

## Tap to Pay on iPhone Implementation - Stripe User Guide

Tap to Pay lets you use an iOS device as a terminal reader. However, implementing a Tap to Pay application is a complex process involving multiple steps that are documented in different places. This guide collects information from the Stripe and Apple documentation to help you build and publish your app.

The following sites provide further details:

- [Tap to Pay \(Stripe documentation\)](#)
- [Tap to Pay on iPhone or Android and Stripe Terminal \(Stripe Support article\)](#)
- [Setting up Tap to Pay on iPhone \(Apple Developer Documentation\)](#)

### Support

For Stripe-related questions, visit our [support site](#).

For Apple-related questions, contact [tpointitlements@apple.com](mailto:tpointitlements@apple.com), and include your case ID if you have one.

### Apple requirements and the entitlement process

For your app to use Tap to Pay on iPhone, you need to request two different entitlements from Apple: a **development entitlement** and a **publishing entitlement**. Follow these steps:

1. Request and receive a development entitlement.
2. Design and build the app to meet Apple's requirements. (Note that for direct users, some requirements don't apply, or only partially apply.)
3. After completing the initial review process, request and receive a publishing entitlement.
4. Submit your app to the App Store through the Apple Developer Program.
5. Publish the app for general availability.
6. Market your app using the [Tap to Pay on iPhone Marketing Guide and Toolkit](#).

### Timeline

The development entitlement is typically auto-approved in most regions, with a turnaround time of one to two business days.

The approval process for publishing entitlements can vary based on how well the app matches Apple's requirements. You can normally expect a response from Apple's operations team within two weeks of your initial publishing entitlement request. In most cases, the approval process involves one or two rounds of feedback with Apple.

If you experience delays exceeding two weeks, you can reach out to Stripe's support team.

### Requirements checklist

**Note:** Certain requirements might not apply, or might only partially apply, to direct users due to the variations in app distribution between Connect platforms and individual users. Make sure to communicate to Apple whether your app is for a direct user or for connected accounts when you request the publishing entitlement.

#	Description	Support	Status
1	Learn how to accept contactless payments on a compatible iPhone or Android device.	<a href="#">Tap to Pay   Stripe Documentation</a>	<input type="checkbox"/>
2	Set up an Apple Developer account.	<a href="#">Become a member   Apple Developer Program</a>	<input type="checkbox"/>
3	Request Apple development entitlement.	<p>The development entitlement allows you to build and test your Tap to Pay on iPhone application.</p> <p>To begin the process, visit the following link:</p> <p><a href="#">Setting up Tap to Pay on iPhone   Apple Developer Documentation</a></p> <p>Log in to your Apple Developer account and complete the application form. Specify Stripe as your payment service provider, and specify the regions where you plan to deploy your app.</p>	<input type="checkbox"/>
4	Design and build your app to meet the requirements. Use the Stripe Terminal iOS SDK.	<a href="#">Set up your integration   Stripe Documentation</a>	<input type="checkbox"/>
4.1	<b>General requirements</b>		<input type="checkbox"/>
4.1.1	<b>Required:</b> Your app offers Tap to Pay on iPhone to compatible devices only (iPhone XS and later).	<a href="#">SCPTerminal Class Reference</a>	<input type="checkbox"/>
4.1.2	<b>Required:</b> If Tap to Pay on iPhone is the main feature of your app, you must set the iOS deployment target in Xcode to the minimum version supporting your Tap to Pay on iPhone configuration. Confirm the minimum version with Stripe's SDK documentation.	<a href="#">Configuring a new target in your project   Apple Developer Documentation</a> <a href="#">Stripe's latest SDK version</a>	<input type="checkbox"/>
4.1.3	<b>Required:</b> For iOS versions prior to 17.6, your app handles the <code>PaymentCardReaderError.osVersionNotSupported</code> error by displaying a message to inform the user that they must update to the latest version of iOS.	This requirement is automatically met through the use of Stripe's SDK.	<input type="checkbox"/>
4.1.4	<b>Required:</b> Your app triggers the Tap to Pay reader connection at launch or when your app comes into the	If you encounter challenges in meeting this requirement, we recommend that you contact	<input type="checkbox"/>

	foreground.	<p>Apple for assistance during the entitlement process.</p> <p><a href="#">Tap to Pay   Stripe Documentation</a></p> <p><a href="#">Connect to a reader   Stripe Documentation</a></p>	
4.1.5	<b>Conditional:</b> If your app supports Tap to Pay on iPhone as the sole payment acceptance method, ensure that only compatible devices can download the app by including <code>iphone-ipad-minimum-performance-a12</code> in the <code>UIRequiredDeviceCapabilities</code> information property.	<a href="#">UIRequiredDeviceCapabilities   Apple Developer Documentation</a>	<input type="checkbox"/>
4.1.6	<b>Required:</b> Retrieve the status of a merchant's acceptance of the Tap to Pay on iPhone Terms and Conditions from Apple, not from a local variable in your app.	Stripe doesn't provide native support for this. However, you can use Stripe's <a href="#">API</a> to configure whether to display the Terms and Conditions when connecting.	<input type="checkbox"/>
4.1.7	<b>Recommended:</b> Apps on the public App Store should leverage Face ID or Touch ID at checkout.	<a href="#">Logging a User into Your App with Face ID or Touch ID   Apple Developer Documentation</a>	<input type="checkbox"/>
4.1.8	<b>Conditional:</b> Apps on the public App Store must adhere to the <a href="#">Human Interface Guidelines</a> .	<a href="#">Tap to Pay on iPhone   Apple Developer Documentation</a>	<input type="checkbox"/>
4.1.9	<b>Conditional:</b> Apps on the public App Store must adhere to the Tap to Pay on iPhone Developer Marketing Guidelines and the <a href="#">Tap to Pay on iPhone Marketing Guide and Toolkit</a> .	<p><a href="#">Marketing guidelines for developers   Tap to Pay on iPhone</a></p> <p><a href="#">Apple marketing tools</a></p>	<input type="checkbox"/>
4.2	<b>Connected account onboarding requirements</b>	<b>New user flow recording required</b>	<input type="checkbox"/>
4.2.1	<b>Required:</b> The process to create an account or gain access to Tap to Pay on iPhone is easily discoverable to a new user who has just downloaded your app.	Apps must provide clear onboarding and in-app guidance, including a simple signup process, a prominent "Get started" or "Enable Tap to Pay" button, and concise instructions within the app itself. Additionally, visual cues like icons and animations can help guide users.	<input type="checkbox"/>
4.2.2	<b>Required:</b> The process to create an account or gain access to Tap to Pay on iPhone triggers from within your app and can be fully completed in an iPhone web browser.	<p>We recommend using Connect embedded components to onboard accounts: <a href="#">Get started with Connect embedded components   Stripe Documentation</a>.</p> <p>Users who leverage <a href="#">unlisted apps</a>, <a href="#">custom apps</a>, or the <a href="#">Apple Developer Enterprise</a></p>	<input type="checkbox"/>

		<a href="#">Program</a> (ADEP) are exempt from this requirement.	
4.2.3	<p><b>Required:</b> Digital onboarding takes less than 15 minutes* for most of your users, from information submission (or access request for existing businesses) to enabling payment with Tap to Pay on iPhone.</p> <p>*In compliance with local merchant onboarding regulations.</p>	<p>For connected accounts, while users can complete the application in approximately 15 minutes, payments might not be enabled until Stripe verifies their identity. That can require uploading IDs or additional business information.</p> <p>Users who leverage <a href="#">unlisted apps</a>, <a href="#">custom apps</a>, or the <a href="#">Apple Developer Enterprise Program</a> (ADEP) are exempt from this requirement.</p>	<input type="checkbox"/>
4.3	<b>Tap to Pay enablement requirements</b>	<b>New user and existing user flow recording</b>	<input type="checkbox"/>
4.3.1	<b>Required:</b> Your app implements a highly visible and easily discoverable communication for Tap to Pay on iPhone.	To make sure that customers easily recognize the Tap to Pay feature on iPhone, the app should prominently display an easily recognizable icon or visual cues indicating the availability of this payment method.	<input type="checkbox"/>
4.3.2	<b>Recommended:</b> Your app implements a full-screen modal (splash screen) for Tap to Pay on iPhone. (This also meets requirement 4.6.2.)		<input type="checkbox"/>
4.3.3	<b>Required:</b> Your app displays communications for Tap to Pay on iPhone to all eligible users at least once. This can be accomplished with a push notification. (This also meets requirement 4.6.3.)		<input type="checkbox"/>
4.3.4	<b>Required:</b> Your app displays an obvious way to enable Tap to Pay on iPhone at the end of every new merchant onboarding flow.	This ensures that businesses can quickly activate this feature when they join your platform.	<input type="checkbox"/>
4.3.5	<b>Required:</b> Your app provides communications for Tap to Pay on iPhone and includes a clear action to trigger acceptance of Tap to Pay on iPhone Terms and Conditions.	To comply with Tap to Pay on iPhone regulations, your communications must present the terms and conditions in a way that requires a clear and unambiguous user action to indicate their acceptance.	<input type="checkbox"/>
4.3.6	<b>Required:</b> Your app provides a way to enable Tap to Pay on iPhone outside of the communications and the checkout flow (for example, in your app settings).	See Apple's documentation for examples: <a href="#">Tap to Pay on iPhone   Apple Developer Documentation</a>	<input type="checkbox"/>
4.3.7	<b>Required:</b> Your app provides a trigger to enable Tap to Pay on iPhone within your checkout flow or requires Tap to Pay on iPhone enablement prior to the checkout flow.		<input type="checkbox"/>

4.3.8	<b>Required:</b> Tap to Pay on iPhone Terms and Conditions must only be accepted by an administrator user or otherwise authorized party for the merchant. If you only provide one user type for the merchant, ensure that the correct user is accepting Terms and Conditions on behalf of the merchant.		<input type="checkbox"/>
4.3.9	<b>Required:</b> If an unauthorized user attempts to enable Tap to Pay on iPhone, a message instructs them to contact their administrator.		<input type="checkbox"/>
4.3.10	<b>Conditional on deployment type:</b> For enterprise deployments (custom app, unlisted app, or Apple Developer Enterprise Program), you must support the ability to accept Terms and Conditions outside of the app using Apple Business Connect.	<p>Stripe's API enables platforms to implement a feature on their websites that redirects users to a custom Apple URL, allowing them to accept the terms and conditions.</p> <p><a href="#">Connect to a reader   Stripe Documentation</a></p>	<input type="checkbox"/>
4.3.11	<b>Recommended:</b> After the user has accepted the Terms and Conditions and viewed the education on how to use Tap to Pay on iPhone, your app presents a dedicated screen prompting the user to try Tap to Pay on iPhone. This screen also displays the progress of Tap to Pay on iPhone readiness as the device is being configured.	<p>See Apple's website for examples:</p> <p><a href="#">Tap to Pay on iPhone   Apple Developer Documentation</a></p> <p>Stripe doesn't natively support this screen. However, you can test Tap to Pay on iPhone using Stripe's test mode and test cards: <a href="#">Test Stripe Terminal</a>.</p>	<input type="checkbox"/>
4.3.12	<b>Required:</b> Your app provides a configuration progress indicator using <code>PaymentCardReader.Event.updateProgress(_ :)</code> (or the equivalent call from Stripe's SDK). This screen displays the progress of Tap to Pay on iPhone readiness as the device is being configured. This screen informs the user whether Tap to Pay on iPhone is ready for use, both during initial setup and during typical usage while the reader is preparing.	<a href="#">SCPTapToPayReaderDelegate Protocol Reference</a>	<input type="checkbox"/>
4.4	<b>Merchant education requirements</b>	<b>New user and existing user flow recording</b>	<input type="checkbox"/>
4.4.1	<b>Recommended:</b> If your app targets iOS 18 or later, use <code>ProximityReaderDiscovery</code> to provide merchant education.	With iOS 18 and later, the <a href="#">ProximityReaderDiscovery API</a> allows you to surface merchant education for Tap to Pay on iPhone with just a few lines of code. Apple	<input type="checkbox"/>

4.4.2	<b>Required:</b> Your app presents education screens after the user accepts Terms and Conditions.	ensures that the content is up to date and localized for your merchant's region.	<input type="checkbox"/>
4.4.3	<b>Required:</b> Your app makes merchant education screens accessible for reference in either the Settings or Help section.	<p>For businesses on older versions of iOS, use the Apple-approved assets and copy available in the <a href="#">Tap to Pay on iPhone Marketing Guide and Toolkit</a> to create the merchant education experience for your app.</p> <p>If you distribute your app with programs such as <a href="#">unlisted apps</a>, <a href="#">custom apps</a>, or the <a href="#">Apple Developer Enterprise Program (ADEP)</a>, the education requirements are strongly recommended, but not required if you provide merchant education through other channels.</p>	<input type="checkbox"/>
4.4.4	<b>Required:</b> If you don't use ProximityReaderDiscovery, use the Apple Tap to Pay on iPhone Marketing Guide and Toolkit for your merchant education screens.		<input type="checkbox"/>
4.4.5	<b>Required:</b> Merchant education demonstrates how to accept contactless cards with Tap to Pay on iPhone.		<input type="checkbox"/>
4.4.6	<b>Required:</b> Merchant education demonstrates how to accept Apple Pay and other digital wallets with Tap to Pay on iPhone.		<input type="checkbox"/>
4.4.7	<b>Conditional on region:</b> If the deployment region supports PIN entry, merchant education must demonstrate support for it, with explicit mention of available accessibility options on the PIN screen. See the <a href="#">Regional requirements section</a> for more details.	<p>Apple's API in iOS 18 enables relevant educational materials.</p> <p>A customer might be prompted to enter their card PIN in certain circumstances. In regions where PINs are common during checkout, your merchant education must provide PIN entry and accessibility details. Refer to the <a href="#">Tap to Pay on iPhone Marketing Guide and Toolkit</a> for Apple-approved in-app PIN assets and copy.</p>	<input type="checkbox"/>
4.4.8	<b>Conditional on region:</b> If the deployment region requires a fallback payment method, your merchant education must demonstrate fallback to an alternate payment method when Tap to Pay on iPhone is unable to read a payment card.	<p>In regions where a significant number of cards cannot be used with contactless under specific circumstances (such as offline PIN markets), the app must support at least one accepted alternative payment method and fulfill the following requirements:</p> <ul style="list-style-type: none"> <li>- All businesses must be set up with at least one accepted alternative payment method, so that all transactions can be completed in person.</li> <li>- We recommend either using Stripe Terminal hardware, which uses the same SDK and integration as Tap to Pay on iPhone, or integrating <a href="#">Payment Links</a>.</li> </ul>	<input type="checkbox"/>

		<ul style="list-style-type: none"> <li>- Ensure that the app provides a seamless transition between the Tap to Pay on iPhone UI and the alternative payment method.</li> <li>- Provide details on the fallback mechanism as part of your merchant education.</li> <li>- If you create your own merchant education screens, use the following copy about fallback payment methods: <i>“Some cards can’t complete contactless transactions using a PIN. If this occurs, ask the customer if they have an alternative contactless card or digital wallet, and continue the transaction using Tap to Pay on iPhone.”</i></li> </ul>	
4.5	<b>Checkout requirements</b>	<b>Checkout flow recording</b>	<input type="checkbox"/>
4.5.1	<b>Conditional on deployment type:</b> The app displays an obvious, prominent button to invoke a Tap to Pay on iPhone payment during checkout.	<p>These elements are required for apps published to the public App Store, but only recommended for <a href="#">Unlisted apps</a>, <a href="#">custom apps</a>, or the <a href="#">Apple Developer Enterprise Program</a>.</p> <p>See Apple’s <a href="#">Human Interface Guidelines (HIG)</a> and the <a href="#">Regional requirements section</a> for details on how to display the Tap to Pay on iPhone button in your app.</p>	<input type="checkbox"/>
4.5.2	<b>Conditional on deployment type:</b> The button that invokes Tap to Pay on iPhone is visible without scrolling at the time of checkout. If there are multiple payment options, Tap to Pay on iPhone must be at the top of the list.		<input type="checkbox"/>
4.5.3	<b>Conditional on deployment type:</b> The button that invokes Tap to Pay on iPhone must never be altered, grayed out, or otherwise hidden based on whether the user has enabled Tap to Pay on iPhone. If the user hasn’t enabled it, the button must invoke the enablement process.		<input type="checkbox"/>
4.5.4	<b>Conditional on region:</b> If there are multiple payment options, the button that invokes Tap to Pay on iPhone must use the proper copy for your region.	See the <a href="#">Regional requirements section</a> for details.	<input type="checkbox"/>
4.5.5	<b>Conditional on button style:</b> If the button uses iconography, it must use <code>wave.3.right.circle</code> or <code>wave.3.right.circle.fill</code> from Apple’s SF Symbols library.	<a href="#">SF Symbols   Apple Developer</a>	<input type="checkbox"/>

4.5.6	<b>Required:</b> When the Tap to Pay on iPhone screen closes after a successful read of the card, a Processing screen must indicate that the transaction is being processed.	Stripe's SDK supports these requirements: <a href="#">Tap to Pay on iPhone   Apple Developer Documentation</a>	<input type="checkbox"/>
4.5.7	<b>Required:</b> A clear communication of the outcome (such as approved, declined, timed out) follows the Processing screen.		<input type="checkbox"/>
4.5.8	<b>Required:</b> Regardless of a transaction's outcome, it must be possible to confidentially send a digital receipt to the customer. For example, the digital receipt can be sent by SMS, email, QR code, or using activity views.	<a href="#">Activity views   Apple Developer Documentation</a>  <a href="#">Provide receipts   Stripe Documentation</a>	<input type="checkbox"/>
4.5.9	<b>Conditional on region:</b> Comply with regional requirements based on where you deploy your app.	See the <a href="#">Regional requirements section</a> for details.	<input type="checkbox"/>
4.5.10	<b>Conditional:</b> If the user is not authorized to accept Tap to Pay on iPhone Terms and Conditions, display a message telling them to contact an administrator to enable Tap to Pay on iPhone.	See also 4.3.9.	<input type="checkbox"/>
4.6	<b>Marketing requirements</b>		<input type="checkbox"/>
4.6.1	<b>Required:</b> At launch, send a dedicated launch email to all eligible users.	This must leverage the “launch” email referred to in the <a href="#">Tap to Pay on iPhone Marketing Guide</a> .	<input type="checkbox"/>
4.6.2	<b>Required:</b> At launch, display an in-app splash screen to all eligible users at least once.	This must leverage the “hero” in-app banner referred to in the <a href="#">Tap to Pay on iPhone Marketing Guide</a> .	<input type="checkbox"/>
4.6.3	<b>Required:</b> At launch, deploy an in-app push notification to all eligible users.	This must leverage the “value proposition” copy from the “push notification” copy guidelines in the <a href="#">Tap to Pay on iPhone Marketing Guide</a> .	<input type="checkbox"/>
4.7	<b>Details for user flow recordings</b>		<input type="checkbox"/>
4.7.1	<b>Required:</b> Provide a recording of the “new user flow.”	For all recordings, use a video recording from another iPhone. If that isn't possible, you can use a screen recording from the same device.	<input type="checkbox"/>
4.7.2	<b>Required:</b> Provide a recording of the “existing user flow.”		<input type="checkbox"/>
4.7.3	<b>Required:</b> Provide a recording of the “checkout flow.”		<input type="checkbox"/>
5	Submit your app to Apple for review, and request a	Request the publishing entitlement by	<input type="checkbox"/>



	publishing entitlement.	replying to the email you received from <a href="mailto:tppointitlements@apple.com">tppointitlements@apple.com</a> .  Because the publishing entitlement is required to use <a href="#">TestFlight</a> , your app must meet the in-app user experience requirements before you can distribute it outside of your development teams.	
6	App review	After you submit your app, Apple reviews it to ensure that it fulfills all necessary requirements.	<input type="checkbox"/>

## Recording information

Record a demonstration of each of the required user flows. If possible, create a video using a second device. Otherwise, record a screen capture on the same device.

### New user flow

- Demonstrate the path for new merchant onboarding, including account creation and Know Your Customer (KYC), where applicable.
- Once a merchant account is approved, demonstrate how Tap to Pay on iPhone is introduced to the new merchant in this path.
- Demonstrate how a user will accept Terms and Conditions for Tap to Pay on iPhone.
  - If Terms and Conditions are accepted outside of the app, provide details of the process.
  - If the Terms and Conditions don't appear because your account has already accepted them, unlink your account so you can repeat the process in your recording. For help, see [Apple's support page](#).
- Demonstrate the merchant education provided after accepting Terms and Conditions.
  - If Terms and Conditions are accepted outside of the app, provide details of how merchant education will be shared within the app.
- Demonstrate where a user can find merchant education at a later time; this is typically accomplished with a section for Tap to Pay on iPhone in the settings of your app.
- Demonstrate that a progress indicator is displayed while Tap to Pay on iPhone is being configured.
- Demonstrate the completion of the configuration.
- If your app does not provide an in-app onboarding path for new users, provide details of how new users onboard to become businesses.
- For applicable regions, you must demonstrate a PIN transaction and a fallback payment method transaction. See the conditional requirements for more details.

## Existing user flow

- Before accepting Terms and Conditions, demonstrate that the Tap to Pay on iPhone payment acceptance button exists and is visible to existing users.
- Demonstrate where an existing user will become aware that Tap to Pay on iPhone is available.
  - Typically, this is accomplished with a full-screen modal after the existing user has logged in or launched the updated app.
- Demonstrate how a user will accept Terms and Conditions for Tap to Pay on iPhone.
  - If Terms and Conditions are accepted outside of the app, provide details of the process.
  - If the Terms and Conditions don't appear because your account has already accepted them, unlink your account so you can repeat the process in your recording. For help, see [Apple's support page](#).
- Demonstrate the merchant education provided after accepting Terms and Conditions.
  - If Terms and Conditions are accepted outside of the app, provide details of how merchant education will be shared within the app.
- Demonstrate where a user can find merchant education at a later time; this is typically accomplished with a Tap to Pay on iPhone section in the settings of your app.
- If the user completes or skips merchant education before configuration is complete, demonstrate that a progress indicator informs the user of the configuration status.
- For applicable regions, you must demonstrate a PIN transaction and a fallback payment method transaction. See the conditional requirements for more details.

## Checkout flow

- Demonstrate adding items to a cart or entering a currency amount for payment.
- Demonstrate the payment options.
- Demonstrate the Tap to Pay on iPhone button.
- Initiate and complete a Tap to Pay on iPhone transaction.
- Demonstrate a PIN (in regions where applicable—see [Regional requirements](#) for more details).
- Demonstrate fallback payment mechanisms (in regions where applicable—see [Regional requirements](#) for more details).

## Regional requirements

### Localization strings for Tap to Pay on iPhone

Region	Long Form	Short Form
US, UK, AUS, CAN (EN), IE, UAE (EN)	Tap to Pay on iPhone	Tap to Pay
CAN (FR)	Paieement rapide sur iPhone	-
FR, CH (FR)	Tap to Pay sur iPhone	Tap to Pay
DE, AT, CH, LI	Tap to Pay auf dem iPhone	Tap to Pay
NL	Tap to Pay op iPhone	Tap to Pay
BR, PT	Tap to Pay no iPhone	Tap to Pay
US (ES)	Pagos sin contacto en iPhone	-
ES, CL	Tap to Pay en iPhone	Tap to Pay
TW	iPhone卡緊收	-
JP	iPhoneのタッチ決済	-
UA, BG	Tap to Pay на iPhone	Tap to Pay
UAE (Arabic)	iPhone مرر للدفع على	-
RO	Tap to Pay pe iPhone	Tap to Pay
SE	Tap to Pay på iPhone	Tap to Pay
CZ, SI	Tap to Pay na iPhonu	Tap to Pay
PO	Tap to Pay pe iPhone	Tap to Pay
HU	Tap to Pay az iPhone-on	Tap to Pay
FI	iPhonen Tap to Pay	Tap to Pay
SK	Tap to Pay na iPhone	Tap to Pay

### Local requirements

Requirement	Region
PIN Entry in Education	All regions except JP, TW
Fallback Payment Method	UK, IE, CAN
Surcharging	BR, AU
IFR	FR, DE

## Cartes Bancaires connected account requirements

Depending on your integration, you might need to [meet additional requirements](#) to enable Cartes Bancaires for your connected accounts. French regulations require that the payment process give customers the option to choose their [card brand](#) before initiating the tap transaction.

Neither Stripe nor Apple offers prebuilt user interfaces for implementing card brand choice, so you must build this functionality into your own application. If you choose to support card brand selection, make sure to set the appropriate [requested\\_priority](#) on the PaymentIntent.

## Disclaimer

While this document aims to assist users with the application process, meeting Apple's requirements is ultimately the user's responsibility, and the final decision to approve or decline an application belongs to Apple.