Article

Donating Reservations

Inform Siri of reservations made from your app.

Overview

Siri Event Suggestions integrates reservations made in your app seamlessly with the system, sharing the information with certain apps and features, such as Calendar, Maps, Spotlight, and the Lock Screen. These apps and features can use the information to remind the user to check-in for a flight, get directions to a hotel reservation, return a rental car, or any of several other options. For a complete list of supported reservation types, such as flight, hotel, and car rental, see Siri Event Suggestions. Siri learns about reservations made from your app from donations that your app makes to Siri.

Donate Reservations at the Right Time

Donate a reservation each time the user views a reservation in your app. For instance, when a user creates a new reservation, donate the reservation after the user views and confirms the information. Make one, and only one, donation per reservation as the user views the reservation. If the user views the reservation again using your app, make another donation. For example, if the user makes a reservation through your website and then views the reservation in your app, make a donation. However, don't make more than one donation for an action at the time the user performs the action.

Donate an Interaction

Your app donates an `INInteraction` object that contains information about the reservation using the `INGetReservationDetailsIntent` and `INGetReservationDetailsIntentResponse` classes, along with the relevant `INReservation` subclass.

To donate the intent, create an instance of the intent and intent response classes and set the parameter values. To recommend a voice phrase that the user may want to add to Siri, set the intent's `suggestedInvocationPhrase` property to a string containing the phrase.

After creating the intent, create an instance of `INInteraction` and call its `donate(completion:)` method, passing in the intent. Passing the intent tells Siri about the reservation.

Handle User Interaction

Apps offering reservations with check-in capabilities can integrate with Shortcuts Suggestions on the Lock Screen and allow users to launch your app to complete the check-in. Create the appropriate reservation class with an array of `INReservationAction` objects that contain the user activity and relevant time frame. Siri provides users with the ability to launch your app with the reservation and check-in information. For example, a flight reservation app may provide users the ability to check-in for a flight starting 24 hours before the departure time.

Siri may launch your app with an `NSUserActivity` that contains an `INInteraction` object after a user interacts with a previously donated reservation through Calendar or Spotlight. Similarly, Siri may launch your app with the previously donated `NSUserActivity` provided in the `INReservationAction` object when the user interacts with a check-in Shortcuts Suggestion on the Lock Screen. Your app uses the interaction information to configure itself and display information related to the specified reservation. For example, your app displays the user's flight information and provides a check-in button for their next flight.

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.

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.

Localizing Your Vocabulary for Chinese Dialects

Apply emphasis markers to your pronunciation tips to assist Siri with Chinese dialects.