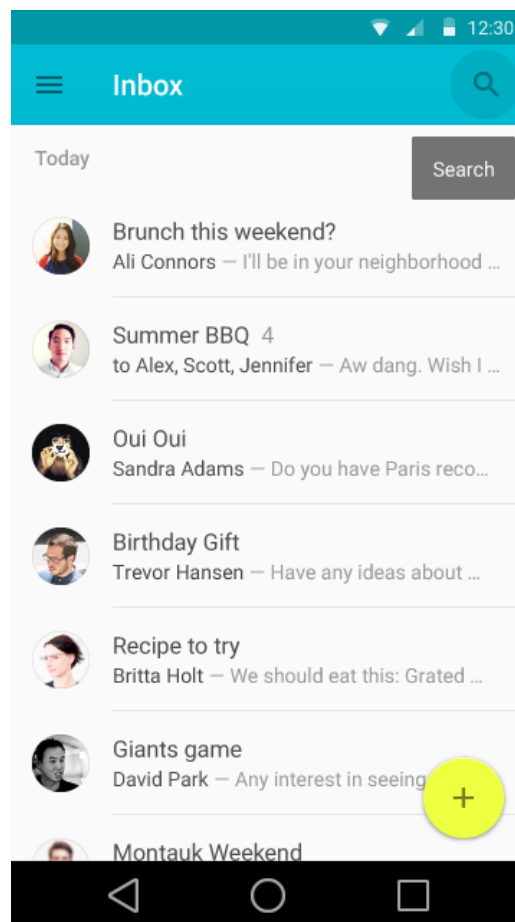
背景填充：90%不透明度



触摸屏UI的工具提示

文本：Roboto Medium 14 sp

背景填充：90%不透明度



Normal



Press + Hold



Refresh

6.1 选择

强烈建议列表和网格容器支持多重选择。当然，不是必须的，有一些选中动作就是需要一次性。

(比如拨号键盘一样，拨打是唯一的动作。),或者一些情境下用户依据自身喜好带有强烈主观意识操作。(就像移动Android主屏幕上的icon一样。)

当启动项目时，使用长按和双指触摸等手势来扩展选择多个项目。拖拽起始点和结束点来将其他内容选中。在桌面端，在空白区域进行简单的拖放也可以选中多个项目（例如，从列表左侧边缘开始拖拽，向右下延伸，就可以选中列表中的项目了）。

一旦选中项目后，可以通过其他操作来改变：

- 触击未选中项目将它选中，或者将已选中的项目取消选中。
- 按住Shift+触摸/点击来在已选中加入其他项目或选中所有项目。

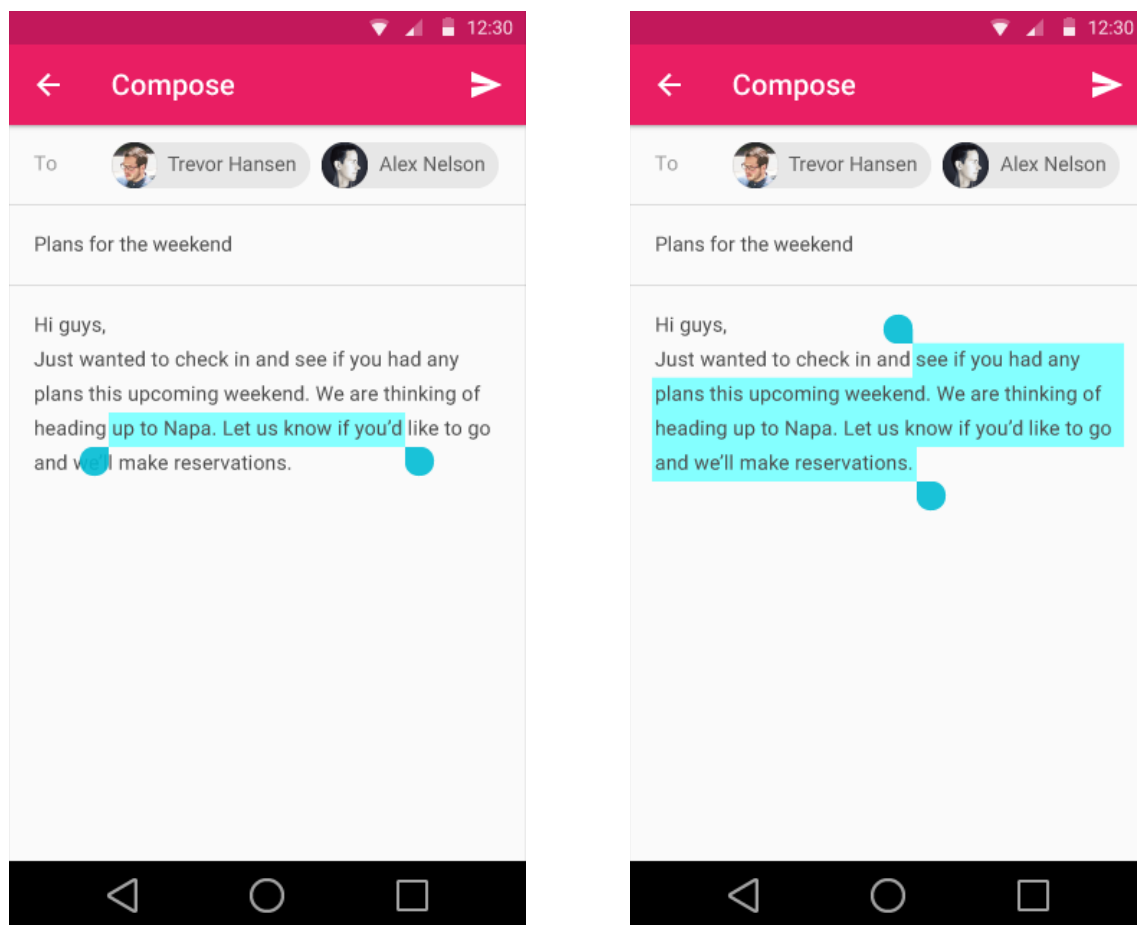
文字选中

被选中的文本会以高亮的方式显示。

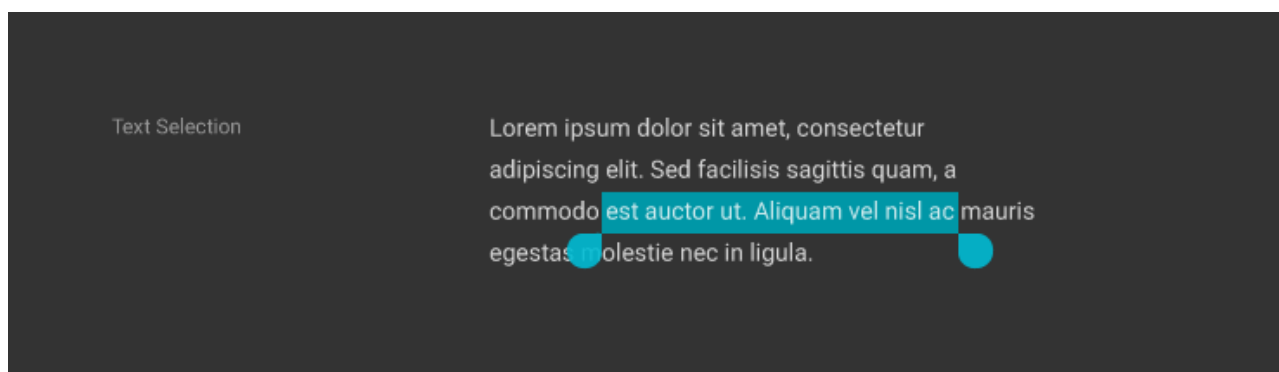
移动平台中，在选中文字的开始和末尾添加上手柄。每当用户激活选中文字时，手柄会出现在文字的上方。

选中文本时

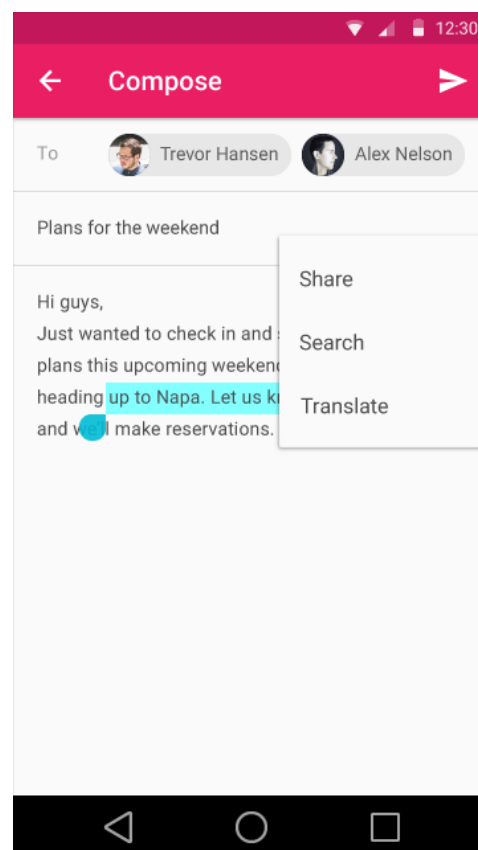
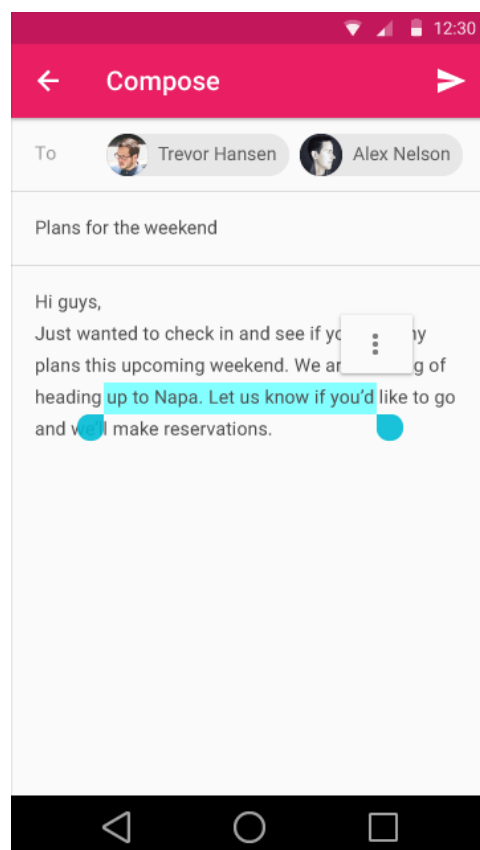
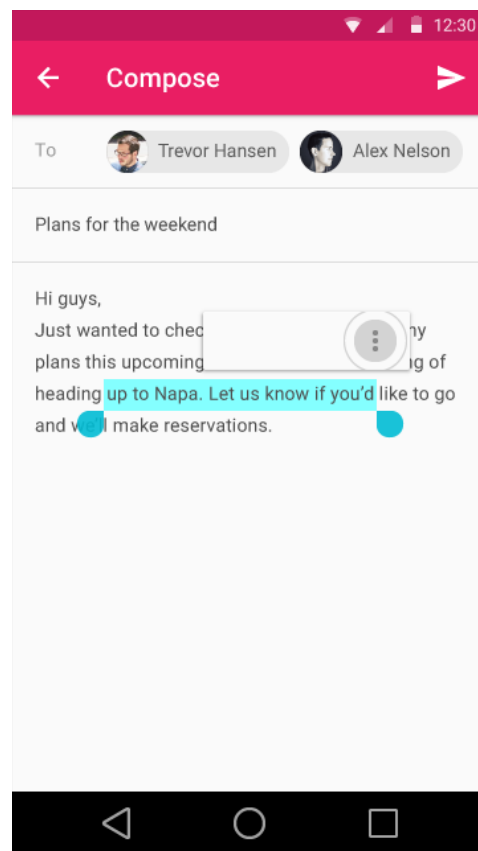
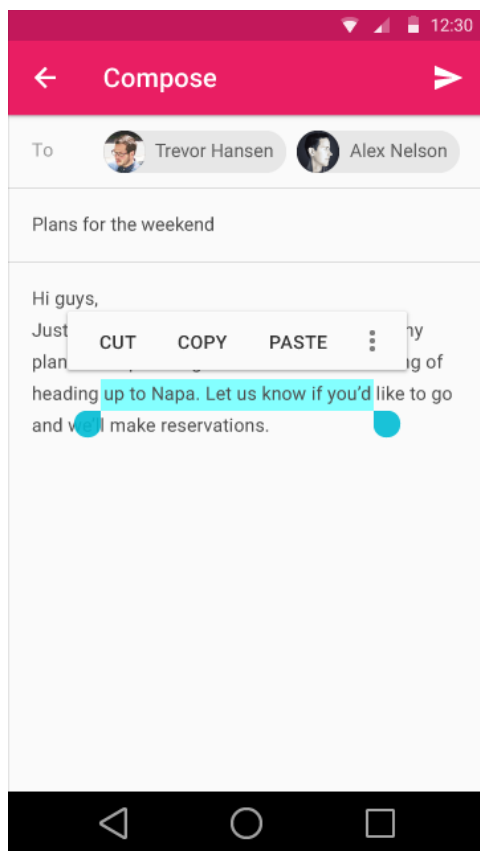
在亮色主题中选中文本的效果

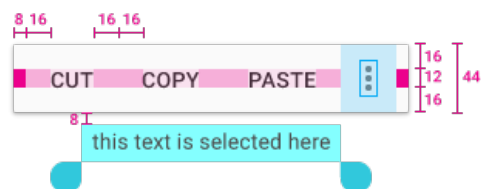


在暗色主题中选中文本的效果



弹出剪切、复制、黏贴和更多选项菜单在选区附近。当用户点中更多按钮时，先收缩，再从图标中心弹出菜单项。文本选项菜单中的字体是Roboto 中号，14 sp（安卓视觉规范单位），全部大写。





文本选择实际上是单向的选择，选中的文字必须是前后连续的。然而，选择的范围可以通过一些操作来改变：

- 初始化选择的过程中拖动来增加选区。
- 拖动选区两端的把手(handle)来增加或者减少选区。
- 在选区内多次点击来增加选区(单词>段落>全文)。

键盘快捷键：

- 按住Shift+左键/右键将字符逐个选中
- Shift+上键/下键将行选中
- Ctrl/Command+A来执行全选

6.2 手势

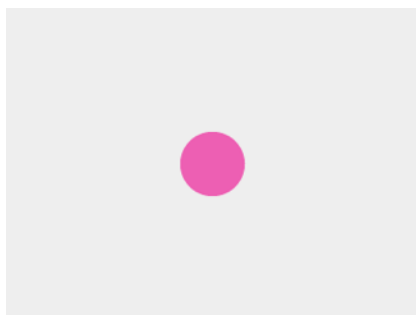
交互手势分为触摸方式（手指在屏幕上的动作）触摸动作（在特殊情境下使用特定的手势来操作）

- 这是因为接触操作（触摸）根据使用情境的不同，可能会有不同的情况（轻击、取消、启用/关闭），而一个触摸动作（放大）也可以通过多次接触来完成（捏放、双击、双击拖移等等）

注意：拖移、滑动和抛掷这些手势依据使用情境会产生不同的效果

触发动作

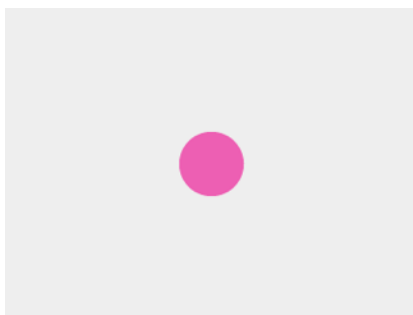
触摸方式：用户在屏幕上如何使用手指操作。



触摸

单根手指按下，拿起

案例：选择

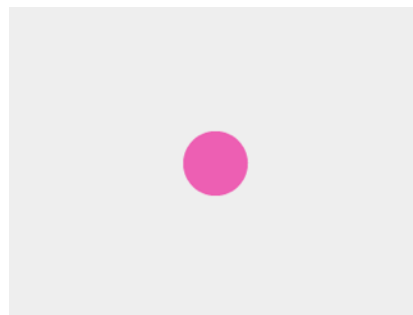


双击

两根手指按下，拿起 / 一根

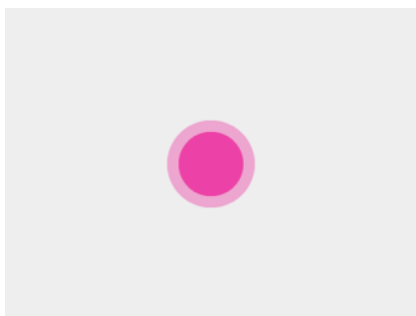
手指按下,拿起

案例：放大



拖拽/滑动/抛掷

一根手指按下后移动，拿起



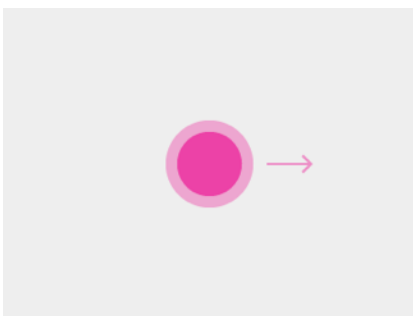
长按

一根手指按下,等待，拿起

案例：选中一个元素，比如

列表中的一个项目

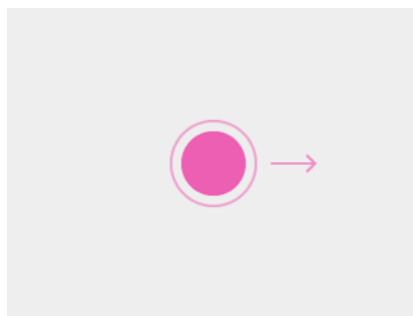
长按不用于显示情境菜单。



长按拖移

一根手指按下，等待，移动，
拿起

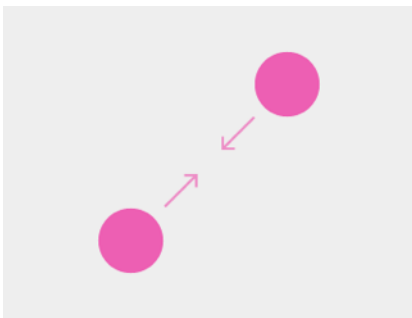
案例：拿起移动/选中多个项目



双击拖移

按下，拿起，按下，拖移，拿起

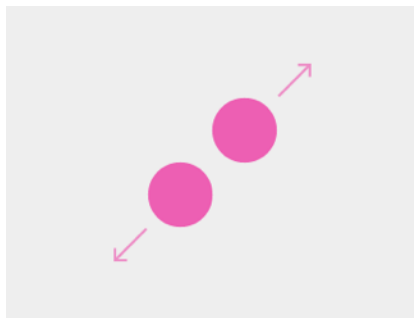
案例：放大/缩小



捏放

两根手指按下，向外移动，
拿起

案例：放大



捏合

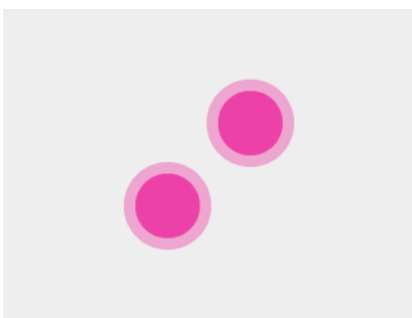
两根手指按下，向内移动，
拿起

案例：缩小



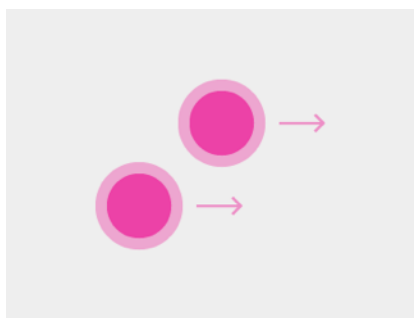
双指触摸

两根手指按下，拿起
案例：缩小



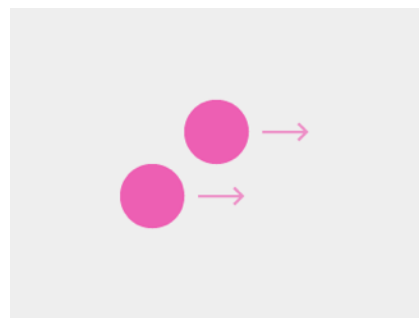
双指拖移，滑动，抛掷

两只手指按下，移动，拿起
案例：选中多个项目/转动视角



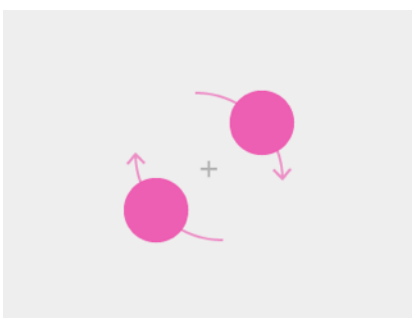
双指长按

两根手指按下，等待，拿起
案例：暂无



两指长按拖移

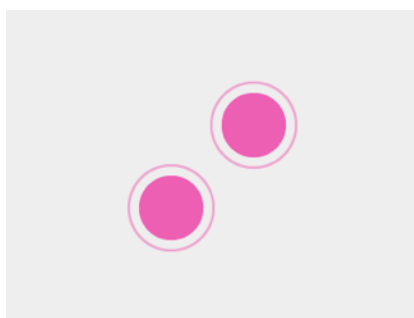
两指按下，等待，移动，拿起
案例：拿起后移动



双指双击

两指按下，拿起，按下，拿起

案例：缩小



旋转

双指按下，在屏幕上旋转，
拿起

案例：旋转内容，比如地图

触摸动作

根据特定情境下使用特定的手势进行操作

轻击

激活屏幕内的元素，像是一个按钮

触摸动作：触摸

取消或者退出

取消或退出当前的任务，比如对话框、菜单

触摸动作：触摸

启用/关闭状态

隐藏或显示Chrome窗口

触摸动作：触摸

拖移/滑动/抛掷

请阅读文章的下半部分来详细区别滚动，滚动显示，拖移（慢速运动），解除，滑动刷新，侧滑，分页滑动，滑出消除，菜单打开和倾斜的不同

触摸动作：拖移/滑动/抛掷

内容选中（当没有任何内容被选中时）

选中一个单独的元素

触摸动作：长按/双指触摸

内容选中（当已经有内容被选中时）

在选择模式下选择额外的内容，可以使用多种动作组合。

触摸动作：触摸/双指触摸

多项内容选中拖移

通过手势启动选择框，手指可以调整宽度和高度。最后选中的范围基于手指拿起时选定的范围。

触摸动作：双指滑动或拖移/当没有内容被选中时，长按拖移

拿起和移动

用户修改选中的单个项目或项目群，可以用在：

- 重新排列视图
- 将一个项目放入容器中（译者注：类似将程序放入文件夹中）或者目标上
- 重新排列项目在列表中的位置或收藏夹

触摸动作：双指长按拖移/长按拖移选中项目

放大

将内容的尺寸放大

触摸动作：

- 双击
- 双击拖移（向下）
- 捏放（向外）
- 缩放至合适

对于窗口内的内容，缩放至合适窗口

触摸动作：双击

缩小

缩小内容

触摸动作：

- 在最大化时双击
- 双击后拖移（向上）
- 捏合（向内）
- 双指触摸
- 双指双击

展开

展开折叠的内容

触摸动作：捏放（向外）

折叠

将展开的内容折叠

触摸动作：捏合（向内）

旋转

旋转目标内容

触摸动作：旋转

拖移，滑动和抛掷

由于滑动手势受具体的使用情境影响较大，故本节介绍一些主要的滑动手势和他们之间的区别

手势的速度（从低到高）是拖移，滑动和抛掷之间最大的区别，根据不同的使用情境，可能会造成不同的结果：

- 拖移：精确手势，目的性更明确，通常是控制屏幕上的一个目标
- 滑动：模糊手势，快速，通常不限定在屏幕上一个特别的目标。
- 抛掷：模糊手势，屏幕上没有特定目标

滑动与抛掷的区别就在于速度和操作是否有超过设定的数值。

一般来说，手势动作的速度有一定的范围，来监听用户是否取消操作：拖拽操作当用户将手指拉回时则会取消操作；而如果是抛掷动作，会在手指速度达到一定值时激活，防止取消操作。

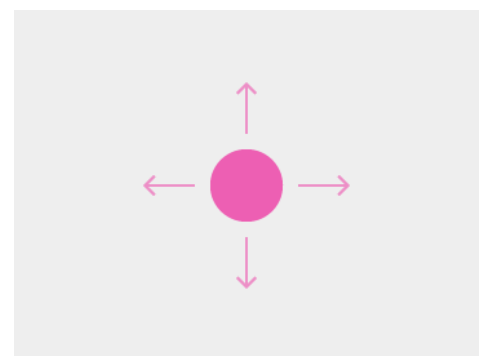
滚动

在内容内垂直或水平滑动。

滚动的距离基于手势的速度：拖动（慢）、滑动、抛掷（快）

一般来说：

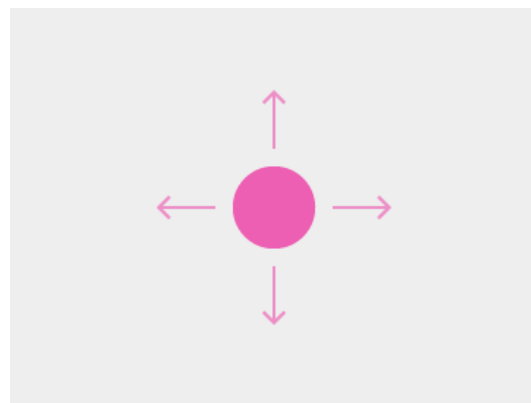
- 滚动的方向与手势是相反的
- 适用于内容尺寸为100%时



滑动显示

随着内容页面的滚动，窗口显示额外的内容，比如向上滚动Chrome显示网址栏。

通过滚动自然方向，来恢复原先的状态。



拖移（慢速运动）

1到2根手指全方面的滚动

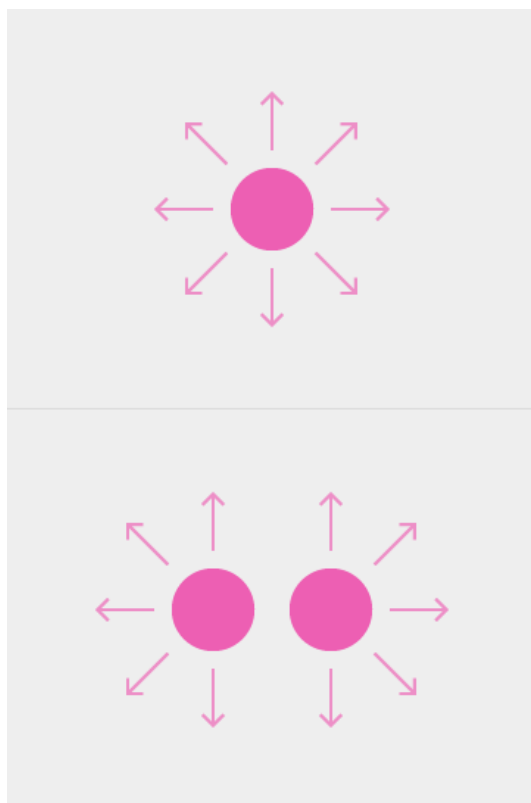
通常用于：

- 无限的内容（地图类）
- 使用双指进行操作（双指捏合放大缩小或旋转）比如在地图上使用双指进行操作。

双指拖移手势控制视角上下变化。

拖移就是典型使用拖移手势进行操作。

延续抛掷手势，将会使内容随着原先的轨迹继续运动。



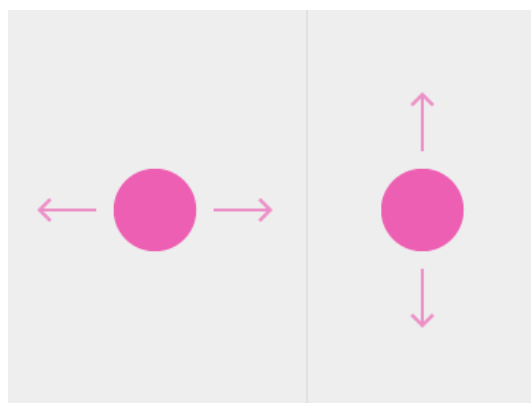
解除

通常设置在列表中单独一项的滑动操作

垂直方向滚动查看

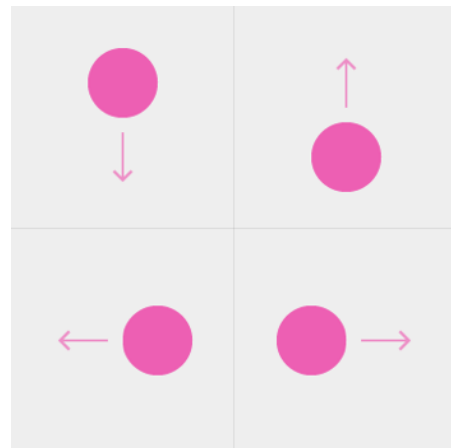
手势通常是在同一水平高度的水平方向

解除动作是基于一定的动作范围来激活。



滑动刷新

通常是在列表中或者界面边缘执行滑动操作激活
方向一般是垂直向下的

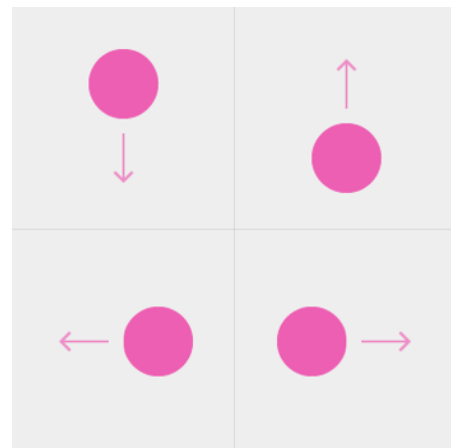


边缘滑动

从屏幕外部滑动，显示相关的其他内容，简单来说就是将内容分开，通过不同的样式展现。

通常如果边缘滑动没有定义，默认情况下从边缘滑动的手势作为翻页滑动

边缘滑动同样也是基于一定的动作范围来激活



翻页滑动

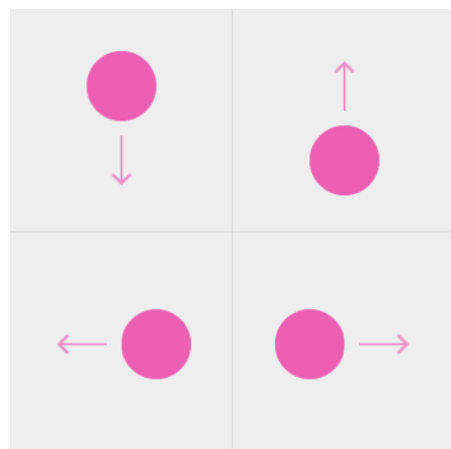
在屏幕内容上滑动，转向余下的内容

不要在使用翻页滑动时同时给滑动手势加入滑动操作标签（Swipeable），每一次手势操作翻动一个页面/标签。

在页面比例大于100%时，在内容上执行pan可能是查看额外的内容或一个额外的页面

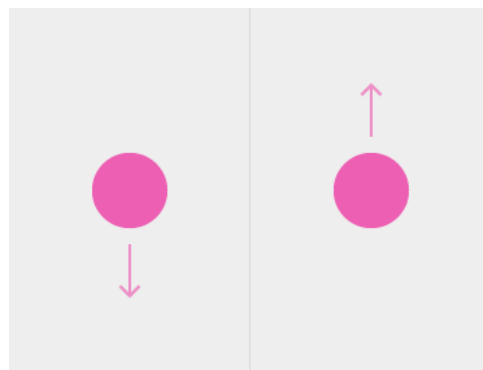
参见：边缘折叠

翻页滑动基于一定的监听范围



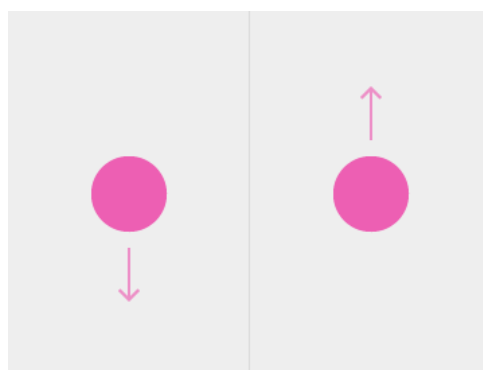
边缘折叠

用于在子层级结构中的显示导航
从顶部或底部滚动出导航或相关内容
同样侧边折叠也有一定的监听范围



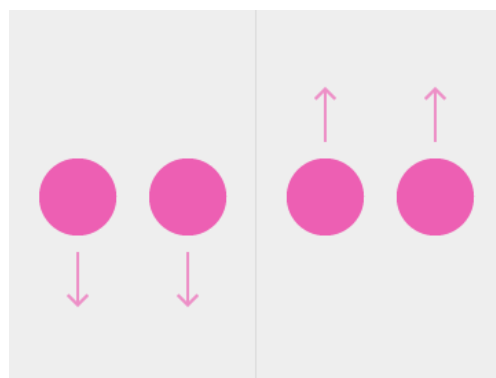
打开菜单

从菜单或者选项中拖拽出现菜单内容，菜单出现后突出显示菜单项。
菜单也可以通过触摸后出现
该动作可以通过拖拽激活



Tilt

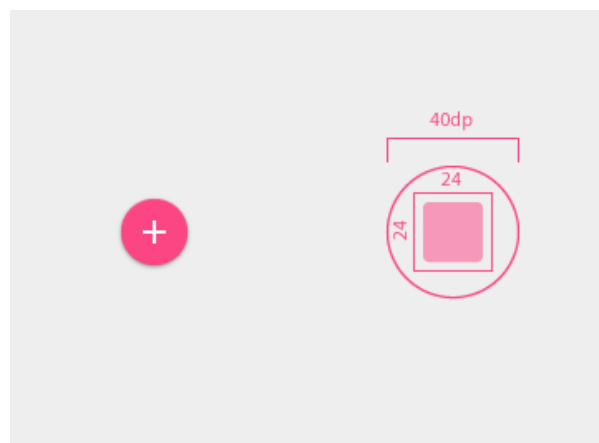
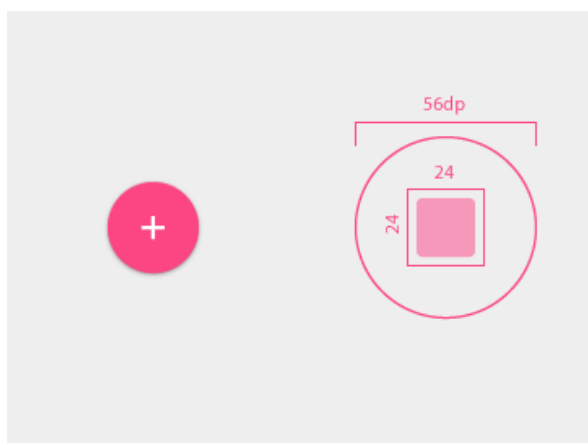
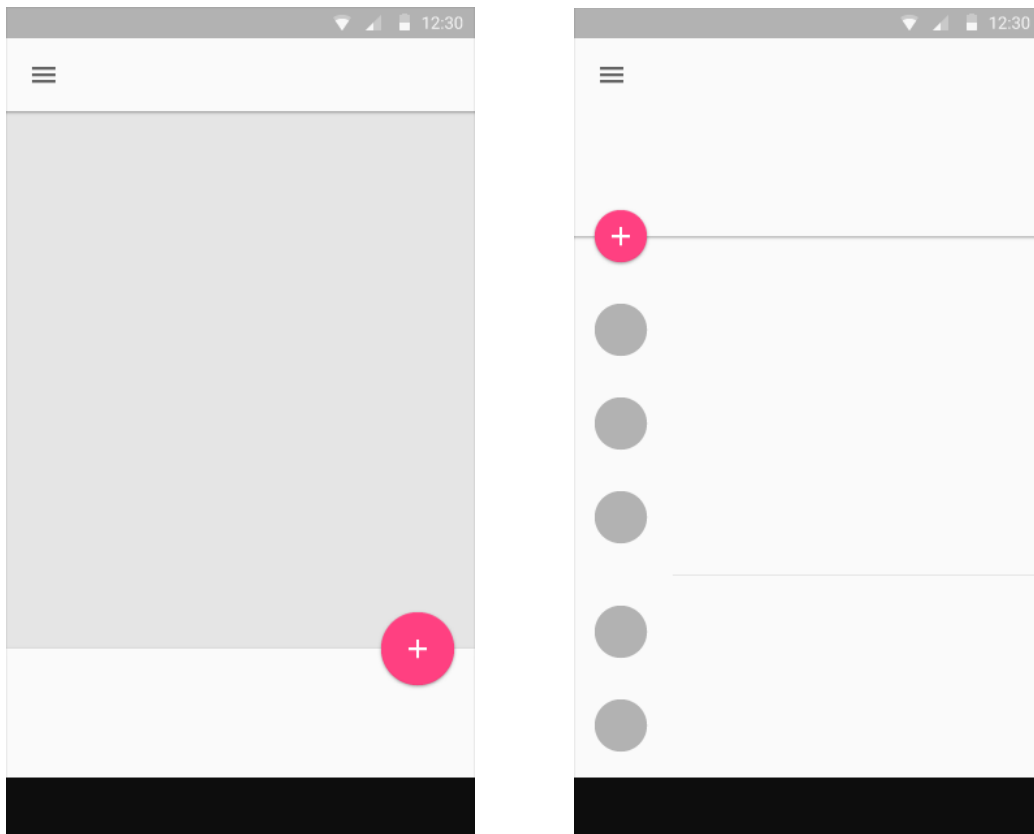
通常用于在3D内容上前进和后退
适用于两指操作手势（比如两指缩放或旋转）比如在地图中的应用，通过两指操控来平移。
通常是和拖拽相配合



6.3 浮动操作按钮

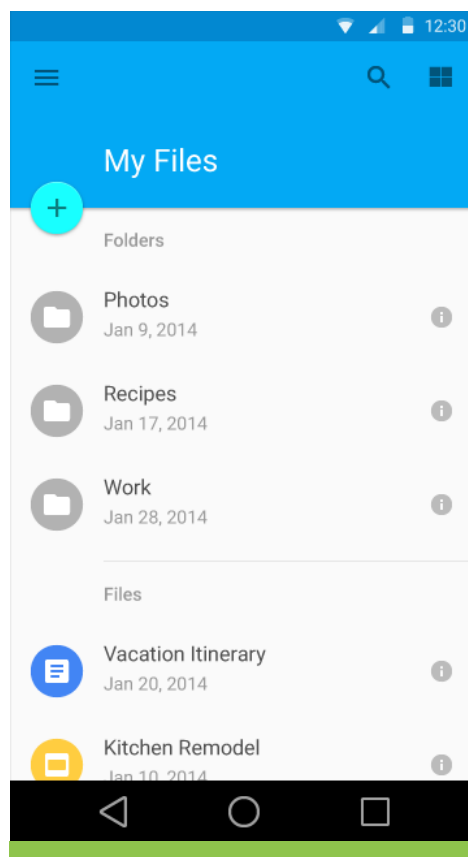
浮动操作按钮一般作为进阶操作的开关，在用户界面中通常是一个漂浮的小圆圈，它有自身独特的动态效果，比如变形、弹出、位移等等。

有两种尺寸的浮动操作按钮：默认大小和较小的，使用哪种尺寸需要根据屏幕中的元素来决定。

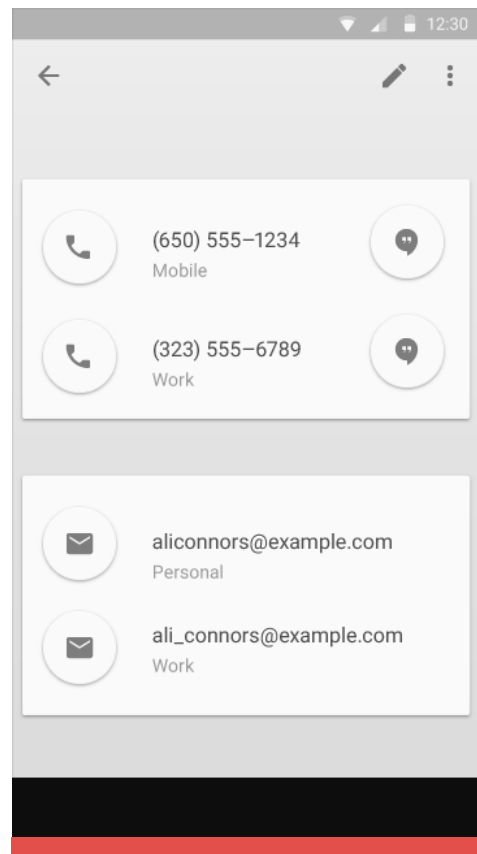
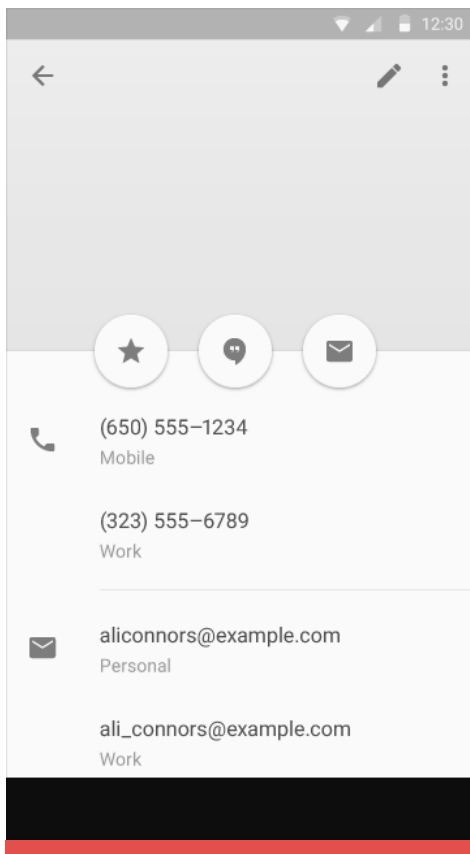


内容相关

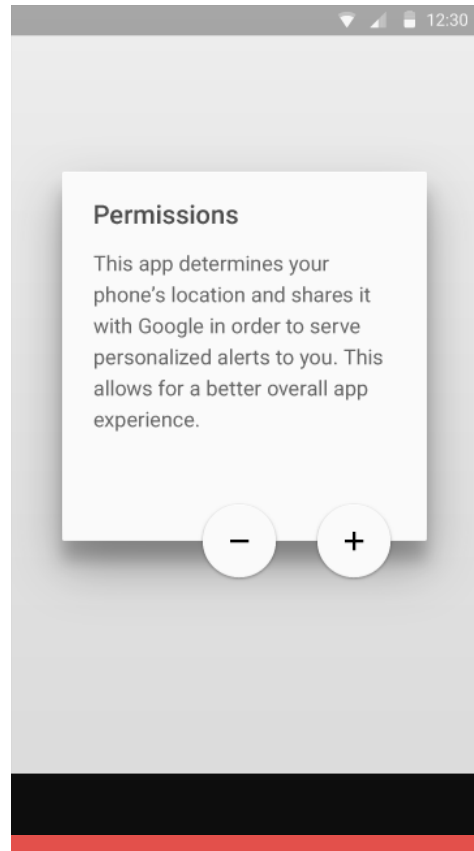
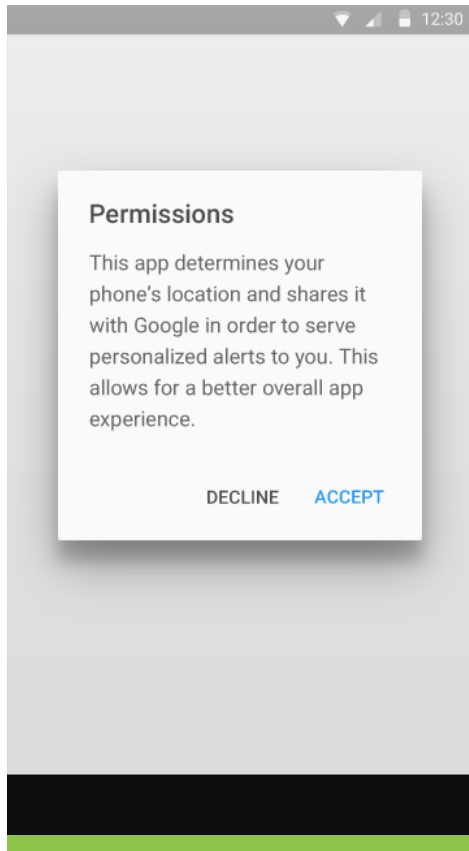
并不是每一个屏幕都需要浮动操作按钮。一个浮动操作按钮应该能够代表这个APP中的主要操作。在左边主要是相册集和打开的图片，并不需要浮动按钮。而在右侧，主要的动作是添加内容，所以添加浮动操作按钮是合理的。



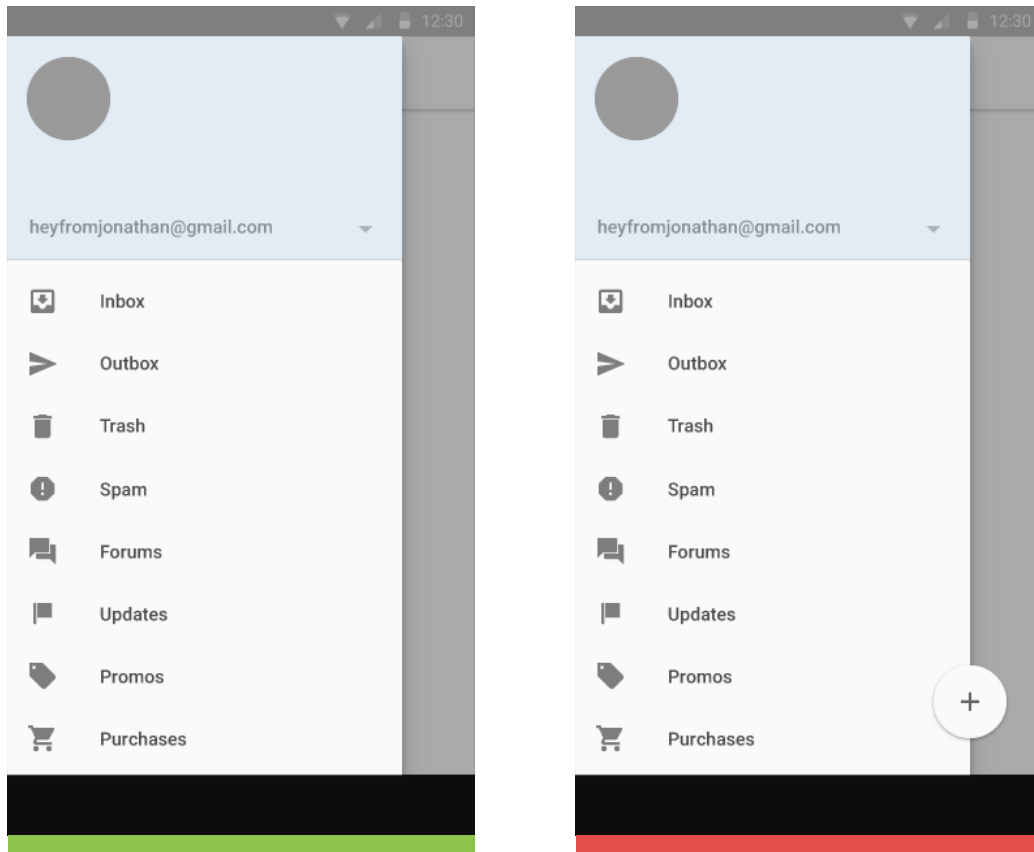
尽量只在屏幕中摆放一个浮动按钮，因为它是整个屏幕中最突出的按钮



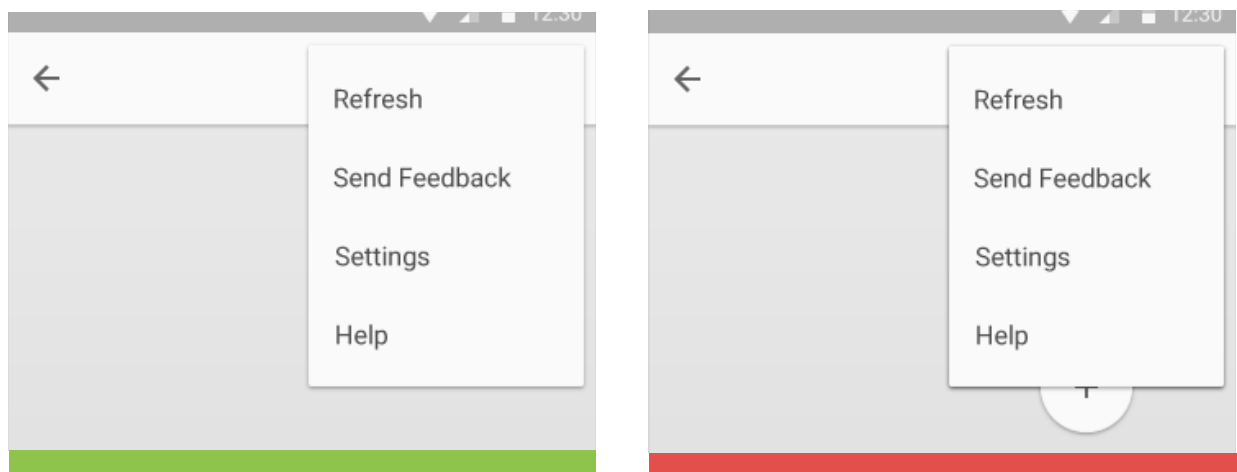
不要在对话框中使用浮动按钮，去使用扁平的按钮



不要在侧边的导航栏中加入浮动按钮，他会分散用户的注意力，侧边栏的主要任务就是导航

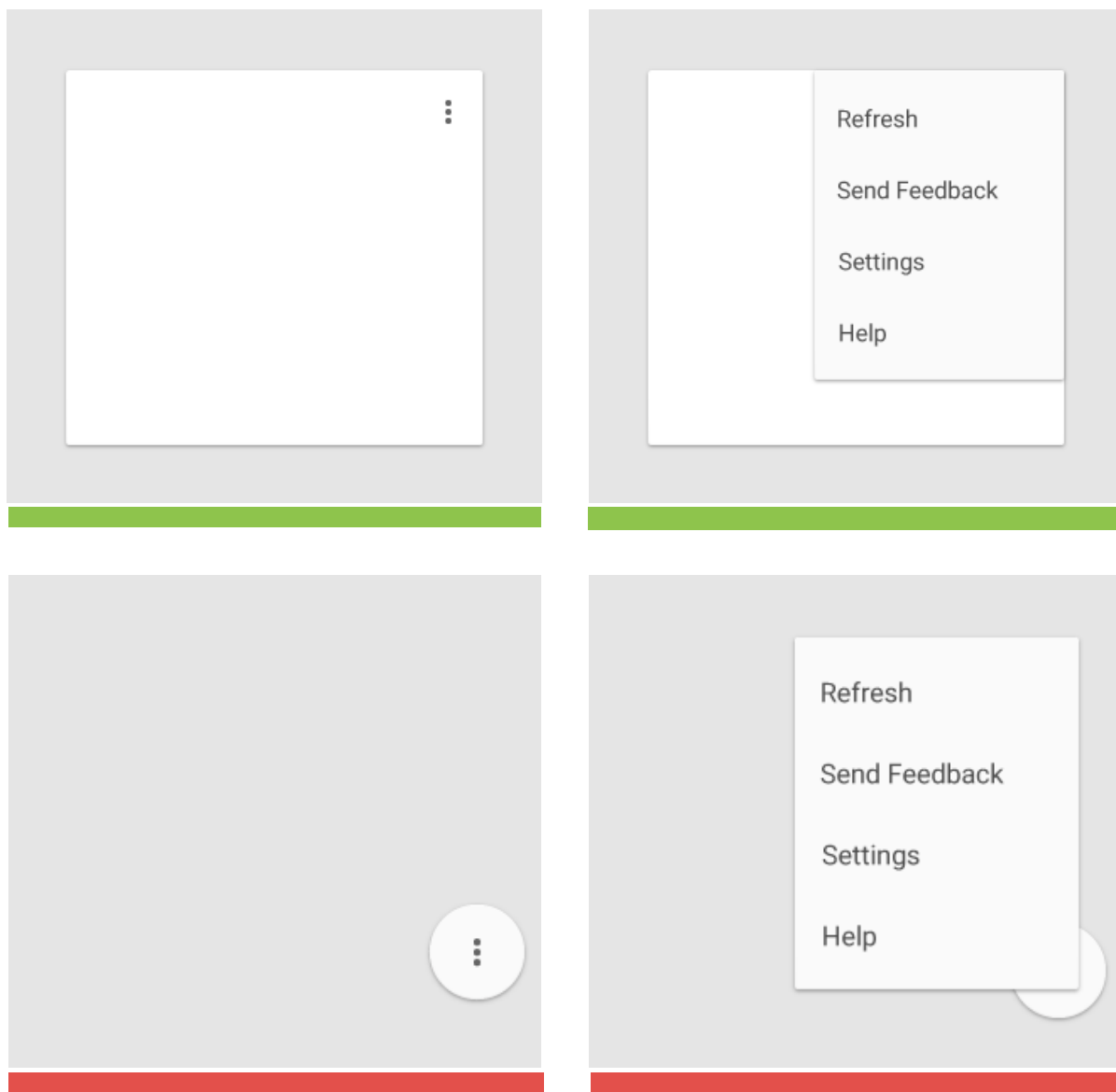


不要把浮动操作按钮摆放在下拉菜单会覆盖的面积



操作相关

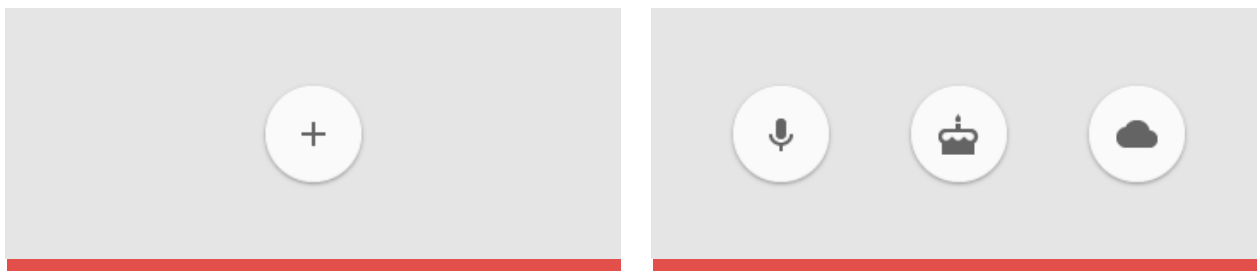
不要在浮动操作按钮中设置弹出菜单选项



如果是应用程序内添加文件类型的标志，可以让浮动操作按钮在被点击后变化为相关的按钮



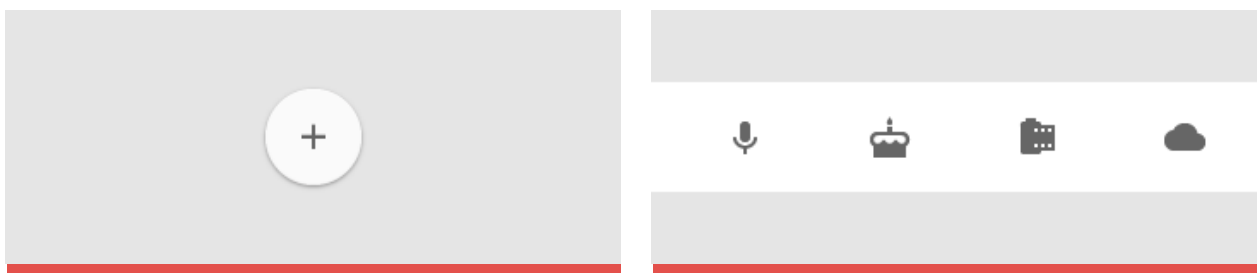
然而，如果是一组动作按钮，之间并没有关联，那么你应该把这些按钮放入到菜单中，而不是通过浮动操作按钮激活



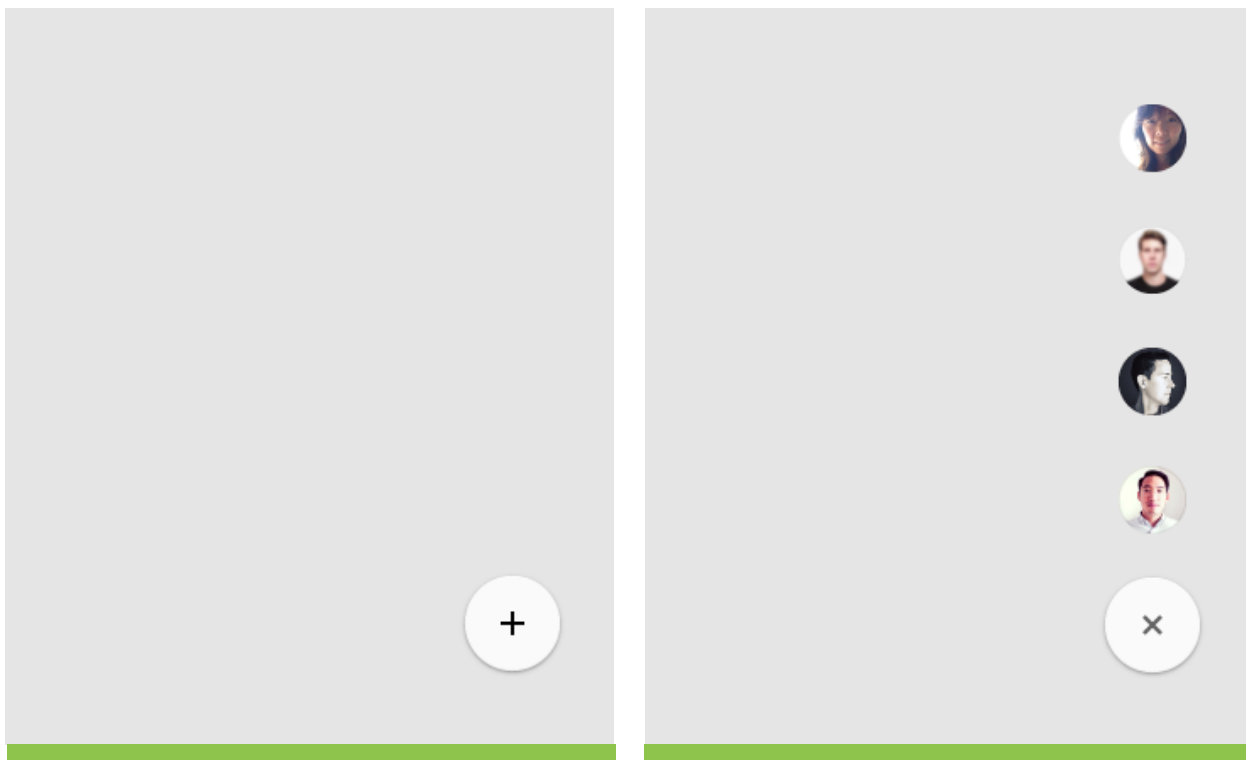
如果在点击浮动操作按钮过后，出现工具栏，那么工具栏中应该摆放相关的按钮。在这个例子中，是让用户选择添加的媒体类型



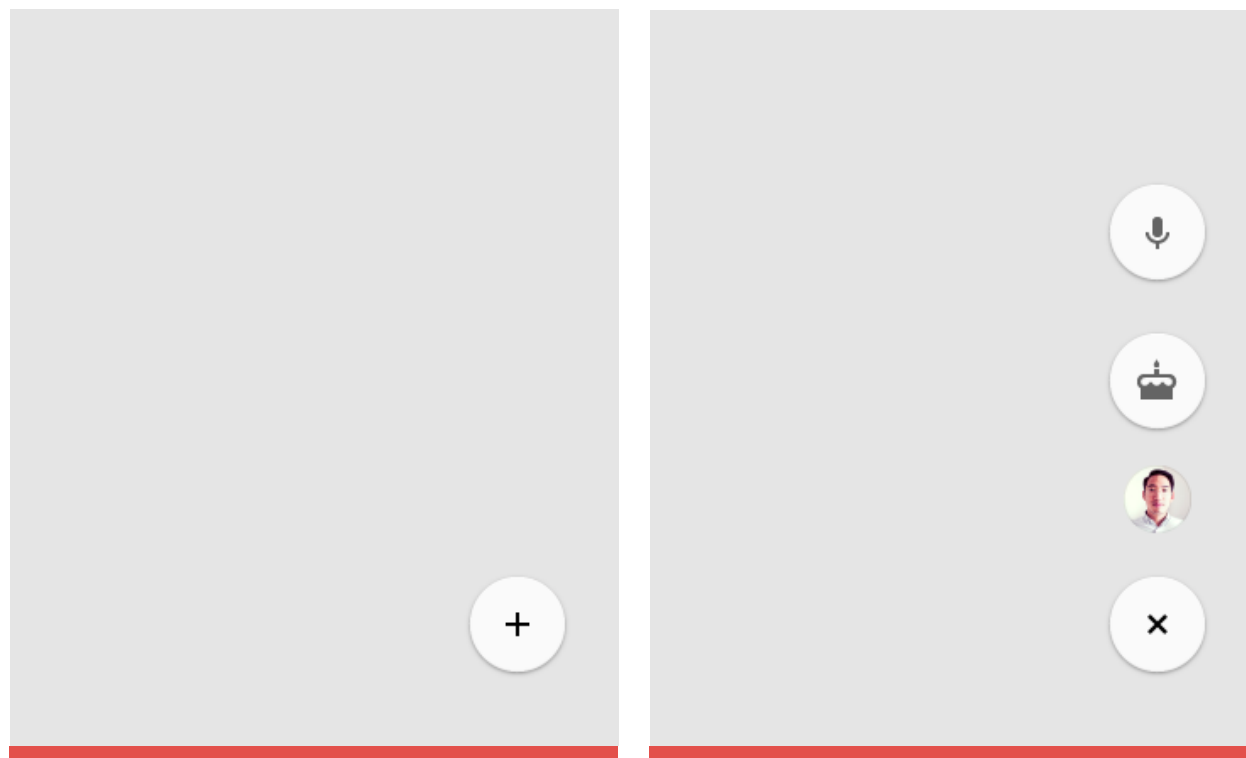
不要将毫无关联的图标按钮添加到由浮动操作按钮激活的工具栏中



一个浮动操作按钮可以包含联系人列表

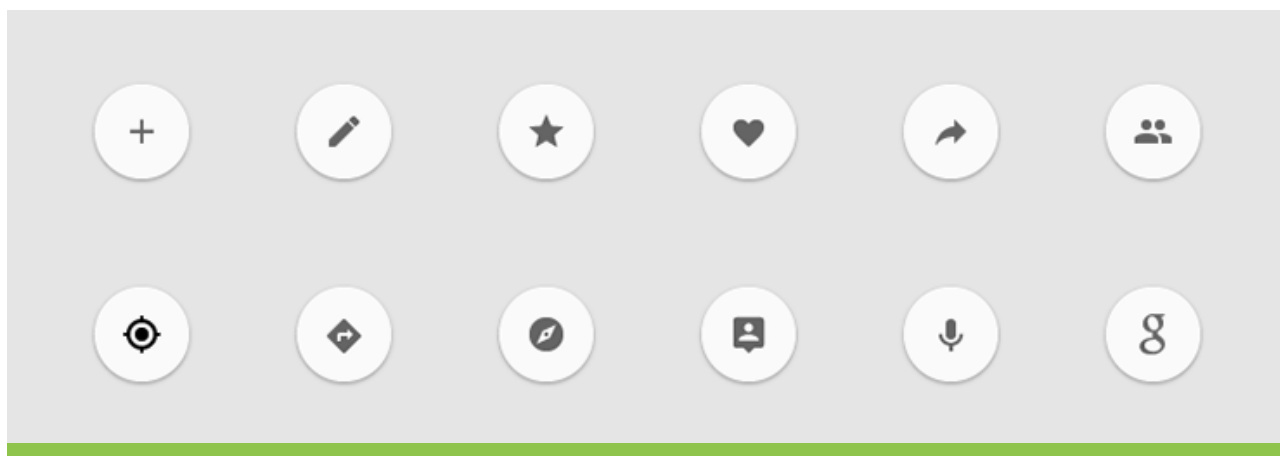


但是不要将各类功能和联系人或其他元素都放进按钮中

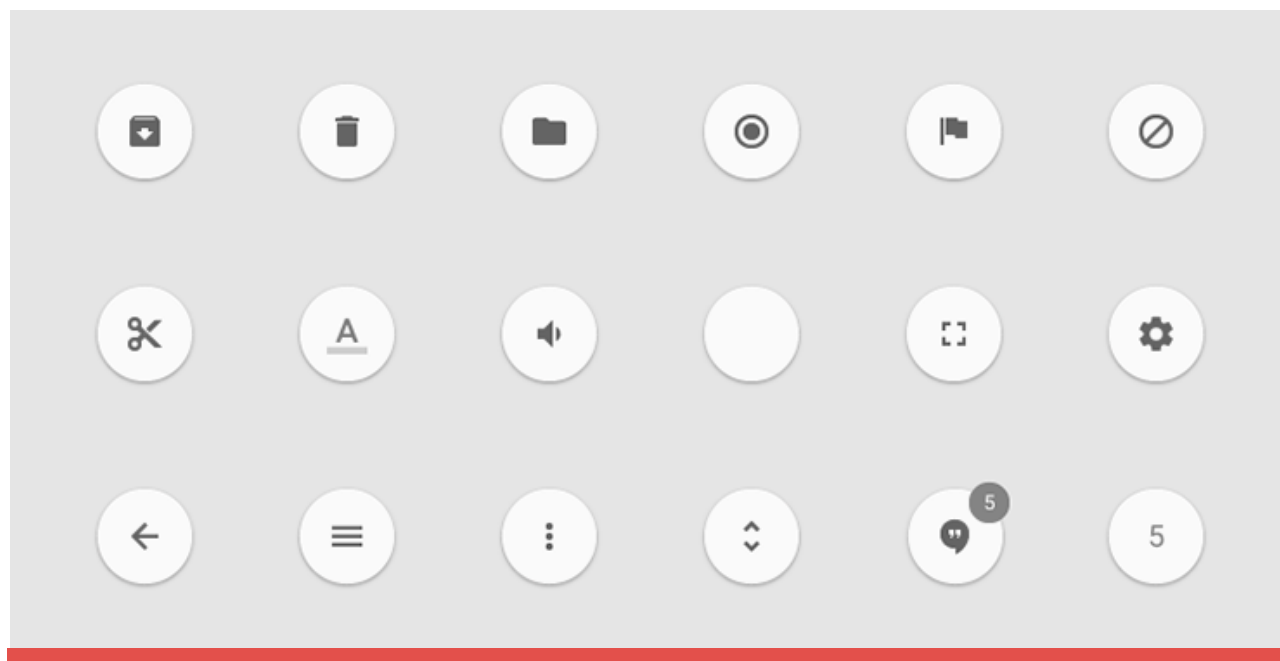


品质

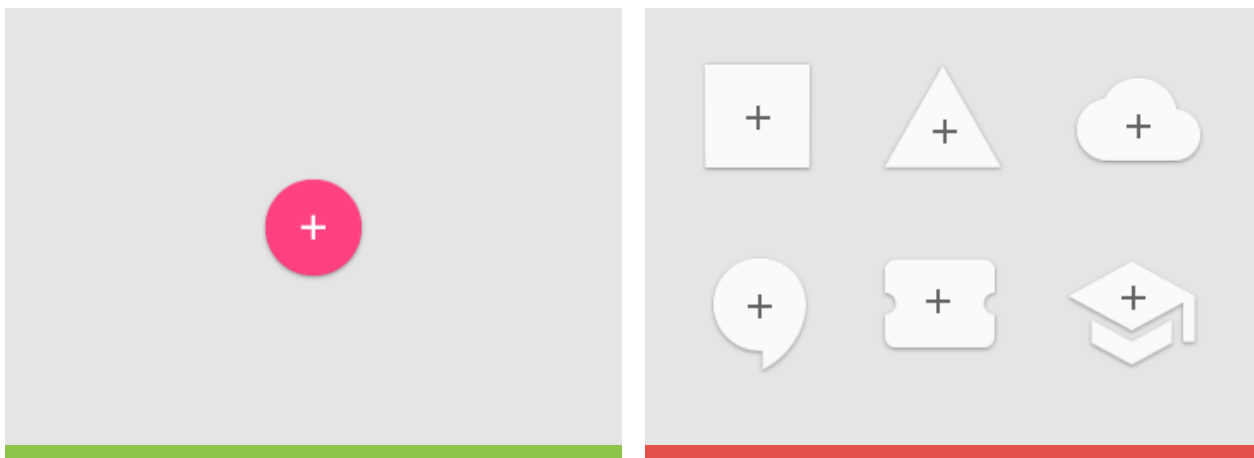
浮动按钮通常可以作为：添加、编辑、喜欢、分享、搜索等等按钮出现



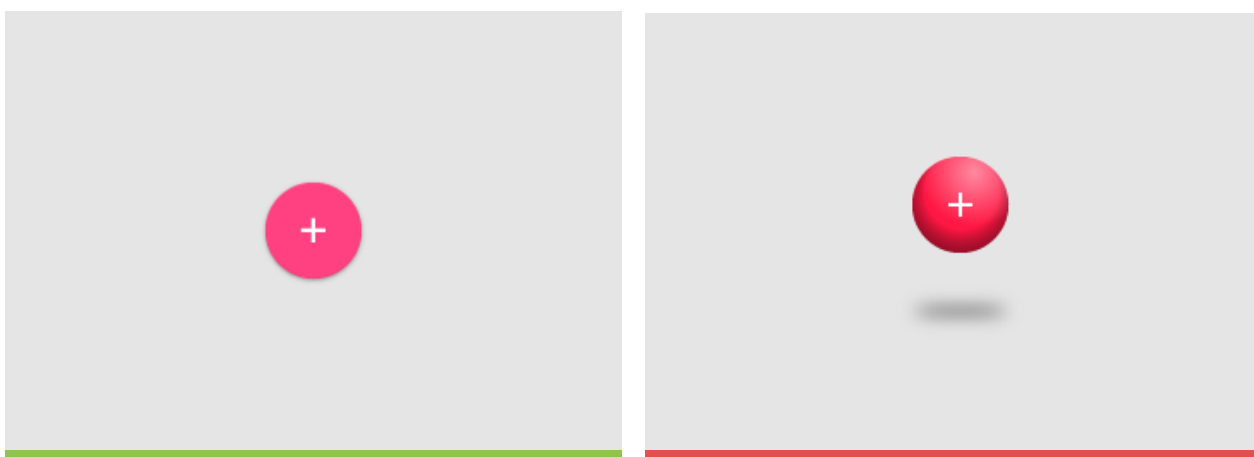
一般来说，我们会避免在浮动操作按钮上设置会对内容操作按钮，比如下载、删除、新建文件夹等等；或者不具体的动作，比如警告或错误；应该在工具栏或控件栏中的按钮，比如音量控制、字体颜色等等。另外浮动操作按钮不应该包含应用程序栏图标或状态栏上的通知图标，还有注意的是，不要在浮动操作按钮上再添加气泡或其他元素。



使用圆形作为按钮的外形，以免混淆用户

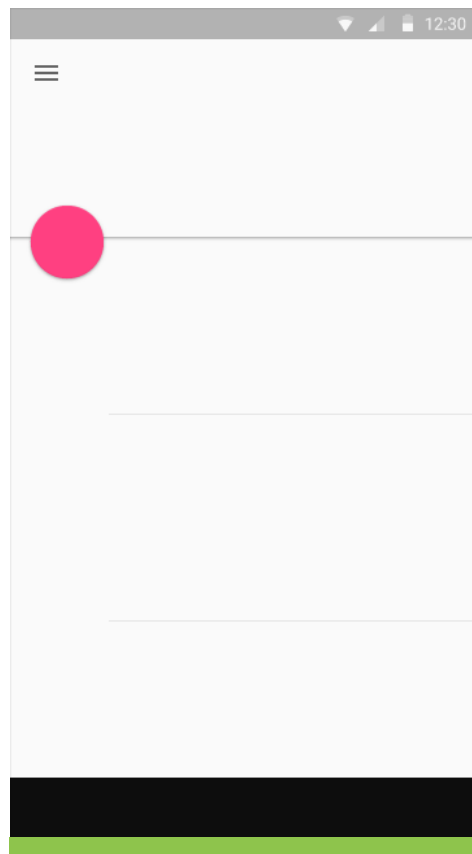
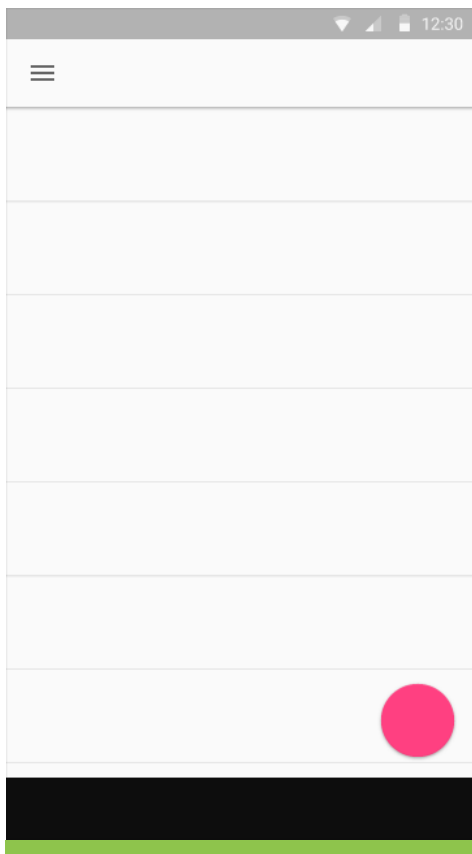


不要在浮动操作按钮中加入反弹动画

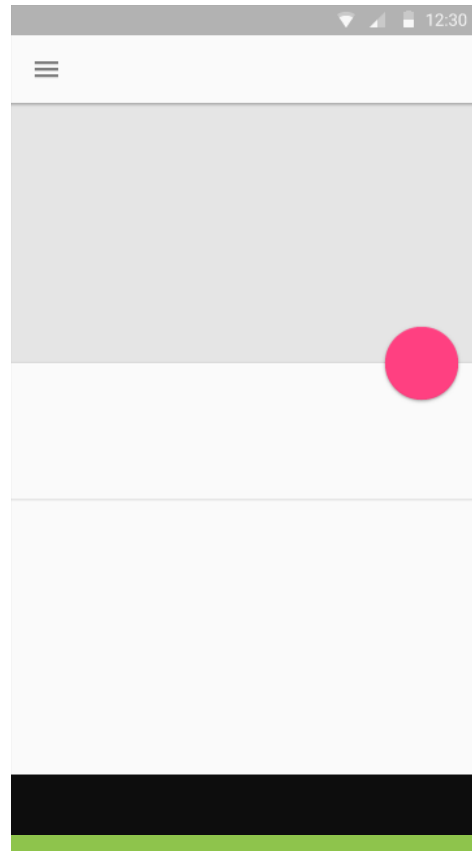
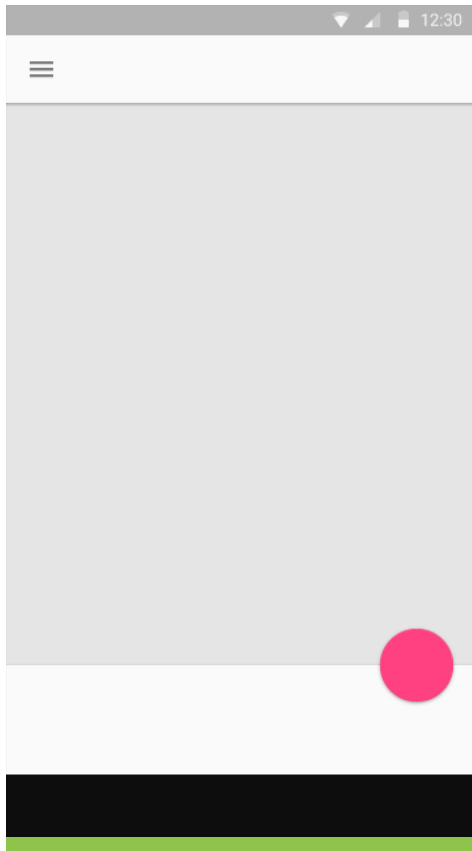


位置

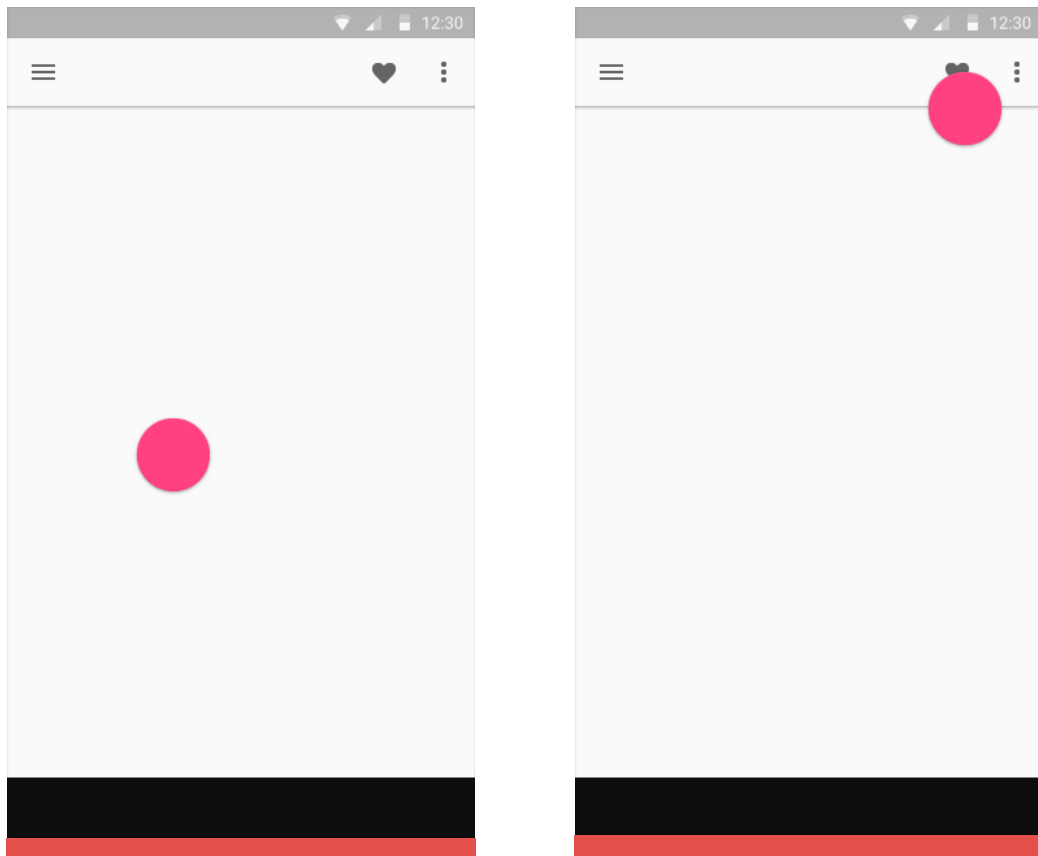
浮动操作按钮可以放在界面中明显的分割线上或附加在扩展栏中



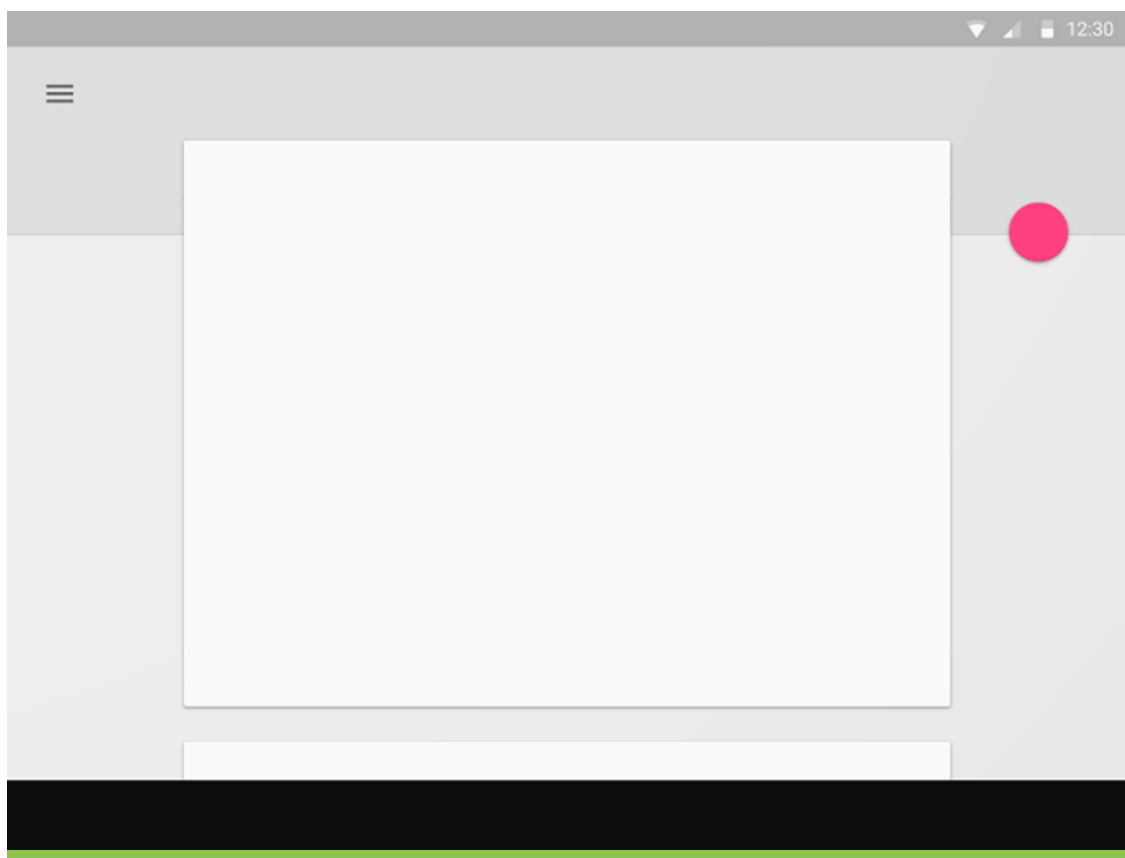
也可以将浮动操作按钮放在页脚或者展开的列表上



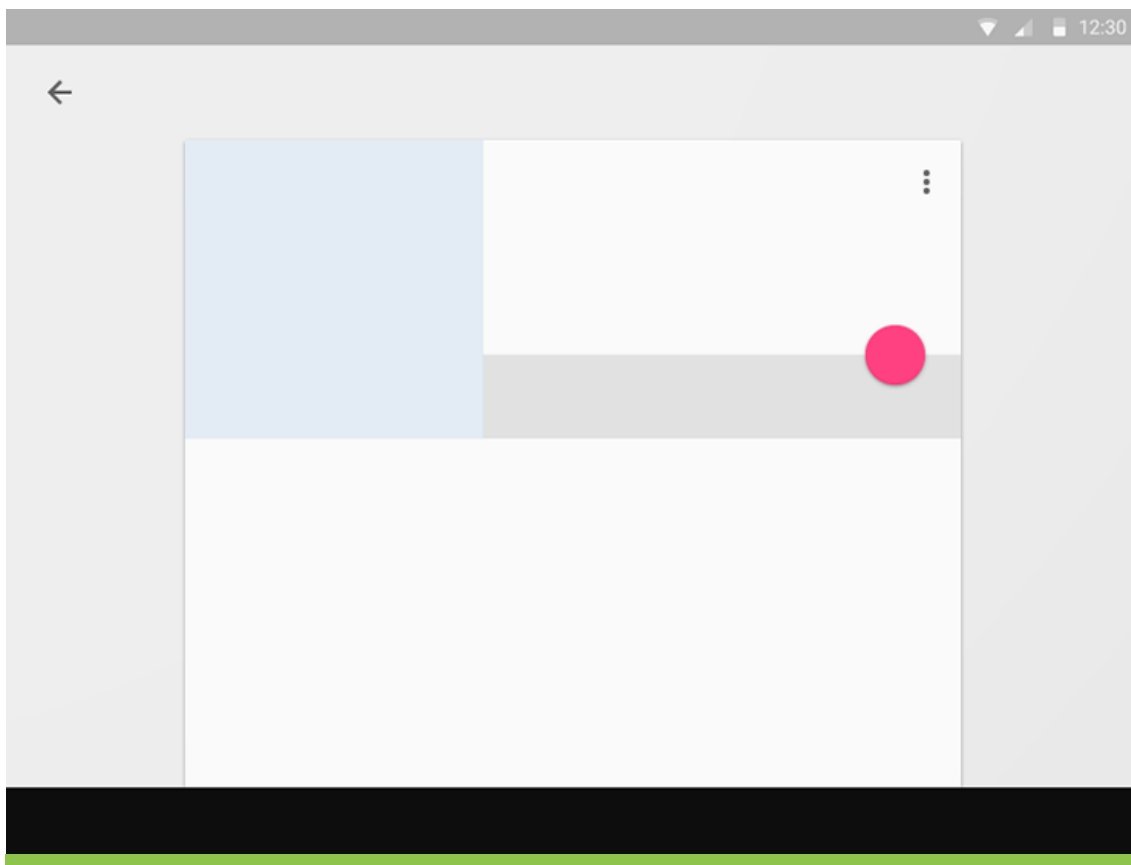
不要将浮动操作按钮随意的悬浮在一个位置上，同时也要注意不要让浮动操作按钮遮挡住其他按钮



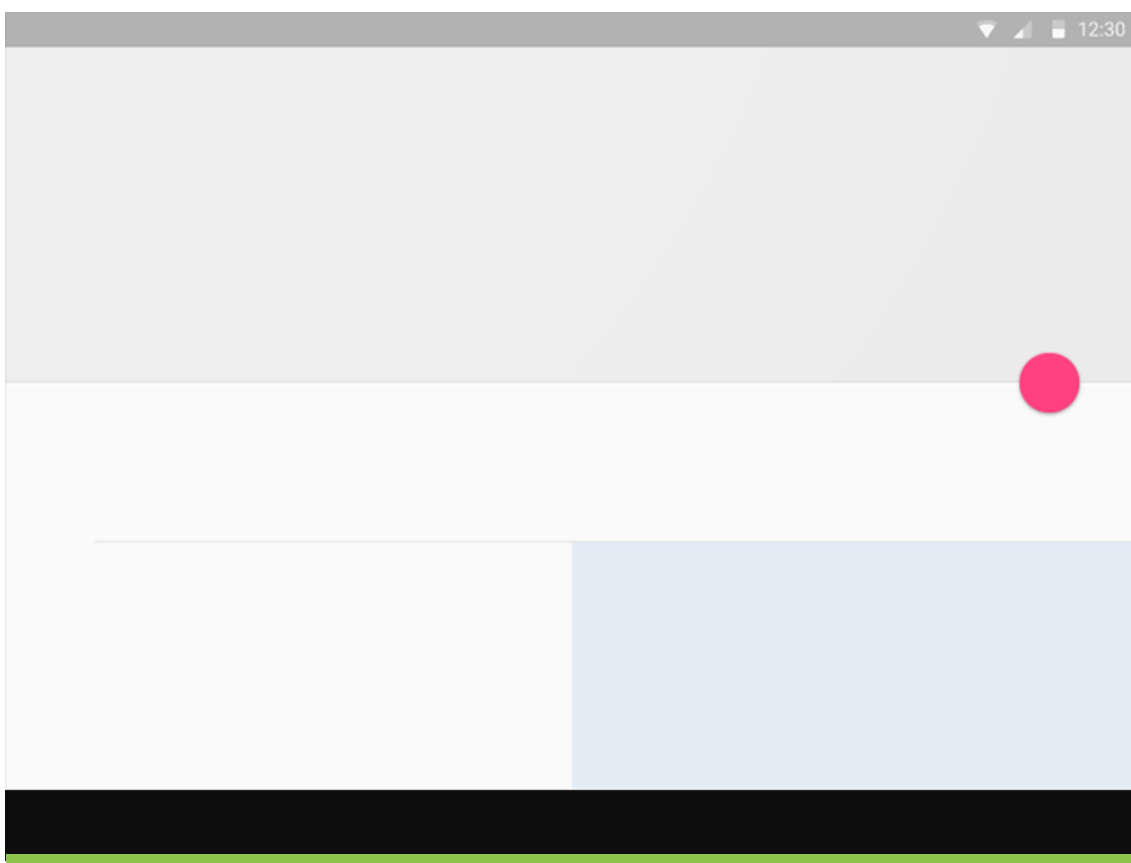
一个浮动操作按钮可以链接到一个新的窗口



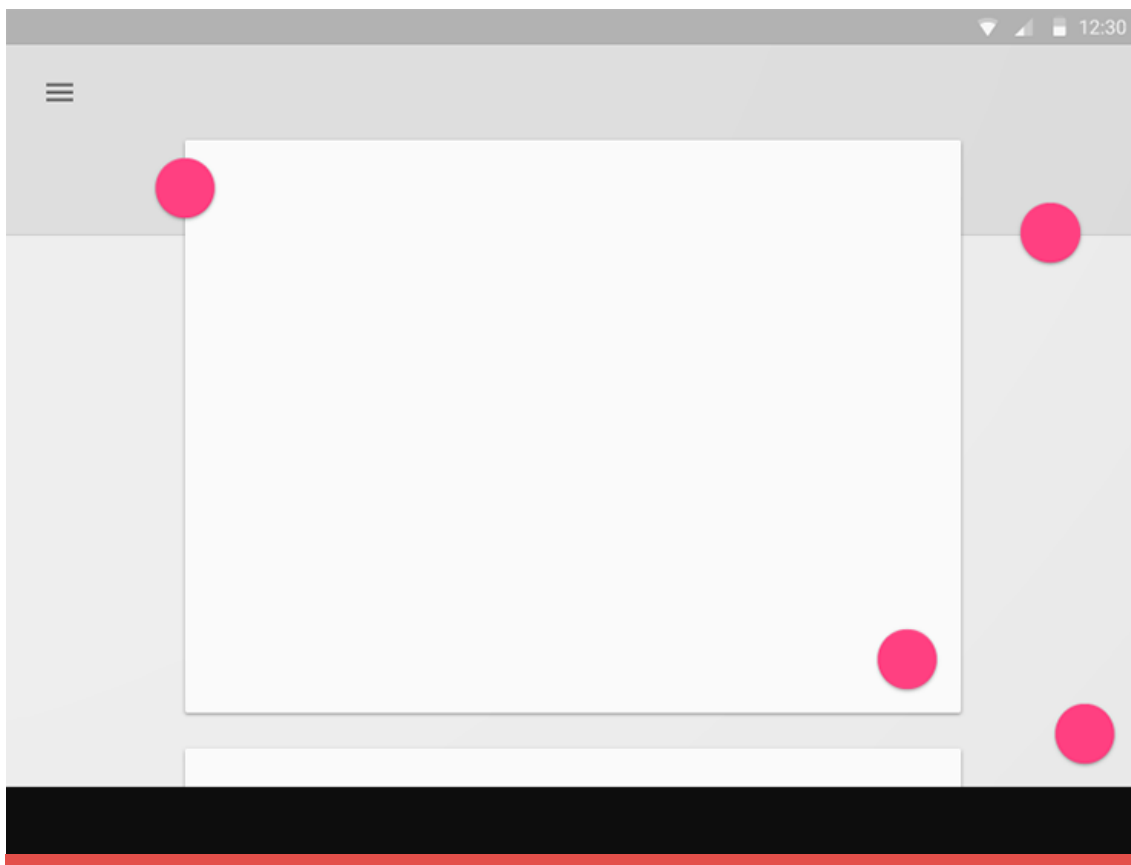
浮动操作按钮可以被放置在工具栏或者页面分隔上（只要不遮挡其他元素）



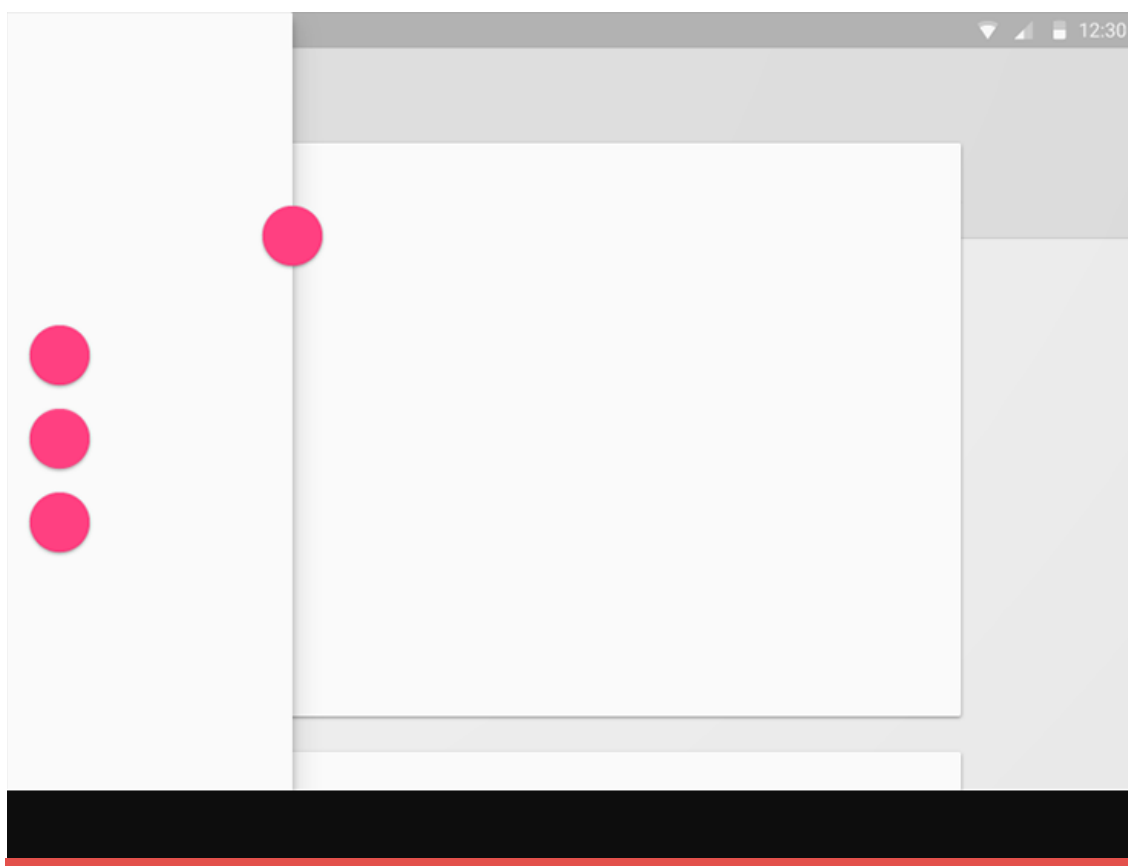
浮动操作按钮可以附着在区域的边缘



一个屏幕中只允许出现一个浮动操作按钮



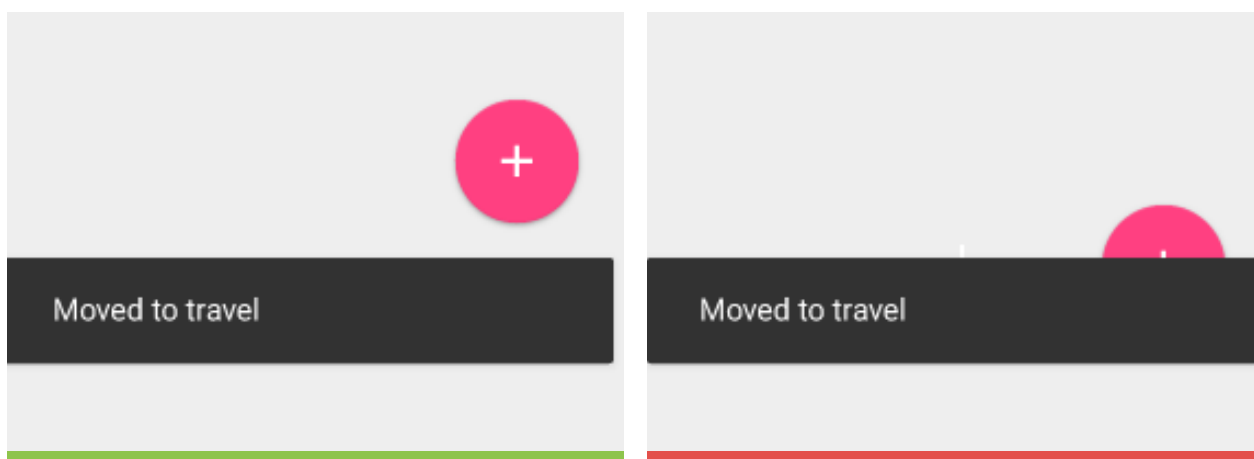
不要将浮动操作按钮隐藏在侧边栏或附着在侧边栏上



不要给界面中每一个元素上都放置一个浮动操作按钮



不要让状态通知或其他元素，遮挡住浮动操作按钮



6.4 设置

应用程序中的设置允许用户根据自己的喜好自定义程序的样子。他们给予用户真正意义上的控制，同时也可以避免很多不断出现的问题。

访问设置

因为设置并没非常高的使用率，所以它的UI并不用那么起眼。当你的程序存有设置时应当是，在任何情况下，“设置”按钮就简单地放上“设置”，这样就够了。如果当前屏幕支持侧边栏，那么将“设置”放在其他内容之下，比如“帮助与反馈”。如果这个页面有一个工具栏，则将“设置”按钮放在悬浮工具条上。

使用恰当的设置

每当用户访问设置时，大部分情况下他们的体验与我们所希望给用户的并不一样。

对用户来说设置页面应当是条理清晰并且容易理解。特别要注意，避免压倒性的给用户一大堆设置选项。注意不要因为外界因素而改变初衷，“我们需要的只是，他作为一个设置，仅仅只是设置，这样就够了。”

对于每个设置项你都应该仔细考量后再放进去，每当要添加一个新的设置项时，都要问自己：这个为用户真正需要的吗？信息和按钮并不需要用户去特别设置的。如果不能更改，那么就不要再让他们能够设置。如果是关于程序的参数信息（比如版本号、服务条款、许可证等等），将这些内容整理好放进帮助页面就可以了。如果他是一个按键（比如刷新、更改账户等）在APP中找到一个合适的位置安放他们，因为这些按钮并不会经常被用户换来换去。

我们应该让设置项更快捷的进行操作，可以通过放在工具栏或者浮动窗口。

对于一个设置项如果不到20%的用户会去操作，那么就不用为这个项开放设置了。这个方法同样也适用于新增加的和已有的设置项。

对于现有的设置，最后的一个问题可能就是：如果这个设置被删除了，会对小部分的用户造成影响吗？如果会，那么说明这个设置项是合理的。

成组的设置

当你有许多的设置功能项时，将一长串的列表利用集群分割开来。当然如何安排，还是取决于设置内的功能项总数。

七个或更少的设置选项

你不需要群组它们。

8~10个设置项

试试将1~2个设置组规划到一级组群内，当然，如果你有“独立项”（这个设置项与其他项毫无关联，并且无法放到分隔选区中）那么把他们：

- 如果这个功能设置项内含有非常重要的设置，那么将他们放在设置页面列表的顶部，单独分割出来。
- 否则，将他们放进“其他”组放在页面的底部，里面的列表则按照功能的重要性排列。

11~15个设置项

同上面的建议一样，但尽量保持在2~4个分隔选区。

此外，消除重复：如果两个设置项之间具有联系，那么就将他们合并为一个。比如，你可以将两个有联系的勾选框设计成一个选择框。

设置项大于16个时

如果你的设置项组大于4个的话，建议你将内容放到二级层中，再在二级层中对应上面的准则进行安排布置。

制定好默认选项

每个用户都不希望使用一个APP还要自己调整设置的内容。下面的几个问题可以帮助你制定合适的默认设置

- 如果没有默认设置，那么大多数用户会选择哪一个？
- 这种选择是最中性还是中性的？
- 这种选择是风险最小的，有争议的还是过度了的？
- 哪种选择会利用较少的设备电量或者手机数据？
- 是否只有当用户需要时，它才会出现并且用户能够非常容易的找到？

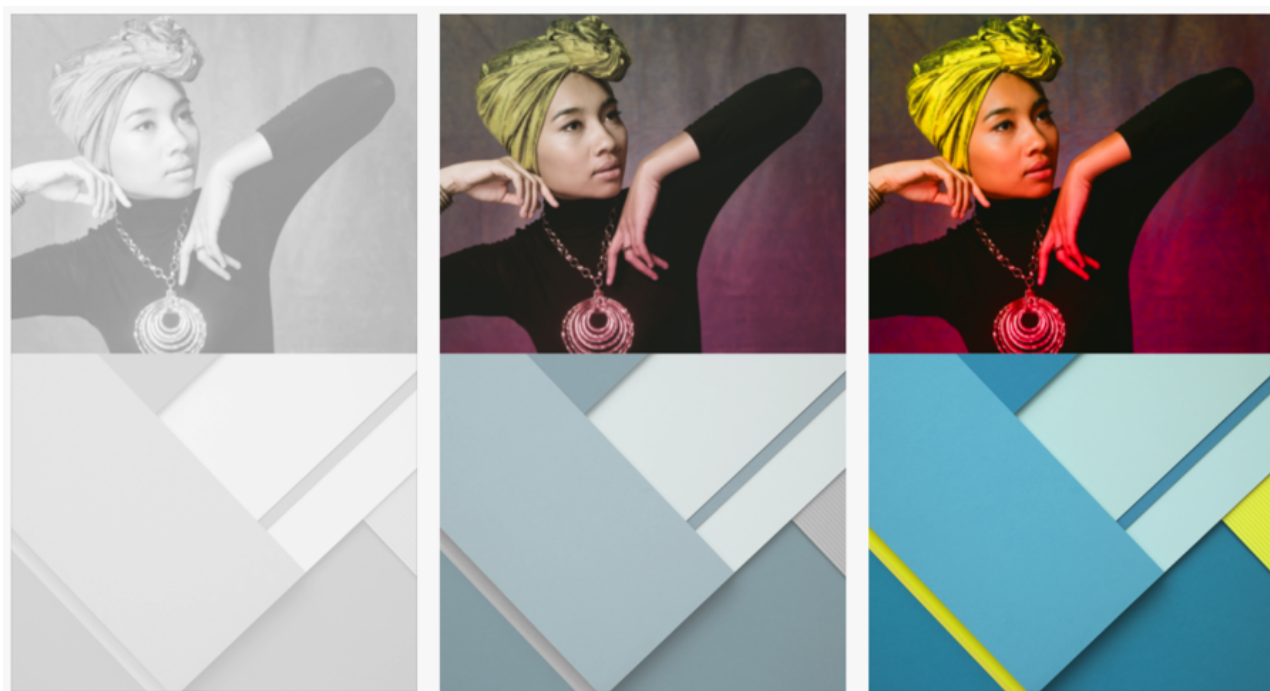
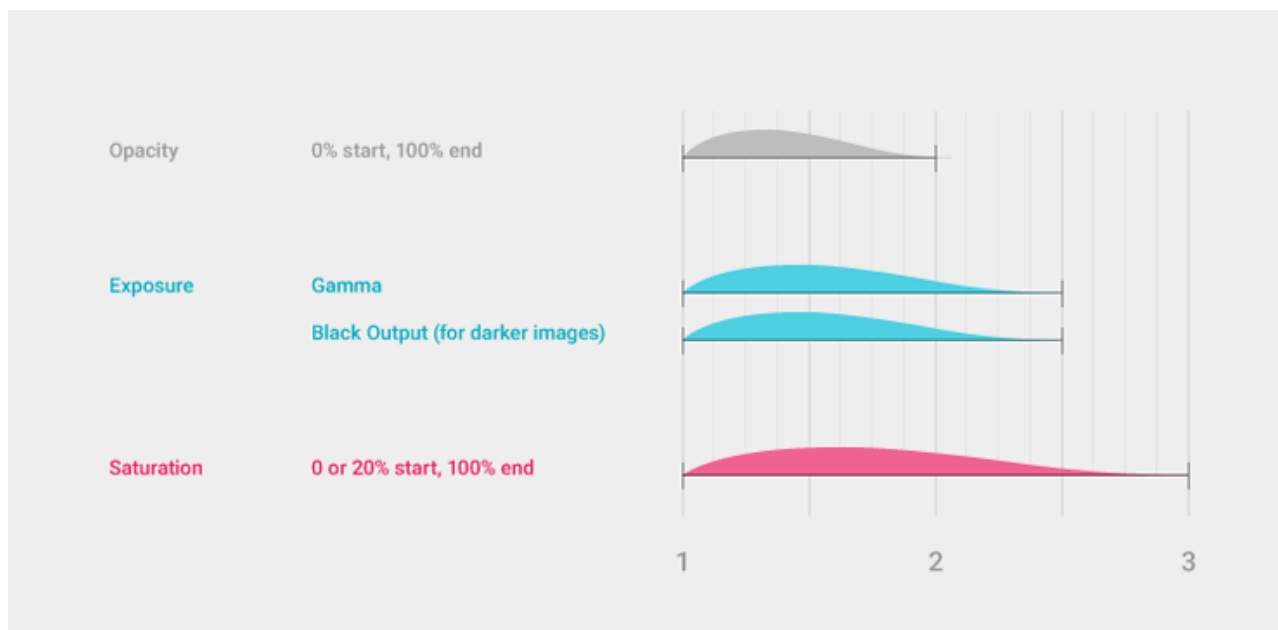
6.5 图像处理

图像载入

通过透明度的变化，可以让插图或者照片在加载和转换的呈现3个阶段。

最初使用低对比度和低饱和色彩，而后将透明度调整到100%后，饱和度再提到最高。

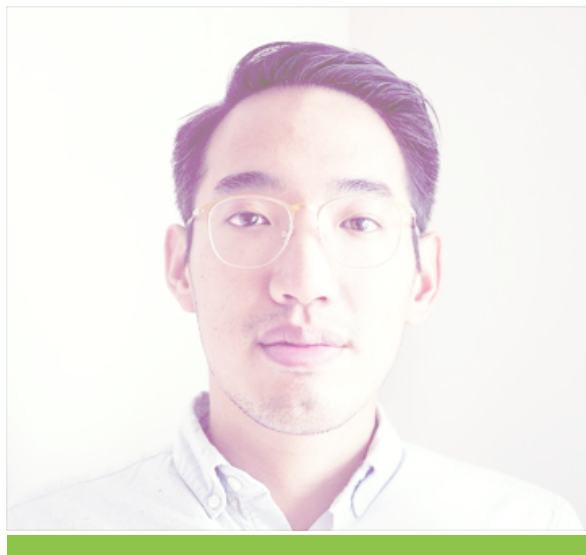
对于比较暗的图像，通过调整伽马和黑色的比率来降低图像的对比度。



Aperture vs Development

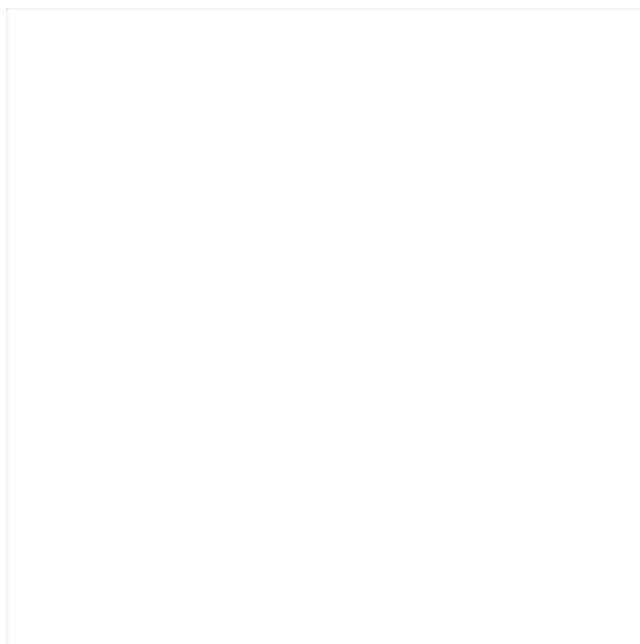
不要去调整图像的色阶，这样会让图像出现过曝。

这个效果就像光穿过镜头光圈到相机中一样，会用户看起来像是在自己冲洗胶片一样。



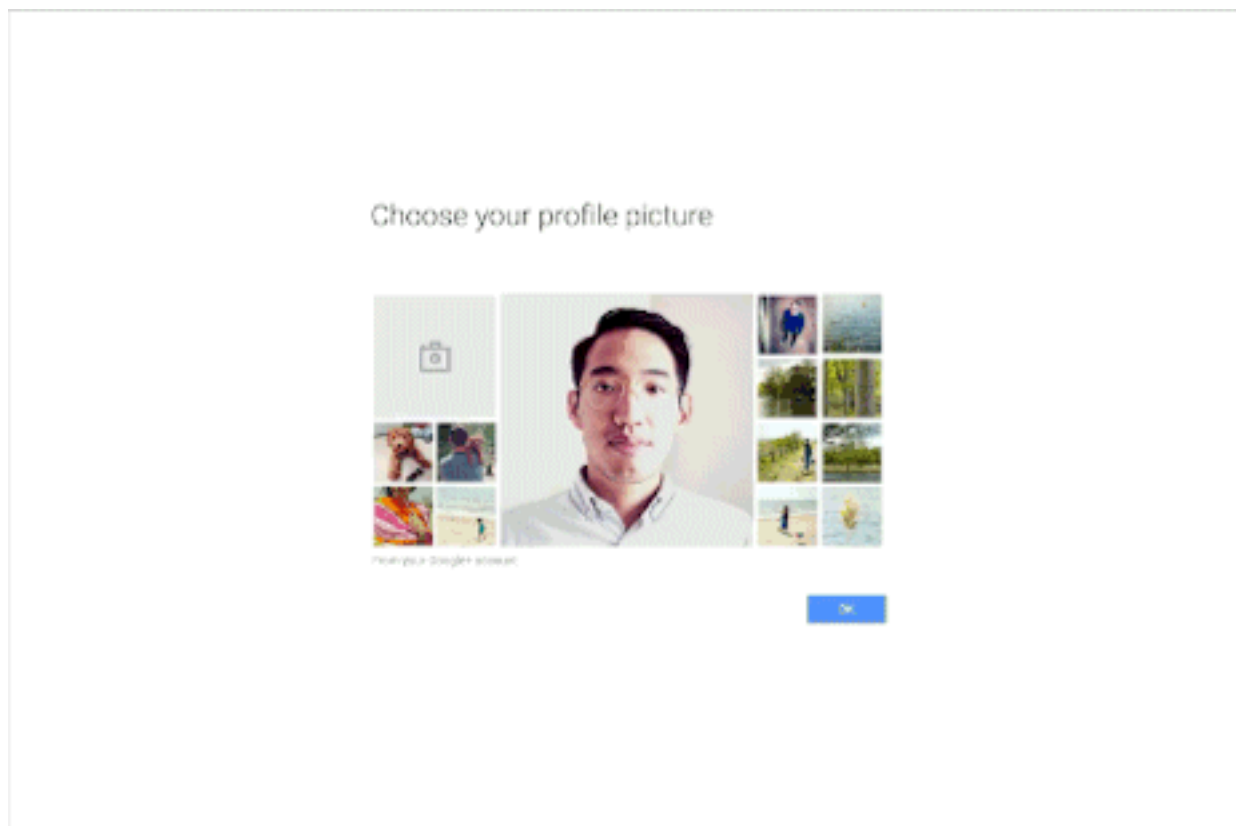
加载和转换

调整使用这三个阶段的参数（透明度、对比度和饱和度）来丰富你的图像加载动画，建议增长加载的持续时间、缩短转换的时间。



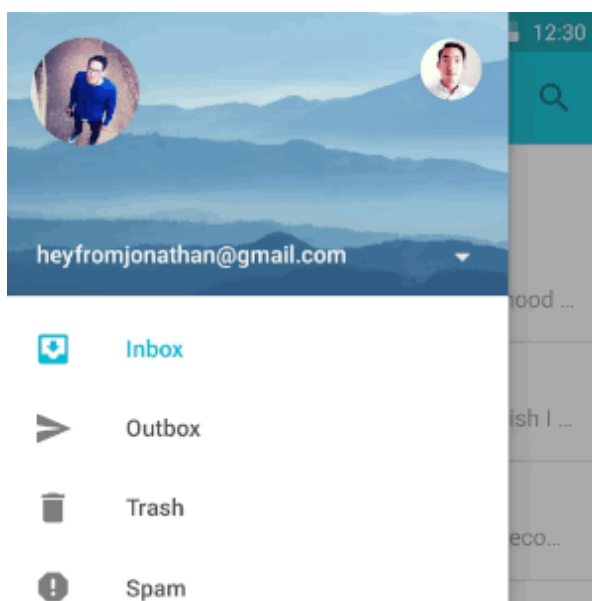
在大屏幕中载入

这是应用在大屏幕中的过程，范例中是在Chrome OS中加载壁纸



加入动画

添加一个小小的动画，让图像看起来像是一个开关



6.6 搜索

在APP内部搜索

当一个应用程序内部有大量的信息时，用户希望能够通过搜索来快速找到需要的内容。

搜索的基本运行结构：

- 启动搜索输入框
- 输入和提交搜索请求
- 显示反馈搜索结果

然而，其实搜索体验可以更加的有趣比如加入下面的一些功能：

- 提供语音搜索功能
- 在执行搜索前，基于搜索历史为用户提供搜索建议
- 提供即时匹配，在用户开始输入时就提供应用程序内的数据

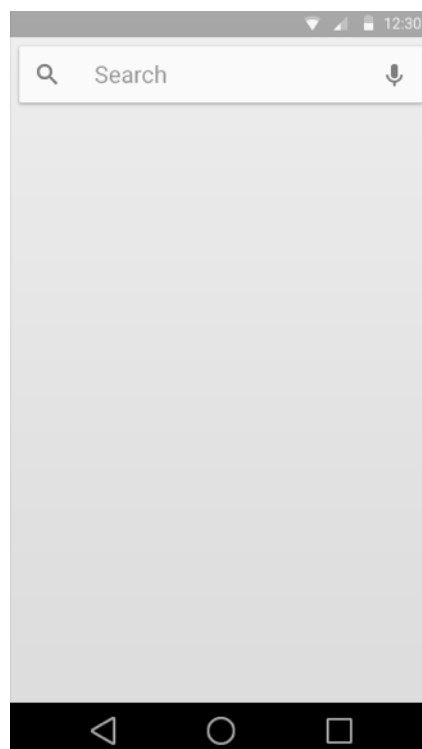
应用程序内部搜索主要有两种模式：持续式搜索和伸缩式搜索

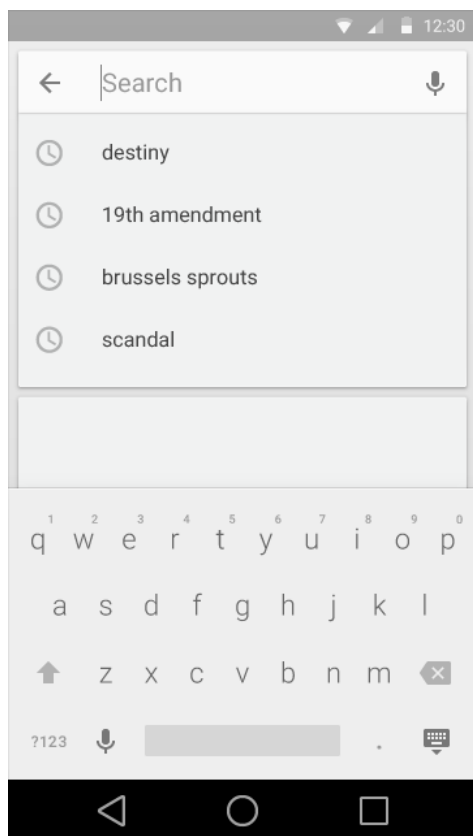
持续式搜索

持续式搜索是在需要的时候在你的应用程序内部执行搜索

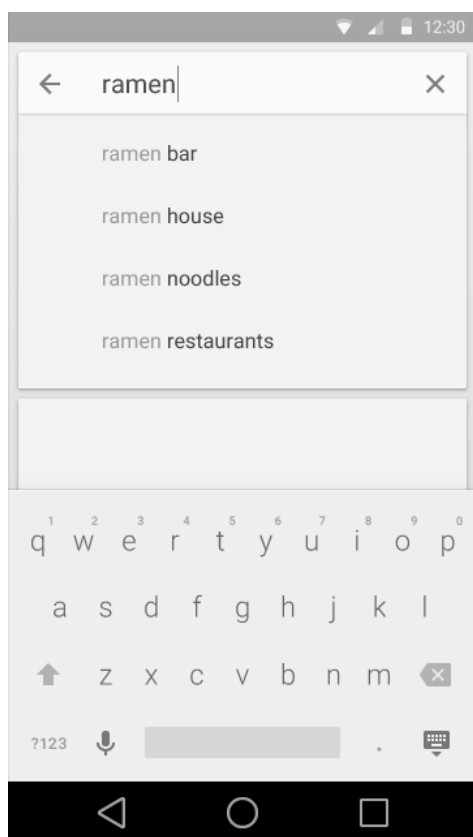
将搜索框和文本融合，随时准备接收用户的激活

用户可以通过点击麦克风的图标启动语音搜索

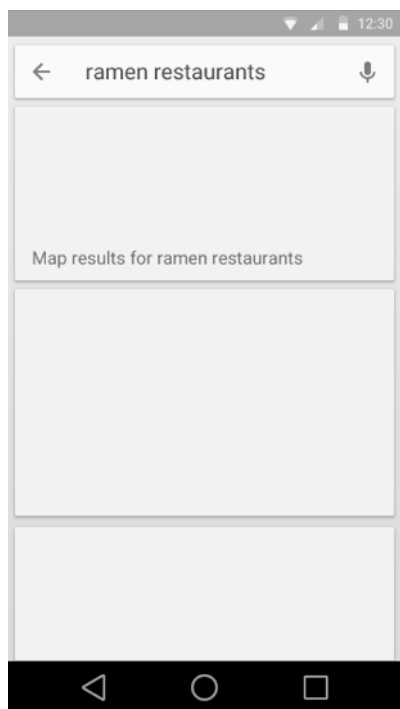




当搜索文本框被激活时，搜索框将会为用户提供搜索历史建议。如果需要，屏幕中的键盘也将出现。可以直接选择建议中提供的内容执行搜索。也可以点击上方的箭头取消搜索和收回键盘。



当显示搜索结果时，搜索框仍然可见但不处于激活状态，将屏幕键盘隐藏起来，这样可以在屏幕上显示更多地搜索结果。

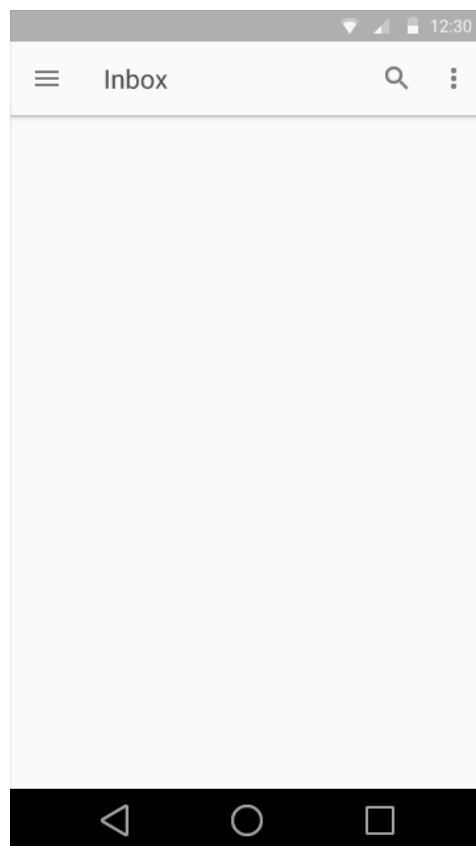


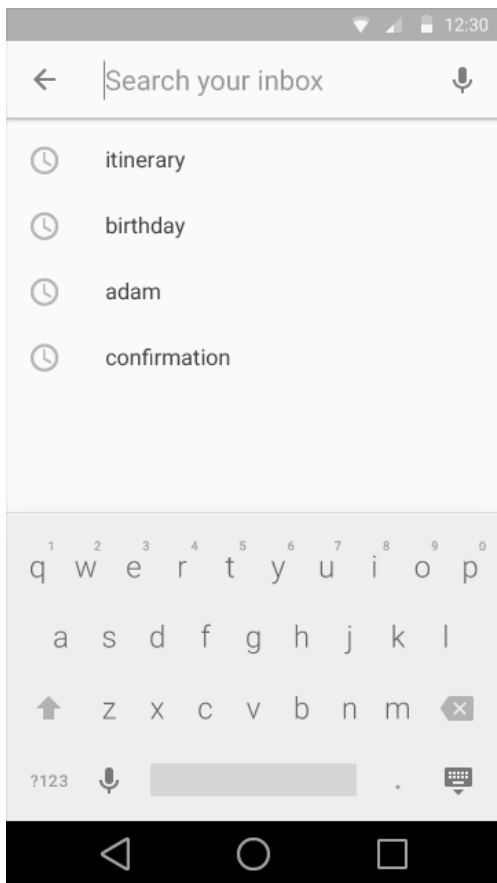
将搜索获得的结果放置在统一的卡片规范中，这个尺寸中可以容纳各种不同类型的搜索结果

可扩展搜索

扩展式搜索适合当用户需要时，再在APP中执行搜索，而不是主要的功能。

界面中并不是显示“搜索/Search”搜索框，而是将搜索按钮（放大镜图标）放在工具栏中

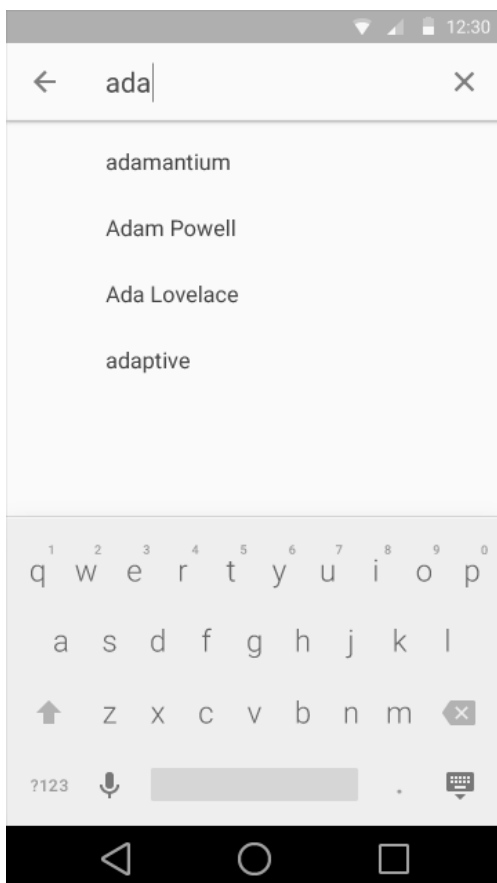




点击搜索图标时，工具栏会自动转换到搜索文本框并将其他内容隐藏。如果支持语音搜索，麦克风图标也会出现。

搜索框会接收用户激活指令，如果需要，键盘会自动出现。历史搜索建议可以取代前面所显示的内容，在工具栏下方出现。用户可以选择结果建议来提交搜索请求。

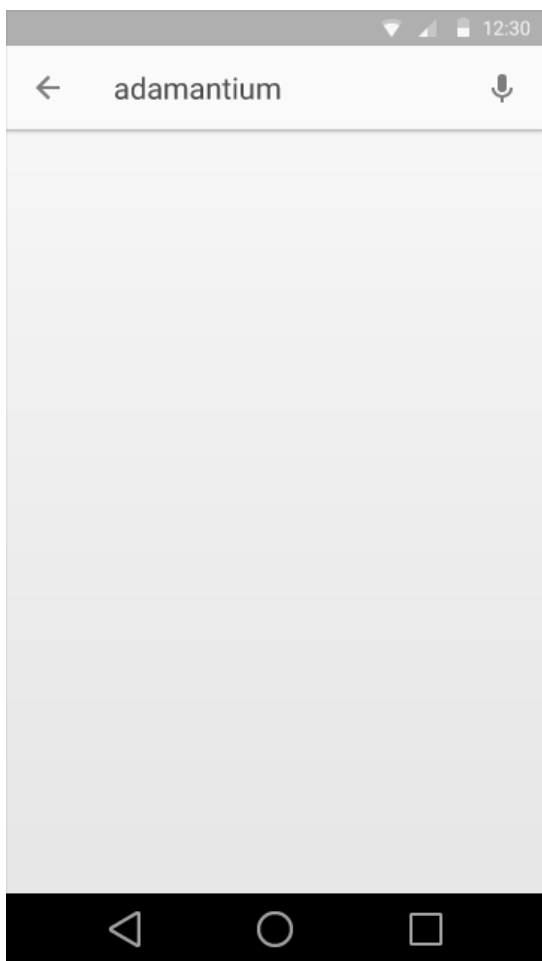
点击上方的方向键来关闭搜索模式，恢复到最初的工具栏



当用户开始输入搜索关键词时，会提供搜索建议并自动完成。

点击搜索框中的“X”按钮将会清空关键词

选择一个结果建议或者按回车就可以执行搜索任务了



当显示搜索结果时，搜索框仍然可见但不处于激活状态，将屏幕键盘隐藏起来，这样可以在屏幕上显示更多地搜索结果。

搜索结果显示，建议和前面一样，出现在页面工具栏的下方

7 可用性

无障碍

一个能被称为无障碍的产品，指的是无论身体是否健全的用户，都可以浏览理解它，并用它来实现自己的目标。一个真正成功的产品要让尽可能多的人方便使用。

设计师想要了解无障碍，这些普遍准则是一个很好的起点。设计完全无障碍的产品是一个复杂的课题，需要深入研究。欲了解更多信息，请访问谷歌无障碍站点。

你的产品如何为残疾人用户提供最好的服务？

首先，思考用户将如何用辅助技术与你的产品进行交互。想象在下列环境中使用你的产品：

- 没有声音
- 没有颜色
- 开启了高对比度模式
- 屏幕被放大
- 屏幕阅读器（没有可视屏幕）
- 只能语音控制
- 上述情况的组合

（译者注：屏幕阅读器是一款帮助盲人、不识字、学习障碍人群使用的应用，能将文字转化成语音或者盲文。随着图形界面 GUI 的发展，屏幕上显示的图形内容取代纯粹的文字表述，使屏幕报读软件的运作变得更加复杂。当你的产品覆盖到这些特殊人群时，应该考虑使用系统文字而不是图片）

然后考虑以下关键区域对无障碍的影响：导航、可读性、引导和反馈。

导航

让用户可以迅捷高效的的操作。是否能够迅捷高效地浏览网页（例如，跳转到关键页面或者返回主导航）？首先呈递最重要的信息。在设计中可以尝试一些细微的变化，能让导航更加迅捷。

可点击区域至少48×48像素。在任何屏幕尺寸小，都推荐使用48×48（px, dp）作为可点击区域的最小尺寸。同时检查你的移动应用设计中各元素之间的间隔是多少。在大多数情况下，应该是8 dp或以上。

支持鼠标和标准手势导航。设计的每个部分，都要兼顾网页界面中键盘输入的可达性和移动设备中手势输入的可达性。盲人用户无法使用鼠标，无法用可视化的方式来浏览你的界面。他们取代鼠标的方案是，使用键盘进行导航或通过划动移动设备上的元素来浏览。要确保不使用鼠标的用户也能获得鼠标悬停中的信息。确保键盘快捷键与平台标准相一致。

注意用户的视觉焦点。你确定用户和他们的焦点从来没有在导航时弹出窗口、提示、以及各种屏幕切换之间迷失？也要想用户会如何关闭弹窗后，返回到屏幕上。要确保他们的焦点能返回到弹窗打开之前。

可读性

确保产品在较大的字体尺寸下仍具可读性。当用户放大屏幕或放大字体，你的文字是否仍清晰可辨？必不可少的元素是否仍然可见、可用，而不是重叠？你应该亲自测试看使用各种内置操作系统/浏览器/应用辅助工具缩放字体是否会出现问题。

确保关键文本有足够的对比度。在大多数情况下，“足够的对比度”是指对比度比例为的4.5:1。背景和文本或关键要素之间的反差足以让所有的用户和那些视力特别差的用户，能更方便地阅读你的信息。较小的文本需要很强的对比度，而大标题则可以使用更多的颜色和背景。

不能只用色彩来传达关键信息。这对于所有色盲用户来说尤为重要。如果色彩在设计中传达的是特定信息（例如，可视化流量：红=高流量，绿色=无流量）提供的另一种方法为用户获取相同的信息是很重要的。除了使用颜色外，添加其他元素例如形状、图案、纹理、或文本。

要提供关于空间关系的线索。空间信息传达和布局，每次只能导航一个元素，这对于盲人用户来说并不是很清晰。因为盲人使用辅助技术不能标明元素之间的距离，所以需要提供的附加的线索让元素之间联系起来，比如共享一个共同的标题。

给声音出一个视觉替代方案，反之亦然。如果你有声音元素，你提供了隐藏式字幕、文字记录、或其他视觉替代方案吗？本指导原则也适用于系统警报声音。任何发光闪烁都需要被转换成声音，反之亦然。

引导和反馈

让交互式控件清晰可见。是否所有的交互式控件都有相关的文字标签、工具提示、或占位符文本来表示它们的目的是什么？你专用术语在整个应用程序中是否始终如一？在辅助技术中首先要提供最相关的信息。当命名元素时，确保你在整个应用程序中使用的术语是统一的。

图像和视频要提供可替代文本。你依靠图形元素来传达信息，是否以书面形式提供同样信息吗？对于特定的任务标签是否清楚的表达具体内容？（例如，不只是“下载”，而是“下载的晚餐菜单”）为所有图片和图标提供替代文本，在标准工具能运行时应避免使用图像表示文本。

提供指导和帮助。当用户不知道一个元素是什么意思时，他/她能快速找到帮助吗？如果一个关键元素超时，有没有办法让用户重新激活它？包括清晰并容易查找的帮助，以及提供相关情境下的帮助，这样使您的用户可以查找到哪些键盘快捷键和手势是能用的，并且知道如何访问使用这些功能。

为你的链接命名。每个链接的目的都清楚明了吗？如“单击此处”这个通用的锚文本链接并不能说明链接的目的。锚文本要能精确的描述所指向页面的内容。更好的解决方案是，把“点击这里”换成一个具体的链接，例如“设备设置”。一些辅助技术模式让用户只需扫描的链接，忽略其他内容，使导航更高效。

8 资源下载

一、布局模板

移动端

10屏的布局模板。可开关标注、边距设置。

Layout Mobile Whiteframe 2 MB (.ai)

平板

包含了14屏的平板布局模板。可开关标注、边距设置。

Layout Tablet Whiteframe 3 MB (.ai)

桌面端

包含了4屏的平板布局模板。可开关标注、边距设置。

Layout Desktop Whiteframe 96 MB (.ai)

白框图

包含了多种布局结构，提供了阴影、表面层、阴影的设计方法

Whiteframes 15 MB (.ai)

二、设计便贴

构件

构件便贴包含了多种元素，可以迅速装点你的应用布局。包含了2种主题，深色和浅色，包含状态栏，工具栏、卡片、下来菜单、搜索域、分割线、侧边导航、对话框、悬停状态按钮等元素。

Structural Elements 1.2 MB (.ai)

容器

下载下列文件

Cards Stickersheet 346.84 MB (.ai)

Dialogs Stickersheet 14.88 MB (.ai)

Dividers Stickersheet	111.5 MB (.ai)
Grids Stickersheet	235.61 MB (.ai)
Lists Stickersheet	506.47 MB (.ai)
Bottomsheet Stickersheet	41.64 MB (.ai)
Subheaders Stickersheet	58.88 MB (.ai)
Tabs Stickersheet	183.18 MB (.ai)

UI元素

请下载下列文件

Buttons Stickersheet	50.86 MB (.ai)
Chips & Tokens Stickersheet	22.16 MB (.ai)
List Controls Stickersheet	81.06 MB (.ai)
Sliders Stickersheet	0.77 MB (.ai)
Snackbar & Toast Sticker sheet	5.28 MB (.ai)
Switches Stickersheet	0.34 MB (.ai)
Text Fields Stickersheet	34 MB (.ai)
Tooltips Stickersheet	32.9 MB (.ai)

三、Roboto字体

最新版的Roboto字体，移动端和网页端均可使用。

1.21 MB (.zip)

四、配色板

其中包含了可安装到AI和PS的配色板，还包含了Read-Me PDF文档，教你如何安装.aco和.ase

Color Swatches 0.13 MB (.zip)

编者的话：

感谢谷歌研发团队对安卓的视觉规范做出统一的设定，让各厂商与设计师有所参考和学习，让混乱的安卓程序有所改进！由于文章最后将转成PDF格式，规范中的一些动画效果将不可见，望读者见谅，可以到下面的参考地址中观看在线版。

由于翻译的理解不同，时间紧促，可能导致内文有误，欢迎大家把错误的地方指正出来，我会尽快修正版本给大家，我的邮箱：larens@qq.com

感谢国内各平台的支持和推广：

极客学院，优设-UISDC，UI中国-专业界面设计平台，eoeAndroid开发者社区，七牛云存储，多说评论框，Github，Jekyll，Calibre，Semantic UI

感谢参与翻译和校对的童鞋：

来自UI.cn的朋友们：MartinRGB，王帆，Daqian.Dong，Hoscar，Larens

来自1sters的朋友们：iceskysl，zhenbeiju，7heaven，Jingsha，com360，arjinmc，acely，Haoxiqiang，lhyqy5，vincent4j，CodeDiving，donlianggit，ThatMrL，Kris，SamanthaChou，marshluca，ZeroLu，fortianwei

参考网站地址：

谷歌官方设计网站：<http://www.google.com/design/spec/material-design>

UI中国翻译集合：<http://www.ui.cn/Material/index.html>

1sters翻译集合：<http://design.1sters.com/>