## What you’ll want to do next when you have more info from Noda (post bank holiday):

1. **Complete the API integration details:**
   * Confirm exact request parameters needed to create a payment session.
   * Implement authentication headers and signature as required.
   * Parse the response to get the payment redirect URL.
2. **Implement webhook handling:**
   * Create an endpoint in your plugin to receive payment status updates from Noda.
   * Verify webhook authenticity.
   * Update WooCommerce order status (processing, completed, failed, etc.) based on the webhook data.
   * Add order notes and trigger emails as appropriate.
3. **Test thoroughly in sandbox mode:**
   * Use Noda’s sandbox API keys.
   * Test payment flows, including success, failure, and cancellation.
   * Verify order status updates and customer notifications.
4. **Polish UX and error handling:**
   * Show clear messages on payment errors.
   * Handle timeouts or API failures gracefully.
   * Log API requests and responses for debugging.

## Helpful references from Noda and WooCommerce docs:

* [Noda WooCommerce integration guide1](https://noda.live/articles/how-to-accept-payments-with-woocommerce)
* [Noda WooCommerce plugin overview2](https://noda.live/plugins/woocommerce)
* [WooCommerce Payment Gateway API3](https://developer.woocommerce.com/docs/woocommerce-payment-gateway-api/)

## Summary

Your current codebase has the **core payment initiation and redirect flow skeleton**. Once you get the full API details from Noda, you’ll:

* Flesh out the API call with real parameters and authentication.
* Add webhook processing to finalize orders.
* Test and go live.

Feel free to push your current skeleton to GitHub and revisit after the bank holiday. When you have the API specs or need help finishing the integration, just share the details and I can assist with coding or troubleshooting!

Close to a complete, professional Noda WooCommerce payment gateway integration.