Build a Payment Gateway

🎯 The Challenge

Build a payment gateway system that merchants can integrate into their websites.
Time Limit: 3 days

📋 What You Need to Build

1. **A Payment Gateway API** - Backend service that processes payments
2. **An SDK** - Library that provide a widget making it easy for merchants to use your gateway
3. **A Demo** - Show it working with a simple checkout page
4. **A Merchant Portal –** A portal for merchant to view transactions

---

💭 Things to Consider

You're building a system where:

- Merchants need to accept payments from customers using credit card only
- Merchants want an easy way to integrate your gateway
- Payments need to be tracked and recorded

🎨 Your Choice

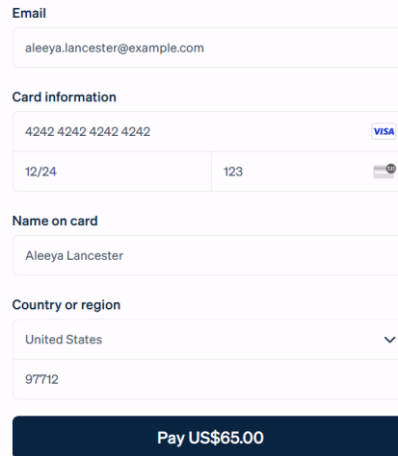**Tech Stack to include:** Java / Springboot / html / css
Tools: Any, no restrictions

📋 Business Requirements

You're building PayFlow - a payment gateway for e-commerce merchants.

As a Customer, I want to:
1. Make a payment on a merchant's website by embedding the widget into the merchant cart checkout page

Sample widget

2. See confirmation when my payment succeeds or fails

   Build a simple checkout page that demonstrates:
   a. Product Page - Shows a product (any item) with "Buy Now" button

   b. Checkout Page - Form with:
      - Amount (read-only)
      - Customer email input
      - Customer name
      - Card number input (use test card: 4242424242424242)
      - Expiry date (MM/YY) - CVV
      - Country
      - Postcode
      - "Pay" button

   c. Confirmation Page - Shows payment success/failure with transaction ID

As a Merchant, I want to:
1. Register and receive API credentials to use the gateway
2. Process customer payments through a simple integration
3. Receive notifications when payments are completed via webhook if URL is set.
4. View all my transactions with their statuses

   Build a web portal that allow merchant to
   a. Register as a merchant
   b. Login
   c. View transactions with their statuses

    d. Set webhook URL

📦 **What to Submit**
1. **Code** - GitHub repository
2. **README** - How to run it, what you built, why you made key decisions