3

Tiny Touchscreen

DESIGNING FOR SIZE AND TOUCH

This means your project is not only a challenge of visual and graphic design but of industrial design, too. Just as surely as if you were soldering circuit boards, molding plastic, or shuffling die-cast buttons, you're designing an app to be manhandled—or at least hand-handled. There are honest-to-god ergonomic issues to account for. Considerations of size and touch combine in iPhone design to present new challenges for interaction designers.

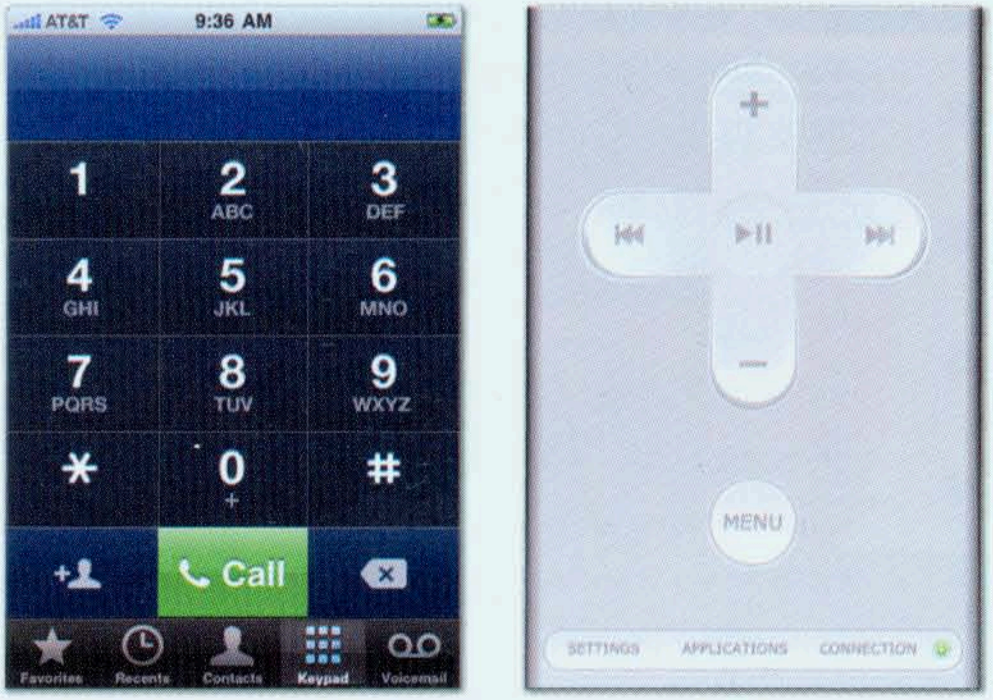
You've already seen how manner and mind-set shape the broad environment for your app and the features you should pursue. The rest of this book deals with a much more immediate set of constraints: the form and conventions of the device itself. This chapter kicks off the process by reviewing the big picture of designing for a tiny touchscreen. You'll explore how fingers and thumbs roam the screen, discover some ergonomic guidelines for comfortable tapping, and wrap up with some good practices for tap-friendly screen layouts.

A Physical Feel

More than simply how the app looks, you have to consider how it *feels*. How well does your interface work when used one-handed? Are the most common tap targets within easy thumb range—and what about lefties? Are buttons chunky enough for easy tapping, or does it take surgical precision to hit them?

The tactile nature of iPhone apps is reinforced by the powerful illusion of real-world physics conjured by the iPhone operating system. The iPhone's interactive metaphors are all about tapping, sliding, and flicking—direct manipulation of onscreen objects that respond with lifelike realism. Flick a scrolling list and watch it slow as "friction" takes hold. Fling it even harder against the top or bottom of the screen and the thing actually bounces. Tug the screen down and watch it snap back into place with rubber-band realism. Everything on the iPhone responds to the familiar rules that apply to the everyday physical world: inertia, momentum, friction, elasticity, follow through. The result is an irresistible impression that you're working with real-world objects.

Don't break that spell. More than just tending to appearances—*wow, it looks so real!*—creating an interface that "feels" like an actual device matches expectations created by the iPhone operating system as well as the concrete reality of tapping away at the hardware itself. (Chapter 8 has more on phone physics and visual feedback, starting on page 257.) Some apps even go all the way with the physical-device metaphor by mimicking the look of familiar gadgets. There are undeniable usability benefits to cribbing a real-world interface that's been used for decades or even centuries, especially when your app performs a similar role to that of the original gizmo. Few will scratch their heads to figure out how to make a call with the built-in Phone app. Its push-button interface conveys its meaning immediately (even three-year olds, for better or worse, know how to dial a phone), and the keypad layout also has the benefit of proven ergonomic success in a handheld device.



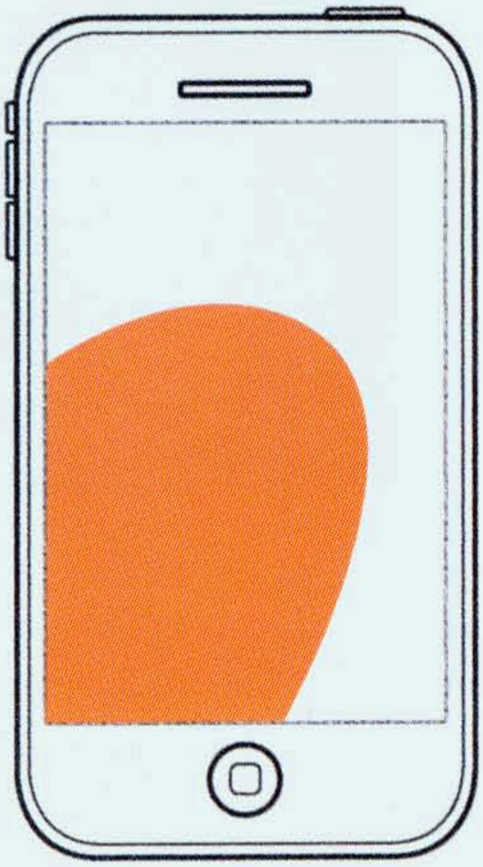
Interfaces that mimic physical devices benefit from intuitive familiarity and proven ergonomics. Here, Phone and Rowmote, respectively, crib the telephone keypad and Apple's physical remote control.

You'll find more discussion of interface metaphors in Chapter 6, but whether or not your app actually apes the look of a mechanical device, that's exactly what it is in practice. Your app's screen design is the sole interface for the iPhone, a gizmo that works by hand, and you have to address the demands of a physical device. When fingers do the walking, designers have to clear the way for them.

Organizing the layout of an iPhone interface means organizing for fingers. More precisely, it means organizing for *the thumb,* since that's the digit that gets the workout when you work the iPhone one-handed. The iPhone is sized perfectly for use in a single hand, allowing your thumb to sweep easily from one corner to the other with only a modest stretch. And because you *can* use the iPhone with one hand, most of us very often do. Whether you're hoisting a coffee, hauling a baby, or eating your lunch with that other hand, you can still keep tapping away with one thumb to answer email, make calls, or browse the web. Your design should optimize for this one-thumbed tapping.

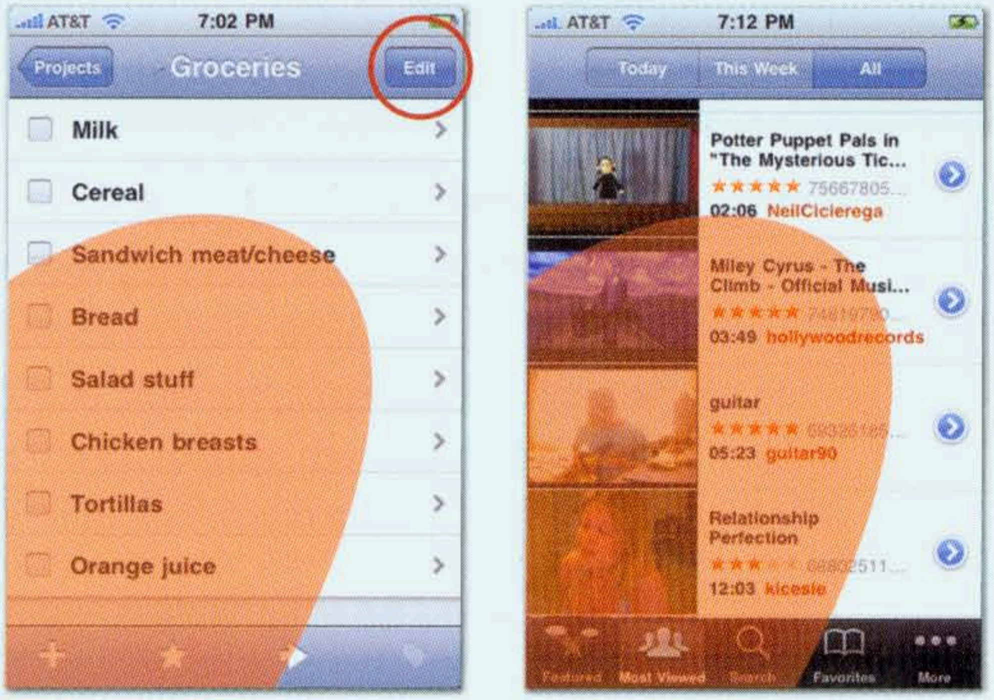
Rule of Thumb

Thumbs are marvelous. It's our thumbs, along with our affection for celebrity gossip, that separate us from the beasts, but they do have limited range and flexibility. While a thumb can manage to sweep the entire screen, only about a third of the screen is in truly effortless territory—at the side of the screen *opposite* the thumb. For a comfortable ergonomic experience, you should place primary tap targets in this thumb-thumping hot zone. (We'll focus for now on right-handed users, but hang in there, lefties, we'll get to you in a sec.)



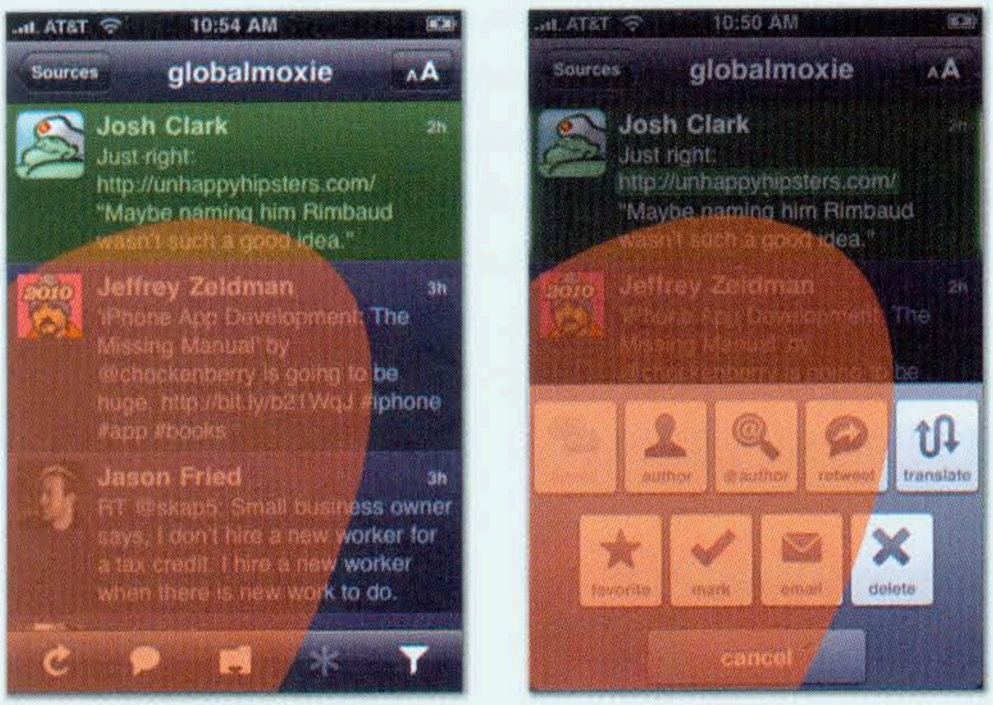
The comfort zone for the right thumb falls on the opposite side of the screen, at the left edge and bottom of the screen. (The top right and bottom right corners are the toughest thumb zones for right-handed users.)

That's an important reason why toolbars and tab bars always go at the bottom edge of the iPhone screen—the opposite of what we're accustomed to for traditional screen interfaces. Desktop software conventions put menus at the top of the screen or window, and websites typically position primary navigation at the top of pages. Our limited thumbspan, however, flips that convention on its head. Navigation and primary tap targets sink to the bottom on the iPhone. This tap zone gives you hints about how to organize the visual hierarchy of tap targets. Frequently used buttons and navigation tabs should occupy the bottom left of the screen, while lesser used buttons and those that make changes to data can be tucked safely away at top right. The Edit button for changing, deleting, or reordering list items, for example, is conventionally placed at top right, putting it in easy view but also in an isolated and relatively difficult spot to tap, making accidental changes less likely.



The standard iPhone toolbar (left) and tab bar (right) always go at the bottom edge of the iPhone screen in convenient thumb-tapping range. To-do list app Things (left) puts the Edit button at top right safely out of accidental tap range. At right, YouTube's left-edge video thumbnails are chunky tap targets in the hot zone of right-handed users.

Let your thumb point the way in laying out your screens according to the most common use cases for your app. Twitterrific, an app for Twitter users, organizes its buttons according to this thumb-thinking hierarchy. The two left toolbar icons respectively refresh and post tweets, reflecting the two most common Twitter-related activities, reading and posting. When you want to do something with an individual tweet, tapping the toolbar's asterisk icon summons a set of buttons whose thumb convenience likewise reflect their importance—commonly tapped buttons at left, less common at right. "We think a lot about where things get placed based upon ergonomics," explains Craig Hockenberry, Twitterrific's lead developer. "The Delete button is off to the right, the hardest location to tap for right-handed users, and we put other options where your thumb has to work less in order to get to more commonly used actions." (For more about Twitterrific's design, see the developer close-up on page 205.)



Josh Clark 2h
Just right:
http://unhappyhipsters.com/
"Maybe naming him Rimbaud wasn't such a good idea."

Jeffrey Zeldman 3h
iPhone App Development: The Missing Manual" by @chockenberry is going to be huge. http://bit.ly/b21WqJ #iphone #app #books

Jason Fried 3h
RT @skap5: Small business owner says, I don't hire a new worker for a tax credit. I hire a new worker when there is new work to do.

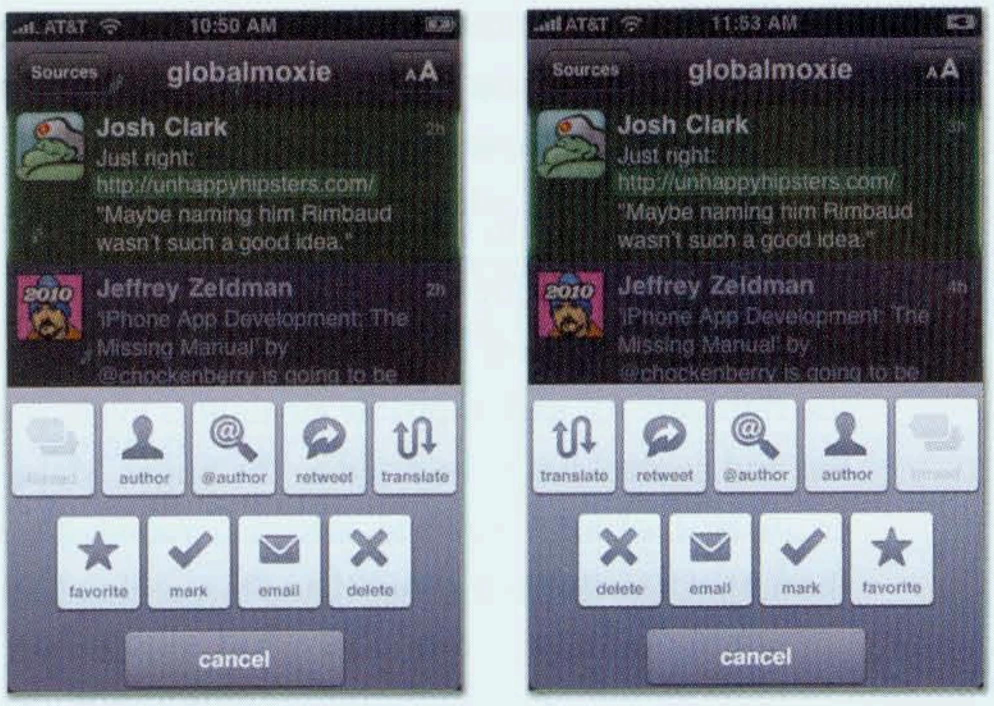
Josh Clark 2h
Just right:
http://unhappyhipsters.com/
"Maybe naming him Rimbaud wasn't such a good idea."

Jeffrey Zeldman 2h
iPhone App Development: The Missing Manual" by @chockenberry is going to be

Twitterrific orders its toolbar icons so that the most frequently tapped buttons appear in the right thumb's hot zone, highlighted here in red.

Happily, this organization happens to coincide with the way we read—our eyes naturally scan onscreen menus from left to right, just like the written word. So, for right-handed use at least, physical and visual considerations align. But what about lefties? The hot zone for the left thumb is, naturally, the mirror image of that of the right, which means left-to-right organization makes for awkward tapping for southpaws. When you optimize for right-handed people—about 85 to 90 percent of users—you're actively inconveniencing left-handed folks (as well as righties who switch to the left hand while they scribble a note).

Some conscientious apps, including Twitterrific, go so far as to include a setting for lefties to flip the layout, putting common controls back in the thumb's hot zone. For very tap-intensive apps, this may be a worthwhile strategy. Consider a calculator, for example, which is all about constant tapping. Like Twitterrific, the scientific calculator app PCalc includes an option to flip the keypad layout to accommodate lefties (see page 69). There are downsides to this approach, though. Ordering the most important tap targets from right to left reverses the way we visually process information, asking lefties to burn a little extra brain power to take in your interface. (This effort is repaid with improved ergonomics for frequent taps once they grok the layout, of course.) Perhaps more important, a left-handed option adds an additional preference to your app's settings, which as you'll see starting on page 176, should be pared to a minimum. Finally, there's the addition of a modest amount of code to do the visual switcheroo on your app's controls. Whether it's worth the extra work to ease the thumbstrain of 10 percent of your



Josh Clark 2h
Just right: http://unhappyhipsters.com/ "Maybe naming him Rimbaud wasn't such a good idea."

Jeffrey Zeldman 2h
'iPhone App Development: The Missing Manual' by @chockenberry is going to be

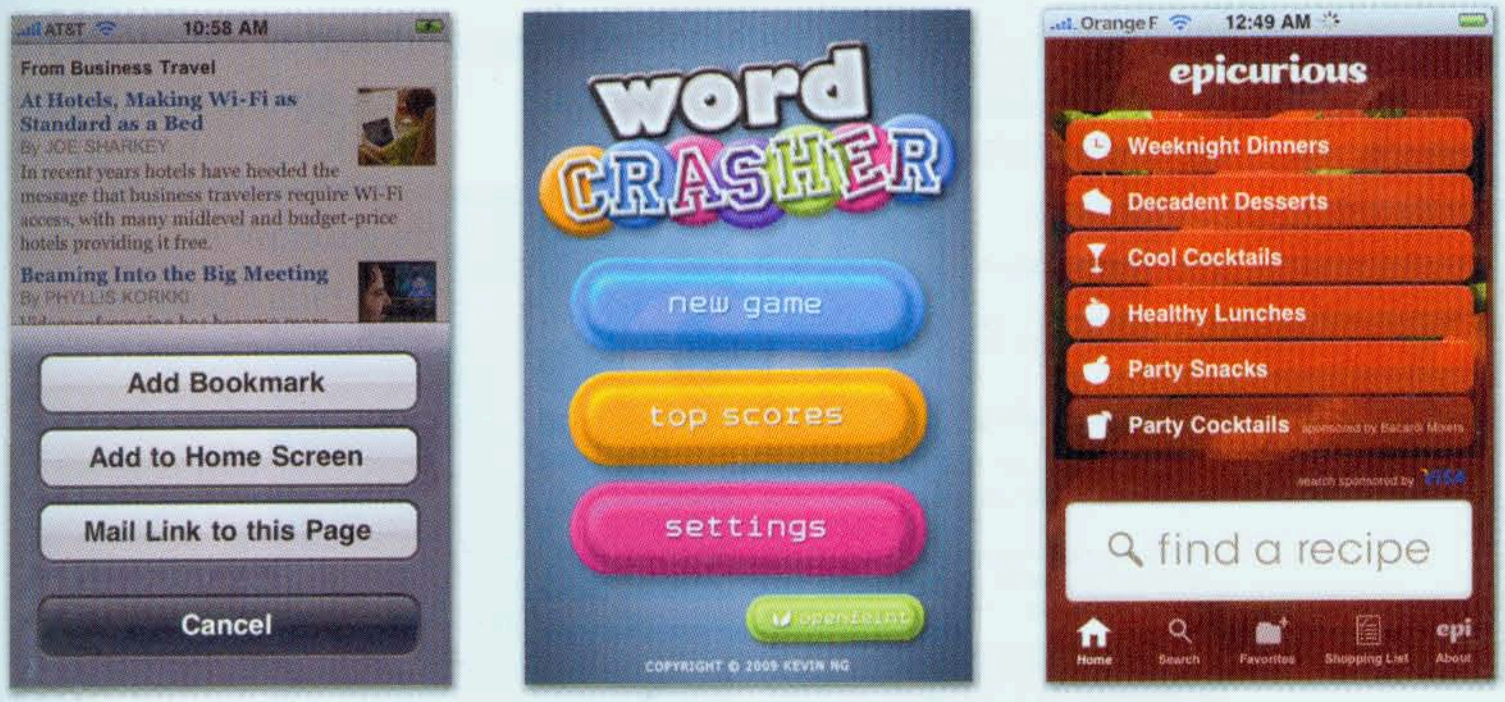
Josh Clark 3h
Just right: http://unhappyhipsters.com/ "Maybe naming him Rimbaud wasn't such a good idea."

Jeffrey Zeldman 4h
'iPhone App Development: The Missing Manual' by @chockenberry is going to be

Twitterrific offers a setting for left-handed controls, flipping the normal button layout (left) to a mirror layout (right) better suited to lefties.

users depends, as always, on the app. A complex, button-heavy, tap-intensive interface might suggest more mercy for lefties than apps that require less manual manipulation.

In interface design as in politics, sometimes it's better just to meet in the middle. Many standard iPhone controls, including buttons and list items, span the entire width of the screen, an equal-opportunity layout for both left and right thumbs. When space allows, full-width controls are the way to go—an important reason, for example, that wide buttons are cooked into the layout of *action sheets,* the



From Business Travel

At Hotels, Making Wi-Fi as Standard as a Bed
By JOE SHARKEY
In recent years hotels have heeded the message that business travelers require Wi-Fi access, with many midlevel and budget-price hotels providing it free.

Beaming Into the Big Meeting
By PHYLLIS KORKKI

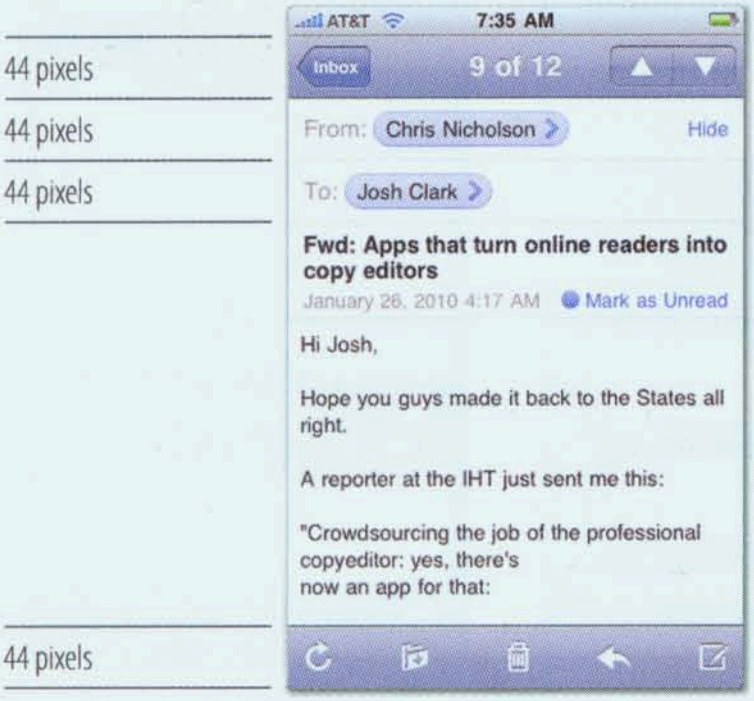
Full-width buttons, like the ones in the standard action sheet (left) or the WordCrasher and Epicurious apps, make for easy tap targets no matter what hand you're using.

iPhone's standard multiple-choice button views (page 167). Big chunky buttons not only give clear guidance to users, they also provide can't-miss tap targets no matter what hand you're using.

Size always matters in interface design. Big text, small text, giant buttons, or tiny ones—they all provide visual cues to what's important on the screen, gently guiding the eye to the next appropriate action. On touchscreen devices, though, fixing the right size for each tappable element is even more important, since every button and control has to be fitted to the finger. Make the buttons too small and you create an exercise in tap-and-miss frustration. The more important or frequent the action, the larger the associated target should be. Big buttons win.

The Magic Number Is 44

But just how big is big enough when it comes to iPhone tap targets? Well, what's the size of a fingertip? Apple pegs it precisely at 44 pixels and this measure appears reliably throughout the standard iPhone controls. In portrait orientation, 44 pixels is the height of buttons in the Calculator app, of the keys of the iPhone's virtual keyboard, of items in a standard list display, of the screen-topping navigation bar, and the list goes on. (With the iPhone's 163 ppi screen resolution, 44 pixels is about 7mm, or just a hair over ¼ inch.)



From: Chris Nicholson Hide

To: Josh Clark

Fwd: Apps that turn online readers into copy editors

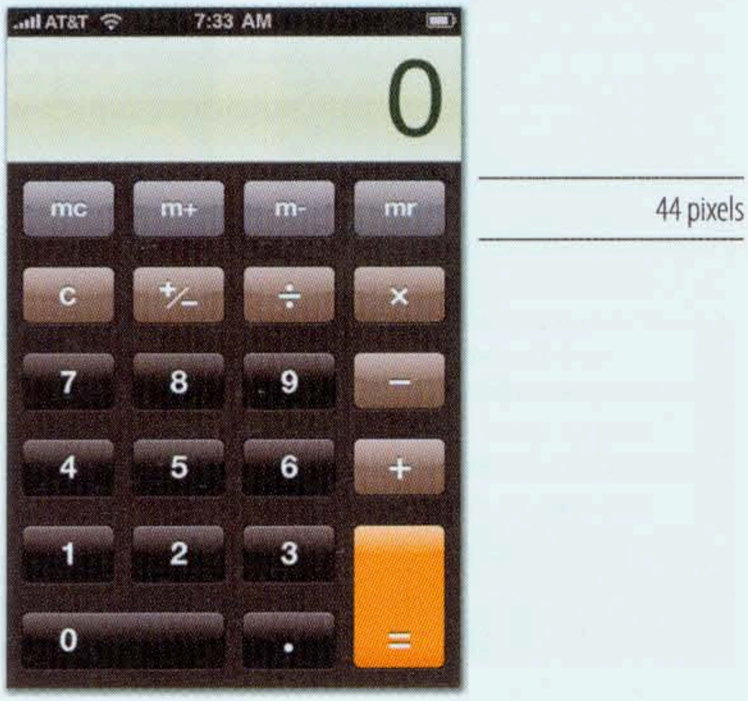
January 26, 2010 4:17 AM Mark as Unread

Hi Josh,

Hope you guys made it back to the States all right.

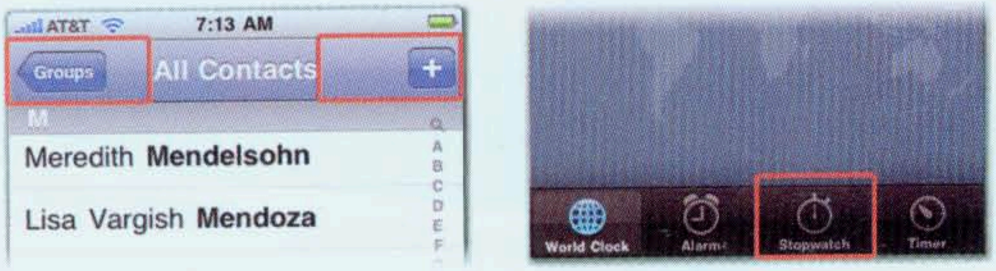
A reporter at the IHT just sent me this:

"Crowdsourcing the job of the professional copyeditor: yes, there's now an app for that:



The 44-pixel block is, in many ways, the basic unit of measurement for the iPhone interface, establishing the visual rhythm of many iPhone apps. That metric is significant as the recommended minimum size to make a tap target (like a button or list item) easily and reliably tappable. Put another way, it's the spread of a fingertip pressed to the screen. By building its standard controls in proportion to that measure, Apple created a device that's not only built *for* the hand, but *of* the hand—measured out in finger-sized units. (In fact, it's wise to craft your overall design to a 44-pixel rhythm, a topic you'll explore on page 75.)

Ideally, any button or other tap target should be at least 44 × 44 pixels. That doesn't necessarily mean that what you see—the outline of the button itself—has to spread over that entire area. Buttons inside the standard navigation bar, for example, are only 29 pixels high, but their *tap area* extends to the full 44-pixel height of the navigation bar. Even if you tap just above or below the button, it still catches the tap as long as you're still inside the navigation bar. Likewise, taps immediately to the left or right are treated as taps on the button itself. Even though the button is *visually* smaller, its tappable footprint honors the 44-pixel minimum, making the button effectively larger than its outline suggests.



Buttons in screen-topping navigation bars (left) have tap areas much larger than their visual footprint. The active tap areas are outlined in red here and span the entire height of the navigation bar, reaching horizontally to the title text. Similarly, icons in screen-bottom tab bars (right) have tap areas that extend several pixels above the tab bar's visible outline.

Apple helps you get this right by providing a whole stable of standard controls that stick to this standard height (you'll explore these built-in views and controls in the next two chapters). When you use Apple's prefab navigation bar, toolbar, or keyboard in your app, its controls automatically use these finger-friendly dimensions. It's only when you start working with custom button sizes and other homebrew controls that you'll need to start counting pixels to make sure you hit the 44-pixel threshold.

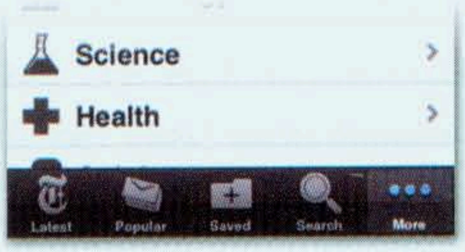
As in most things, compromise is sometimes necessary. Even the iPhone's standard controls fudge the 44-pixel rule from time to time. In the keyboard, for example, keys are 44 pixels tall but only 30 pixels wide—similarly, in landscape view, the buttons are 44 pixels wide but 38 pixels high. Apple doesn't have much choice here; it's crucial to include the full QWERTY keyboard in this view, but all the keys just won't fit as 44 × 44 buttons. Something's gotta give. When limited space puts the squeeze on tap targets, here's the rule I've found works best: as long as a tap target is at least 44 pixels high or wide, you can squeeze the other dimension as small as 30 pixels if you really must (these are the same dimensions as a keyboard key). That means, *the practical minimum size for any tap target is 44 × 30.*

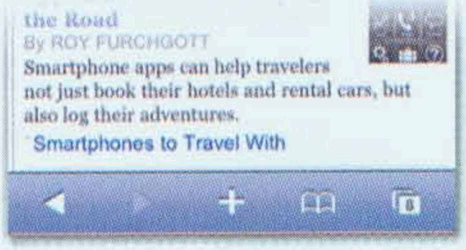
Photo: jonrawlinson.com

Don't Crowd Me

Your faithful author spent many years of his misspent youth with a svelte Casio calculator watch strapped to his wrist before finally retiring it in 1985. The problem wasn't just its tiny controls or its dampening effect on my prom king prospects. *The buttons were too close together.* Forget factorials or logarithms, the hardest thing about doing math on this thing was just hitting the right button. You'd aim for a five, but come up with a two or an eight, who knows—it was more wheel of fortune than calculator. Button size, in other words, isn't the only determining factor of tap accuracy; you have to consider spacing, too.

The iPhone's 320 × 480 pixels make for cozy quarters, and you'll inevitably be tempted to deal with that challenge by crowding the interface. "I'll just nudge these a little closer. I'll just add one more button to this toolbar." To quote a popular phrase of the calculator-watch heyday: "Just say no."

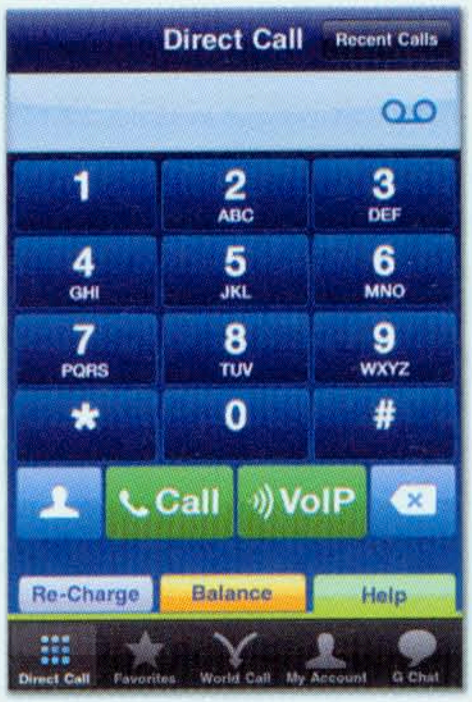


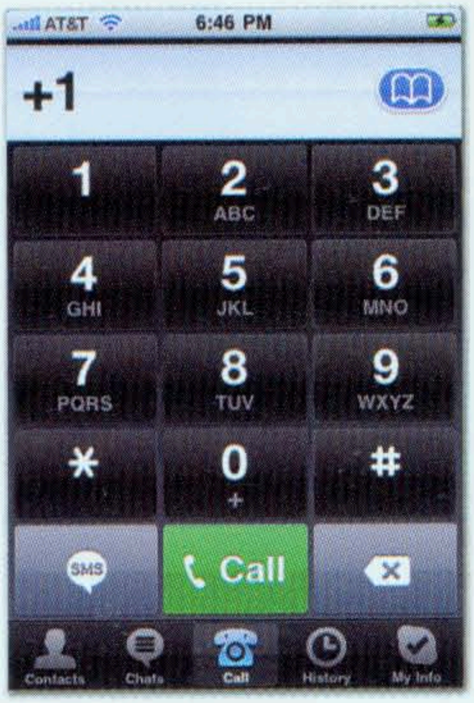

the Road
By ROY FURCHGOTT
Smartphone apps can help travelers not just book their hotels and rental cars, but also log their adventures.
Smartphones to Travel With

The iPhone's standard tab bar ensures comfortable spacing by limiting app designers to five tabs—no crowding allowed. If you add more than five, the tab bar displays only four of them, adding the More icon which takes you to a screen with additional options, as shown at left in the New York Times app. Apple's guidelines suggest limiting toolbars to just five icons, as shown at right in Safari.

Once again, Apple's standard controls help you do the right thing. For example, when your app uses a standard tab bar at screen bottom to switch between modes or views, the operating system automatically spaces them out for you and limits you to a maximum of five tabs. Toolbars don't impose the same automatic limit, but Apple nevertheless recommends that you limit toolbar icons to five, too. (You'll learn all about tab bars and toolbars on page 106 and page 143 respectively.) In both cases, the screen's 320-pixel width is technically large enough to cram up to seven 44-pixel icons, but they'd bump right up next to each other, introducing the calculator-watch problem for your audience.

It's especially important to give fingers some breathing room at the bottom of the screen. Usability testing reveals this to be a clumsy area prone to mistaps when targets are placed too close to an app's tab-bar navigation. The frustration is





Call Global App and Skype both include phone keypads with buttons that press right up against the tab bar. Call Global (left) makes things especially difficult by making the adjacent buttons narrow and tough to hit, with frequent missed taps. When you want to see your balance, a mistaken tap sweeps you away to the app's World Call screen. In Skype (right), the problem is less pronounced because the big buttons are tough to miss.

compounded by the fact that accidentally tapping a tab bar icon often takes you to an entirely new screen. If you must place targets near the tab bar or toolbar, make sure they're large enough to hit easily. (See how USA Today worked around this problem on page 94.)

Pointed Design

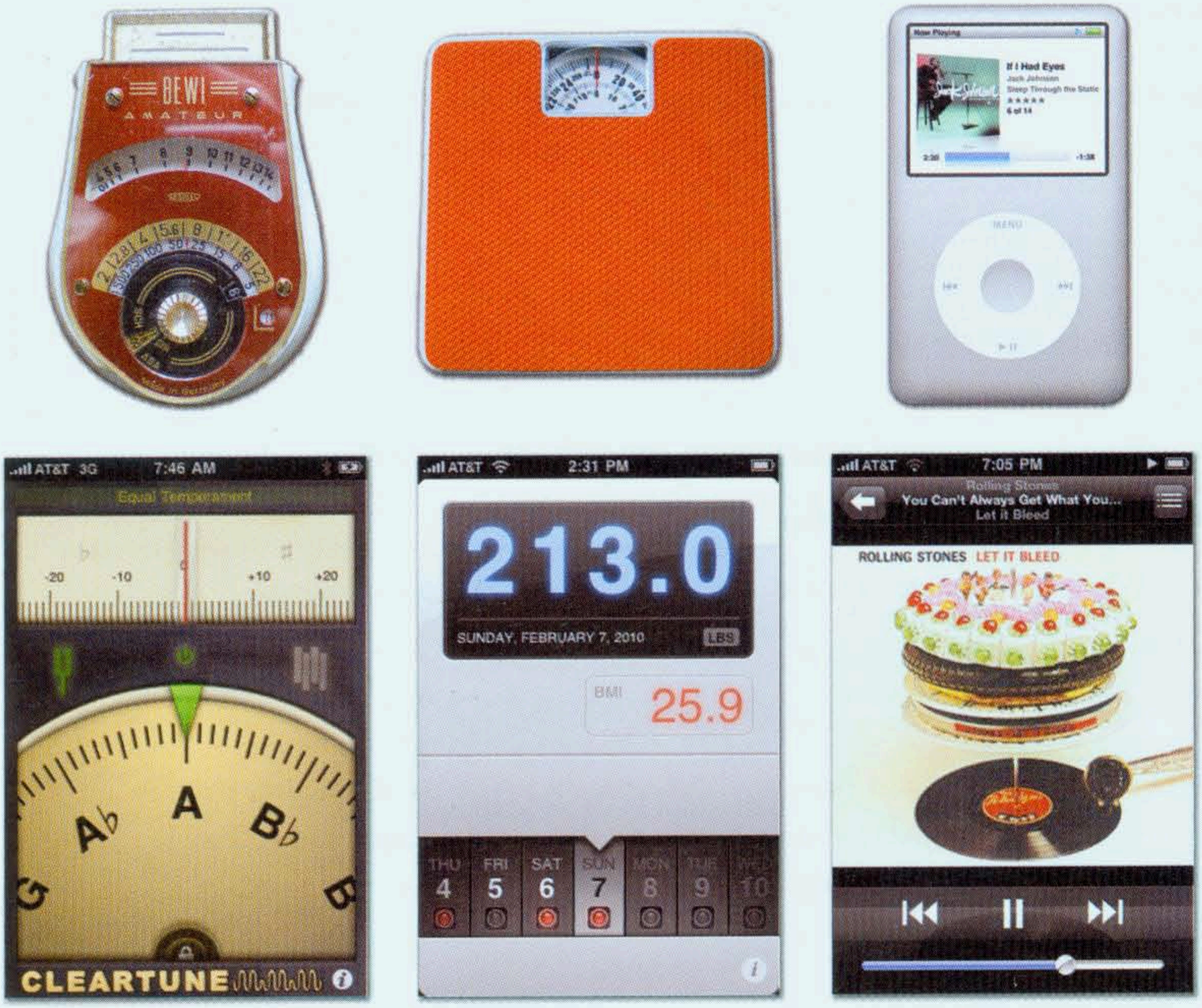
So far, this chapter has emphasized the physical more than the visual, pushing the ergonomic angle of app design. As always, though, form follows function, and all this talk of tap targets and thumb-friendly layouts influences more than just button sizes. When you plan the placement of your app's touch controls based on comfort and ease of use, other aspects of your design naturally fall into place. The best iPhone apps also share a common visual hierarchy broadly following these tap-friendly guidelines:

- Place important info at the top and sink controls to the bottom.
- Design to a 44-pixel rhythm.
- Where appropriate, create at-a-glance displays that avoid scrolling.
- Whittle onscreen elements to the bare minimum.
- Push advanced tools and controls to the background.

Like any set of guidelines, these aren't ironclad laws, just useful practices that can help you quickly block out your design in an appealing, efficient, and accessible layout. The next few sections give you the gist.

Take It From the Top

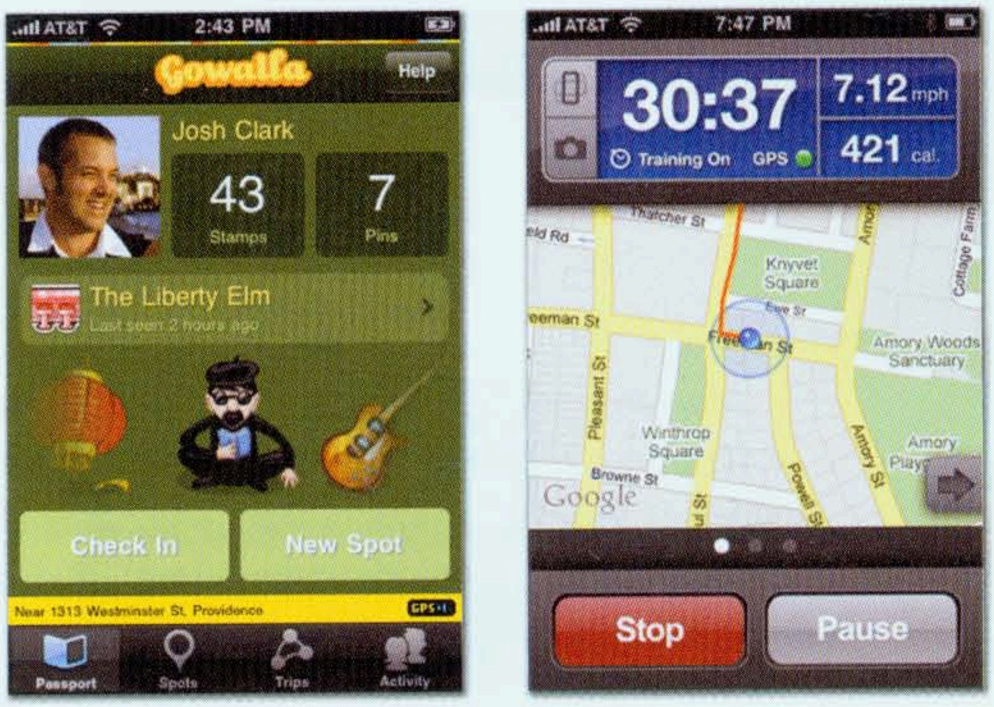
The most important or frequently used info should float to the top of the screen above the app's primary buttons and controls. This meets our expectations not only of graphic design—headlines at the top—but also the way we hoist and handle just about any physical device. The screen bottom is the most comfortable thumb zone for a handheld gadget, but that's also where the screen is most likely to be obscured by hovering hands. To keep info in clear view, position it above your app's controls. This is a familiar, common-sense layout that applies to most physical devices—iPods, calculators, cell phones, bathroom scales, you name it. Here again, though, it's the opposite of what we'd expect from Web and desktop software where toolbars and menus stake out the top of document windows with primary content below.



Tapworthy apps follow an interface convention familiar with all kinds of mechanical devices. Even when they don't directly mimic real-world gadgets, they organize controls at the bottom and content on top. From left, Cleartune, Weightbot, and iPod.

For primary app navigation, Apple helps you do the right thing by cautioning designers to place tab bars and toolbars at screen bottom (see page 106 and page 143). Likewise, when you summon a keyboard by tapping a standard text field, the built-in keyboard automatically slides in at screen bottom, nudging the typed results to the top of the screen. Follow Apple's lead here: everywhere it makes sense, let the primary controls sink to the bottom and main content float to the top.

This guideline is easy to follow when content and controls are separate, but not all iPhone apps make such a ready distinction. In many apps, the primary display info doubles as a view's main controls. That's especially true for list content;



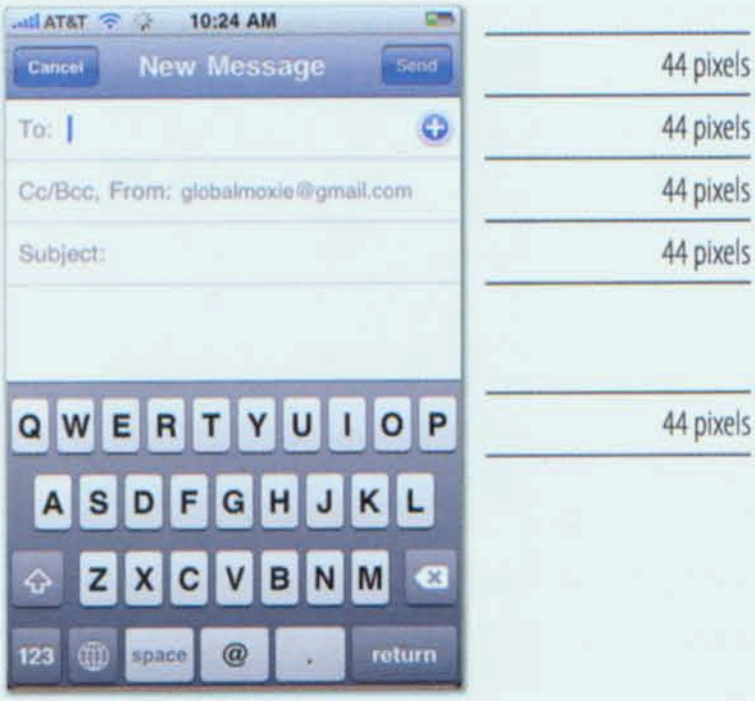
In Gowalla's Passport view (left) every display element doubles as a touch control, but the Check In and New Spot buttons anchor the screen as the most frequent tap targets. Similarly, while the stats at the top of RunKeeper Pro (right) are tappable, the main Stop and Pause buttons remain at bottom, leaving the stats display prominently visible at all times.

whether you're scrolling through contacts, to-dos, songs, or email, you tap list items to see their detail views. In those cases—as well as examples like Home-screen icons or photo album thumbnails—the entire display turns content into tap target from top to bottom. This double duty encourages you to directly tap anything you want to know about, a natural interaction and a good design. Even in these cases, though, it's a good practice to place *the most frequent* tap targets at the bottom, as discussed earlier.

Design to a 44-Pixel Rhythm

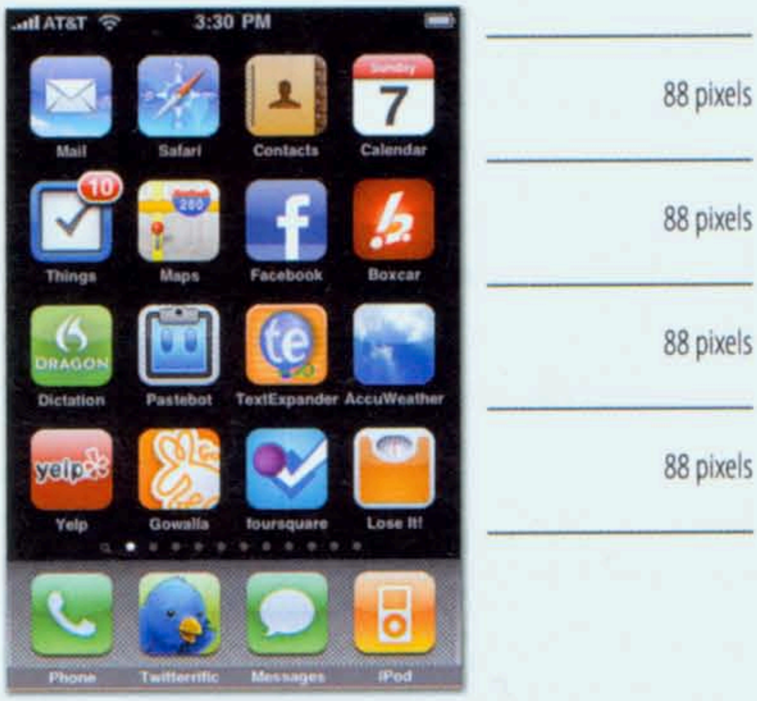
The iPhone looks good, sure, but it *feels* good, too. The industrial design gurus at Apple carefully designed the iPhone hardware to feel right in your hand, with weight and proportions that are as appealing as the device's glossy sheen. The same should go for your app design, with screens that are not only pretty but are constructed in proportion to the hand, with a sense of balance in each element's visual weight.

Part of that, as you've seen, is a matter of using finger-sized tap targets placed in easy reach. But the size of a fingertip can do more than just determine button sizes: as a unit of measure lifted from the hand itself, fingerspan provides a visual building block for giving your interface design a consistent rhythm in natural proportion to your paw and its pointer. By loosely blocking out your design in a



44-pixel grid, you ensure that interface elements are sized in harmony with one another as well as the fingers that work the device. As you saw earlier, several standard iPhone elements—navigation bars, toolbars, list items, keyboard keys—are sized to 44 pixels, establishing the vertical rhythm of many screens.

Go with that flow, at least loosely. You don't have to rigidly stick to aligning every single element to a 44-pixel grid. In fact, you *can't,* since the iPhone's 320 × 480 dimensions don't round neatly to 44, and some of the built-in controls like the tab bar at screen bottom veer into slightly different sizes. Instead, the point is that the



The iPhone Home screen builds on the 44-pixel visual rhythm by organizing icons into 88-pixel rows.

44-pixel measure provides a natural size to build with, one that rhymes with the size of several standard controls. Creating interface elements based on that proportion gives your designs visual balance and stability.

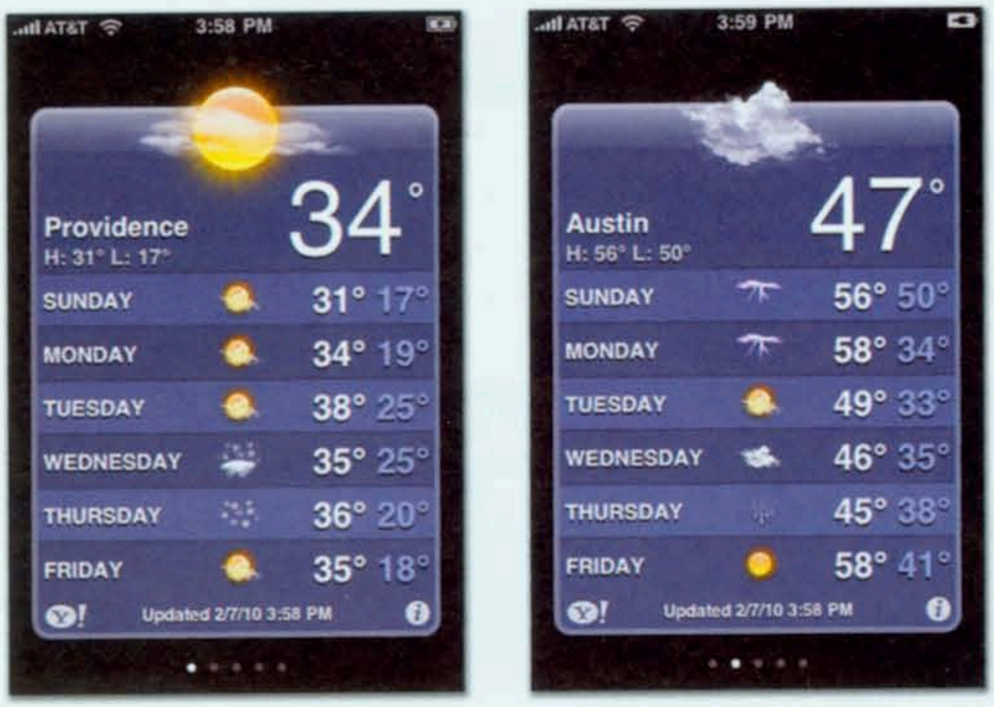
Be a Scroll Skeptic

Another way to reinforce an impression of sturdiness and weight is to build screens that stay put—no scrolling required. When it's possible to comfortably fit a screen's content into a single no-scroll view, you should go for it. The "out of sight, out of mind" effect is especially strong on the iPhone, where distracted mobile users speed through apps. When info isn't front and center, chances are good they won't see it at all. It's a matter of both brain and strain: scrolling requires extra thought as well as extra swipes, and one of your jobs as app designer is to reduce both.

This might seem subtle, but just asking users to *figure out* that they need to scroll requires them to fire up brain cells: "Wait, what's missing, and how do I get to it?" By contrast, taking in a screen's complete content at a single glance lets users focus exclusively on the content without making their gray matter do any background processing about what's offscreen. This might seem like coddling—seriously, it'll break their brains to scroll?—but the best apps bend over backward to reduce the overhead required to work the app itself. No-scroll screens require less brainwork while also reinforcing the illusion of your app as a physical, not virtual, device. A fixed screen gives a sense of solidity.

The point here is not to avoid scrolling at all costs (this section is titled "Be a Scroll *Skeptic*"—not a zealot, reactionary, or dogmatist). Eliminating the scroll is just one tactic for designing a display that's easy to absorb, not a goal in itself, but when it suits your app's content, it's a design tactic that encourages healthy restraint.

There's a whole category of apps devoted to the single-screen display. Apple dubs them *utility apps,* narrowly focused tools that provide quick summaries or perform a simple task, almost always in simple no-scroll screens. The built-in Weather app is the quintessential utility app. It provides streamlined forecasts for up to 20 cities, each tidily presented in a get-it-quick, no-scroll view—you flip



AT&T 3:58 PM

Providence 34°

H: 31° L: 17°

SUNDAY	31° 17°
MONDAY	34° 19°
TUESDAY	38° 25°
WEDNESDAY	35° 25°
THURSDAY	36° 20°
FRIDAY	35° 18°

Updated 2/7/10 3:58 PM

AT&T 3:59 PM

Austin 47°

H: 56° L: 50°

SUNDAY	56° 50°
MONDAY	58° 34°
TUESDAY	49° 33°
WEDNESDAY	46° 35°
THURSDAY	45° 38°
FRIDAY	58° 41°

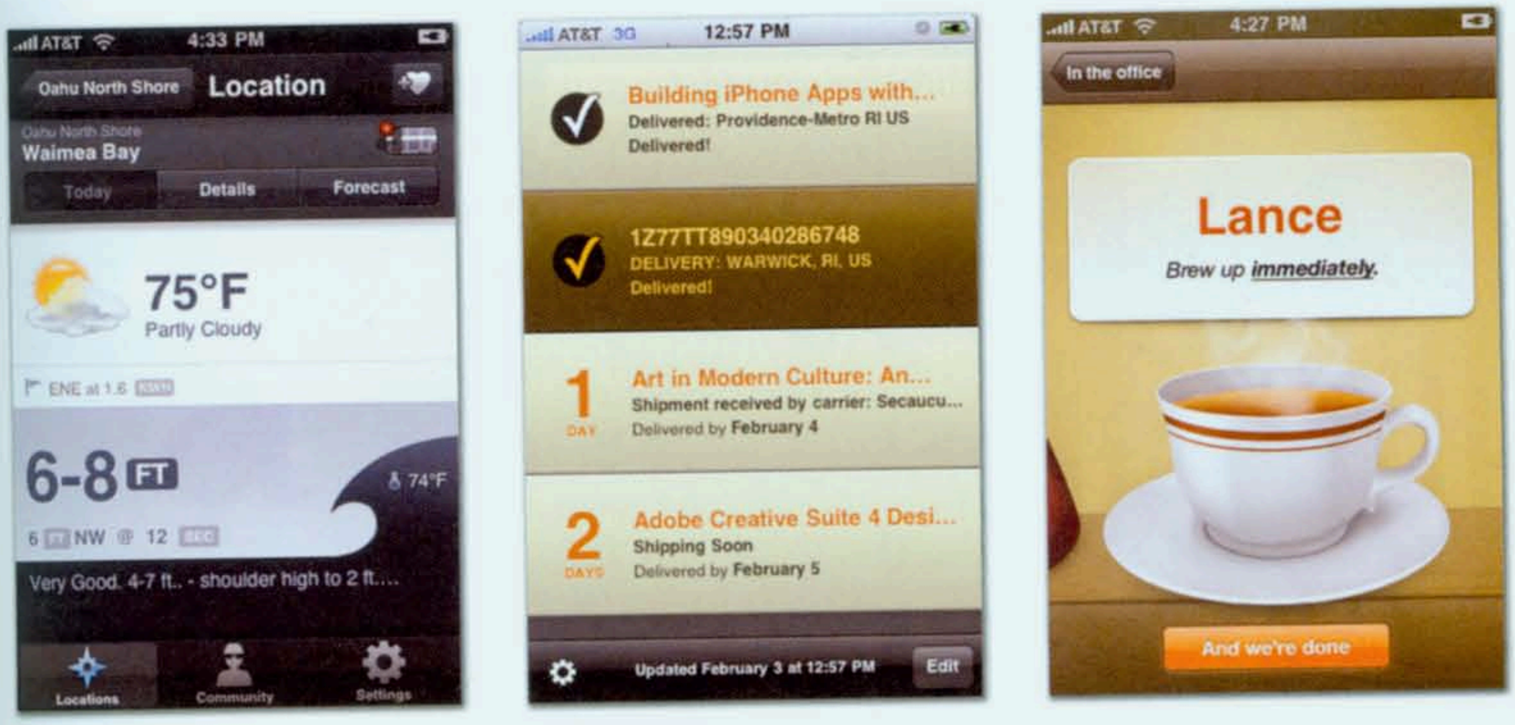
Updated 2/7/10 3:58 PM

The Weather app offers a compact weather forecast, summarizing a week's weather in a single screen. The border around the info creates a solid container to signal that there's no additional scrolling content below. The page indicator dots at screen bottom indicate that more screens await (see page 103 for more info about page indicators).

through them like cards. The app's presentation lets you absorb the week's forecast for each city with just a glimpse. The carefully contained layout makes it instantly clear that there's no additional info below the fold, no "Is there more?" thinking required.

The Weather app pulls off its single-screen layout by stripping content down to the bare minimum. A few icons and high/low temperatures hardly tell the whole story of your local weather, a topic that occupies entire websites and 24-hour weather channels. Instead of indulging in complex detail, the app focuses on giving the quick gist in a simple and efficient display. This just-the-basics approach is a good strategy for utility apps, the simplest class of iPhone software. Like Weather, the best of these apps rely on graphically rich displays to telegraph simple info quickly, with big can't-miss text and images that sink in with just a quick peek. Contrary to what you might expect, the success of compact interfaces often relies on big text and chunky images cushioned with generous surrounding space. Apps pass *the glance test* when you can hold them at arm's length and still soak up their info effortlessly.

The glance test reinforces an essential principle of tapworthy app design: *clarity trumps density.* A crowded screen creates more work and confusion for your audience. This doesn't mean you're obliged to chuck your app's detailed info in order to have a beautiful and intuitive interface. It's not an either-or bargain. Complexity itself isn't bad; the trick is making complexity seem uncomplicated.

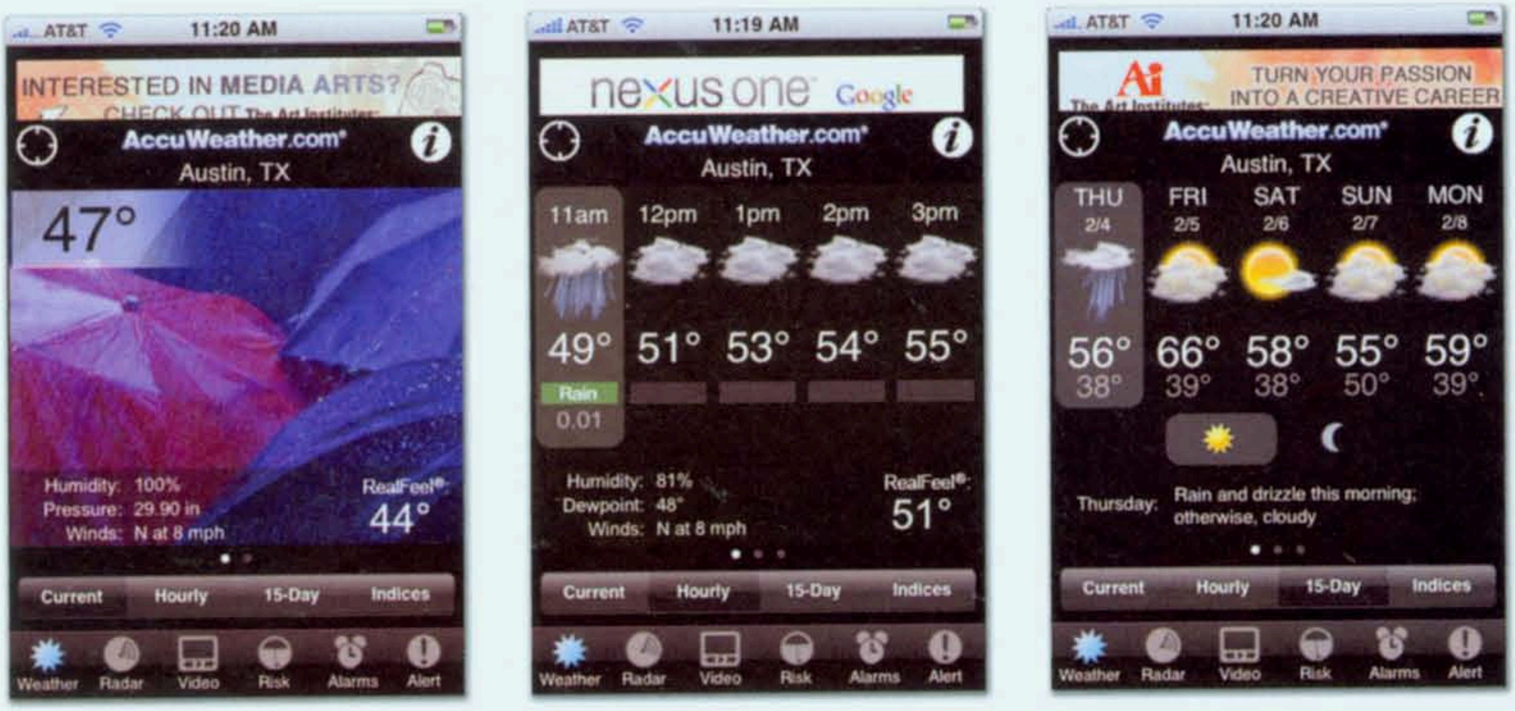


From left, apps like Surf Report, Delivery Status Touch, and Tea Round pass the glance test with high-impact text and graphics that are easy to read even from several feet away. Surf Report shows water conditions, Delivery Status Touch shows package whereabouts, and Tea Round shows whose turn it is to brew the tea.

Even within the iPhone's tiny screen, it's possible for apps to reveal complex information in a simple display—and yes, without scrolling.

The AccuWeather.com app, for example, is a sophisticated alternative to the Weather app that steps beyond the utility category by providing several screens of detailed weather info. Each of these screens (see page 80) is self-contained in a no-scroll display, providing dense information layered in multiple but uncomplicated views. Want to find out what the humidity will be in a few hours? Tap the desired hour in the app's 15-hour forecast and the details pop up immediately. The main screen shows the big-picture overview, and the app tucks additional details behind tabs or icons instead of piling it all into the same screen at once. It's at once intuitive and information-rich, no scrolling or crowded layout required.

Healthy scroll skepticism means recognizing that you don't have to reveal all your information in one shot. It's a bit like the art of conversation: don't be that guy who drones on and on without pausing to check if he's saying something that actually interests his suffering listeners. Especially when there's lots to be said or explored, the best conversations are interactive, with listeners allowed to ask questions instead of passively receiving info. In an app, a tap is effectively a question in the conversation. The best apps provide top-level, need-to-know information


AT&T 11:20 AM
INTERESTED IN MEDIA ARTS? CHECK OUT The Art Institutes
AccuWeather.com
Austin, TX
47°
Humidity: 100%
Pressure: 29.90 in
Winds: N at 8 mph
RealFeel®
44°
Current Hourly 15-Day Indices
Weather Radar Video Risk Alarms Alert
AT&T 11:19 AM
nexus one Google
AccuWeather.com
Austin, TX
11am 12pm 1pm 2pm 3pm
49° 51° 53° 54° 55°
Rain 0.01
Humidity: 81%
Dewpoint: 48°
Winds: N at 8 mph
RealFeel®:
51°
Current Hourly 15-Day Indices
Weather Radar Video Risk Alarms Alert
AT&T 11:20 AM
Ai The Art Institutes TURN YOUR PASSION INTO A CREATIVE CAREER
AccuWeather.com
Austin, TX
THU 2/4 FRI 2/5 SAT 2/6 SUN 2/7 MON 2/8
56° 38° 66° 39° 58° 38° 55° 50° 59° 39°
Thursday: Rain and drizzle this morning; otherwise, cloudy
Current Hourly 15-Day Indices
Weather Radar Video Risk Alarms Alert

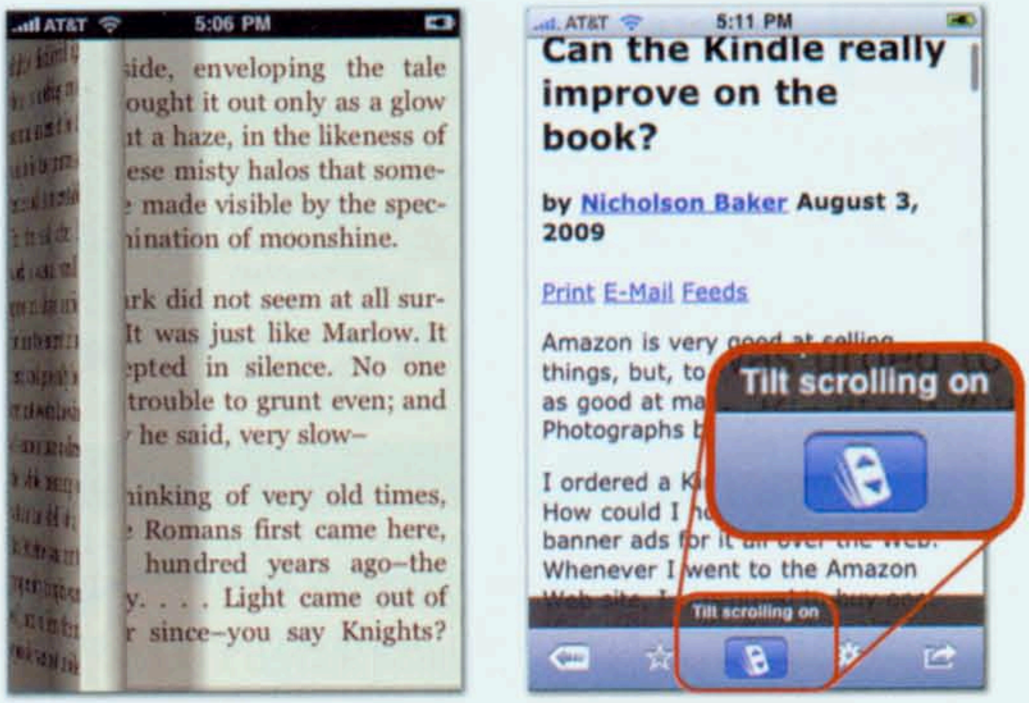
AccuWeather.com's simple tab control lets you flip between current, hourly, and daily weather forecasts. On the hourly and daily views, you can tap the day or hour to get more info—an intuitive way to make data available without crowding too much into the display at once.

at a glance and, from there, let users tap something to "ask" for more information about it. Chances are, you don't need to know what the local wind speed will be at 3 p.m. every time you launch AccuWeather.com, but when you need it, the answer is just a tap away. In the meantime, the main view still gives you the basic at-a-glance info that you'd get from the built-in Weather app.

Are all these extra taps really better than scrolling? Just on the basis of physical comfort, a tap is an easier gesture than the swiping scroll. A more important consideration than tap quantity is tap *quality*. As long as there's an appropriate reward waiting on the other side of every tap, extra taps are okay—and usually less of an imposition than a long scrolling screen. If you focus the main screen on the most important tools and info, you can safely tuck secondary content into another view. If done right, tapping through fixed screens requires you to do less visual scanning than it takes to locate content on a scrolling page. Content and controls on a fixed screen remain reliably in the same place visit after visit, so it's easy to duck in and out of an app to get the info you want without pausing to get your bearings.

Even long-form content allows creative alternatives to scrolling. Most apps for reading books, for example, use a page-turning metaphor instead of scrolling to advance the text. This lets you tap the screen just once to flip the virtual page to a fresh screen of prose, sparing you constant swiping while also reinforcing the illusion of handling a physical object—convenient and familiar. Instapaper Pro, an app for saving and reading lengthy online articles, likewise offers an option to page through screens a tap at a time. The app also offers scrolling but with a clever ergonomic gimmick to spare you swipe-swipe-swipe tedium: you can scroll just by tilting the phone back and forth to advance the text.

Stanza and Instapaper Pro offer thumb-sparing alternatives to swiping through long text. Like most ebook readers, Stanza (left) lets you tap the screen to flip to a new page of text. Instapaper (right) offers a tilt-scrolling feature to advance text by physically tipping the device.



AT&T 5:06 PM

ide, enveloping the tale
ought it out only as a glow
it a haze, in the likeness of
ese misty halos that some-
made visible by the spec-
nination of moonshine.

rk did not seem at all sur-
It was just like Marlow. It
pted in silence. No one
trouble to grunt even; and
he said, very slow–

inking of very old times,
Romans first came here,
hundred years ago–the
y. . . . Light came out of
r since–you say Knights?

AT&T 5:11 PM

Can the Kindle really improve on the book?

by Nicholson Baker August 3, 2009

Print E-Mail Feeds

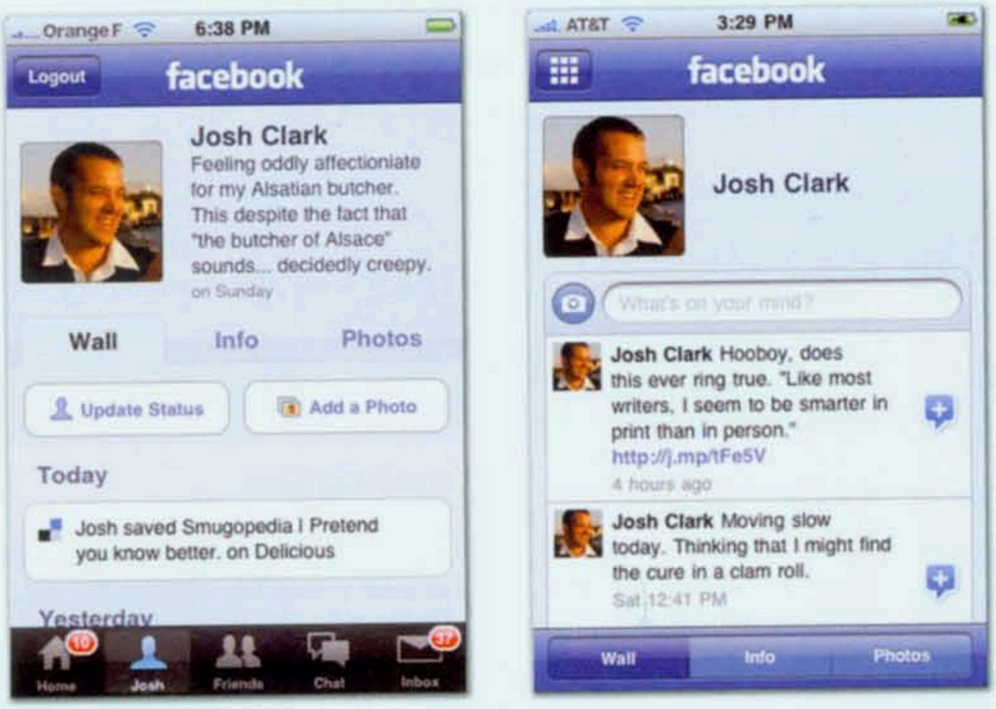
Amazon is very good at selling things, but, to
as good at ma
Photographs b

I ordered a K
How could I
banner ads for it all over the Web.
Whenever I went to the Amazon
Web site, I

But let's not be strident anti-scrollers. While it's good to avoid scrolling where appropriate, it's not like it's inherently evil. It's part of the fun of the iPhone's physics, and it's obviously essential to some apps. To-do lists, news feeds, articles, and emails inevitably run long, and scrolling is (usually) the best way to handle those kinds of long-form content. List-based interfaces that try to wriggle out of a scrolling screen often just feel awkward (see the interfaces the USA Today designers experimented with on page 91).

When your app does require scrolling, just be sure to keep the primary app controls anchored in one place. In early versions of the Facebook app, for example, the tab control for flipping through a friend's profile content was itself part of the content area, which meant it scrolled out of view with the rest of the page. When you finished reading through a pal's status updates, for example, you had to scroll back up to the top of the screen to switch over to your friend's photos—assuming that you could even remember where you last saw those tab controls. Facebook 3.0

remedied this where'd-it-go problem by anchoring the wayward tabs to the bottom of the screen, saving both time and head scratching. The lesson: scrolling or no, a view's primary controls should never skitter offscreen. Anchored elements create a sense of stability and consistency.



Josh Clark
Feeling oddly affectioniate for my Alsatian butcher. This despite the fact that "the butcher of Alsace" sounds... decidedly creepy.
on Sunday

Today

Josh saved Smugopedia | Pretend you know better. on Delicious

Yesterday

Josh Clark

What's on your mind?

Josh Clark Hooboy, does this ever ring true. "Like most writers, I seem to be smarter in print than in person." http://j.mp/tFe5V
4 hours ago

Josh Clark Moving slow today. Thinking that I might find the cure in a clam roll.
Sat 12:41 PM

In Facebook 2.0 (left), the control for switching between wall posts, profile info, and photos was part of the scrolling content area and could disappear offscreen. In Facebook 3.0 (right), the tab control is always anchored to screen bottom.

Ergonomic and visual simplicity should be important goals for your app design. Whether or not you ultimately decide to include scrolling screens, approaching the scroll with skepticism asks you to be more discerning about what you include. Committing to the iPhone's 320 x 480 footprint puts limitations on your design as firm as if you were designing a real-world device. When there's only so much room for your tools, controls, and content, you have to ask yourself a useful question: "Do I really need all this stuff?"

Edit, ~~Edit, Edit~~

The Guinness world record for "most multifunctional" pocket knife belongs to Wenger, the company behind the storied Swiss Army knife. The company says the knife's 87 gadgets (including a laser pointer, cigar cutter, and golf reamer) can be used for no fewer than 141 functions. The $1400 gadget is a nifty feat of Swiss ingenuity and—who knew!—Swiss humor, too. Alas, weighed down by its three pounds of gizmos, this "most multifunctional" knife has no practical function at all, a pocket knife that doesn't fit in your pocket. This slice-n-dice Goliath

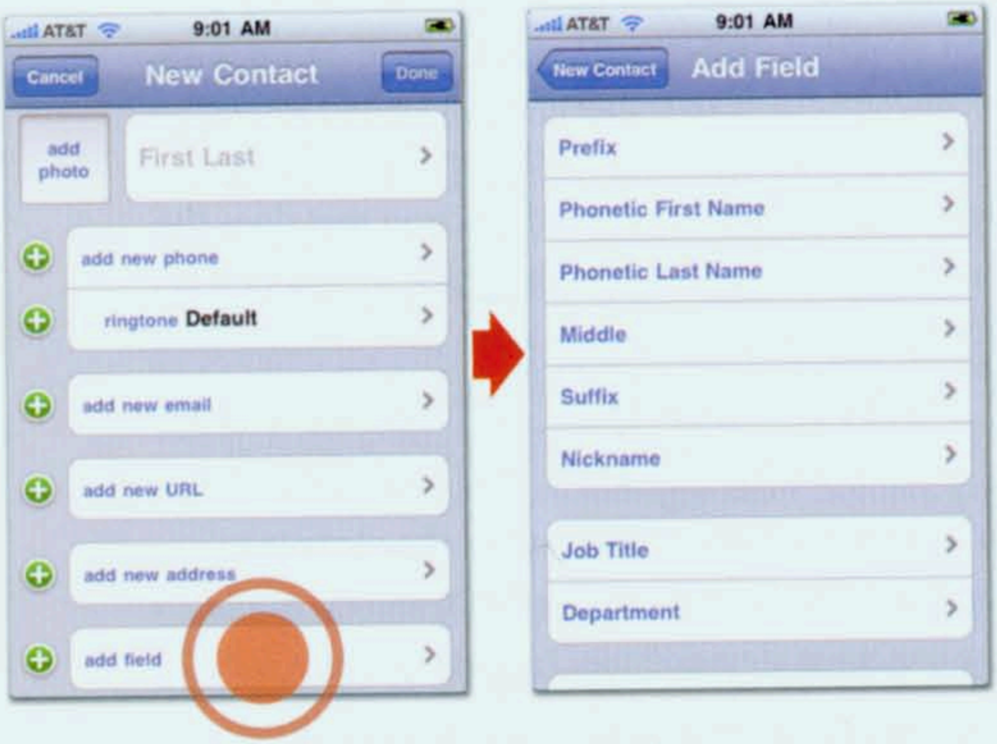
was, of course, never really designed to be used. It was a novelty created for the company's 100th anniversary, a whimsical project to bring together every gadget the company ever included in its knives. At a certain point, as Wenger's craftsmen added more and more tools, the knife suddenly stopped being a knife, and it just became a doorstop.

In geekspeak, you might call this a crowded interface. While the knife is obviously (and intentionally) ridiculous, it's a winking reminder that somewhere in the reptile part of our brains, a misguided instinct tells us that more is always better. More features, more preferences, more flexibility—the gizmo that does the *most* is always the best. In the end, of course, the best gizmo is the thing that lets us do what we need to do with the greatest ease. In mobile devices—whether it's a Swiss Army knife or an iPhonε –that almost always means removing features and *chrome*, the buttons, icons, and other controls that often crowd an interface.

The last chapter encouraged you to trim your app's functions in the planning stage, whittling the app to a sharp focus on the most important features. Apply the same tapworthy filter to every onscreen element, too. Include controls only if they're likely to be used most of the time by most of your audience. Be ruthless when you consider every button and icon: Does this element invite attention? Is it clear what it does? Does it deliver something meaningful? Every tool should be tightly related to the primary task at hand, and auxiliary tools and content should be dispatched to a secondary screen or sliced out altogether. Text labels should be terse and words shouldn't be repeated.

It might seem harmless to add just one more icon, but every onscreen element comes with a cognitive cost for your users. It takes longer to scan the screen, longer to absorb the possible options, longer to figure out what you're supposed to do. Don't make 'em think. When every pixel is precious, your app doesn't have room for question marks that will snag users as they hustle through. It's like the pocket knife: it's easy to choose a blade to use when there's only one of them, but it takes forever when there are 87.

Just as you saw in the AccuWeather.com app, you don't have to show every last scrap of information all at once. Consider whether second-tier content and controls might be dumped entirely and, if not, nudge them over to a separate view in your app. The built-in Contacts app does this, for example, when you add or edit a contact. The main screen is a no-scroll view of the most common contact info; the app doesn't pester you about less frequent fields and instead provides an "add field" button at the bottom of the screen. From there, you can choose from among 12 additional fields to round out your contact info when the standard fields don't do the trick.



When you add a new contact, the Contacts app displays only a few primary fields (left). Tap the "add fields" button at screen bottom to go to a secondary screen (right), which offers 12 less common fields.

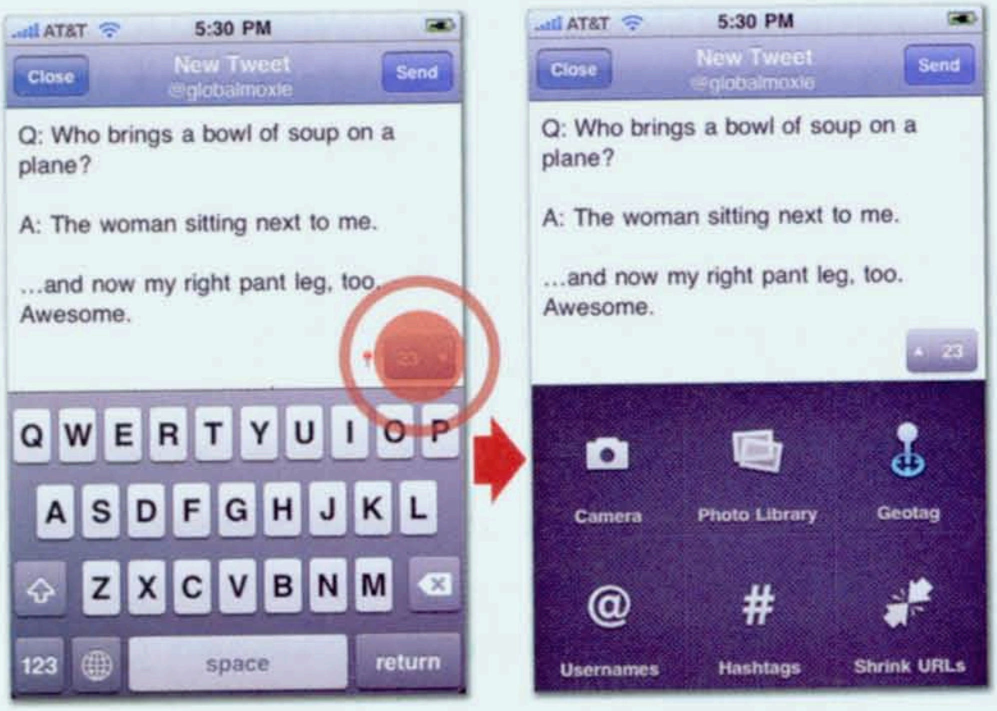
Apple is particularly clever about doing more with less. The company's software has a well-deserved reputation for both elegance and ease of use. It looks great, works intuitively, and drips with good taste. Like all good editors, Apple's designers do this in part by removing distractions and plucking out features that aren't relevant to mainstream users in order to focus more screen real estate and polish on the features that matter to the most people. What's not included in Apple's built-in iPhone apps is as important as what is.

Secret Panels and Hidden Doors

Here's the thing: leaving stuff out inevitably drives some people nuts. Power users will, in particular, pine for all the tools and flexibility that mainstream users don't miss. For many, the basic Weather and Stocks apps just don't cut it. If your app addresses an audience with more advanced needs, then you'll almost certainly have to add more features, tools, and content than your app's mainstream counterpart. That doesn't give you a pass to create mind-numbingly crowded interfaces. The same rules apply, and the pocket knife still has to fit in the pocket. If your app really does have to include those extra features and controls, you've still got to prevent the extra interface chrome from elbowing aside the main content.

This is a challenge that the many iPhone Twitter apps wrestle with. On the surface, Twitter couldn't be simpler: post terse updates and read those posted by others. Yet an enormously complex infrastructure of tools, social customs, and third-party services has emerged around this basic concept. Your tweets can link to photos or videos, point to your physical location, mention other Twitter users, "retweet" someone else's post, embed #hashtag metadata, and more. The dilemma for Twitter apps is how to make a dead-simple interface for firing off a quick tweet while also surfacing the many power features that make Twitter megausers tingle. The brute-force approach is to pile a bunch of icons and controls into the tweet-posting screen. Trouble is, when you add a keyboard to the mix, all that extra chrome quickly squeezes out the most important thing: the text you're typing. It's a sign of trouble when controls leave little room for the content.

The Twitter app (formerly known as Tweetie) solves this problem by hiding all those add-on controls behind the keyboard. The button that displays the


Q: Who brings a bowl of soup on a plane?

A: The woman sitting next to me.

...and now my right pant leg, too. Awesome.

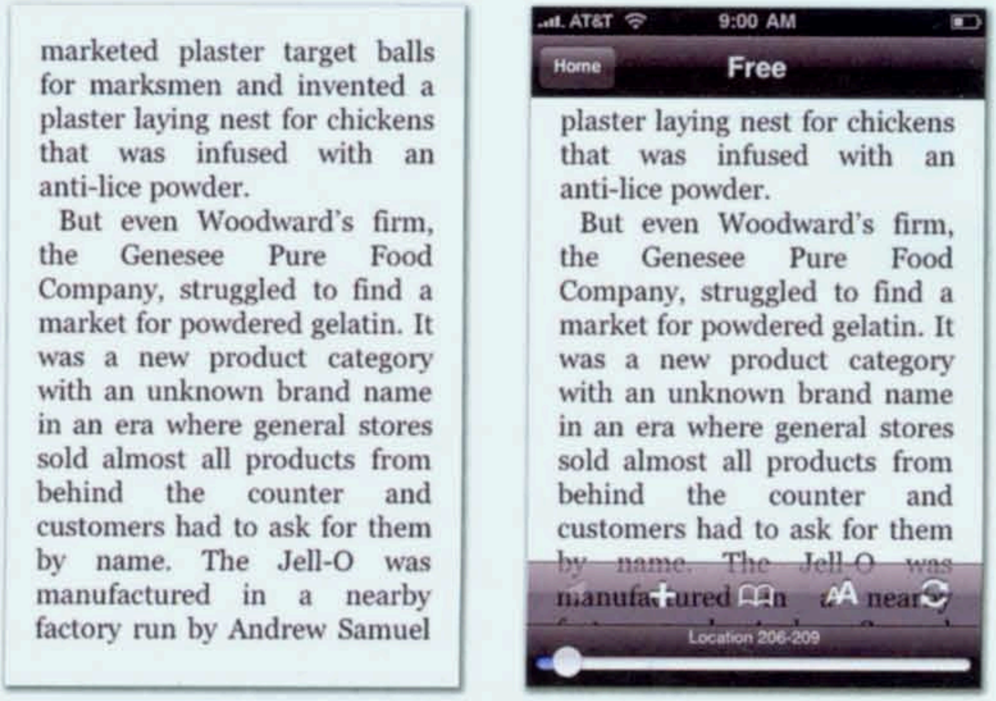
In the Twitter app, tapping the character-count button (left) slides away the keyboard to reveal a control panel (right), a compact means to put advanced features within reach without crowding the screen.

remaining character count for your tweet doubles as a toggle switch that slides the keyboard out of the way to reveal all the options you can apply to your tweet—a secret control panel for power users. When you're done, the keyboard slides back into view, and you're back to typing, no extra chrome in sight.

Other apps use similar tricks to move interface elements out of the way. Ebook apps Stanza, Kindle, and Eucalyptus all hide their main controls completely to allow book text to fill the screen when you're reading. Ditto for the Photos app when flipping through full-screen pictures. Hiding primary controls like this admittedly brings a risk of disorienting people, leaving them uncertain about how to escape the current view. In all of these examples, though, tapping the screen summons the controls again, a tactic that even panicked users will try soon enough. It's a legit approach, but one that should be used sparingly and only by apps that get a big benefit from an unsullied full-screen view.

Advertisements also count as interface chrome, and unlike primary controls, most users wouldn't mind seeing them slide away for good. Ads present a business and usability dilemma: they take up valuable real estate and present content to which many users are actively hostile, yet many free apps rely on them to fund the whole operation. The sliding panel approach provides a useful compromise. In the USA Today app, for example, a 50-pixel ad banner slides up from the bottom of the screen when you first arrive at an article screen. It holds its place for a

Kindle banishes its controls to turn over the full screen (left) to your book. Tapping the screen brings the controls back (right).



marketed plaster target balls for marksmen and invented a plaster laying nest for chickens that was infused with an anti-lice powder.

But even Woodward's firm, the Genesee Pure Food Company, struggled to find a market for powdered gelatin. It was a new product category with an unknown brand name in an era where general stores sold almost all products from behind the counter and customers had to ask for them by name. The Jell-O was manufactured in a nearby factory run by Andrew Samuel

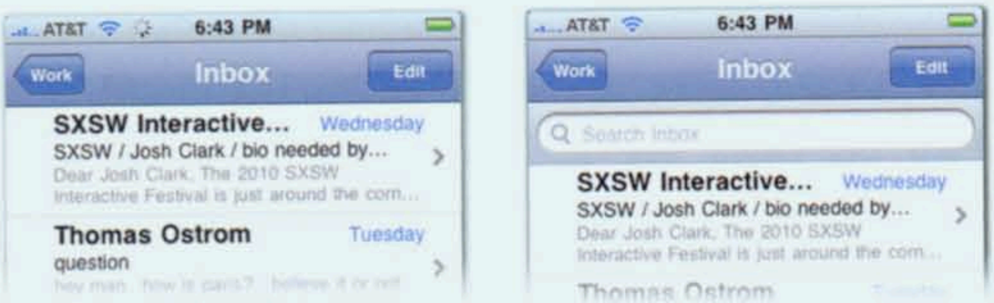
plaster laying nest for chickens that was infused with an anti-lice powder.

But even Woodward's firm, the Genesee Pure Food Company, struggled to find a market for powdered gelatin. It was a new product category with an unknown brand name in an era where general stores sold almost all products from behind the counter and customers had to ask for them by name. The Jell-O was manufactured in a nearby

few seconds before sliding out of view—a reasonable compromise for both advertiser and reader.

The point here is that for all of this chapter's talk about designing your app to feel like a real-world device, it's okay to bend the rules to take advantage of your app's virtual environment. Virtual keyboards and sliding panels let your app do things no physical gadget could do by swapping out the interface for a new set of tools or content. These "cheats" let you layer secondary tools and content into your app without crowding the main screen. Turns out it's not all that different from the Swiss Army knife after all, a handy gadget that can constantly become something new. While pesky considerations like size and weight prevent the Swiss Army knife from offering too many tools, the iPhone's virtual interface gives you a theoretically unlimited interface surface area. Just keep flipping through secret panels as you need them, right?

Well, yes and no. The trouble with hidden content is, you guessed it—it's hidden. You'll continue to bump against the "out of sight, out of mind" problem. The more secret panels you spirit away, the more overhead you ask people to take on to remember how to find them. You have to take care to provide visual cues to help them find their way. If you decide to use a hidden door, that means you have to put the latch in clear sight to make it easy to open. Ideally, this trigger should be well-labeled or at least in such close proximity to the content you're working



SXSW Interactive... Wednesday
SXSW / Josh Clark / bio needed by...
Dear Josh Clark, The 2010 SXSW Interactive Festival is just around the corn...

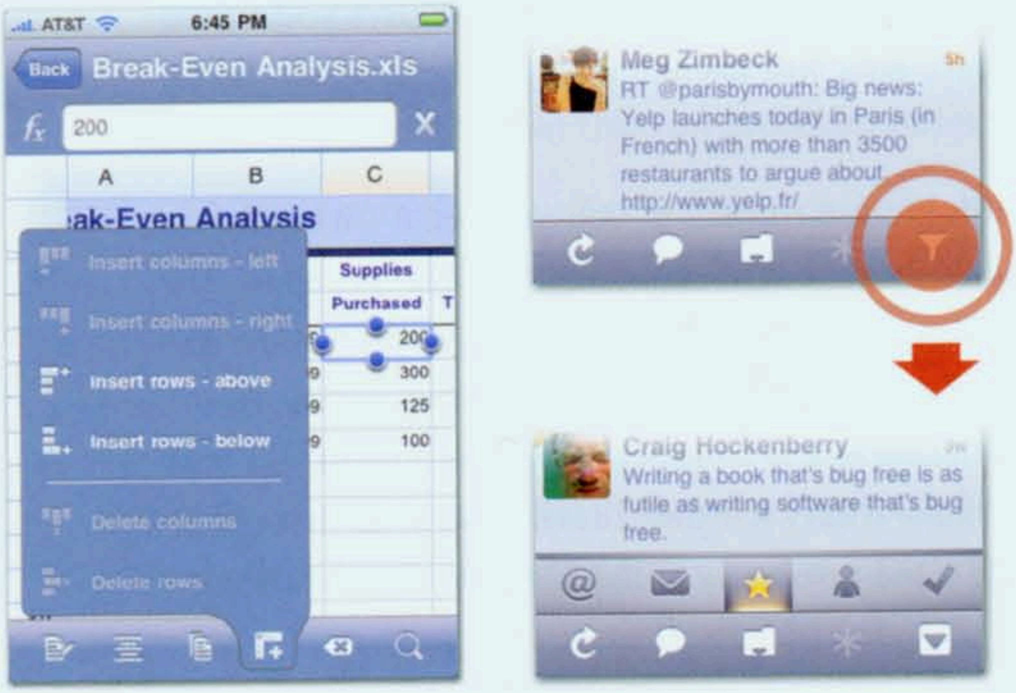
Thomas Ostrom Tuesday
question
hey man , how is paris? , believe it or not

The search bar is hidden offscreen in the Mail app until you tug the screen down to reveal it.

with that it suggests what you'll find on the other side. The latch for your hidden door should look tapworthy and invite action.

A more discoverable approach is to hide interface chrome in plain sight. Most apps that have a search feature position their search bars at the top of the screen, premium real estate. The built-in Mail app, among many others, reclaims this valuable space by scrolling the view so that when you first land on a screen, the search bar is tucked away just out of view. If you want to search, you tug the screen down or tap the screen-topping status bar to zip the search bar into view.

Another strategy for discoverable secondary controls is to offer toolbars that emerge temporarily from the app's main navigation. Quickoffice's iPad-style popover menu provides tools for formatting text, for example, and Twitterrific's filter bar lets you choose the type of tweets you'd like to view in your Twitter timeline. As always, the goal is to keep your interface visually uncomplicated, no matter how complex your app may be. Limit interface chrome, but hide only as much as you have to. Give all of your features and controls a hard look before you include them in your design to make sure they're really tapworthy. If they pass muster, be



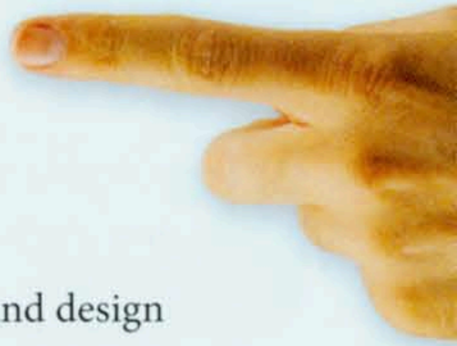
Meg Zimbeck
RT @parisbymouth: Big news: Yelp launches today in Paris (in French) with more than 3500 restaurants to argue about http://www.yelp.fr/

Craig Hockenberry
Writing a book that's bug free is as futile as writing software that's bug free.

Quickoffice (left) offers secondary tools in popover menus when you tap an icon in the main toolbar. Twitterrific (right) similarly reveals icons in a second toolbar when you tap the funnel-shaped Filter icon at far right.

sure to place each element carefully in intuitive, easy-to-find locations, hidden or not.

Touchpoints



✓ Ergonomics matter: consider how your app feels in the hand.

✓ Put primary controls in the thumb’s “hot zone.”

✓ Forty-four is the magic number. Make tap areas at least 44 pixels, and design to a 44-pixel rhythm.

✓ Be generous with space and don’t crowd your design.

✓ Feature primary content at the top, controls at the bottom.

✓ Keep the main controls within easy reach, and avoid scrolling where practical.

✓ Reduce interface chrome by dispatching power tools to secondary views with secret panels and hidden doors.