

# Application GUI Design – Notes From a Toolkit Developer



**[stosb.com/talks](http://stosb.com/talks)**

Tom Hacoen  
Samsung Electronics Open Source Group  
[tom.hacoen@samsung.com](mailto:tom.hacoen@samsung.com)  
[@TomHacoen](https://twitter.com/TomHacoen)

## What does it do?

- ▶ Essential features

## What does it do?

- ▶ Essential features
- ▶ Nice to have features

## What does it do?

- ▶ Essential features
- ▶ Nice to have features
- ▶ Niche features (<1% of the users)

## What does it do?

- ▶ Essential features
- ▶ Nice to have features
- ▶ Niche features (<1% of the users)
- ▶ Remove all the non-essential features

## Who is it for?

- ▶ CLI power users? Designers?

## Who is it for?

- ▶ CLI power users? Designers?
- ▶ Target environment

## Who is it for?

- ▶ CLI power users? Designers?
- ▶ Target environment
- ▶ Common demographics

## Who is it for?

- ▶ CLI power users? Designers?
- ▶ Target environment
- ▶ Common demographics
- ▶ Application specific classifications

## Who is it for?

- ▶ CLI power users? Designers?
- ▶ Target environment
- ▶ Common demographics
- ▶ Application specific classifications
- ▶ Userbase  $\neq$  you

# KISS

- ▶ Adapt feature list according to your audience



# KISS

- ▶ Adapt feature list according to your audience
- ▶ Keep focus on the more important features



# KISS

- ▶ Adapt feature list according to your audience
- ▶ Keep focus on the more important features
- ▶ Avoid creating complex UIs



# KISS

- ▶ Adapt feature list according to your audience
- ▶ Keep focus on the more important features
- ▶ Avoid creating complex UIs
- ▶ Keep option lists (combo box) short, simple and if there's no choice, split to categories



## Have a rough sketch

- ▶ Make a general storyboard sketch

## Have a rough sketch

- ▶ Make a general storyboard sketch
- ▶ Have all the major interactions and features there

## Have a rough sketch

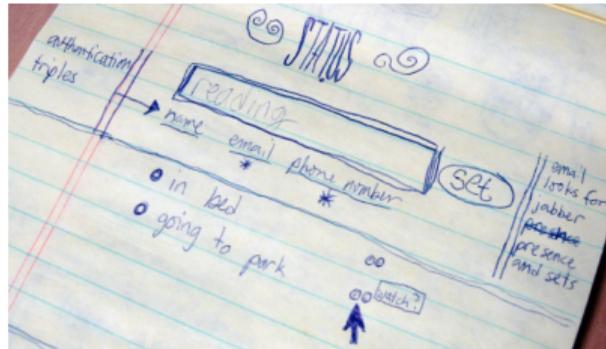
- ▶ Make a general storyboard sketch
- ▶ Have all the major interactions and features there
- ▶ Can be really rough – it's more about the purpose of the “pages”

## Have a rough sketch

- ▶ Make a general storyboard sketch
- ▶ Have all the major interactions and features there
- ▶ Can be really rough – it's more about the purpose of the “pages”

## Have a rough sketch

- ▶ Make a general storyboard sketch
- ▶ Have all the major interactions and features there
- ▶ Can be really rough – it's more about the purpose of the “pages”



## Stick to the basics

- ▶ Don't bother with colour – harder and will be added later

## Stick to the basics

- ▶ Don't bother with colour – harder and will be added later
- ▶ Do not customise available widgets/patterns unless there really is no other way

## Stick to the basics

- ▶ Don't bother with colour – harder and will be added later
- ▶ Do not customise available widgets/patterns unless there really is no other way
- ▶ Develop it around the content – content is king

## I meant it! KISS...

- ▶ Be purposefully simple



## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad



## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad
- ▶ Be consistent (easier when simple)

Next

Next

## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad
- ▶ Be consistent (easier when simple)
- ▶ Make it simple for your users – don't create new usage patterns, and adhere to common ones



## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad
- ▶ Be consistent (easier when simple)
- ▶ Make it simple for your users – don't create new usage patterns, and adhere to common ones
- ▶ Common (all?) usage patterns should be easily available



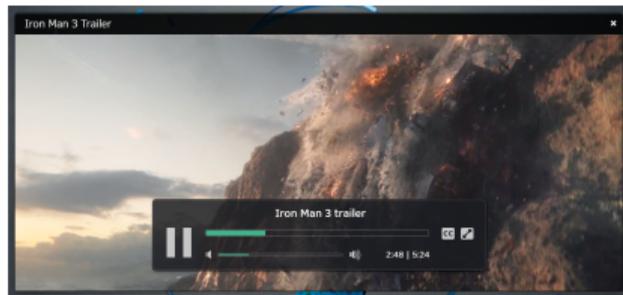
## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad
- ▶ Be consistent (easier when simple)
- ▶ Make it simple for your users – don't create new usage patterns, and adhere to common ones
- ▶ Common (all?) usage patterns should be easily available
  - ▶ Rest should be revealed as needed



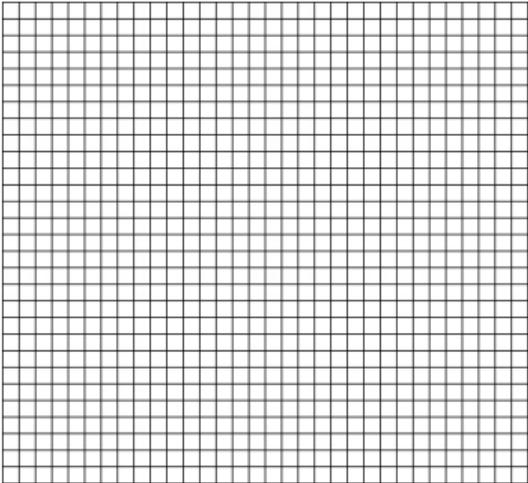
## I meant it! KISS...

- ▶ Be purposefully simple
- ▶ Skeuomorphs are almost always bad
- ▶ Be consistent (easier when simple)
- ▶ Make it simple for your users – don't create new usage patterns, and adhere to common ones
- ▶ Common (all?) usage patterns should be easily available
  - ▶ Rest should be revealed as needed
- ▶ Make self-documenting applications



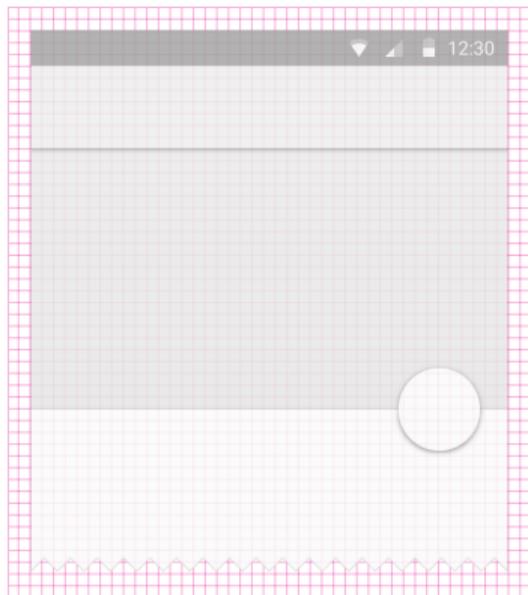
# Spacing

- ▶ Choose a baseline unit size (e.g. 8px)



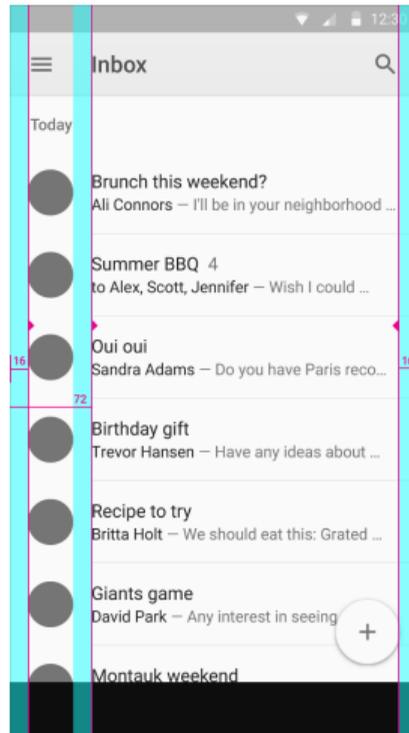
# Spacing

- ▶ Choose a baseline unit size (e.g. 8px)
- ▶ Align everything to that baseline size



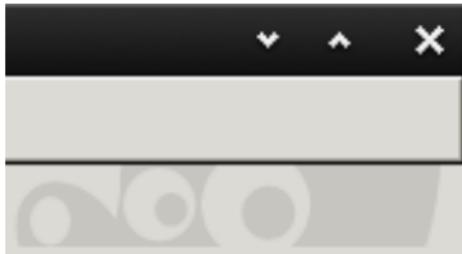
# Spacing

- ▶ Choose a baseline unit size (e.g. 8px)
- ▶ Align everything to that baseline size
- ▶ Use a small set of spacing alternatives (e.g. 2, 3 and 6 units)



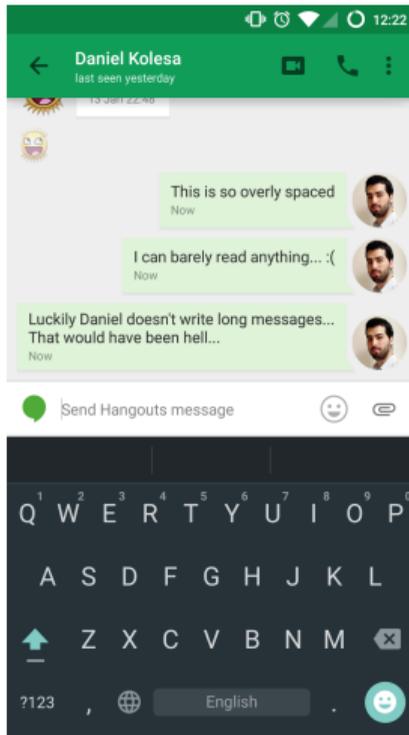
## Spacing

- ▶ Choose a baseline unit size (e.g. 8px)
- ▶ Align everything to that baseline size
- ▶ Use a small set of spacing alternatives (e.g. 2, 3 and 6 units)
- ▶ Make sure touch-targets are spaced enough



# Spacing

- ▶ Choose a baseline unit size (e.g. 8px)
- ▶ Align everything to that baseline size
- ▶ Use a small set of spacing alternatives (e.g. 2, 3 and 6 units)
- ▶ Make sure touch-targets are spaced enough
- ▶ Be generous, but don't overdo it (don't waste my screen estate)



# Organization

- ▶ Content should be at the front of the stage



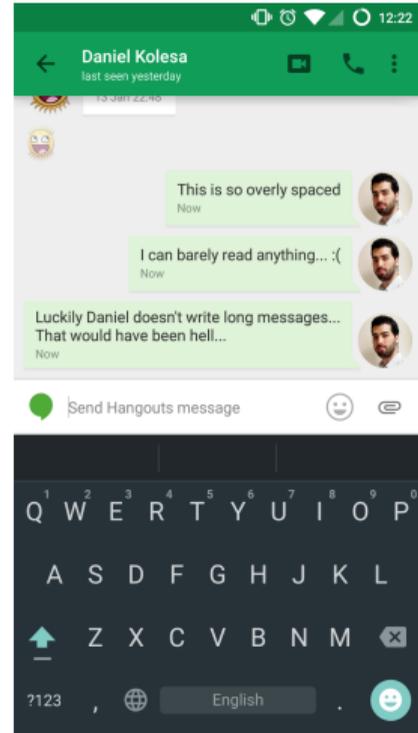
# Organization

- ▶ Content should be at the front of the stage
- ▶ Important functionality in key positions



# Organization

- ▶ Content should be at the front of the stage
- ▶ Important functionality in key positions
- ▶ Make the hierarchy of information clear



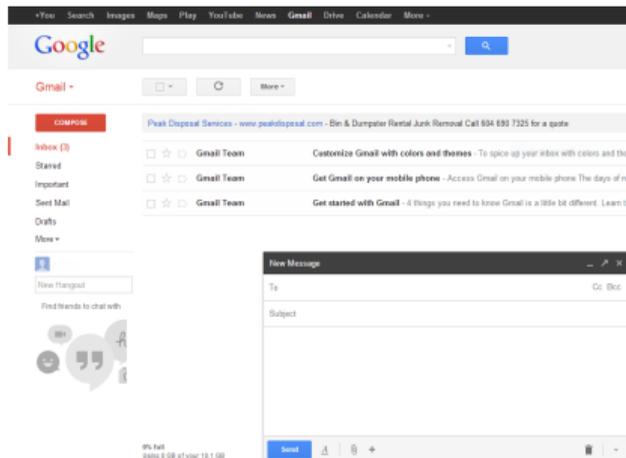
## Organization

- ▶ Content should be at the front of the stage
- ▶ Important functionality in key positions
- ▶ Make the hierarchy of information clear
- ▶ Associate related elements



# Organization

- ▶ Content should be at the front of the stage
- ▶ Important functionality in key positions
- ▶ Make the hierarchy of information clear
- ▶ Associate related elements
- ▶ Help directing the user's focus



## Icons and images

- ▶ Use known icons on buttons



## Icons and images

- ▶ Use known icons on buttons
- ▶ Don't use a known icon for something other than intended





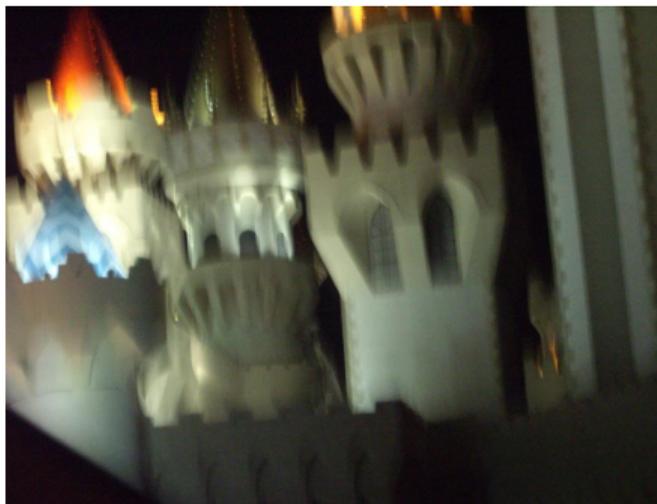
# Icons and images

- ▶ Use known icons on buttons
- ▶ Don't use a known icon for something other than intended
- ▶ Use rich graphics when appropriate (cover-art, mood graphics, etc.)
- ▶ Don't overshadow content



## Icons and images

- ▶ Use known icons on buttons
- ▶ Don't use a known icon for something other than intended
- ▶ Use rich graphics when appropriate (cover-art, mood graphics, etc.)
- ▶ Don't overshadow content
- ▶ Don't use ugly graphics



## Icons and images

- ▶ Use known icons on buttons
- ▶ Don't use a known icon for something other than intended
- ▶ Use rich graphics when appropriate (cover-art, mood graphics, etc.)
- ▶ Don't overshadow content
- ▶ Don't use ugly graphics
- ▶ Have consistent sizing



# Typography

- ▶ Use a readable font size

Text should be big and readable for everyone

# Typography

- ▶ Use a readable font size
- ▶ Space up your text

Alice's Adventures in Wonderland (commonly shortened to Alice in Wonderland) is an 1865 novel written by English author Charles Lutwidge Dodgson under the pseudonym Lewis Carroll. It tells of a girl named Alice falling through a rabbit hole into a fantasy world populated by peculiar, anthropomorphic creatures. The tale plays with logic, giving the story lasting popularity with adults as well as with children.

Alice's Adventures in Wonderland (commonly shortened to Alice in Wonderland) is an 1865 novel written by English author Charles Lutwidge Dodgson under the pseudonym Lewis Carroll. It tells of a girl named Alice falling through a rabbit hole into a fantasy world populated by peculiar, anthropomorphic creatures. The tale plays with logic, giving the story lasting popularity with adults as well as with children.

## Typography

- ▶ Use a readable font size
- ▶ Space up your text
- ▶ Use text attributes (e.g. bold and size) to make text more or less prominent

**Alice's Adventures in Wonderland** (commonly shortened to **Alice** in Wonderland) is an 1865 novel written by English author *Charles Lutwidge Dodgson* under the pseudonym Lewis Carroll. It tells of a girl named **Alice** falling through a rabbit hole into a fantasy world populated by peculiar, anthropomorphic creatures. The tale plays with logic, giving the story lasting popularity with adults as well as with children.

## Typography

- ▶ Use a readable font size
- ▶ Space up your text
- ▶ Use text attributes (e.g. bold and size) to make text more or less prominent
- ▶ Use a small set of fonts (probably one)



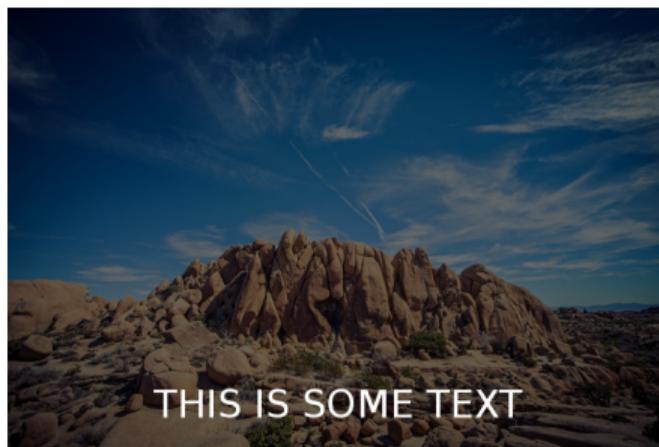
# Overlaying text on images

- ▶ Just put it on (bad)



## Overlaying text on images

- ▶ Just put it on (bad)
- ▶ Black/colour/white-wash the whole image to make text readable



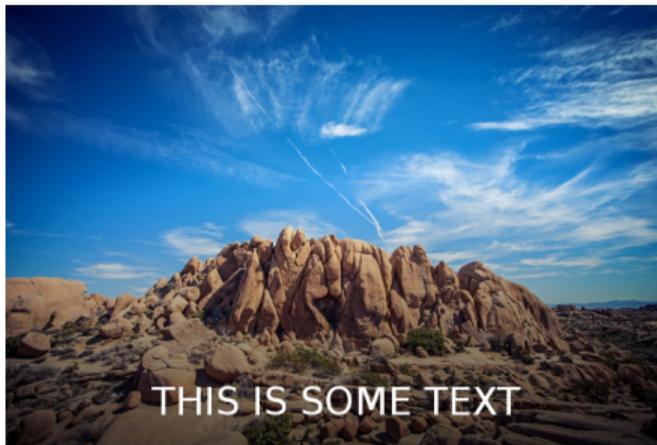
## Overlaying text on images

- ▶ Just put it on (bad)
- ▶ Black/colour/white-wash the whole image to make text readable
- ▶ Add translucent background to the text



## Overlaying text on images

- ▶ Just put it on (bad)
- ▶ Black/colour/white-wash the whole image to make text readable
- ▶ Add translucent background to the text
- ▶ Add a translucent gradient (i.e. partial black/white-wash) so your text area is handled



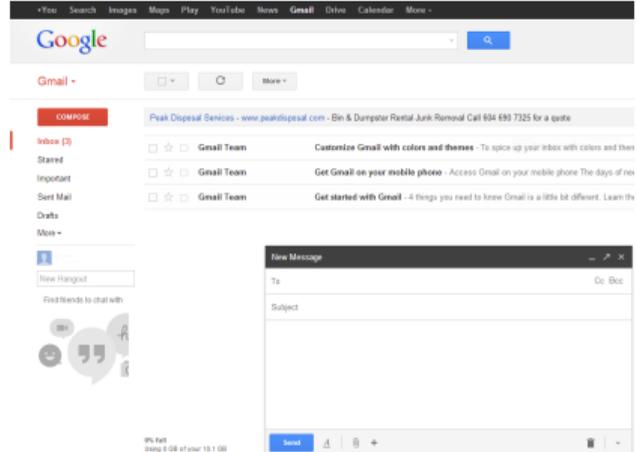
## Overlaying text on images

- ▶ Just put it on (bad)
- ▶ Black/colour/white-wash the whole image to make text readable
- ▶ Add translucent background to the text
- ▶ Add a translucent gradient (i.e. partial black/white-wash) so your text area is handled
- ▶ Use a big font with a shadow and on outline



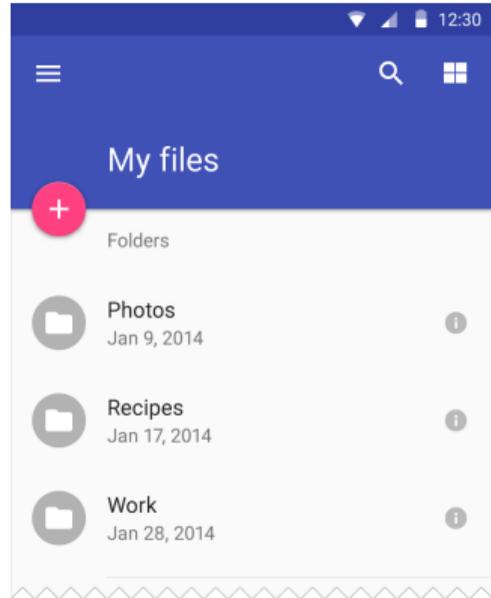
# Colour

- ▶ Option 1 (easier): keep the interface b&w and use colour to direct focus



## Colour

- ▶ Option 1 (easier): keep the interface b&w and use colour to direct focus
- ▶ Option 2: Choose a palette of 2 different hues and use different shades



## Colour

- ▶ Option 1 (easier): keep the interface b&w and use colour to direct focus
- ▶ Option 2: Choose a palette of 2 different hues and use different shades
- ▶ I almost always tint my greys (and not use black)

**Some text!**  
**Some text!**

# Colour

- ▶ Option 1 (easier): keep the interface b&w and use colour to direct focus
- ▶ Option 2: Choose a palette of 2 different hues and use different shades
- ▶ I almost always tint my greys (and not use black)
- ▶ Find a good palette online

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

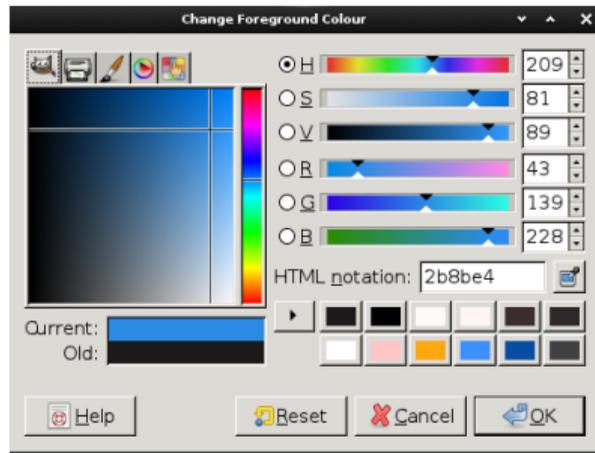
Accent – Pink	
A200	#FF4081

Fallback	
A100	#FF80AB
A400	#F50057

## Colour

- ▶ Option 1 (easier): keep the interface b&w and use colour to direct focus
- ▶ Option 2: Choose a palette of 2 different hues and use different shades
- ▶ I almost always tint my greys (and not use black)
- ▶ Find a good palette online
- ▶ Use HSV rather RGB when choosing colours



## Mimic what works

- ▶ Compare to other applications you/users like better using what we've covered

## Mimic what works

- ▶ Compare to other applications you/users like better using what we've covered
- ▶ Mimic what's good there (e.g. do you need more spacing?)

## Mimic what works

- ▶ Compare to other applications you/users like better using what we've covered
- ▶ Mimic what's good there (e.g. do you need more spacing?)
- ▶ It's usually easier to mimic than to design from scratch

## Mimic what works

- ▶ Compare to other applications you/users like better using what we've covered
- ▶ Mimic what's good there (e.g. do you need more spacing?)
- ▶ It's usually easier to mimic than to design from scratch
- ▶ Don't mimic the bad things (i.e. use them as excuse)

## Mimic what works

- ▶ Compare to other applications you/users like better using what we've covered
- ▶ Mimic what's good there (e.g. do you need more spacing?)
- ▶ It's usually easier to mimic than to design from scratch
- ▶ Don't mimic the bad things (i.e. use them as excuse)
- ▶ Don't copy, learn. . .

## User experience tips

- ▶ Consistent behaviour (with the platform and within the application)

## User experience tips

- ▶ Consistent behaviour (with the platform and within the application)
- ▶ Start instantly and lazy load in the background

## User experience tips

- ▶ Consistent behaviour (with the platform and within the application)
- ▶ Start instantly and lazy load in the background
- ▶ Everything should take a small amount of clicks

## User experience tips

- ▶ Consistent behaviour (with the platform and within the application)
- ▶ Start instantly and lazy load in the background
- ▶ Everything should take a small amount of clicks
- ▶ Discoverable UI (easy to figure out how to do things)

## More user experience tips

- ▶ Make it harder to make mistakes



# More user experience tips

- ▶ Make it harder to make mistakes
- ▶ Avoid interruptions



## More user experience tips

- ▶ Make it harder to make mistakes
- ▶ Avoid interruptions
- ▶ Sort long lists in a predictable, sensible order

## More user experience tips

- ▶ Make it harder to make mistakes
- ▶ Avoid interruptions
- ▶ Sort long lists in a predictable, sensible order
- ▶ If your users do something and expect something to happen, it probably should happen

## Open-source GUI toolkits

- ▶ Opted for a consistent theme-able system, not individually styled apps

## Open-source GUI toolkits

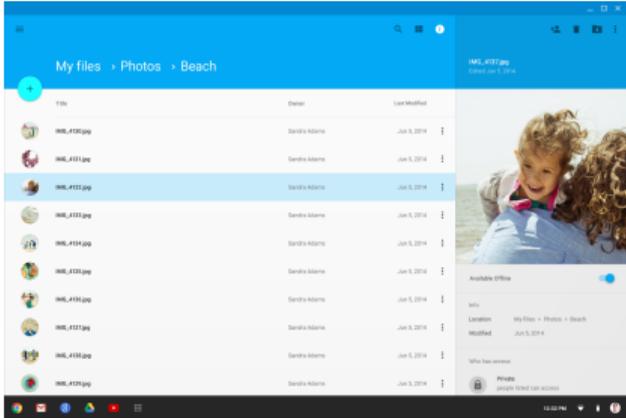
- ▶ Opted for a consistent theme-able system, not individually styled apps
- ▶ Usually look very similar (for better or worse)

## Open-source GUI toolkits

- ▶ Opted for a consistent theme-able system, not individually styled apps
- ▶ Usually look very similar (for better or worse)
- ▶ Mostly similar in their widget offering and can be styled to look similar

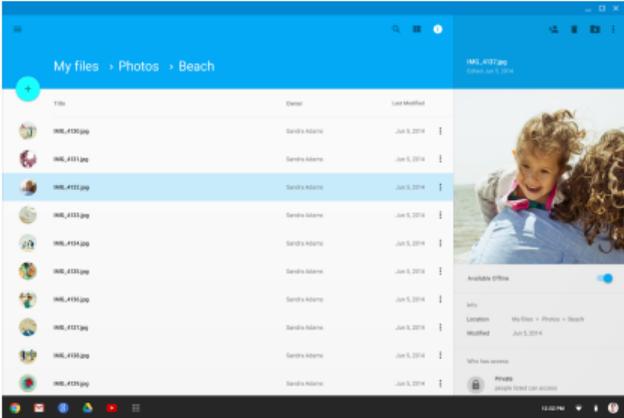
# Material design (Google)

- ▶ Highly talked about cross device design guidelines



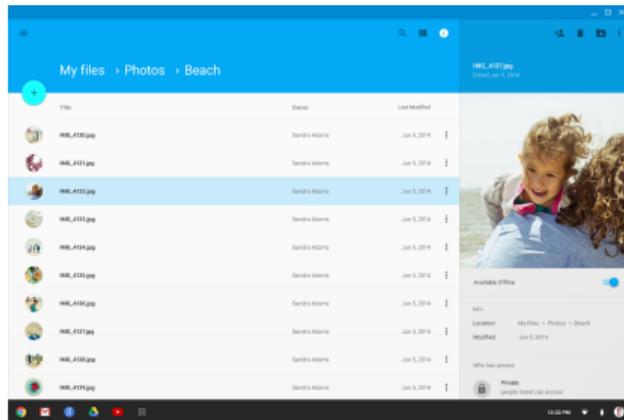
# Material design (Google)

- ▶ Highly talked about cross device design guidelines
- ▶ A fairly good Android adoption rate



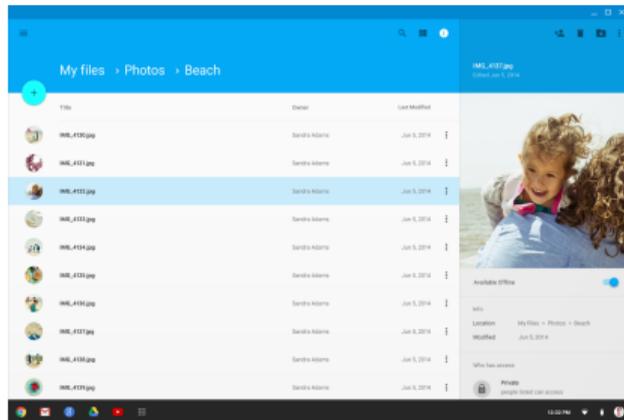
## Material design (Google)

- ▶ Highly talked about cross device design guidelines
- ▶ A fairly good Android adoption rate
- ▶ I personally don't like animations that slow users down



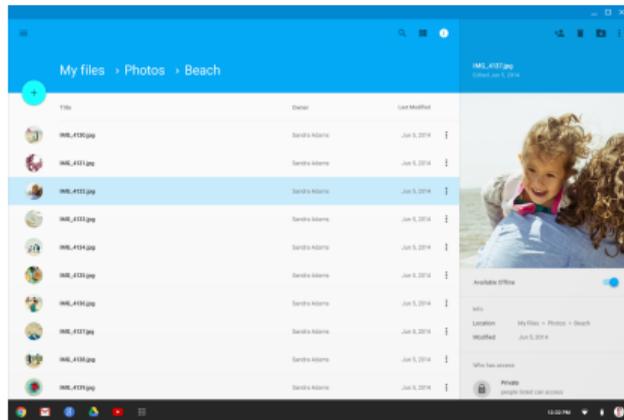
## Material design (Google)

- ▶ Highly talked about cross device design guidelines
- ▶ A fairly good Android adoption rate
- ▶ I personally don't like animations that slow users down
- ▶ Concepts are good, it's implementations that are not always good



## Material design (Google)

- ▶ Highly talked about cross device design guidelines
- ▶ A fairly good Android adoption rate
- ▶ I personally don't like animations that slow users down
- ▶ Concepts are good, it's implementations that are not always good
- ▶ It doesn't feel like they care about low-end and power consumption



## iOS guidelines

- ▶ Old style: horrible skeuomorphism

## iOS guidelines

- ▶ Old style: horrible skeuomorphism
- ▶ New style: very simple and clean, sometimes confusing

## iOS guidelines

- ▶ Old style: horrible skeuomorphism
- ▶ New style: very simple and clean, sometimes confusing
- ▶ Flat and easy to theme

## iOS guidelines

- ▶ Old style: horrible skeuomorphism
- ▶ New style: very simple and clean, sometimes confusing
- ▶ Flat and easy to theme
- ▶ Good general guidelines

## iOS guidelines

- ▶ Old style: horrible skeuomorphism
- ▶ New style: very simple and clean, sometimes confusing
- ▶ Flat and easy to theme
- ▶ Good general guidelines
- ▶ Usually not very discoverable

# EFL

- ▶ Proportional layout (hard to create non-scalable layouts)



# EFL

- ▶ Proportional layout (hard to create non-scalable layouts)
- ▶ We make it harder to not use the system theme



## EFL

- ▶ Proportional layout (hard to create non-scalable layouts)
- ▶ We make it harder to not use the system theme
- ▶ We are mostly flat, use 3d when helps usability (similar to Material)



## Know your toolkit

- ▶ Trying to pixel match a design across toolkits is stupid

## Know your toolkit

- ▶ Trying to pixel match a design across toolkits is stupid
- ▶ Know what takes a performance hit on your toolkit

## Know your toolkit

- ▶ Trying to pixel match a design across toolkits is stupid
- ▶ Know what takes a performance hit on your toolkit
- ▶ Check out the toolkit's common patterns, those are usually best for performance and users

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling
- ▶ Start with black and white, don't colour unless you need it

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling
- ▶ Start with black and white, don't colour unless you need it
- ▶ Saturate your greys and don't use black

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling
- ▶ Start with black and white, don't colour unless you need it
- ▶ Saturate your greys and don't use black
- ▶ Add spacing and make your text bigger

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling
- ▶ Start with black and white, don't colour unless you need it
- ▶ Saturate your greys and don't use black
- ▶ Add spacing and make your text bigger
- ▶ Use common patterns (and take inspiration from the best)

## Easy takeaways

- ▶ It's like API design, it should be simple, self-documenting, consistent and have a familiar feeling
- ▶ Start with black and white, don't colour unless you need it
- ▶ Saturate your greys and don't use black
- ▶ Add spacing and make your text bigger
- ▶ Use common patterns (and take inspiration from the best)
- ▶ Pay the design some thought (from the start)

## Less obvious example

- ▶ The hell that is going to the cinema in London...

Questions?

Tom Hacoen

[tom.hacoen@samsung.com](mailto:tom.hacoen@samsung.com)

<http://stosb.com>

@TomHacoen

- ▶ Page 11, flight-deck.jpg
- ▶ Page 15, twitter-sketch.jpg
- ▶ Page 23, purposefully-simple.png
- ▶ Page ??, skeuomorph.jpg
- ▶ Page ??, self-documenting.png
- ▶ Page ??, layout-baseline-align.png
- ▶ Page ??, layout-spacing-alternatives.png
- ▶ Page 55, direct-attention.png

- ▶ Page ??, imagery-mood.png
- ▶ Page ??, bad-blurry.png
- ▶ Page ??, multiple-fonts.png
- ▶ Page 50, text-overlay-bad.jpg
- ▶ Page ??, two-colours.png
- ▶ Page ??, palette.png
- ▶ Page 69, repo-delete-confirmation.png
- ▶ Page ??, interrupt-programmer.jpg
- ▶ Page 76, material.png