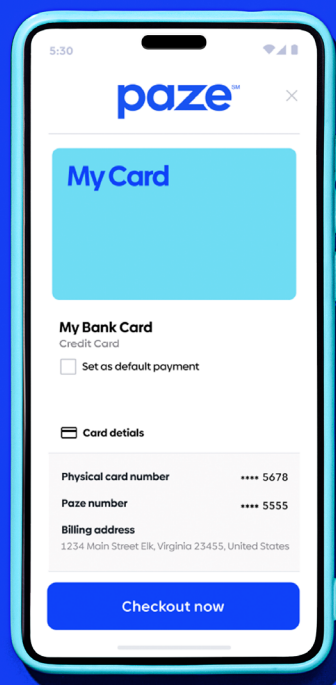




# Paze<sup>SM</sup> Merchant SDK Guide: Simplifying Checkout for E-Commerce

**Streamline Checkout  
with Our Easy-to-Use SDK**



# Contents

1. What is Paze <sup>SM</sup> ? .....	3
2. Overview of the Paze Merchant SDK.....	4
3. Prerequisites for the Paze Merchant SDK .....	6
4. Getting Started with the Paze SDK.....	7
5. Understanding Paze APIs and SDK Interactions.....	8
6. Paze Merchant SDK Backward Compatibility Considerations ....	10
7. Paze Merchant SDK Responses .....	11
8. Get Connected with Paze .....	12

# 1 What is Paze<sup>SM</sup>?

Paze is a first-of-its kind digital wallet offered by some of the nation's largest financial institutions – available with more than 144 million eligible consumer credit and debit cards.



## Bank Provided

All eligible cards appear in one place,  
ready to use at checkout



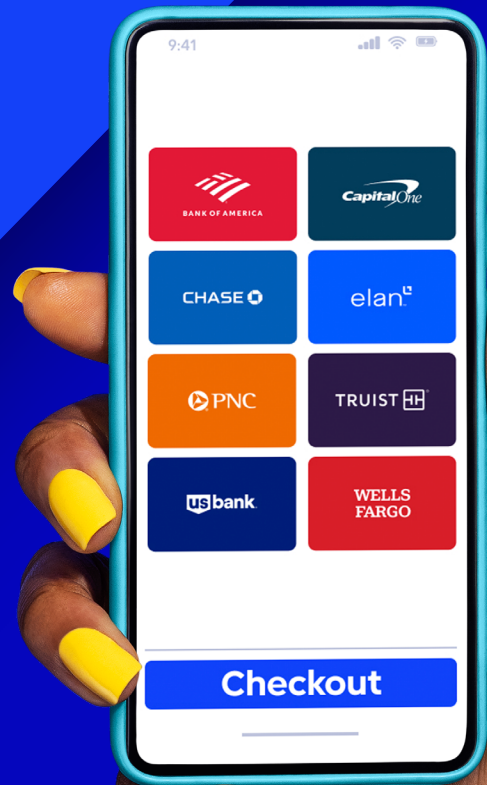
## Elevated Convenience

Card numbers stay up-to-date  
automatically



## Added Security

Network tokenization that keeps  
consumer card numbers private



# 2 Overview of the Paze<sup>SM</sup> Merchant SDK

The **Paze<sup>SM</sup> Merchant SDK** empowers merchants to provide customers with a fast, secure checkout. By using the SDK, e-commerce sites can simplify payment collection, support tokenized data, and offer a user-friendly wallet interface to streamline transactions.

## Key Features



**Quick Integration** - Set up the Paze SDK to display a branded wallet interface that integrates seamlessly into your site, without complex backend modifications.



**Secure Checkout** - Through tokenization, Paze ensures that sensitive data remains secure as full card details are not shared, helping to reduce fraud risk.



**Enhanced User Experience** - Paze adapts to eligible users automatically, offering them a fast checkout with their saved information.



Consumers use Paze by clicking on the Paze checkout button on a merchant's site. From there, a pop-up window appears to complete the payment portion of the e-commerce transaction. Some merchants may require account setup to make purchases. There is no fee to the merchant to adopt Paze.

Digital Transactions, October 2024



# Overview of the Paze<sup>SM</sup> Merchant SDK, Continued

## Step-by-Step SDK Overview

There are four available JavaScript APIs to be used in a merchant's front-end implementation:



1. **Initialize** – Identifies you as the merchant to Paze and initiates fraud controls.



2. **CanCheckout** (optional) – Shows the Paze button only to consumers with an account.



3. **Checkout** – Displays a comprehensive wallet checkout, which includes consumer authentication and card and address selection.



4. **Complete** – Returns payment data securely, providing the necessary data for payment processing.

With Paze, merchants can enhance their checkout offerings with cutting-edge security features and a streamlined shopping experience.

Merchants retain the flexibility to use their existing payment processors. Paze will deliver the consumer's card information and required data for seamless processing.



At Sephora, we continue to put our customers at the center of our business and seek out innovative ways for them to shop and enjoy their favorite beauty products. Our focus has always been on providing a seamless shopping experience, and we're excited to offer Paze to our most loyal shoppers.

**Stefan Jensen,**  
Vice President, Treasurer at Sephora

# 3 Prerequisites for the Paze<sup>SM</sup> Merchant SDK

To begin using our APIs, you'll need to partner with an approved Paze<sup>SM</sup> Distributor. Your Distributor will guide you through the onboarding process, which involves:



**Establishing a Partnership:** Your Distributor will sponsor your integration.



**Exchanging Cryptographic Material:** You or a Technical Integrator working on your behalf will share secure cryptographic keys with Paze to enable secure communication



**Receiving Credentials:** Paze will assign all merchants a unique identifier.

**By completing these steps, you'll be ready to leverage the power of Paze APIs.**

# 4 Getting Started with the Paze<sup>SM</sup> Merchant SDK

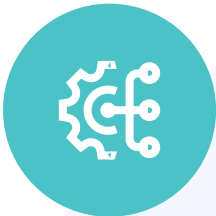
Before requesting Initialize to begin using Paze<sup>SM</sup> you'll need to load our SDK into your website or application. The SDK will create a popup window that will be used to display the various screens such as authentication and card selection.

When the SDK needs to display a popup window, such as for authentication or card selection, it will automatically manage the window's behavior. This ensures a smooth and consistent user experience.



## Data Privacy with Secure Key Management

Asymmetric data encryption keys are supported and require a public certificate from the entity that will be decrypting the data.



## Authentication Endpoints

Paze provides authentication endpoints during the onboarding process to ensure secure communication between your system and our platform. We have separate endpoints for each environment, including sandbox and production.

# 5 Understanding Paze<sup>SM</sup> APIs and SDK Interactions

To enable Paze, you must use the Initialize, Checkout, and Complete APIs. To improve the customer experience, you can also use the optional CanCheckout function.

All JavaScript (JS) SDK APIs use JS Promises to communicate results. This means that instead of events or callbacks, you'll get a direct response from the Paze<sup>SM</sup> Merchant SDK.



## Initialize

Before authenticating the consumer and offering card selection, it's essential to initialize Paze with common state information. This crucial step:

- **Starts the merchant session:** Sets the stage for seamless payment selection.
- **Accelerates fraud checks:** Immediately begins verifying the consumer's device for enhanced security.

**Multiple Sessions, No Problem:** You can run multiple merchant sessions simultaneously to handle complex scenarios and improve efficiency.



## CanCheckout

The CanCheckout API empowers you to tailor the payment experience for each customer. By checking if a specific customer (identified by their email address) is eligible for Paze, you can:

- **Dynamically display the Paze button:** Show the button only to enrolled customers.
- **Enhance the user experience:** Avoid unnecessary steps for eligible customers, leading to faster and smoother transactions.

**Privacy-First Approach:** Rest assured that customer privacy is protected. The SDK securely hashes the customer's email address before transmission, ensuring confidentiality.



## Checkout

When you invoke the Checkout API, your customers will be seamlessly guided through the Paze™ user interface:

1. **Secure Authentication:** Customers will authenticate themselves using their preferred method.
2. **Card Selection:** Customers can choose from their saved cards (cards are pre-provisioned from the consumer's participating financial institutions).
3. **Address Entry (Optional):** If configured, customers will select their shipping address or add a new one. Streamline your checkout by automatically using the customer's address from Paze. No need for you to collect this information separately.

### Simplified Data for Customer Confirmation:

By leveraging the streamlined Paze checkout process, you can significantly improve the customer experience and increase conversion rates. Upon successful checkout, you'll receive essential information:

- **Masked Card Details:** A secure, consumer-recognizable representation of the customer's card details, including art and its last four digits.
- **Customer Information:** Key details about the customer.
- **Billing and Shipping Addresses:** Accurate address information for payment processing and order fulfillment.

Customers can return to the Paze flow if they would like to modify the card they want to use.



## Complete

The Complete API is the final step in the Paze journey. It:

- **Finalizes the transaction:** Closes both the merchant and consumer sessions.
- **Provides essential payment details:** Delivers a payment identifier and network-tokenized card information.
- **Enables saving payment details to your site:** (Optional) Delivers data that can be used to generate a merchant-specific network token.



# 6 Paze<sup>SM</sup> Merchant SDK **Backward Compatibility Considerations**

We're committed to making your integration journey as seamless as possible. Here's what you need to know about our SDK's backward compatibility:



## **What We Consider Backward-Compatible:**

- **Adding New Features:** We may introduce new API resources, methods, or optional parameters without disrupting your existing integration.
- **Expanding Options:** We may add new values to enumerated fields in requests or responses.
- **Relaxing Constraints:** We may loosen restrictions on certain request parameters, offering more flexibility.
- **Enhancing Response Information:** We may include additional information in API responses.
- **Updating Identifiers:** We may change the format of non-human-readable identifiers.



## **Key Principles for Seamless Integration:**

- **Ignore the Unknown:** Your integration should gracefully handle any unexpected fields or enumerated values in API responses.
- **Flexibility with Identifiers:** Don't make assumptions about the format or content of opaque string identifiers.
- **Embrace Future Changes:** Be prepared for potential additions of optional parameters and relaxed constraints in our APIs.

**By following these guidelines, you can future-proof your integration and minimize the impact of future SDK updates.**

# 7 Paze<sup>SM</sup> Merchant SDK Responses



## Successful Responses

A successful Promise means that the task or operation has been completed successfully. In simpler terms, the action you requested has been carried out without any issues.



## Error Responses

An error response indicates that the requested action has failed. When this occurs, a Promise is rejected, and an error object is provided to help you understand the reason for the failure.

By understanding the nature of these errors and following the guidelines provided in the error object, you can effectively troubleshoot and resolve issues.



## Standard Error Codes

To enhance the consistency and reliability of your integration, we've standardized error codes across all our APIs. This means you can expect a uniform approach to error messages, regardless of the specific API you're using.

To aid in troubleshooting, our system prioritizes errors based on processing order. When an error occurs, only the first encountered error is returned, allowing you to quickly identify and address the root cause.



## Business Errors

Business errors are specific to certain APIs and are detailed in their respective documentation.



## Error Handling Guidelines

Adhere to these easy-to-follow guidelines to streamline your error handling process and ensure seamless compatibility with our APIs:

**Focus on the essentials:** Prioritize error handling based on the error reason.

**Avoid unnecessary complexity:** The error message is for informational purposes only and subject to change. Don't parse it or expose it to your customers.

**Handle unexpected errors gracefully:** Implement a default "catch-all" error handler to gracefully handle unexpected errors.

# Get connected with Paze<sup>SM</sup>

## About Paze

Paze is a reimagined online checkout solution that banks and credit unions offer to consumers and merchants, combining all eligible debit and credit cards into a single wallet and eliminating manual card entry. Solving long-standing challenges in e-commerce, Paze provides an easy experience for consumers and merchants alike. Paze is operated by Early Warning Services, LLC, an innovator in financial and risk management solutions.

©2025 Early Warning Services, LLC. All rights reserved. The information contained in this document is the property of Early Warning Services, LLC ("Early Warning"). This document and its contents are provided to you for reference only. Interested Merchants must establish a relationship with an approved Paze<sup>SM</sup> Distributor for additional information. Unauthorized use, copying, display, distribution, modification or reproduction is strictly prohibited. All trademarks referenced in this material are the property of their respective owners.

Due to continuing development, the specifications and capabilities described in this document are subject to change without notice.

