

## The Petal Pouches – Complete Technical Documentation

This document provides a comprehensive technical and design plan for The Petal Pouches — a 3D-first, creative, and high-performance e-commerce experience focused on gifts for teenage girls. It outlines the full technology stack, workflow, and implementation roadmap for the project.

### 1. Core Frontend Stack

Purpose	Tool / Framework	Description
Base Language	JavaScript (ES6+) / TypeScript	Main language of the web; TypeScript recommended for type safety and scalability.
Frontend Framework	React.js	Core UI library for building reusable and interactive components.
Meta-Framework	Next.js (React Framework)	Provides Server-Side Rendering (SSR), Static Site Generation (SSG), routing, and API routes. Ideal for SEO and performance.
Styling	Tailwind CSS	Utility-first CSS framework for responsive design.
Icons	Lucide React / Heroicons	Open-source SVG icon sets.
Animations	Framer Motion	Smooth, modern animations for user interaction.
3D Integration	react-three-fiber (Three.js)	Render 3D objects using WebGL within React.
3D Scene Embeds	Spline	No-code tool for designing 3D scenes, exportable to React.
Lightweight Animation	Lottie React / bodymovin	Animated icons and loaders.
State Management	Zustand / Redux Toolkit	For handling global app state.

Routing	Next.js Router	Handles page navigation and dynamic routes.
Forms / Validation	React Hook Form + Zod/Yup	Efficient form handling and validation.

## 2. Backend Stack

Purpose	Tool / Service	Description
Serverless Backend	Next.js API Routes / Vercel / Netlify Functions	For API endpoints (orders, shipping, payments).
Alternative Backend	Express.js (Node.js)	Optional for more control; host on Render or Railway.
Runtime Environment	Node.js (v18+)	Core JS runtime for backend execution.
Authentication	Supabase Auth / Firebase Auth	Built-in user authentication and management.
Email Service	Resend / Brevo (Sendinblue)	Transactional emails like confirmations and OTPs.
API Security	JWT (JSON Web Token)	Secure user session handling.

## 3. Database and Storage

Type	Tool	Description
Main Database	Supabase (PostgreSQL)	Cloud SQL database with realtime API and file storage.
Alternative Option	Firebase Firestore	NoSQL database for easy scalability.
File Storage	Supabase Storage / Firebase Storage / Cloudflare R2	Store product images, 3D models, and media assets.
ORM / Query Layer	Prisma / Supabase JS SDK / Firebase SDK	Simplifies database queries in JavaScript.

#### 4. Payment Integration

Purpose	Service	Description
Payment Gateway	Razorpay / Stripe	For secure card, UPI, and wallet payments.
Security	Serverless API Validation	Secrets managed securely on backend.
Webhook Handling	Vercel / Netlify Functions	Async confirmation of payments.

#### 5. Shipping and Delivery APIs

Provider	API Function	Description
Shiprocket	Rates, Pincode, Delivery Estimate, Tracking	Aggregator for multiple couriers like BlueDart, Delhivery, etc.
Delhivery API	Shipment Creation, Tracking	For order fulfillment and delivery tracking.
Blue Dart API	Pickup, Estimated Delivery	Courier tracking and scheduling.
AfterShip API	Unified Tracking (Optional)	Tracks multiple couriers via one API.

#### 6. Deployment & Hosting

Component	Platform	Description
Frontend	Vercel / Netlify	Free-tier hosting with CI/CD integration.
Backend APIs	Vercel / Netlify Functions	Serverless endpoints for checkout, tracking, etc.
Database Hosting	Supabase Cloud / Firebase Cloud	Scalable backend services.
Media CDN	Cloudflare CDN / Cloudflare R2 / AWS S3 + CloudFront	Global distribution for heavy assets.
Version Control	Git + GitHub	Source code and deployment control.

## Workflow Overview

- User visits website
- Browses 3D interactive catalog
- Adds to cart and checks out via Razorpay/Stripe
- Backend verifies payment
- Shipment created using Shiprocket/Delhivery API
- User can track shipment via tracking API
- Product delivered