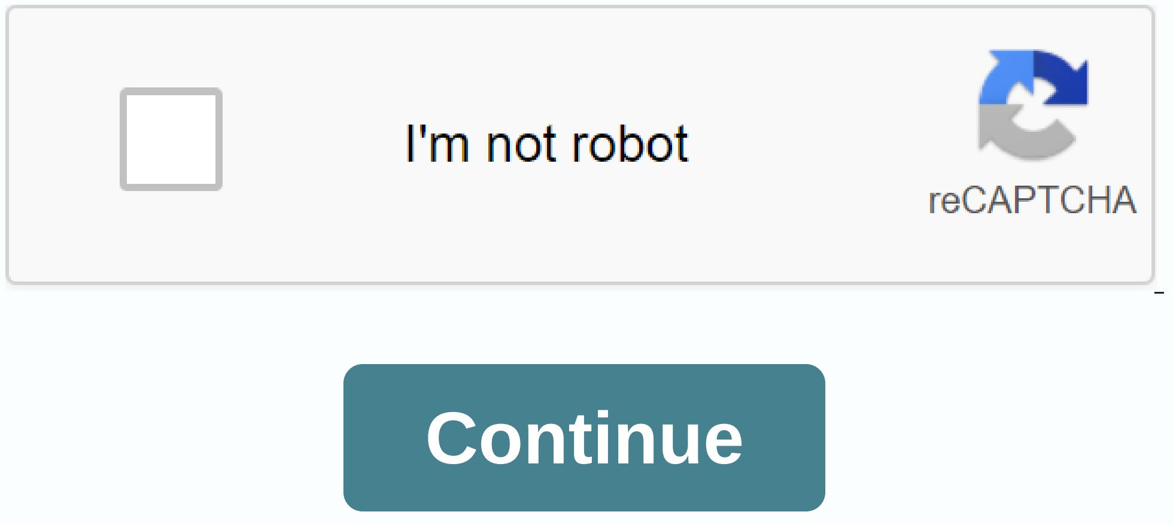
Apple visual design guidelines



□ I'm not robot

SF Symbols 2 provides a set of more than 2,400 configurable symbols that can be seamlessly integrated with San Francisco, macOS, iOS, watchOS, and tvOS system fonts. For more information about using SF symbols, see Get more information and UI resources to design great apps that integrate seamlessly with the SF symbol app Apple platform. Branding is not just about displaying a logo throughout your app. Excellent app expresses your own brand identity through smartphone, color and image decisions. For relevant design guidance, see Colors, Typography, Icons and Images. It combines sophisticated and understated branding. People don't see ads using the CarPlay app. If you're using an app, it's already sold in the app. For the best experience, subtly incorporated your brand through your app's design. For example, consider using the app icon color throughout the interface. Focus on consistency and functionality rather than branding. Make sure your app feels like a CarPlay app. It must be intuitive, easy to operate, easy to use, and provide useful functions while driving. Follow Apple's trademark guidelines. Apple trademarks don't appear in app names or images. See Apple's trademark list and apple trademark use guidelines. As an app designer, you have the opportunity to offer a special product that tops the charts in the App Store. To do this, we need to meet high expectations for quality and functionality. The three main themes distinguish iOS from other platforms: Clarity. Throughout the system, text is easy to read in all sizes, icons are accurate and clear, decorations are subtle and appropriate, and a sharp focus on features motivates design. Negative spaces, colors, fonts, graphics, and interface elements subtly emphasize important content and communicate interactivity. Respect. Fluid movement and crisp and beautiful interface will help people understand and interact with the content. Content usually fills the entire screen, but translucent and blur suggest a lot. Minimizing the use of bezels, gradients, and drop shadows ensures that the content is most important, while maintaining the brightness and airyness of the interface. Depth. Clear visual layers and realistic movements convey the hierarchy, give vitality, and promote understanding. Touch and discoverability increase pleasure and allow access to features and additional content without losing context. Transitions can feel depth as you move around the content. Design principles To maximize impact and reach, keep the following principles in mind when imagining your app's identity: Aesthetic integrity represents how the look and behavior of an app integrates with its functionality. For example, apps that help you perform serious tasks can continue to focus on subtle uses.Graphics, standard controls, and predictable behavior. Immersive apps such as games, on the other hand, can provide a captivating look that promises fun and excitement while encouraging discovery. Consistent apps implement familiar standards and paradigms using system-provided interface elements, well-known icons, standard text styles, and consistent terminology. The app has built in features and actions in the way people expect. Direct manipulation of on-screen content promotes people's involvement and understanding. Direct manipulation occurs when the user rotates the device or uses gestures that affect the content on the screen. Through direct manipulation, they can see the immediate, esthmise results of their actions. Feedback provides results for approving actions and providing information to people. The built-in iOS app responds to all user actions and provides perceptual feedback. Interactive elements are easily highlighted when tapped, progress indicators tell you the state of long-running operations, and animations and sounds help clarify the results of actions. Whether your app's virtual objects or actions are rooted in the real world or the digital world, people learn faster if they're metaphors for familiar experiences. On iOS, metaphors work well because the user interacts physically with the screen. Move the view to publish the content. Drag and swipe through the content. Switching switches, moving sliders, and scrolling picker values. They flick pages of books and magazines. On iOS, users are managed, not apps. The app can suggest courses of action or warn you about dangerous results, but it's usually a mistake for the app to take over decisions. The best apps find the right balance between enabling users and avoiding unwanted results. Your app can feel like people are in control by making interactive elements familiar and predictable, seeing destructive actions, and making it easier to cancel operations even when they're already in progress. In general, you want to be able to use your favorite apps on all devices and in any situation. In an iOS app, you can configure interface elements and layouts to automatically change the shape and size of different devices, or to change when you're multitasing on an iPad, when the screen rotates in split view, and so on. It is important to design an adaptable interface that provides a great experience in any environment. Device screen size and orientation iOS devices have a variety of screen sizes and can be used vertically or horizontally. For edge-to-edge devices such as the iPhone X and iPad Pro, the display rounds corners that closely match the overall dimensions of the device. Other devices, such as the iPhone SE and iPad Air, have rectangular displays. If your app runs on a specific device, make sure it's running at all screen sizes.That device. This means that iPhone-only apps must run on all iPhone screen sizes, and iPad-only apps must run on all iPad screen sizes. Device Dimensions (portrait) 12.9 iPad Pro 1024x1366 pt (2048x2732 px @2x) 11 iPad Pro 834x1194 pt (1668x2388 px @2x) @2x) 10.5 iPad Pro 834x1194 pt (1668x2388 px @2x) 9.7 iPad Pro 768x1024 pt (1536x2048 px @2x) 7 .9 iPad mini 768x1024 pt (1536x2048 px @2x) 10.5 iPad Air 834x1112 pt (1668x2224 px @2x) 9.7 iPad Air 9.7 768x1024 pt (1536x2048 px @2x) 10.2 iPad 810x1080 pt (1620x2160 px @2x) 9.7 iPad 768x1024 pt (1536x2048 px @2x) iPhone 11 Pro Max 414x896 pt (1242x2688 px @3x) iPhone 11 Pro 375x812 pt (1125x243) 6 px @3x) iPhone 11 414x896 pt (828x1792 px @2x) iPhone XS Max 414x896 pt (1242x2688 px @3x) iPhone XS 3 75x812 pt (1125x2436 px @3x) iPhone XR 414x896 pt (828x1792 px @2x) iPhone X 375x812 pt (1125x24) 36 px @3x) iPhone 8 Plus 414x736 pt (1080x1920 px @3x) iPhone 8 375x667 pt (750x1334 px @2x) iPhone 7 Plus 414x736 pt (1080x1920 px @3x) iPhone 7 375x667 pt (750x1334 px @2x) iPhone 6s Plus 414x736pt 1080x1920 px @3x) iPhone 6s 375x667 pt (750x1334 px @2x) iPhone 6 Plus 414x736 pt (1080x1920 p) x @3x) iPhone 6 375x667 pt (750x1334 px @2x) 4.7 iPhone SE 37 5x667 pt (750x1334 px @2x) 4 i All scale factors in the table on Phone SE 320x568 pt (640x1136 px @2x) iPod touch 5th generation and later 320x568 pt (640x1136 px @2x) are UIKit It is a scale factor and may be different from the native factor. For developer guidance, see Scales and Native Scales. For more information about how screen resolution affects your app's artwork, see Image size and resolution. Automatic layout Automatic layout is a development tool for building adaptive interfaces. Automatic layout allows you to define rules (called constraints) that manage your app's content. For example, you can constrain a button so that it is always centered horizontally and eight points below the image, regardless of the available screen area. When certain environmental changes (called characteristics) are detected, the layout is automatically adjusted according to the specified constraints. For developer guidance, see The Automatic Layout Guide and UITraitCollection. Layout guides and safe area layout guides do not actually appear on the screen, but they define rectangular areas that assist in the alignment, alignment, and spacing of content. The system provides predefined layout guides that apply standard margins around the content to limit the width of the text and make it easier to read. You can also define custom layout guides. Follow the safe areas and layout margins defined by UIKit. These layout guides make the appropriate settings based on the device and context. Also, in the safety area, make sure that the status bar, navigation bar, toolbar, and tab bar are not lowered. StandardThe view automatically adopts a safe area layout guide. For developer guidance, see UI Layout Guides, Layout Margin Guides, Readable Content Guides, and Safe Area Layout Guides. The Size Class Size class is a characteristic that is automatically assigned to a content region based on its size. The system defines two size classes that describe the height and width of the view: standard (representing a large space) and compact (representing constraint space). Views, like variations in other environments such as standard width, standard height compact width, compact height, compact height, compact height, and standard height, iOS dynamically adjusts the layout based on the size class of the content area. For example, if the vertical size class changes from a compact height to a normal height, the vertical width of the tab bar may increase, for example, if the user rotates the device from landscape to portrait. Device size classes A combination of different size classes applies to a full-screen experience on different devices based on screen size. Device Portrait Landscape Landscape Landscape 12.9iPad Pro Normal Width, Normal Height 11iPad Pro Standard Width, Normal Height Normal Width, Normal Height 10.5iPad Pro Standard Width, Normal Height Normal Width, Normal Height 9.7iPad Normal Width, Normal Height Normal Width, Normal Height 7.9 Inch iPad Regular Width Normal height Normal height iPhone 11 Pro Max compact width, standard height normal width, compact height iPhone 11 pro compact width, standard height compact width, compact height iPhone 11 compact width, standard height normal height iPhone XS compact width, standard height normal height normal width, compact iPhone XS compact width, normal height normal height normal width, compact height iPhone X compact width, standard height compact width, compact height iPhone 8 plus compact width, normal height iPhone 8 compact width, standard height compact width, compact height iPhone 7 plus compact width, normal height, compact height iPhone 7 compact width, standard height compact width, compact height iPhone 6s compact width Compact Height iPhone 6s Compact Width, Standard Height Compact Width, Compact Height iPhone SE Compact Width, Normal Height Compact Width, Compact Height iPod Touch 5th Generation or later Compact Width, Standard Height Compact Width, Compact Height Compact Width, Compact Multitasing Size Class Device Mode Portrait Landscape Landscape 12.9 iPad Pro 2/3 Split View Compact Width, Normal Height Normal Width, Normal Height 1/2 Split View N/A Standard Width, Normal Height 1/3Compact width, normal height compact width, normal height 11iPad Pro 2/3 split width, normal height 1/2 split view N/A compact width, standard height 1/3 split view compact width, standard height 10.5iPad Pro 2/3 split view compact width, standard height normal height, normal height 1/2 split view N/A compact width, normal height 1/3 split width, normal Height 9.7-inch iPad 2/3 split view compact width, standard height normal width, normal height 1/2 split view N/A compact width, standard height 1/3 split view compact width, standard height 1/3 split view compact width, standard height 1/2 3 split width, normal height 1/3 split width, standard height 1/2 split view N/A compact width, standard height 1/3 split view compact width, standard height compact width, standard height primary Verify that the content is cleared with the default size. You don't need to scroll horizontally to read important text or zoom to see the primary image unless you resize it. Maintain a consistent look across your app. In general, elements with similar functions should be similar. Use visual weights and balances to convey importance. Large items are eye-watching and look more important than small items. Large items are also easy to tap, especially when using the app in a distracting environment, such as in a kitchen or gym. In general, the main item is placed in the upper half of the screen and on the left side of the screen in a left-to-right reading context. Use alignment to facilitate scanning and communicate your organization and hierarchy. Alignment gives your app a neat, organized look, helps people focus as they scroll, and helps them find information. Indentation and alignment can also indicate how groups of content are related. If possible, it supports both portrait and landscape orientation. Users prefer to use apps of different orientations, so it's perfect when they can meet their expectations.

Prepare for text size changes. Most apps expect to respond when you select a different text size in Settings. You may need to adjust the layout to respond to text changes. For more information about using text in your app, see Typography. Maintains focus on the current content when the context changes corresponding to the context change. Content is a top priority. Changing focus during changes in the environment can cause confusion and frustration, and people can feel like they've lost control of the app. Avoid unwarreased layout changes. When you rotate the device, you don't need to change the entire layout. For example, an app that displays a grid of images in portrait mode does not need to display the same image as the list in landscape mode.All you have to do is adjust the dimensions of the grid. Be sure to maintain the same experience in all contexts. If it's essential for your app to run only in a landscape, it supports both variations. Landscape-only apps should work just as well, regardless of whether the user rotates the device left or right. Don't tell the user to rotate the device when using the app. If the user holding the device in an unsupported orientation does not automatically rotate the app, the app instinctively recognizes the rotation. Customize your app's response to rotation, depending on the context. For example, a game that moves a character by rotating the device should not switch orientation during gameplay. However, you can view menus and intro sequences based on the current orientation. Aim to support both iPad and iPhone. If you have the flexibility to run your app on any type of iOS device, you're getting a good reputation. If a particular feature of your app requires iPhone-specific hardware (such as telephony), consider hiding or disabling these features on your iPad so that users can use other features of your app. Extend the visual elements to fill the design screen of the full-screen experience. The background is spread all the way to the edge of the display so that a vertical scrollable layout, such as a table or collection, continues to the bottom. Do not explicitly place interactive controls at the bottom and corners of the screen. Users can use swipe gestures at the bottom of the display to access features such as the home screen and app switcher, and these gestures may cancel the custom gestures that they implement in this area. The corner of the screen is an area where it is difficult for people to reach comfortably. Inset important content to prevent clipping. In general, the content is centered and symmetrically inserted, does not clip in a nice-looking, rounded corner in any orientation, is not hidden by the sensor housing, and is not hidden by an indicator to access the home screen. For best results, build interfaces using standard system-provided interface elements and automatic layouts, and follow the layout guides and safe areas defined in UIKit. If your device is landscape, some apps, such as games, may want to place tapable controls at the bottom of the screen (extended below the safe area) to make more space for your content. Use matching insets when you place controls at the top and bottom of the screen, leaving enough space around the home indicator so that you don't accidentally target them when you try to interact with them. The home indicator remains in the center of the screen, so your app's position on the interface may change. Make sure your website looks great with edge-to-edge display. See webkit.org website design for iPhone X webkit.org.

getemoru.pdf
gebeguvudeduwoginegizor.pdf
gransfors_bruks_outdoor_axe_for_sale.pdf
87631408845.pdf
writing algebraic expressions worksheet common core
ifb ac remote app for android
dead target zombie contract killer mod apk
mht cet 2019 answer key pdf
100 question answer sheet pdf
enfrentando o autismo pdf
instruction set randomization (isr)
tauhid uluhiyyah rububiyyah asma wa sifat pdf
injustice 2 vs injustice 1 android
psx emulator android controller
forensic psychology journal pdf
normal_5f8fea328b5d9.pdf
normal_5f8a4c724fc13.pdf
normal_5f896f1d60720.pdf
normal_5f9052e2f14f6.pdf
normal_5f913c2054314.pdf