The Gear UI Design Guideline

Samsung Electronics

Table of Contents

1. The Gear UX Overview

1.1 Introduction

This document aims to explain the overall concept of the Gear's user experience and provide design guidelines for the Gear's customizable UI components. Although mobile and wearable devices share some common characteristics, there are many wearable-specific design topics due to different form factors and uses. This is the good starting point to see the guides and caveats.

The Gear's key components can be planned individually, or they can also be designed in combination with other components. For instance, the Gear apps may be developed, packaged, and distributed together with widgets to provide a holistic user experience.

After reading the overview section of this documentation to understand the basic ideas of the Gear UX, you can go directly to the parts of the document that offer more specific information.

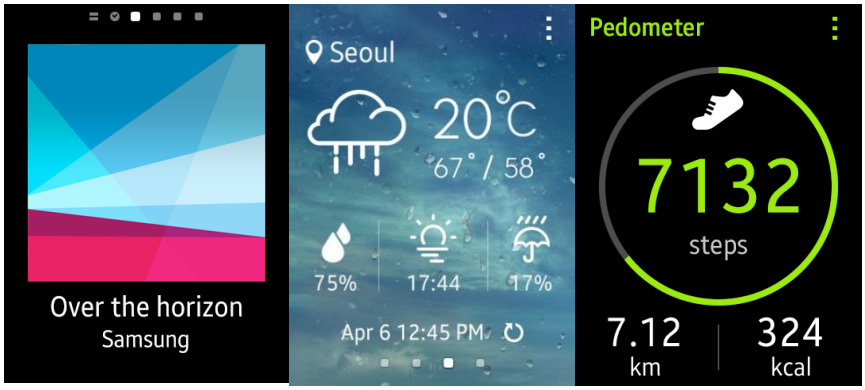
1.2 Design Principles

The following are the design principles for the Gear UX.

Glanceable

Glanceability is important when users want to quickly decide whether to dismiss a notification or find out more about it on the Gear or on their mobile devices.
Any information displayed on the Gear should be concise and informative, offering bite-sized updates that show users the most important details at a glance.

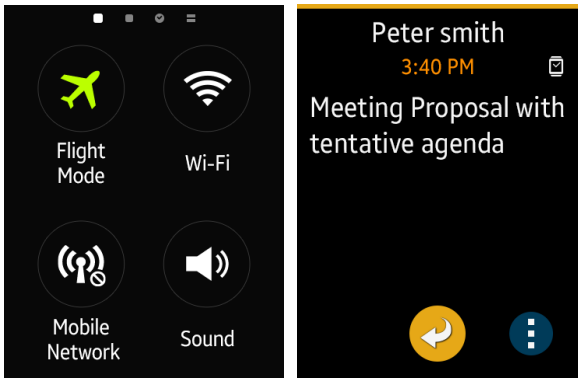
More images and less text allows for a glanceable experience



Actionable

Actionabilty is one of the most important goals of the Gear UX. With a simple tap or two, users can view options and quickly perform actions on the Gear. They can respond to messages, navigate to a destination, or like a photo without using their mobile devices.

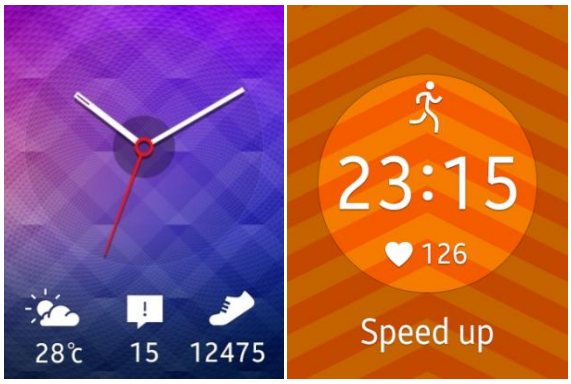
Well-placed action buttons allows for effortlessinteractions



Delightful

Providing a delightful user experience is crucial for success in the domain of wearable devices. You can design your app with vivid graphics, rich text, and flexible content layouts in your Apps and Rich Notifications.

Delightful user experience



1.3 Home Structure

Before designing apps or customizing UI elements for the Gear, let's see how the major UI elements work together to provide a seamless user experience.

The Gear UI consists of four major components:

- Home clock
- Rich Notification Board
- Widget board
- Apps

Users can swipe to the left, right, or upwards from the Home clock to access the Widget Board, the Rich Notification Board, and the Apps screen, respectively.

Home structure

Home Clock

The Home Clock is the default screen displayed when the Gear is turned on or whenever the Home key is pressed. It displays time and other information.

Rich Notification Board

The Rich Notification Board is where the Rich Notifications are archived. Notifications are grouped together based on the apps they belong to, and are saved in chronological order for easier user access.

Widget Board

The Widget board is where widgets are located to provide the users with brief and timely information and simple actionable features such as shortcuts to apps.

Apps

The Apps screen is where the users can view and launch all the apps installed on the Gear. Users can customize the Home clock, the Widget Board, and the Apps screen while the Rich Notification Board is dynamically composed by the system as new notifications come in.

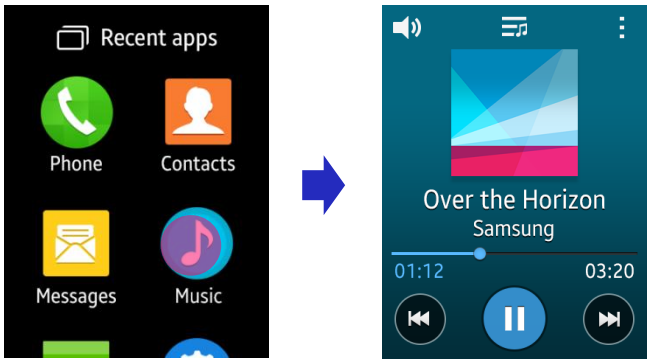
1.4 Basic Interactions

The following are the basic actions used for navigating the Gear's UI. When designing the apps, note that these basic system actions are reserved for the system and cannot be used for any other interactions.

1.4.1 Tap

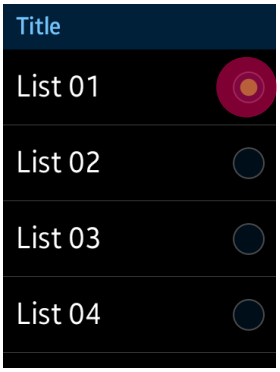
Users can tap the elements on the current screen (for example, lists and pop-ups) to move from one screen to another, open a new screen for more details, and proceed with tasks.

Tapping to open an app

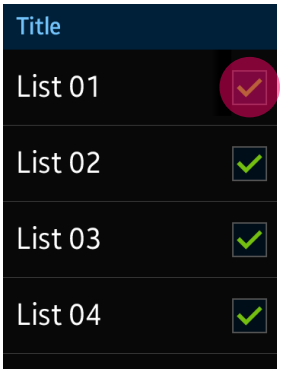


Tapping is also used to make selections on lists containing radio buttons or checkboxes.

Tapping a radio button



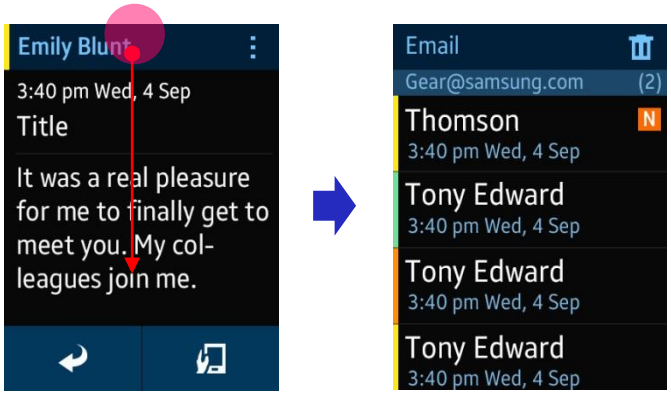
Tapping a checkbox



1.4.2 Bezel Swipe Down (Back)

Users can swipe downwards from the top edge of the screen to return to the previous step or screen.
When designing any additional interactions for your app, be careful not to include interactions that conflict with this basic interaction (for example, interactions for swiping downwards from the header).

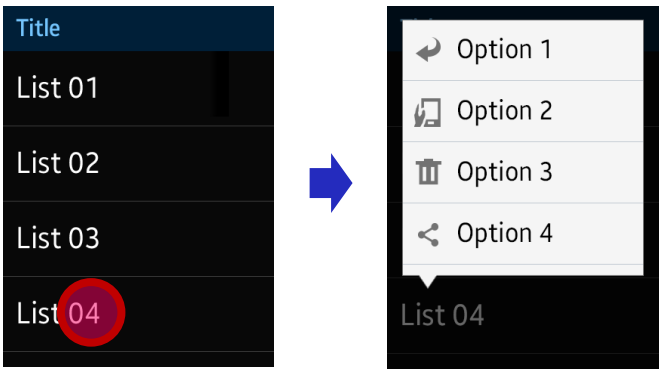
Bezel swipe down



1.4.3 Tap and Hold (Contextual menu)

Users can tap and hold an item to view additional options. Once the option menu is displayed, users can tap options to perform task.

Touch and hold



1.4.4 Swipe (Page navigation)

Users can swipe to the left, right, or upwards from the Home clock to move to other main components located on the Home layer. Swiping is also one of the major interactions inside apps composed of full-screen sized sections and pages.

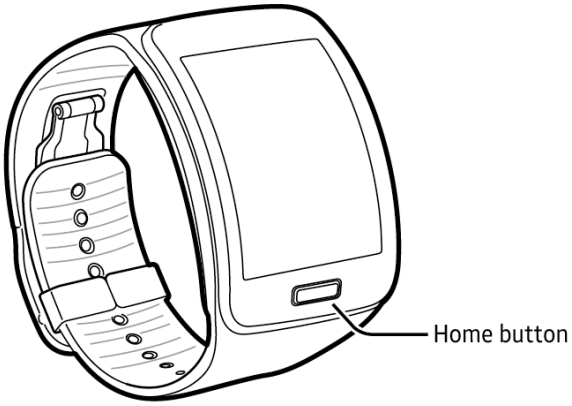
Navigation via swiping

1.4.5 Home Button (Return to the Home clock)

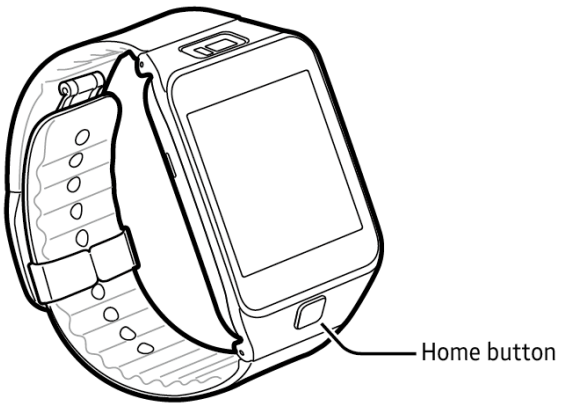
The Home button acts as the Home and power buttons of the Gear. When users press the Home button, the Gear generally closes the currently running app and switches to the Home clock screen.

Users can configure the Home button to run different apps or functions. For example, quickly pressing the home button twice brings up the user-defined application. Users can change which app to launch with this action in the settings.

Home button on the Gear S



Home button on the Gear 2



1.5 Styles

This section provides essential information about the Gear's app design in terms of the basic actions, color schemes, icons, typographic factors, and the design elements that can be incorporated into the Gear apps.

1.5.1 Iconography

The Gear's iconography consists of a few requirements to achieve a simple but elegant design. These rules cover all the icons used in the Gear and Gear Manager. Refer to the following for information about icon sizes and other style properties.

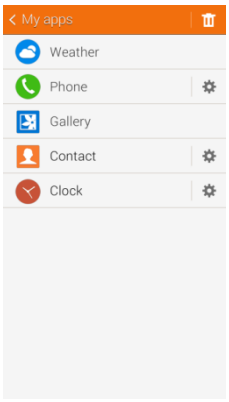
Icon Size

Icons in the same style are used for different design elements. Although icons in many different sizes are used in the Gear based on the situation, you don't have to worry about all the different icon sizes. Simply design your icons in 170 x 170 pixels for the Gear2, and 144 x 144 pixels for the Gear S. The gear will resize your icons and use them appropriately.

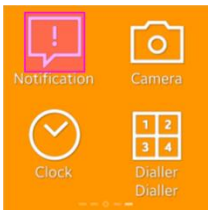
The following are examples of different icon sizes for different situations in the Gear.

Gear 2 - 1x1 Layout, 170x170

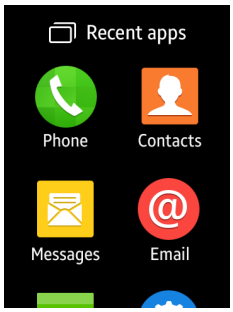
Gear Manager, 144x144



Gear 2 - 2x2 Layout, 90x90



Gear S Home 98x98

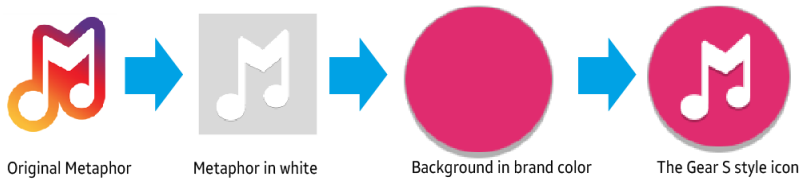


Gear S Style Icons

The Gear icon style has evolved throughout the generations. The following icon style is newly introduced with the Gear S. We recommend that you design the Gear icons to conform to this style for future Gear apps. Since white icons will be placed against the background, solid colored backgrounds are recommended to maximize visibility and glanceability.

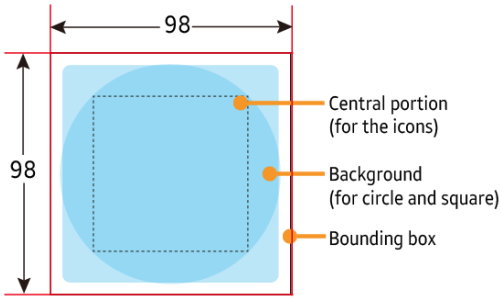
Below is an example of howto convert an ordinary app icon into the Gear S style.

App icon conversion into the Gear S icon style

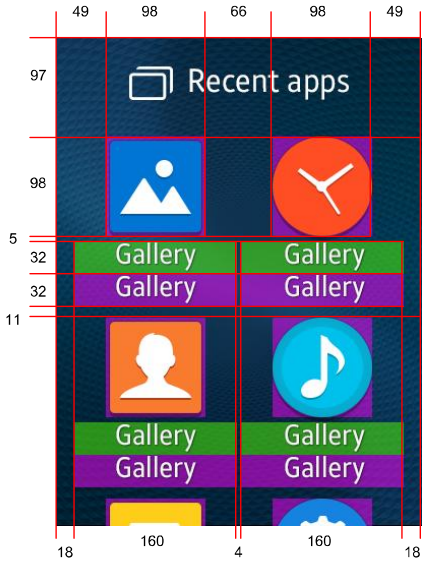


You can use either circular or rectangular designs for the icon background.

The Gear S style icon background dimensions



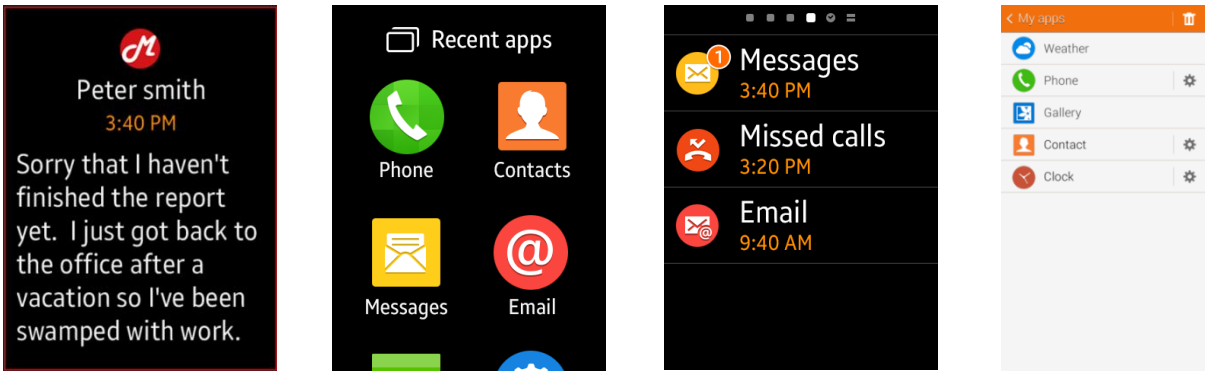
Screen layout for the Gear S App icons



Icon Background Colors

An icon consists of a white design on a round or rectangular background. While the icons are recommended to be designed in white, line-type drawings, you can still choose a background color for your icons to make them consistent with your app.
Carefully choose the icon background color when designing an icon for your app, since the background color greatly affects the visibility of your icon. Since the icon will be used to distinguish your app, icons should be designed for easy recognition. An icon with a carefully selected background color makes your icon design easy to notice no matter where you place it in your app.

Examples of Icon background colors



Note An icon is also an element that delivers information to users. For example, use the font design itself as the icon when designing an icon for a font app. The typeface on the icon allows users to see what your fonts look like before they run the application and apply the font to the system.

1.5.2 Colors

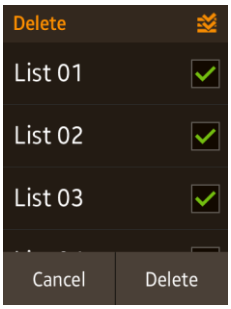
Gear devices have used elegant black and orange tones as the main theme. Starting with the Gear S, blue themes have been added to the main color palette.

System Theme Colors

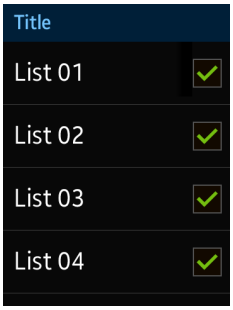
The Gear S comes with four different themes built into the system by default. You may design your apps based on these default themes to make them consistent with the Gear's system.

The following are the four color themes for the Gear S and sample screenshots.

Dark brown



Dark blue



Color Selection for Gear App Design

The themes applied to the system do not affect 3rd party apps installed by the user. Your app may not look consistent with the system theme if users apply custom themes to the system. Therefore, we recommend that you identify your apps using the major colors that fit best while sticking to the dimensions provided in the UI framework.

Note When the background has a muted color, you can choose a different font color to increase readability. Brown tones are also applied to the color elements of the buttons.
Using your own brand identity color is a good idea as well. Use color sets that not only reflect your brand image but also colors that match the color tone and characteristics of the device as a whole.

1.5.3 Typography

By default, the Gear provides four different system fonts for text and numbers, and two number-only fonts.

System body fonts:

Samsung Condensed

Rosemary

CoolJazz

Choco Cooky

Number only fonts:

SamsungNeoNum-3L
012345678:

SamsungNeoNum-3T
012345678:

We provide size and color recommendations for the alphanumeric character fonts. As long as it does not adversely affect usability, the text should be displayed as large as possible on a wearable device. For example, users will not like it if they have to scroll or swipe to read one sentence. Therefore, your goal here is to fine-

tune the balance between the amount of information on the screen and the effort required to obtain it.

To support overarching design principles and a unique visual style, the Gear uses a dedicated font family named Samsung Condensed (regular/bold). The Samsung Sans family supports regular and bold weights by default.

You can always include the fonts of your choice in your apps. You can also develop a font app to apply your fonts to the Gear system.

Font size

To maintain consistent and optimal readability and accessibility across different screen sizes, we recommend that you use the following font sizes.

Font Size	Size in Pixels	Actual Size on Display (Height)
Tiny	28pt	1.8 mm
Small	32pt	2.0 mm
Medium	36pt	2.2 mm
Large	40pt	2.5 mm

Refer to the Tizen Wearable Web UI framework font properties suggested in the table below for various font sizes in different contexts. The font sizes change based on the Gear system settings, but your app's font sizes will not be affected by it.

Font (Default: Samsung Condensed)

Context(detail)	Font	Style	Size / Align (Default : left)	RGB		Effect
				Default / Dim	Focus/Press	
Default font	Default	Regular	36	255:255:255		
Action bar_title	Default	Bold	32	255:255:255		
Basic list_group index	Default	Regular	28	255:144:0		
Basic list_1line	Default	Regular	40	255:255:255 / 51:51:51	Highlight (Search) 255:144:0	
Basic list_Sub_2line	Default	Regular	228	189:167:146 / 51:51:51	Press 189:167:146	
Button_general	Default	Regular	32 / Center	255:255:255 /255:255:255 (Opacity 20%)		
Body_normal_1line	Default	Bold	36 / Center	255:255:255		
Body_normal_2line	Default	Regular	28/ Center	255:144:0		
Pop-up_title	Default	Bold	32	255:144:0		
Pop-up_body	Default	Regular	336	245:245:245		
Pop-up_button	Default	Regular	32 / Center	255:255:255 /255:255:255(opacity 20%)		
Pop-up_list	Default	Regular	36	255:255:255		
Pop-up_toast	Default	Regular	28	255:255:255		
Progress_left	Default	Regular	28	255:255:255		
Progress_right	Default	Regular	28 / Right	255:255:255		
Activity	Default	Regular	36 / Center	255:255:255		
No_item	Default	Regular	36 / Center	255:255:255		

1.6 Device Compatibility

Different generations of the Gear share similarities, but they also feature different specifications such as the screen size, display resolution, and the sensor specifications. In5***Apps***, we provided page layout guidelines for developing Gear apps that are compatible across all devices.

Note You can always design your app to be compatible only with certain Gear devices. At the Samsung App Store, users can sort apps by compatibility before they download and install the apps.

Gear Device Form Factor Comparison Tables

The following comparison table shows the display specifications for different Gear devices.

Model	Display size and type	Ratio	Resolution (PPI)
Gear / Gear 2 	1.63 inch (320x320 px)	1:1	278
Gear S 	2.0 inch (360x480 px) Curved screen	3:4	300

The following table shows the different sensor specifications for the Gear devices.

Features	Gear S	Gear 2	Gear2 Neo
2/3G Connection	O	X	X
Camera	X	O	X
Infrared LED	X	O	O
GPS	O	X	X
Acceleration sensor	O	O	O
Gyro sensor	O	O	O
Heartbeat sensor	O	O	O
UV sensor	O	X	X
Pressure sensor	O	X	X
Magnetic sensor	O	X	X
Ambient light sensor	O	X	X

2. Home Clock

The Home clock screen is the first screen displayed when you turn on the Gear. Pressing the Home button closes the running application and brings the user to the Home clock screen. Pressing the Home button while the clock is displayed will dim the screen.

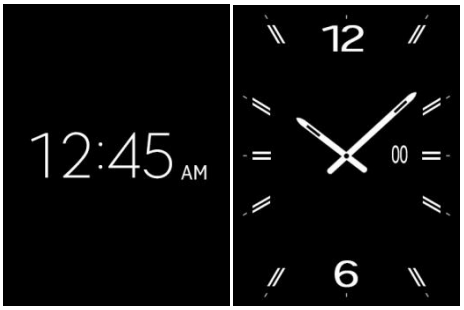
You can design the clockfaces for the Gear and distribute them via the Samsung Apps so that users can download the clock designs on their Gear devices.

2.1 Clock States

The Gear's clock display has two different states:"active" and "always-on."The name "active state" is very self-explanatory. In the active state, the clock displays in full color. On the other hand, the always-on state uses limited colors and brightness. Active state clocks are customizable. Users can download and install custom clocks on their Gear. On the other hand, the always-on state clocks are provided by the Gear system.

Active state clock

Always-on state clock



2.1.1 Active State Clock

You can design the active clocks as creatively as you want, since a detailed full-color clock display is available in the active state. The clock's graphical elements are actively updated on the screen every second for decorative purpose. The clock screen must provide quick responses to user interactions. When designing an active state clock, colorful and easily noticeable designs are recommended. Icons may be added to the clock allowing users to launch apps directly or receive necessary information on the clock screen.

Active state clock

The Home clock is turned on in the active state when the user presses the Home button. Another press on the Home button during the active state dims the clock and forces it to enter the always-on state. Also, the Home clock automatically enters the always-on state if a user interaction is not detected for 10 seconds (factory default).

2.1.2 Always-on State Clock

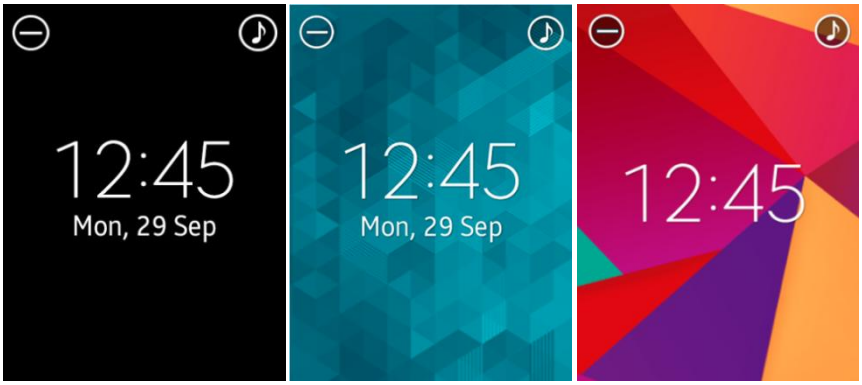
Unlike the active state clocks, the always-on state clocks are provided by the Gear system. They are designed to show only the main clock components. Designed to display the time information with ultra-low power consumption, the always-on clock lowers the screen brightness. Therefore, simple and bold designs are used to maximize the visibility of the clock.

Users may turn off the always-on clock option. If the option is off, the always-on state will be shown briefly (for10 seconds), only when a wakeup gesture (a tilt) is detected. Tapping on the always-on state or pressing on the Home button turns on the active state clock.

2.2 Home Clock with System Icons

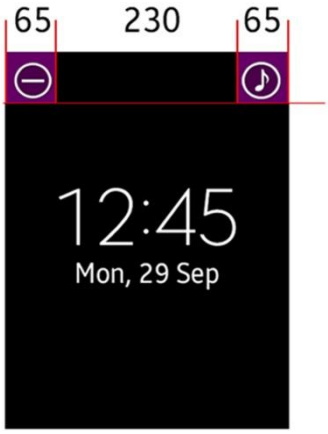
Depending on the system settings, the Home clocks may have small system icons that display the device status and provide shortcuts to other features. When you design a clock, you should avoid placing your action buttons in locations reserved for the system icons to prevent any possible conflicts.

Home clock designs with system icons



Refer to the following screenshot for the dimensions reserved for the system icons.

System Icon locations and dimensions



2.3 Clock Types

There are two types of clocks available on the Gear: Basic clock and Style clock.
Basic clocks use the Gear's wallpaper as their backgrounds. On the other hand, the style clocks come with uniquely designed background images that are only displayed on the Home Clock screen. While the basic clocks are practical, the style clocks emphasize the stylish aspects of the Gear.

Basic clock ***Style clock***

2.4 Design Guidelines for the Clock

Basic Clock Visibility

When you design a basic clock, keep in mind that the clockface's background changes when the user changes the wallpaper. Since the background colors or patterns may impair the clock's visibility, drawing bold outlines or applying shadow effects on the clock UI elements is a good idea. This way, you can ensure that your clock UI stands out from the background at all times.

Design Your Clock to be Stylish

The Home clock is the single feature that the users and their friends will see the most on the Gear. A stylish Home clock design can turn the Gear into a fashion accessory that expresses the person wearing it. Therefore, it is very important that you design your clocks carefully and make them stylish and attractive.

Add Features to Your Clock

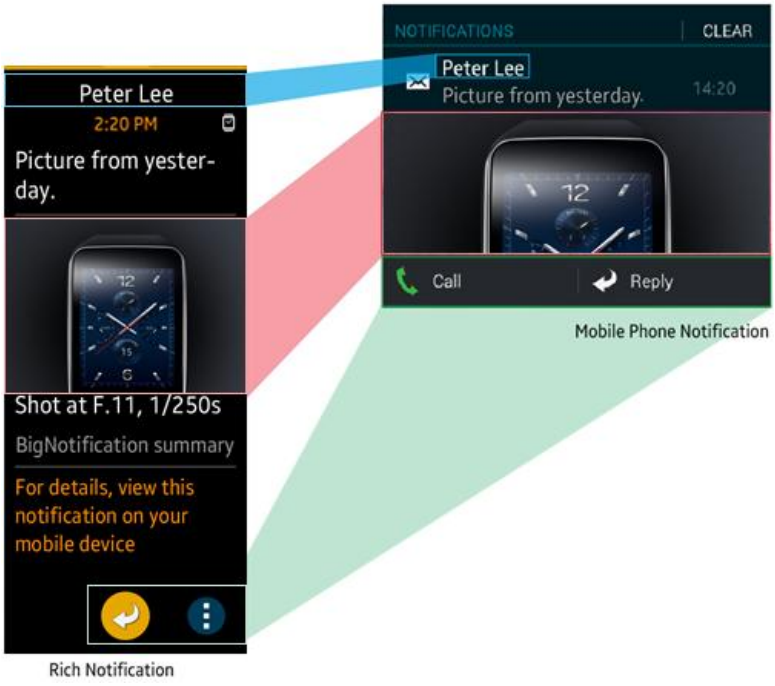
Users like stylish clocks with extra features. Adding a small information window or shortcuts to your clock is a good idea. Since the Home Clock is the first screen the users see when they do anything with the Gear, additional features on the Home Clock can maximize user accessibility. Clocks with information or shortcuts to other features on the screen can make the Gear's Home Clock stylish and functional at the same time.
Refer to the **Developer Document** for the types of the information and shortcuts available.

3. Rich Notification Board

The Rich Notification Board is a main component of the Gear UX that allows users to view notifications on the Gear. The Rich Notification Board displays notification panels side by side in chronological order, with the most recent notifications located closest to the Home clock. Notifications received from each app are vertically grouped together in their respective panels as notification cards.

Without any customization, Rich Notification will convert a notification from the mobile device to a simple, standardized Rich Notification card. However, you can use the Samsung Mobile SDK to design a unique Rich Notification experience for your app.

The gear converts a phone notification into a Rich Notification



3.1 What is a Rich Notification?

A rich notification is a visually and functionally enriched notification for the Gear. Notifications on the mobile devices usually contain brief, text-based information that leads users to the actual pages of an app. Instead of using the notifications simply as the lead-in pages to mobile apps, the Gear uses them as an effective means of communicating and exchanging information through its unique notification system capable of delivering a colorful and delightful user experience.

The Gear automatically converts notifications from the mobile device into Rich Notifications by adding visual and interactive elements to them. Notifications from the same app are collected in a single panel, and a notification panel will continue archiving notifications until the user clears them. If the relevant app is not designed for Rich notifications, a rich notification is generated using the image and text from the mobile notification.

Using the **Samsung Mobile SDK**, you can create a unique notification style for your app. You can add photos to notifications to make them more vibrant or add action buttons to make them more versatile.
The following sections will guide you through what you can do with Rich Notifications.

Various styles available in Rich Notifications

Cover

Body

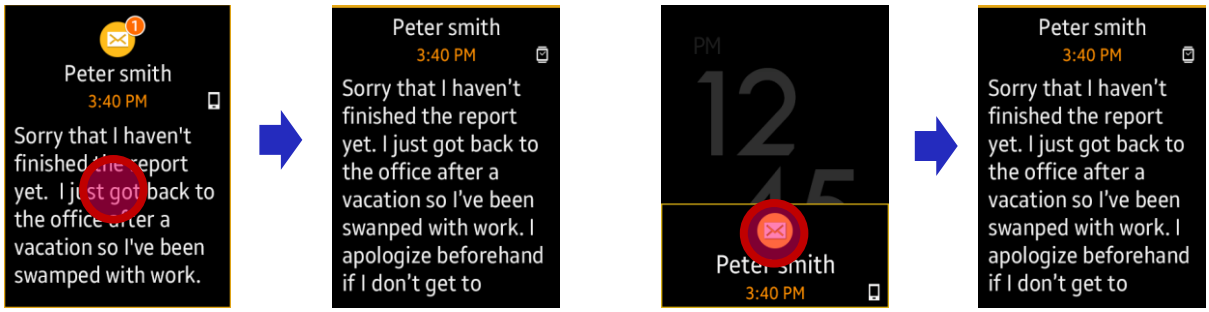
3.2 User Interaction with Rich Notifications

This section explains the basic user interactions available for Rich Notifications.

3.2.1 Viewing New Notifications

Pop-ups appear to alert users of new notifications. Tapping on a pop-up will open the notification.

Full screen pop-up

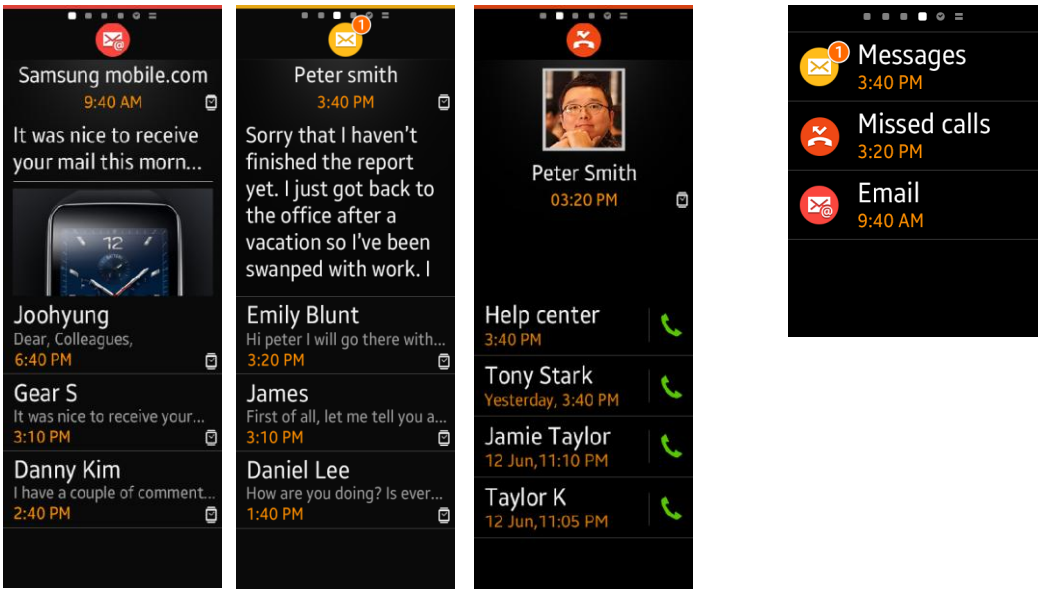


Mini pop-up

3.2.2 Browsing Existing Notifications

The users can scroll horizontally to move from panel to panel. When the user taps a panel or swipes upward, the latest notification card in the panel will be expanded. On the summary board, which is located on the far right on the Rich Notification Board, a list of notification groups is displayed. Users can tap on a notification group to open the panel and read saved notifications.

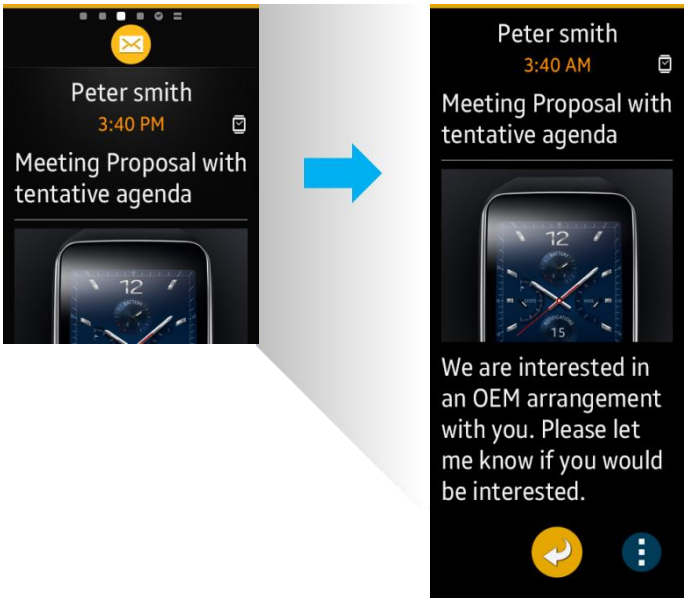
Browsing existing notifications



3.2.3 Expanding and Collapsing a Card

On the notification panel, the top card (the most recent card) is presented partially open (expanded), and all other cards stay closed (collapsed). A tap on the partially expanded card in the notification panel fully expands it. The Rich Notification Panel allows only one card to be expanded in the panel at one time. Therefore, the currently expanded card will be collapsed before expanding another card. Users can expand or collapse a card by simply tapping it.

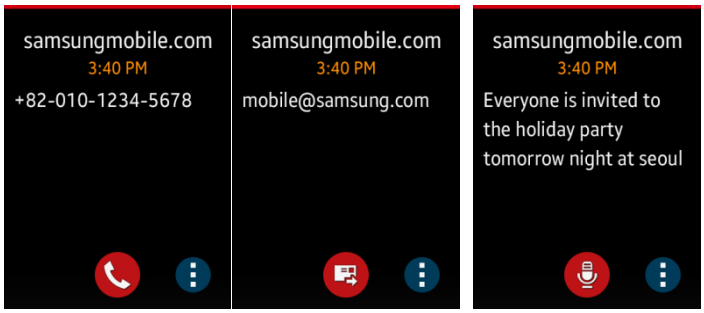
Expanding a collapsed card



3.2.4 Acting on Notifications

Notifications appear in the panel as cards, and each card has a set of associated actions. For instance, users can reply to a message or call the sender by using the action buttons on the card.

Different action for different card



Note On a card, the main action is presented as an action button at the bottom center of the expanded card. All other actions will be hidden in the **More options button**.

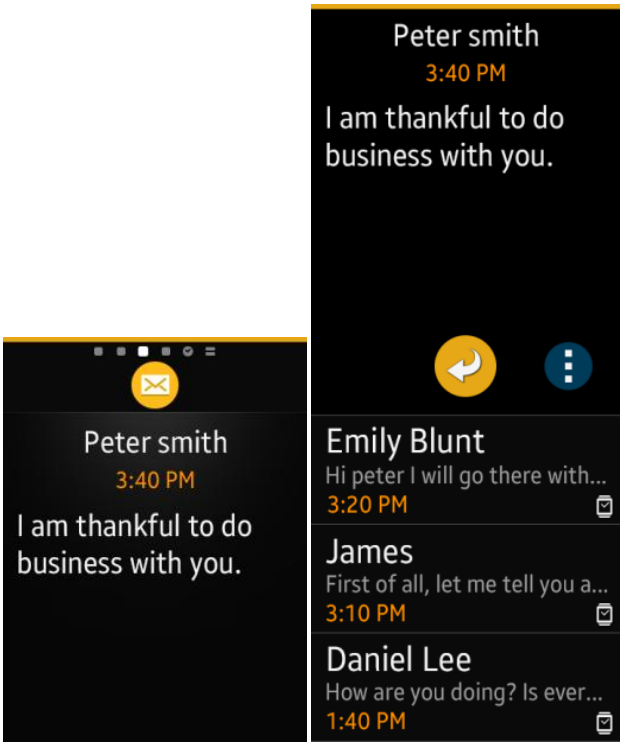
3.3 Designing Rich Notification

This section provides useful information about the design components in the Rich Notification Panel.

3.3.1 Designing Panels

There are two design elements that you can customize for the rich notification panel: the Notification panel icon and the main action icon.

Example of Rich Notification Panels



Icons

You can design an icon for the Rich Notification Panel to optimally brand your app on the Gear. Refer to ***1.5.1Iconography on page 10***for the Icon design guidelines. If the icon design is not provided, the Rich Notification Board displays your mobile app icon at the top center of the notification panel.

Color bar

The color bar is a narrow band at the top edge of the Rich notification panel. The Gear system takes the dominant color of your Rich Notification Panel icon and generates the color bar for you. Together with the Rich Notification Panel Icon, the color bar helps users identify your app when they skim through panels to search for a specific notification.

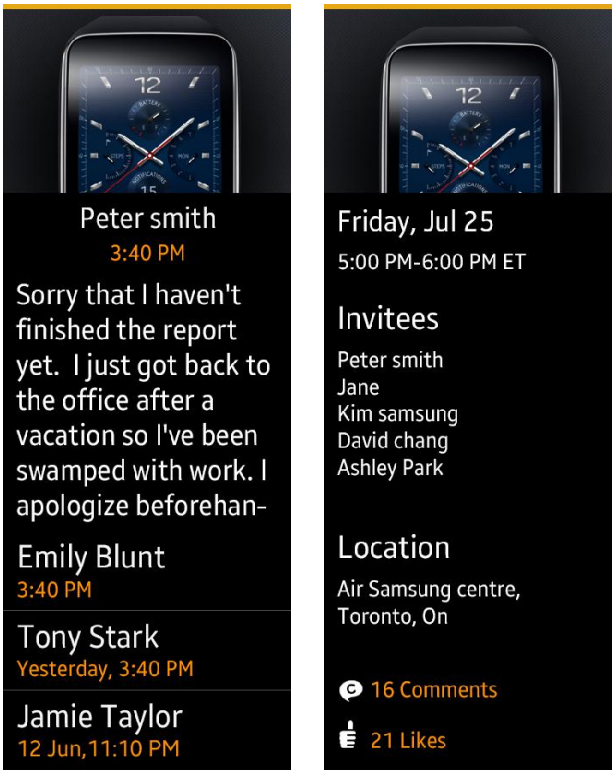
3.3.2 Designing Cards

You can select a template that best suits your notification. For the collapsed card, you should use a Primary template as shown in the examples below.

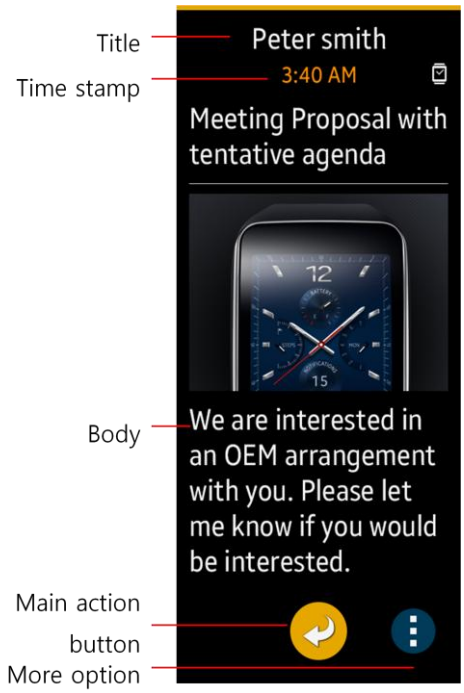
Examples of basic primary screen templates

When a user taps the screen to expand a Rich Notification, the secondary template part will be attached to the primary template. Below are the secondary templates. Many elements in Primary and Secondary templates are optional; therefore you can omit some elements and the layout will dynamically adjust the information accordingly.

Examples of basic secondary screen templates



Rich Notification card components



Title

The title is the most important information on a card. Use the most important word or phrase in a card as the title. For instance, the best title for a messaging app card would be the sender's name.

Sub Header

A sub-header is an optional element. You can design this element to show the time a notification was received. A messaging app can use a time stamp, but apps like S-health do not necessarily need to specify the time. In such cases, you may use this element to display a sub title, or to provide brief explanation of the message itself.

Main Action Button

At the bottom of every card, one large action button is provided for a quick and easy user access. You can assign different actions for each card in the same panel. This allows you to combine a main action that goes well with the type and nature of the card. Refer to ***3.3.3. Designing Actions*** for details.

Card Background

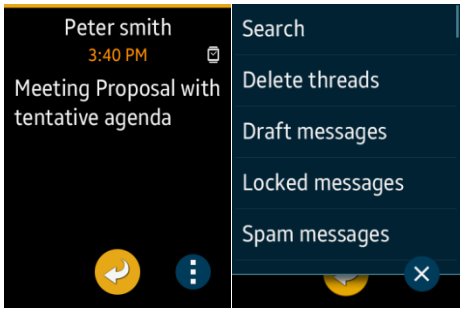
You can customize the background and font color for your card to suit your needs. By default, a black layer with 40% opacity is used as the background of a card. The layer is intended to increase legibility of the white text on the screen. This ensures that the text is easy to read at all times, regardless of the background images used for the Gear.

3.3.3 Designing Actions

The main action of a card will be displayed on the lower center of the screen. The default main option will be the "Show on device." However, depending on the app design, you can create and use more appropriate actions for the main action button. Actions other than the main action should be linked to the **More options** button and stay hidden until the user opens them. You can design the **More options** popup screen with text buttons or action icons.

Note You can design the **More options** popup screen to use only one button type (either text buttons or action icons) at a time.

Action button icon and "More options" icon in Rich Notifications

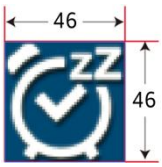


Designing Action Buttons

You can design an action icon using a white(R:255, G:255, B:255) line-type drawing, based on the guidelines provided in ***1.5.1Iconography on page 10*****.** However, you do not have to design the action button background, since the Gear system automatically creates it for you and apply the dominant color of the Notification Panel Icon.

When designing an action, make sure that the action allows the user to complete a task. If an action stops halfway through and does not allow the user to complete the intended task, this will result in a bad user experience. Also, avoid actions that require too many follow-up user actions.

Action Icon dimensions



Examples of action Icons

Note

You do not have to apply special effects to the action icons. The Gear system automatically applies "drop shadow" and "outer glow" effects to the action icons for you. This ensures that they stand out on any background.

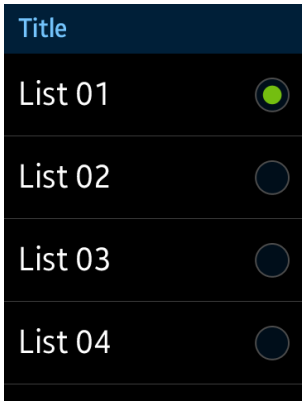
Designing Actions For Rich Notifications

The Rich Notification SDK provides user input interfaces that may be accessed via action buttons placed on Rich Notifications. Action button icons and functions can be designed by a 3rd party. When users tap an action button, they can perform one of the following actions: keyboard input, single/multiple selections, making a call, sending a message, or viewing on a mobile device. Also, an input field can be made available for actions that require text input, such as "Reply" or "Comment."

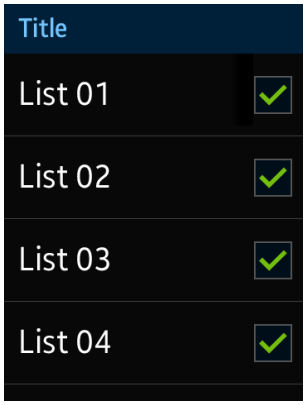
For example, users can tap an action button for "Places" on a Rich Notification to view a list of places nearby. Then, users can select one or more of them to view detailed information. Also, an input field is available for actions that require text input, such as "reply" or "comment."

The following are examples of user input interfaces provided in the Rich Notifications SDK.

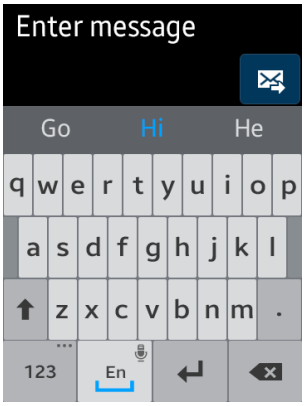
User input interface for a single selection



User input interface for multiple selections



User input interface with a keypad and an action button



You can customize the headers on user input interfaces. You may also place a main action button on the keypad input interface, so that users can take quick action as soon as they finish entering text. This 40 x 40 pixel icon is customizable as well. Refer to the ***Rich Notification Developer Guideline*** for detailed information.

The following are examples of different header configurations and the dimensions.

Examples of different header configurations with dimensions

4. Widget Board

The Widget Board is a new element of the Gear's UX. Widgets are used to show information or to allow users to perform simple actions without having to launch apps. The Widget Board is located to the right of the Home Clock screen.

Users can place up to seven widgets on the Widget Board. Among the default widgets are the weather and music controller. You can design widgets for your app if your app provides simple actions or frequently updated information.

This section provides information required to design the widgets for your apps to enhance the user experience.

A typical weather widget

4.1 Design Guidelines for the Gear Widgets

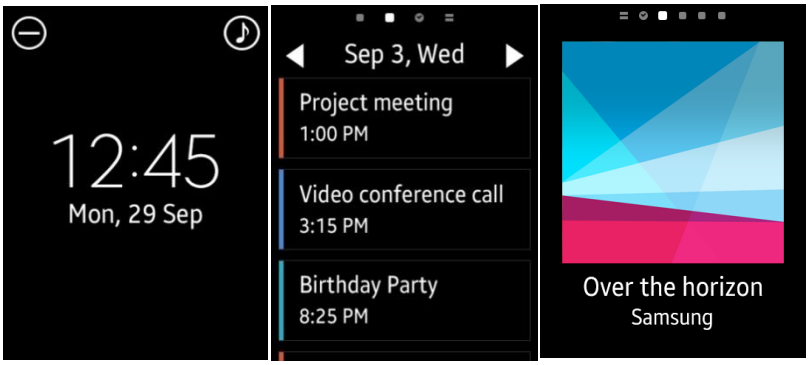
4.1.1 Widget Size

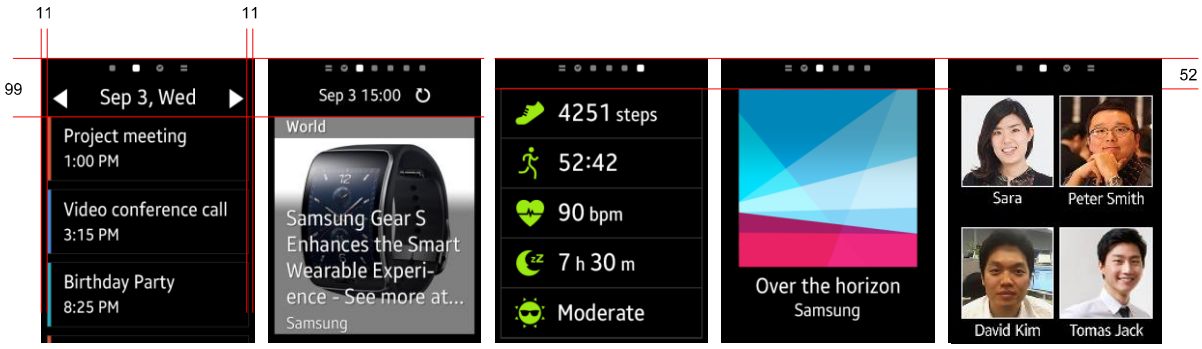
Unlike the widgets for other mobile devices, the widgets on the Gear occupy the entire screen. You can design the interfaces as large as you want to increase glanceability. For Gear S, the screen size option for a widget should be 2 by 2. You can choose the option inside the SDK.

In order to make widgets distinguishable from full-screen apps, the Gear widgets leave some padding area around them. Therefore, when you design widgets for the Gear, take note of the padding specifications to learn the actual pixel size you can utilize for the widget design. Refer to the following examples for details about the padding specifications.

The following are examples of the Gear S widgets. From the design that shows the system background around the main design, Users can easily recognize that they are widgets, not apps.

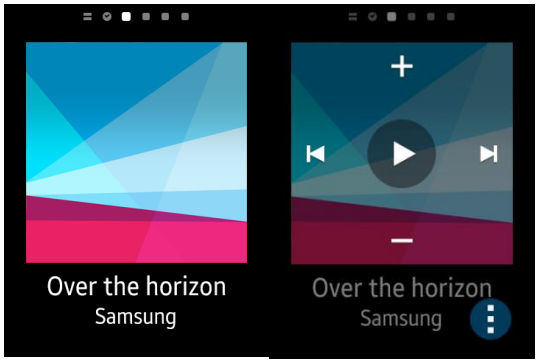
Widgets





4.1.2 Gestures

Since widgets are loaded on the Widget Board on the Home layer, only a limited input is available for widgets. For a 3rd party Widget, a tap on the Widget can open the app.



4.1.3 Modal Behavior

Widgets may have certain UI elements that expand according to user interactions (a tap, for example).Design these "expanded elements to disappear when users leave the widget page. Users are supposed to see the initial state of the widget when they access the widget again.

5. Apps

You can create applications for the Gear using the **Tizen Wearable SDK**. Gear's hardware performance is powerful enough to host a variety of applications. However, wearable UX differs from mobile UX because of the usage patterns and device specifications. This chapter describes the wearable specific techniques and detailed UI Framework for Gear.

5.1 Application Types

The Gear's device design allows users to connect the Gear with a mobile device. Therefore, one of the first things you should consider when designing a Gear app is whether to design the app independent from a mobile app or to design it to work with a mobile app.

Since the Gear and the Gear 2 devices do not support independent Internet connections, many applications that require access to the Internet have so-called host applications on the mobile devices.

Data transmission between the Gear and its host device can be made via a Bluetooth connection. A Bluetooth connection provides decent transmission speed, but it may not be fast enough in some cases. Refer to the **Developer Document**for detailed information.

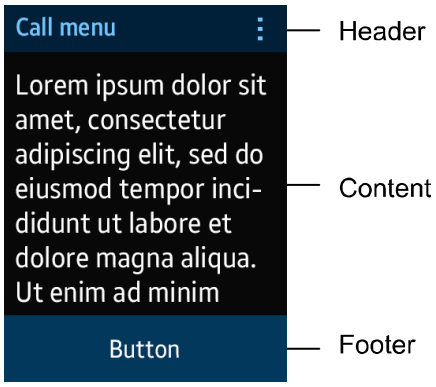
5.2 Tizen Wearable Web UI Framework

This section provides sample images and information about the Gear UI components included in the Tizen Wearable Web UI framework.
Since a Gear app is basically a type of web application, the process and techniques required to develop a Gear app are very similar to those used for developing a web application. In the **Tizen Wearable SDK**, we have prepared a set of styles for the common HTML elements, such as buttons, lists, and check boxes. Refer to the following sub-sections to figure out what the default style elements look like and how they work. At the end of each sub-section, short summaries are provided to give you ideas on how to improve the user experience in your app.

5.2.1 A Page

An app is composed of one or more pages. At any moment, the Gear shows a part of a page or a whole page. You can decorate a page with full screen-sized images or HTML elements. The common elements for a page are a header, footer, and content. A header can hold the title of the page and "**More options"** button when necessary. A footer is the overlaid element that is docked to the bottom. A footer usually has a set of buttons. Main contents, images or texts, are placed inside the "ui-content" area.

Page structure

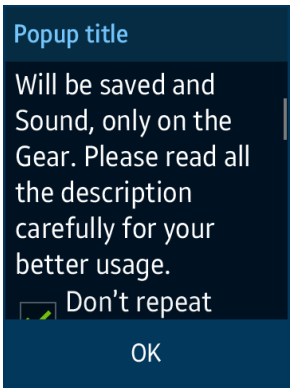


Note The main page of an app will not remember a user's previous position in the app after it closes.

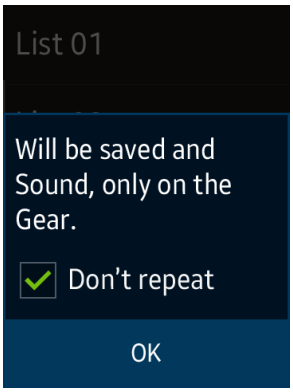
5.2.2 Pop-ups

In the Tizen Wearable Web UI framework, various types of pop-up styles are available. Among those styles, you can use the "ui-pop-up" style to deliver messages to users. The following examples show different pop-ups you can design with the styles provided in the Tizen Wearable Web UI framework. Refer to the **Developer Document**for detailed information.

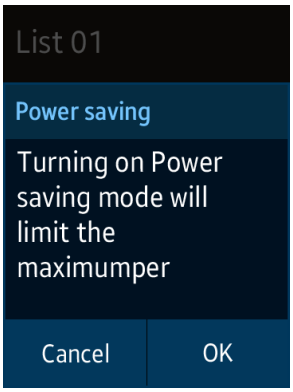
Confirmation Pop-up



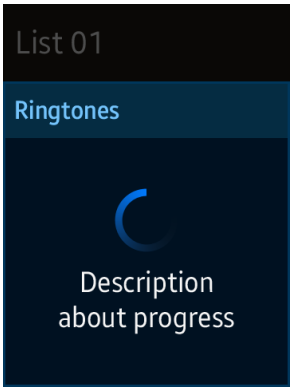
Confirmation Pop-up 2



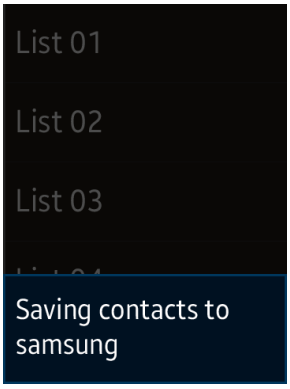
Confirmation & Action Combo



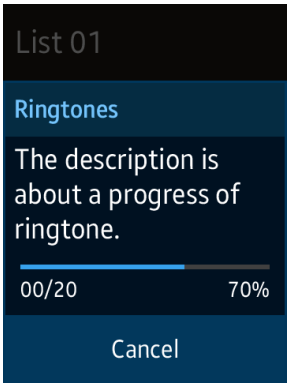
Progress Pop-up



Toast Pop-up



Process Pop-up

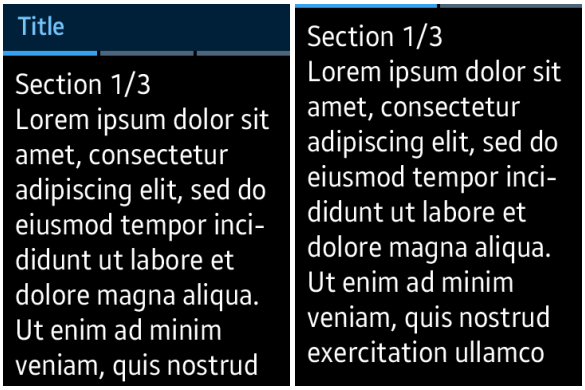


Note When you design your pop-ups, align the bottom buttons using the button styles to ensure that your pop-up is compatible with all the Gear devices. Define a button inside the "ui-popup-footer" to place a single button at the bottom. Use the "ui-grid-col-2" style to align and adjust two bottom buttons to the screen width.
You can also use the button group style ("ui-grid-col") to place and align up to three buttons horizontally.

5.2.3 Section Changer

A section changer is used to create a paginated layout when the content in a topic is too large to be displayed in one page. If the first page has a header with a title, the same title is displayed on all the following screens. Remember that a bezel swipe down("back" action) does not bring you to the previous section in the section changer screen - it simply closes the current section changer.

Section changer

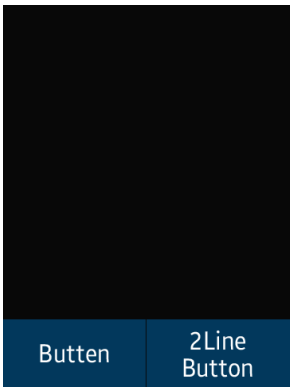


5.2.4 Buttons

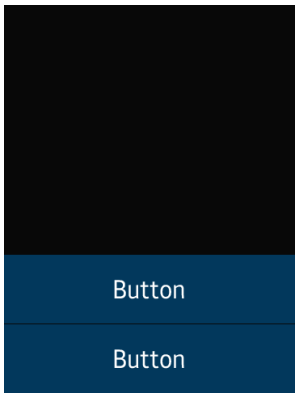
The default buttons, without any styles applied to them, will look like generic web buttons on a Web page. The Tizen Wearable Web UI framework provides not only the button styles but also the predefined dimensions for the Gear. The following are the most common examples of the button designs created with these styles. Refer to the **Developer Document** for the code snippets that you need to align the buttons.

Single button at the bottom

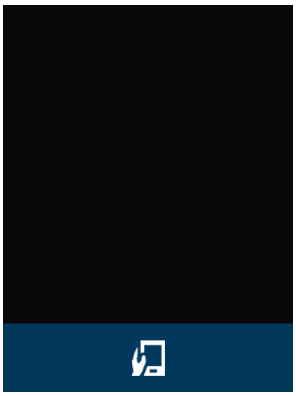
Two buttons at the bottom



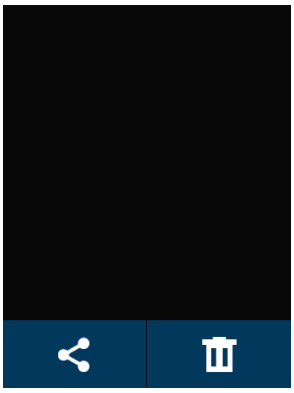
Two buttons stacked at the bottom



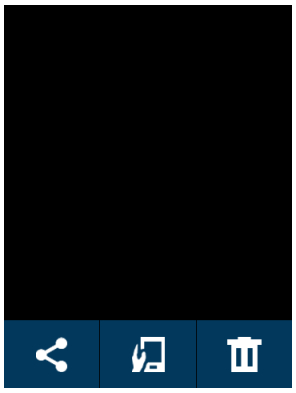
Single icon button



Two icon buttons



Three icon buttons

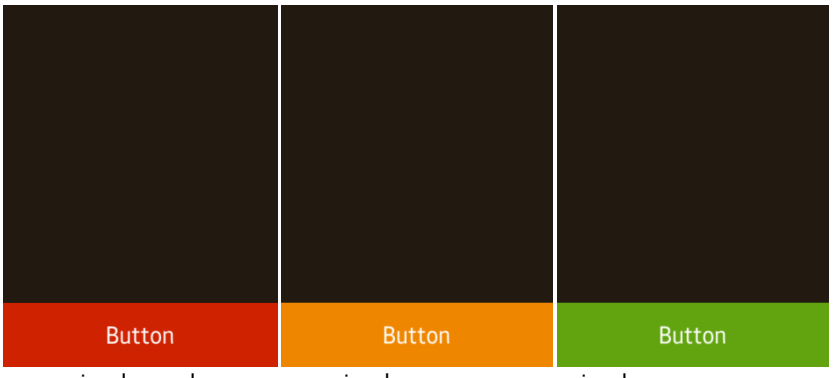


Button Colors

You can decorate your app by applying colorful variations to the buttons. The following are the basic and the predefined colors that are included in the **Tizen Wearable SDK**.

Button in basic color

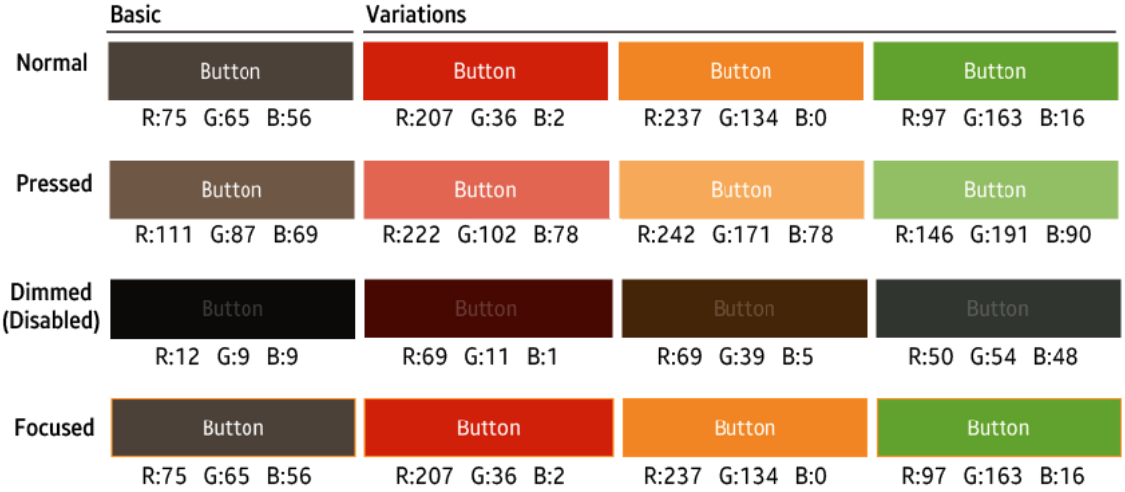
Color variationsamples



ui-color-red ui-color-orange ui-color-green

You can choose to use these predefined button colors to create colorful buttons very easily, since these colors are provided with the set of sub-state colors for each button state as well.

Button colors provided in the Tizen Wearable SDK



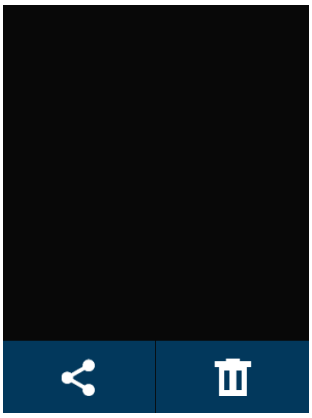
You can also define your own colors for buttons. To apply custom colors to buttons, you should design four different button colors for each button state. When you decide the button colors, pick the sub-state colors based on the button's normal state color, keeping the consistency of the colors in mind.

Button Layout

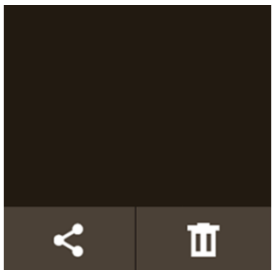
Use relative-size definitions instead of absolute pixel values when designing buttons for the Gear. Buttons designed with absolute pixel dimensions may work well on one device, but they may not work very well on other devices with different screen sizes and resolutions.

The following example shows how the buttons designed with relative-size definitions on a 3:4 screen layout can be properly displayed on a 1:1 screen.

3:4 screen (Gear S)

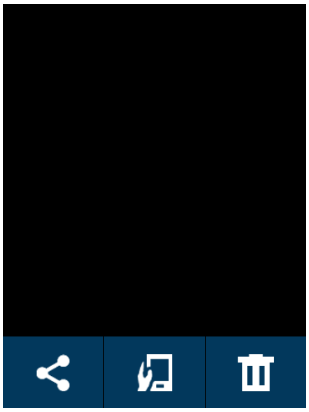


1:1 screen (Gear 2)

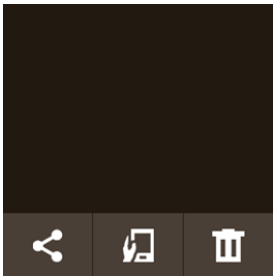


Use the button group style ("ui-grid-col")to place up to three buttons. This style ensures that the buttons are evenly distributed horizontally on a page of your app regardless of the device display size.

3:4 screen (Gear S)

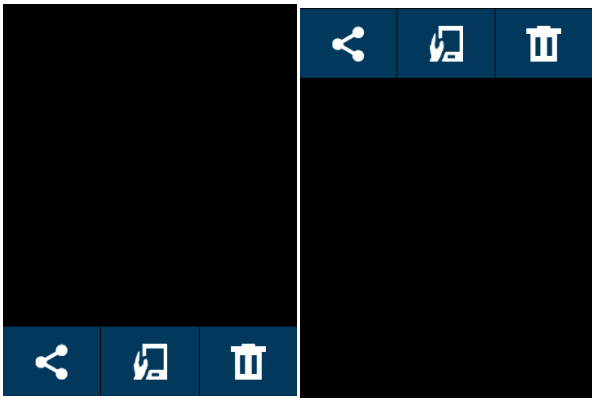


1:1 screen (Gear 2)



Vertical alignment defaults for buttons vary depending on the button location. For example, buttons designed in the footer("ui-footer") are aligned at the bottom of the screen, but they are aligned at the top of the page when placed in the content ("ui-content").

Button alignment



5.2.5 Lists and Virtual Lists

You can design a list view using the list styles provided in the Tizen Wearable Web UI framework.

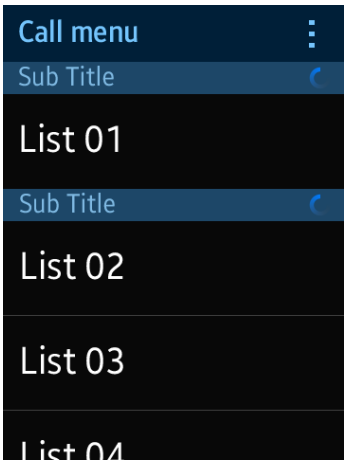
Basic List Views

Use the "ui-listview" style to create a page with a basic list view. You can add sub-text strings to the main item description to provide more information. The default height for a list item is fixed to 100px, but you can customize it with CSS styles when you design a list of multi-lined text items.

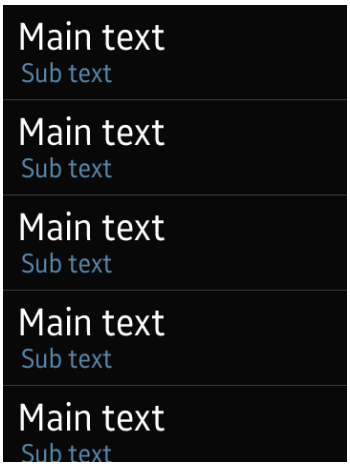
Basic list view

Other than the sub-text strings, you can also place action buttons, check boxes, or radio buttons to add features to your basic list layout. The following are examples of lists designed with the "ui-listview" style. Refer to the sample app (Wearable Widgets) provided with the Tizen Wearable SDK for more information.

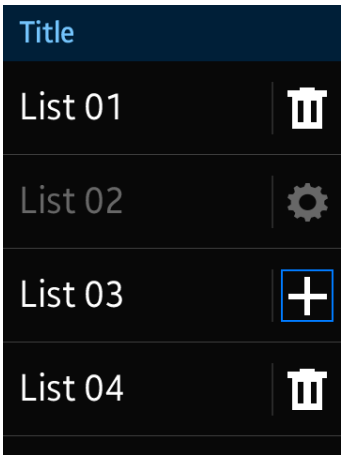
Basic list



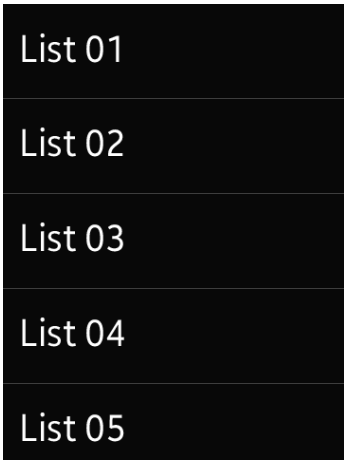
List with sub-text



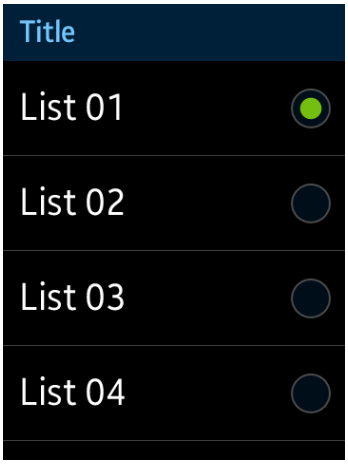
List with action buttons



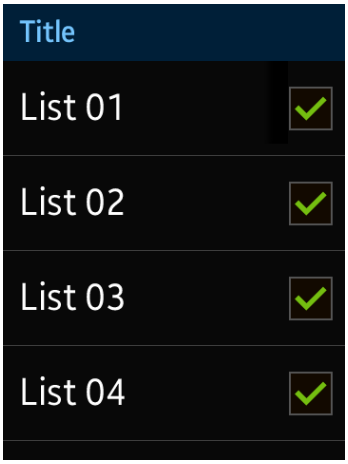
List with more options



List with radio buttons



List with check boxes



Virtual List Views

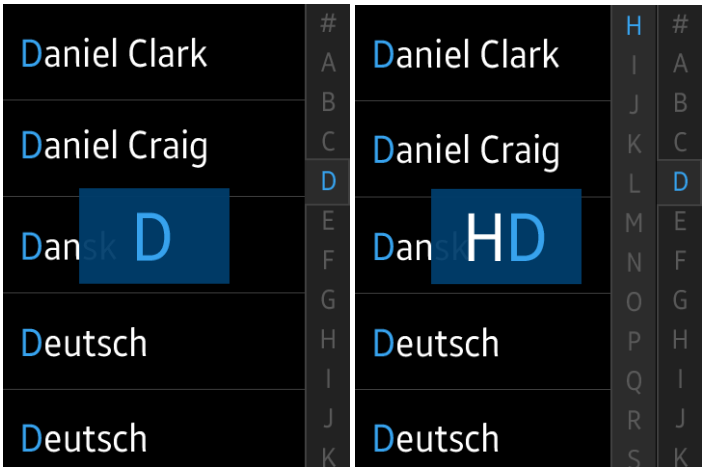
Use the "ui-virtuallistview" style for a very long list that contains a lot of items. The list created with the "ui-virtuallistview" style automatically manages the device memory to optimally display the list items. Refer to the Developer Document for detailed information about virtual lists.

Note If you design a very long list that contains a lot of items without this style, your app's running speed may be impaired.

Index Scroll Bar

When designing a long list of text items, consider including an index scroll bar in the list. The Index scroll bar ("ui-indexscrollbar") displays index characters on the list page, as defined by the designer. The index allows users to jump directly to the list items they want to view.

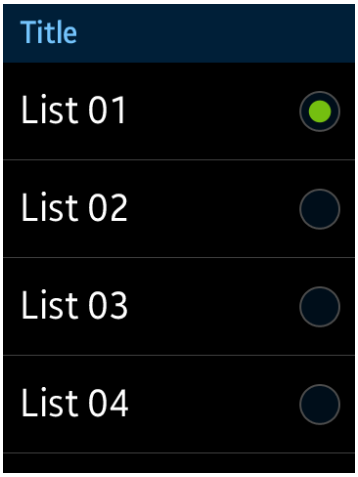
List with indexes



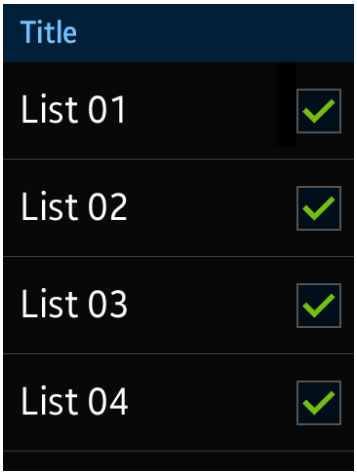
5.2.6 Check Box and Radio Button

Use the [CSS style names for check box]or the [Radio buttons?] style to create pages with radio buttons or check boxes. By default, the Gear's key color schemes are applied to the radio button, check boxes and radio buttons.

Radio buttons



Check boxes



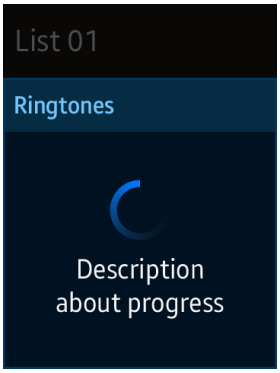
When you design a list of check boxes or radio buttons, make the whole area of each entry tappable, rather than designing only the check boxes or the radio buttons to be tappable. This will provide an improved user experience by making list selections easier for users.
Refer to the **Developer Document** for detailed information about enhancing the usability of these UI elements.

5.2.7 Progress

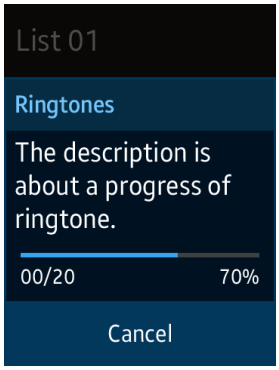
You have three different styles to inform the users that a task is in progress:"ui-progress-indeterminate,""ui-progress," and "ui-processing."The "ui-progress-indeterminate" style shows an indefinite process where the progress rate is unknown, with a rotating circle icon. You can design your app to stop the rotation when a job is done. On the other hand, the "ui-progress" style can be presented with either a percentage bar or a fraction indicator to show the detailed progress of a task.

ui-progress-indeterminate

ui-processing



ui-progress



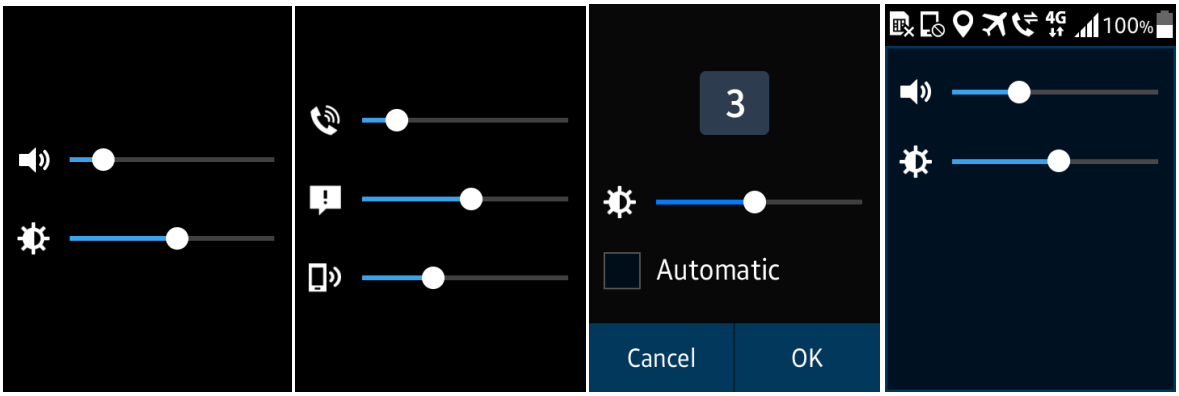
5.2.8 Slider and Toggle switch

Sliders

You can implement sliders in the Gear's style simply by adding the web standard components to your design. The Gear automatically applies the system style to the components to make your designs consistent with the Gear system.

The following are examples of sliders and toggle switches presented in the Gear's system style.

Sliders

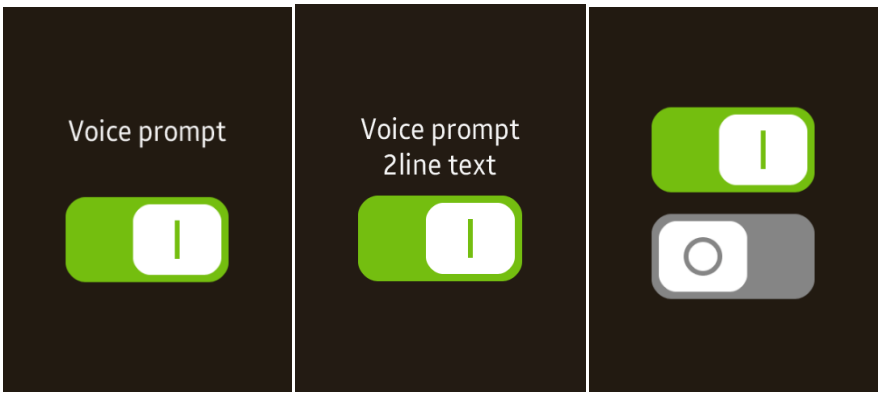


Still, if you want to design your own components, you can apply the predefined CSS styles provided in the Tizen Wearable Web UI framework. Refer to the Developer Documentation for detailed information about customizing sliders.

Toggle Switches

You can design toggle switches for your app to turn a feature on or off. Refer to the Developer Documentationfor detailed information about how to define toggle switches.

Toggle switches



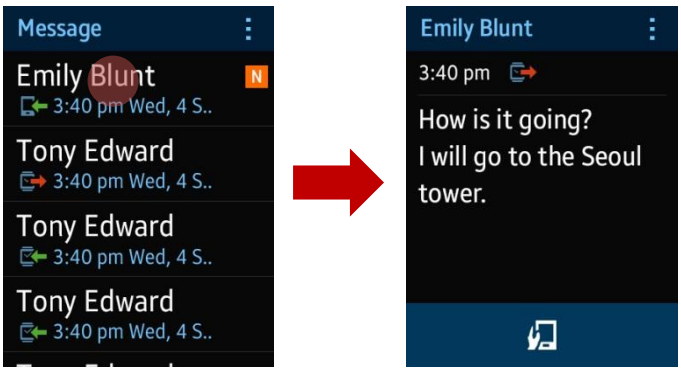
5.3 Design Guidelines

This section provides suggestions to make the most out of the Tizen Wearable Web UI framework that we have covered in the previous section.

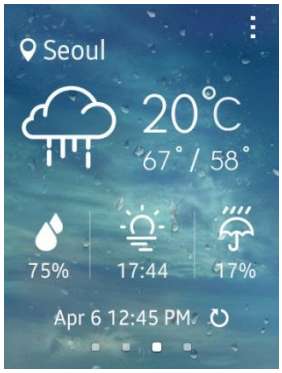
5.3.1 What is a Header for?

A Header element (ui-header) on the Gear may take up 60 pixels (on Gear S) at the topmost part of the display. It provides the users with context. Small and simple apps made up of only a few pages may work just fine without headers. However, you should consider adding headers to your page designs if you are designing apps with more pages and a complex structure. Using such apps, users will be required to move to different pages, and chances are high that they will get lost in your app without headers.

Example of headers: Headers help users understand the context.



Example of a simple app



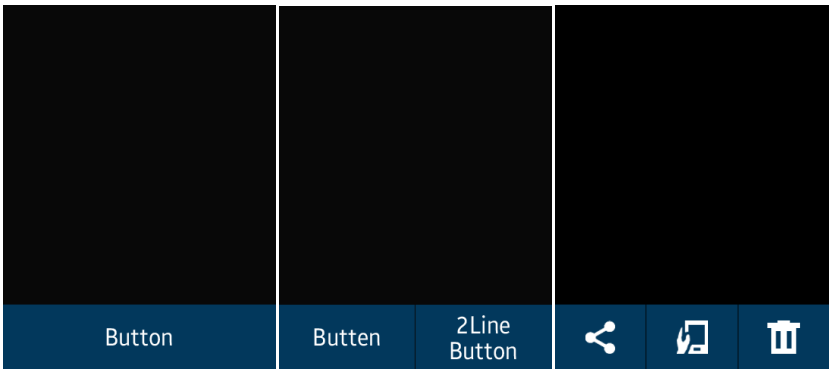
5.3.2 How Many Bottom Buttons are Good on a Page?

The display sizes vary among the Gear devices, and the display specifications will continue to change in future Gear devices. To stay on the safe side, the number of buttons we recommend at the bottom of any page is two. We would advise you against using three buttons at the bottom of a page. However, if you need a three button design for your app, use button icons instead of button texts.

Notes

- In the Gear UI, up to two 10-character strings may be used for a button text, though it is always best to keep them as short as possible.
- Use the button group ("ui-grid-col") style to ensure that your button designs are compatible with various display sizes. These styles can automatically align a maximum of 3 buttons.

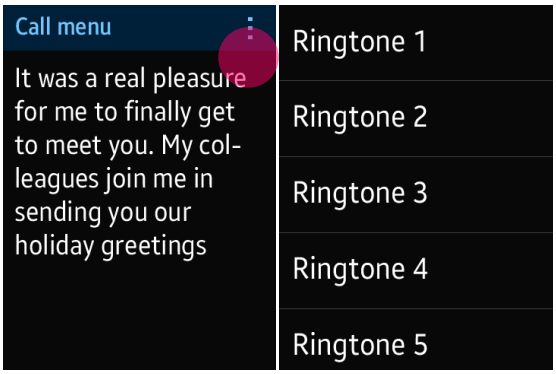
Different numbers of buttons



5.3.3 What is the "More options" Button for?

Considering the small display size, it is natural that having fewer menu options is preferable when designing pages for your wearable app. However, some pages require more than two or three options. You can always add another page, but this can distract the user and they may lose track of the current task.
Consider using a "More options" button on the page to solve this problem. "More options" buttons are widely used across the native Gear apps. They allow you to place multiple options on one page. Users can tap the "More options" button to see a list of available options, keeping the focus on the current task.

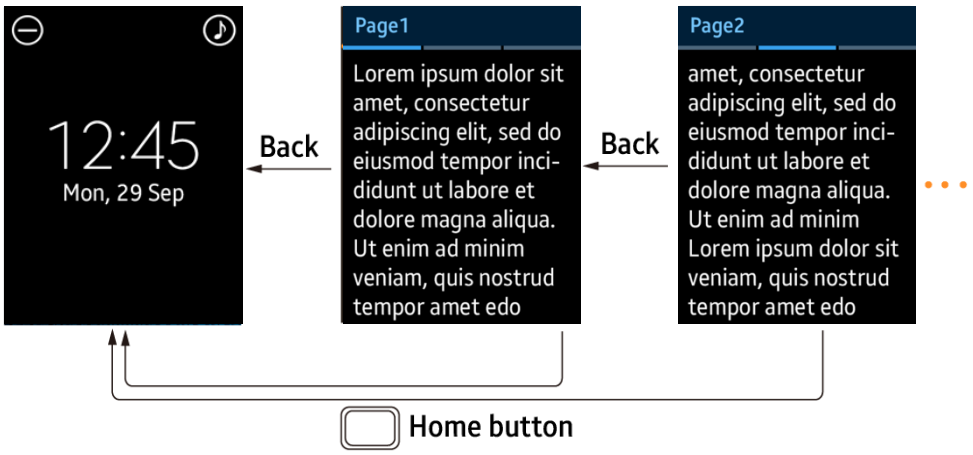
Example of the more options button



5.3.4 How do I Design My App to Close?

When users press the Home button or perform a bezel swipe down (the "back" gesture) from the main page (the first page) of your app, your app should close. If you have many pages inside your app, it will keep returning to the previous page each time the gesture is made. Then it will close the app when another "back" gesture is made on the main page.

Note Your app pages may include several sections (section changer designs). In this case, remember that the "back" gesture is used to display the previous page, and not the previous section. Therefore, performing back gestures on the main page will close the app regardless of the section that is currently being accessed.



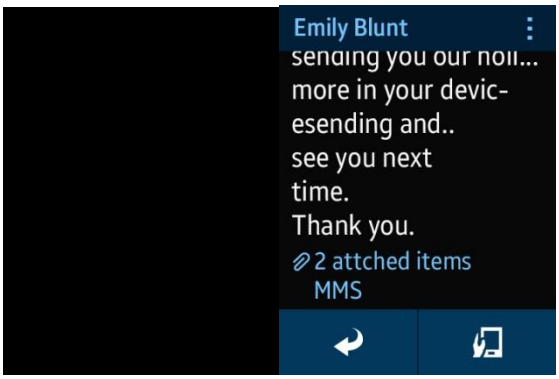
We recommend that you design your Gear app to close without a confirmation pop-up, since asking for confirmation may annoy the users. Also, avoid designing your app to run in the background when an action is performed to close the app, since stable background operation is not guaranteed.

5.3.5 What Do I Need to Know to Create Compatible Designs?

In order to ensure that your app is compatible with other Gear devices, use responsive designs when developing an app. The Gear devices have been developed and marketed in different display sizes and resolutions so far, and such a trend may continue as the relevant technology advances.

1:1 Screen Layout

3:4 Screen Layout



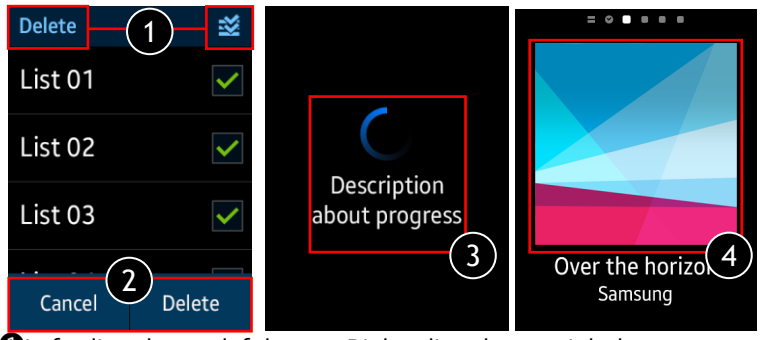
If you use the assets in the [Tizen Wearable SDK](), apps will be composed to match the screen ratio, but the layout may vary according to the screen ratio and size.

The Tizen Wearable Web UI framework elements are designed to shrink or expand according to the screen size. Also, note that the UI styles provided in the framework can be applied to display shapes other than the rectangular designs (circular display, for example).

Consider designing your screen based on a 1:1 ratio layout whenever it is practical. Then, find different ways to utilize the additional space provided by other screen ratios. While a design based on a 1:1 screen ratio screen layout will fit on any 3:4 ratio screen, one designed for a 3:4 screen ratio will not fit on a 1:1 ratio screen, nor can it be adjusted well enough to provide a natural look (if it is adjustable at all).

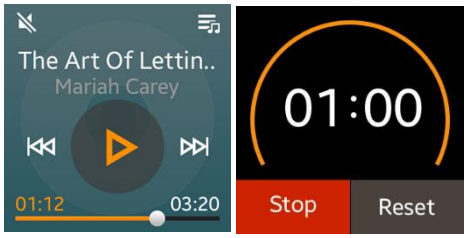
If you decide not to use the provided design elements and to design an independent UI, make sure to follow the basic design principles below.

Basic design principles

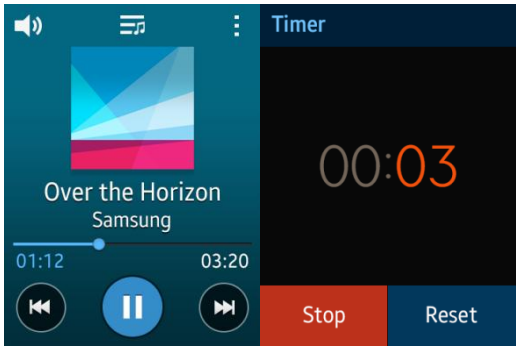


❶Left-align the top left button.Right-align the top right button.
❷Bottom-align the bottom button and fix the width to 100%.
❸Content in the body can be center-aligned, or it can use a different alignment depending on the design.
❹If a background image is used, the width and the type of alignment should be set accordingly.

1:1 Screen Layout



3:4 Screen Layout



If you design an app where the layout cannot be modified (i.e. clock) and therefore cannot utilize the full screen, the app can be displayed as shown below. Set the horizontal width to 100% and leave blanks at the top and the bottom of the screen. The color for the blanks can be configured to match the main display, or separate images may be used to cover up the blanks.

1:1 Screen Layout

3:4 Screen Layout