

[SiriKit / Offering Actions in the Shortcuts App](#)

Article

Offering Actions in the Shortcuts App

Suggest shortcuts users may want to add to Siri or combine with other actions in their own shortcuts.

Overview

When the user performs an action in your app, donate a shortcut that accelerates user access to the action. Siri shows these donated interactions to the user in places such as Spotlight search and the Lock Screen.

However, sometimes there are actions in your app the user hasn't performed that might be of interest to them. For example, perhaps your soup-ordering app features a special soup every day. The user has never ordered the daily soup special, but they might be interested in the option to add a *soup-of-the-day* shortcut to Siri. Your app can provide this option by making a shortcut suggestion.

Note

A donated interaction or suggested shortcut containing a system intent defines a specific action. The user can't customize the parameters of such an action in the Shortcuts app. Instead, define a custom intent. For more information on defining custom intents, see [Adding User Interactivity with Siri Shortcuts and the Shortcuts App](#).

Suggest a Shortcut

To suggest a shortcut to an action that the user hasn't performed but may want to add to Siri, create an `INShortcutReference` object with either an `INIIntent` or `NSUserActivity` object that defines the action. Then add the shortcut to an array. Repeat for each suggestion your app wants to make. After creating the list of shortcut suggestions, call `setShortcutSuggestions(_:_:)`, passing in the shortcuts.

```
import Intents

// Add a user activity to the list of suggestions.
var suggestions = [INShortcut(userActivity: orderFavoriteBeverageUserActivity)]

// Add an intent to the list of suggestions. To create
// a shortcut from an intent, the intent must be valid.
if let shortcut = INShortcut(intent: orderSoupOfTheDayIntent) {
    suggestions.append(shortcut)
}

// Suggest the shortcuts.
INVoiceShortcutCenter.shared.setShortcutSuggestions(suggestions)
```

Update Suggestions

Provide shortcut suggestions that represent actions that pertain to the user. This list may change over time for reasons such as:

- The addition or removal of features from your app.
- A change in the way the user interacts with your app.

To update the shortcut suggestion list, replace the existing list by calling [setShortcutSuggestions\(_ :\)](#) and passing in a new list of suggestions. If you want to remove all suggestions made by your app, call the same method, passing in an empty array.

Changes to the list of shortcut suggestions don't affect shortcuts the user adds to Siri. For instance, if the user adds the suggested *order favorite beverage* shortcut to Siri and the app removes the suggestion from the list some time later, that shortcut is still available to the user.

See Also

Articles

-  [Adding User Interactivity with Siri Shortcuts and the Shortcuts App](#)
Add custom intents and parameters to help users interact more quickly and effectively with Siri and the Shortcuts app.
-  [Defining Relevant Shortcuts for the Siri Watch Face](#)
Inform Siri when your app's shortcuts may be useful to the user.

- 📄 Deleting Donated Shortcuts

Remove your donations from Siri.
- 📄 Dispatching intents to handlers

Provide SiriKit with an intent handler capable of handling a specific intent.
- 📄 Improving Siri Media Interactions and App Selection

Fine-tune voice controls and improve Siri Suggestions by sharing app capabilities, customized names, and listening habits with the system.
- 📄 Improving interactions between Siri and your messaging app

Donate app-specific content, use Siri's contact suggestions, and adopt the latest platform features to create a more consistent messaging experience.
- ☰ Registering Custom Vocabulary with SiriKit

Register your app's custom terminology, and provide sample phrases for how to use your app with Siri.
- 📄 Confirming the Details of an Intent

Perform final validation of the intent parameters and verify that your services are ready to fulfill the intent.
- 📄 Handling an Intent

Fulfill the intent and provide feedback to SiriKit about what you did.
- 📄 Resolving the Parameters of an Intent

Validate the parameters of an intent and make sure that you have the information you need to continue.
- 📄 Generating a List of Ride Options

Generate ride options for Maps to display to the user.
- ☰ Handling the Ride-Booking Intents

Support the different intent-handling sequences for booking rides with Shortcuts or Maps.
- 📄 Donating Reservations

Inform Siri of reservations made from your app.
- 📄 Specifying Synonyms for Your App Name

Provide alternative names for your app that are more familiar or easier for users to speak.
- 📄 Intent Phrases

The keys that you include in your global vocabulary file to show how users engage your app from Siri.