

Vrindavan Ras Desh

Website Development Roadmap & Checklist

Generated: 8 December 2025

This document outlines the current state of the Vrindavan Ras Desh website, future enhancement opportunities, and detailed requirements for implementing an e-commerce store with Razorpay integration.

Table of Contents

1. Current Implementation Status (Completed Features)
2. Future Enhancements & Roadmap
3. E-Commerce Store Implementation Plan
4. Technical Requirements & Effort Estimation
5. Timeline & Resource Allocation

1. Current Implementation Status

' Core Infrastructure

- & Next.js 16 framework with App Router
- & TypeScript configuration
- & PostgreSQL database (Vercel Postgres)
- & Prisma ORM for database management
- & Vercel deployment with CI/CD
- & Environment variable management

' Authentication & Security

- & NextAuth.js integration
- & Secure password hashing (bcryptjs)
- & Role-based access control (ADMIN, SUPERADMIN)
- & Protected admin routes
- & Session management

' Frontend Features

- & Responsive design with vanilla CSS
- & Premium spiritual theme (saffron & gold)
- & Hero carousel with 4 image slots
- & Navigation bar with smooth scrolling
- & Footer with social links
- & Mobile-responsive layout

' Content Management

- & Admin dashboard
- & Site settings CMS (hero, videos, contact)
- & Image upload to Vercel Blob storage
- & YouTube video embedding (3 featured videos)
- & Instagram post embedding (3 posts)
- & Dynamic content updates

' Blog System

- & Blog post creation interface
- & Rich text content support
- & Publish/unpublish functionality
- & Author attribution
- & Slug-based URLs
- & Blog listing page
- & Individual blog post pages

' Database Schema

- & User model (authentication)
- & Post model (blog system)
- & SiteSettings model (CMS)
- & Database seeding script

2. Future Enhancements & Roadmap

Ø=Ý Content & Media Enhancements

- & Photo gallery section for temple darshan images
- & Video gallery with categories (Kirtan, Darshan, Discourses)
- & Audio player for bhajans and kirtans
- & Live streaming integration for special events
- & Podcast section for spiritual discourses
- & Multi-language support (Hindi, English, Sanskrit)

Ø=Ý User Engagement Features

- & Newsletter subscription system
- & User comments on blog posts
- & Social media sharing buttons
- & Event calendar for temple festivals
- & Donation system (separate from e-commerce)
- & User testimonials/experiences section
- & Prayer request submission form

Ø=Ý SEO & Performance

- & SEO optimization (meta tags, structured data)
- & Sitemap generation
- & Image optimization and lazy loading
- & Performance monitoring (Core Web Vitals)
- & Analytics integration (Google Analytics)
- & Search functionality for blog posts

Ø=Ý Admin Panel Improvements

- & User management interface
- & Blog post editor with rich text (WYSIWYG)
- & Media library management
- & Analytics dashboard
- & Bulk operations for content
- & Content scheduling (publish later)

Ø=Ý Mobile Experience

- & Progressive Web App (PWA) capabilities
- & Offline mode for reading blog posts
- & Push notifications for new content
- & App-like navigation on mobile

3. E-Commerce Store Implementation Plan

Transform the website into a comprehensive platform where devotees can purchase spiritual items, books, and prasadam offerings.

Ø=ÞØ Store Features - Phase 1 (Essential)

- & Product catalog with categories
 - Categories: Books, Idols, Puja Items, Prasadam, Clothing, Audio/Video
- & Product detail pages with images and descriptions
- & Shopping cart functionality
- & Checkout process
- & Razorpay payment gateway integration
- & Order management system
- & Order confirmation emails
- & Basic inventory tracking

Ø=ÞØ Store Features - Phase 2 (Enhanced)

- & User accounts for order history
- & Wishlist functionality
- & Product reviews and ratings
- & Advanced search and filters
- & Related products suggestions
- & Discount codes and coupons
- & Multiple payment options (UPI, Cards, Wallets)
- & Shipping cost calculator
- & Order tracking system

Ø=ÞØ Store Features - Phase 3 (Advanced)

- & Subscription boxes (monthly prasadam delivery)
- & Gift cards and vouchers
- & Bulk order discounts
- & International shipping support
- & Multi-currency support
- & Advanced analytics and reporting
- & Automated inventory management
- & Vendor/supplier management

Ø=Ü³ Razorpay Integration Details

Razorpay will serve as the primary payment gateway for all transactions.

- Payment Methods Supported:
 - Credit/Debit Cards (Visa, Mastercard, RuPay, Amex)
 - UPI (Google Pay, PhonePe, Paytm, etc.)
 - Net Banking (all major banks)
 - Digital Wallets (Paytm, Mobikwik, etc.)
 - EMI options for high-value purchases
- Key Integration Features:
 - Razorpay Checkout (hosted payment page)
 - Webhook integration for payment status updates
 - Automatic refund processing
 - Payment link generation for custom orders
 - Subscription/recurring payments for monthly boxes
 - Smart routing for better success rates
- Security & Compliance:
 - PCI DSS Level 1 compliant
 - 3D Secure authentication
 - Fraud detection and prevention
 - Encrypted payment data

Ø=Üæ Database Schema Extensions

New models required for e-commerce functionality:

- Product Model:
 - id, name, description, price, compareAtPrice
 - category, subcategory, images[], stock
 - weight, dimensions, sku, isActive
 - createdAt, updatedAt
- Category Model:
 - id, name, slug, description, image
 - parentId (for subcategories)
- Order Model:
 - id, orderNumber, userId, status
 - subtotal, tax, shipping, total
 - paymentId, paymentStatus, razorpayOrderId
 - shippingAddress, billingAddress
 - items[] (relation to OrderItem)
- OrderItem Model:
 - id, orderId, productId, quantity
 - price, total
- Customer Model (extends User):
 - phone, addresses[], defaultAddressId
 - orders[] (relation to Order)
- Address Model:
 - id, customerId, name, phone, addressLine1
 - addressLine2, city, state, pincode, country
 - isDefault

4. Technical Requirements & Dependencies

ðŸ“œ New NPM Packages Required

- razorpay - Official Razorpay Node.js SDK
- @razorpay/razorpay-js - Client-side Razorpay integration
- react-hook-form - Form management for checkout
- zod - Schema validation for forms and API
- react-query / @tanstack/react-query - Data fetching and caching
- zustand or redux - State management for cart
- nodemailer - Email notifications for orders
- react-toastify - User notifications
- date-fns - Date formatting for orders
- sharp - Image optimization for products
- slugify - Generate product slugs

ðŸ“œ Third-Party Services

- Razorpay Account (Business/Merchant)
- Email service (SendGrid, Resend, or AWS SES)
- SMS service for order updates (optional - Twilio, MSG91)
- Shipping API integration (Shiprocket, Delhivery)
- Google Analytics for e-commerce tracking

&™ Environment Variables to Add

- RAZORPAY_KEY_ID - Razorpay API key
- RAZORPAY_KEY_SECRET - Razorpay secret key
- RAZORPAY_WEBHOOK_SECRET - Webhook signature verification
- SMTP_HOST, SMTP_PORT, SMTP_USER, SMTP_PASS - Email config
- NEXT_PUBLIC_STORE_URL - Store base URL
- TAX_RATE - GST/Tax percentage
- SHIPPING_RATE - Base shipping cost

ðŸ“œ UI/UX Components to Build

- Product card component
- Product detail page with image gallery
- Shopping cart sidebar/modal
- Checkout form (multi-step)
- Payment status page
- Order confirmation page
- Order history page
- Product filters and search
- Category navigation
- Admin product management interface
- Admin order management dashboard

5. Effort Estimation & Timeline

Estimated effort for a single full-stack engineer working full-time (8 hours/day):

Phase 1: E-Commerce Foundation (Essential Features)

Task: Database Schema Design & Migration

Effort: 2 days

Details: Create new Prisma models, migrations, seed data

Task: Product Catalog & Management

Effort: 5 days

Details: Product CRUD, categories, image uploads, admin UI

Task: Shopping Cart System

Effort: 3 days

Details: Cart state management, add/remove/update, persistence

Task: Checkout Process

Effort: 4 days

Details: Multi-step form, address management, validation

Task: Razorpay Integration

Effort: 4 days

Details: Payment gateway setup, webhook handling, testing

Task: Order Management System

Effort: 4 days

Details: Order creation, status tracking, admin dashboard

Task: Email Notifications

Effort: 2 days

Details: Order confirmation, status updates, templates

Task: Store Frontend Pages

Effort: 5 days

Details: Shop page, product pages, cart, checkout UI

Task: Testing & Bug Fixes

Effort: 3 days

Details: End-to-end testing, payment testing, fixes

Phase 1 Total: 32 working days (~6.5 weeks)

Phase 2: Enhanced Features

Task: User Accounts & Order History

Effort: 3 days

Task: Product Reviews & Ratings

Effort: 3 days

Task: Wishlist Functionality

Effort: 2 days

Task: Advanced Search & Filters

Effort: 3 days

Task: Discount Codes System

Effort: 3 days

Task: Shipping Integration

Effort: 4 days

Task: Order Tracking

Effort: 2 days

Phase 2 Total: 20 working days (~4 weeks)

Phase 3: Advanced Features

Task: Subscription System

Effort: 5 days

Task: Gift Cards & Vouchers

Effort: 4 days

Task: International Shipping & Multi-currency

Effort: 5 days

TOTAL EFFORT SUMMARY

Task: Phase 1 (Essentials): 22 days (~6.5 weeks)

Phase 2 (Enhanced): 20 days (~4 weeks)

Phase 3 (Advanced): 21 days (~4.5 weeks)

Task: Inventory Management Automation

Effort: 3 days

GRAND TOTAL: 73 working days (~15 weeks / ~3.5 months)

Phase 3 Total: 21 working days (~4.5 weeks)

Recommended Approach

- Start with Phase 1 to get a functional store live quickly
- Gather user feedback and analytics from Phase 1
- Prioritize Phase 2 features based on user demand
- Implement Phase 3 features as business scales
- Consider hiring additional resources for faster delivery

Resource Recommendations

- 1 Full-Stack Engineer (primary developer)
- 1 UI/UX Designer (for store design and product photography)
- 1 QA Tester (for payment and order flow testing)
- 1 Content Manager (for product listings and descriptions)

6. Additional Considerations

Legal & Compliance

- & Terms & Conditions page for e-commerce
- & Privacy Policy (GDPR compliance if applicable)
- & Refund & Return Policy
- & Shipping Policy
- & GST registration and invoice generation
- & FSSAI license (if selling prasadam/food items)

Business Requirements

- & Product photography and descriptions
- & Pricing strategy and margins
- & Inventory sourcing and management
- & Packaging and shipping materials
- & Customer service workflow
- & Return/exchange process

Security Considerations

- & SSL certificate (already on Vercel)
- & Payment data encryption (handled by Razorpay)
- & User data protection
- & Rate limiting on checkout API
- & Fraud detection mechanisms
- & Regular security audits

Marketing & Growth

- & SEO optimization for product pages
- & Social media integration for sharing products
- & Email marketing for promotions
- & Abandoned cart recovery emails
- & Loyalty program for repeat customers
- & Referral program

7. Next Steps

1. Review and approve this roadmap
2. Set up Razorpay merchant account
3. Finalize product catalog and categories
4. Begin Phase 1 development
5. Prepare product content (images, descriptions, pricing)
6. Set up email service for notifications
7. Plan soft launch with limited products for testing
