# **PRODUCT REQUIREMENTS DOCUMENT**

## **Vendora Bangladesh Marketplace**

### **Version 3.0 - Core + Plugin Architecture Edition**

**Date:** October 16, 2025

## **DOCUMENT CONTROL**

* **Project:** Vendora Bangladesh Marketplace
* **Version:** 3.0 (Core + Plugin Architecture)
* **Status:** Final for Development
* **Prepared By:** Product Team
* **Reviewed By:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* **Approved By:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### **Change Log:**

* v1.5 (Oct 15, 2025) - Initial draft
* v2.0 (Oct 16, 2025) - Enhanced with critical modules and Bangladesh features
* v3.0 (Oct 16, 2025) - Restructured with Core + Plugin Architecture

## **ARCHITECTURAL PHILOSOPHY**

### **Core + Plugin Architecture Benefits**

**Performance First:**

* Core loads only essential features (60-70% faster initial load)
* Plugins load on-demand reducing memory by 40%
* Unused features have zero performance impact
* Better resource utilization on shared hosting

**Business Flexibility:**

* Launch MVP in 3 months with core only
* Add features without touching core code
* Different plugin bundles for subscription tiers
* Enable/disable features per vendor
* A/B test new features safely

**Development Advantages:**

* Parallel development by multiple teams
* Independent testing and deployment
* Lower risk of breaking changes
* Easier maintenance and debugging
* Third-party developer ecosystem potential

## **TABLE OF CONTENTS**

1. **EXECUTIVE SUMMARY**
2. **OVERVIEW**
   * 2.1 Purpose & Goals
   * 2.2 Market Context & Competitive Analysis
   * 2.3 Success Criteria
3. **CORE OBJECTIVES**
4. **ARCHITECTURAL DESIGN**
   * 4.1 Core System Components
   * 4.2 Plugin System Architecture
   * 4.3 Hook System & Events
5. **USER ROLES & PERMISSIONS**
6. **CORE MODULES** (Essential - Always Loaded)
7. **PLUGIN MODULES** (Optional - Load on Demand)
8. **PLUGIN DEVELOPMENT GUIDELINES**
9. **TECHNOLOGY STACK**
10. **BANGLADESH MARKET SPECIFICS**
11. **SECURITY & COMPLIANCE**
12. **PHASE PLAN** (Revised for Core + Plugin)
13. **SUCCESS METRICS**
14. **APPENDICES**

## **1. EXECUTIVE SUMMARY**

### **Project Overview**

Vendora Bangladesh is a **plugin-based** multivendor marketplace built on Laravel 11 and Inertia + Vue 3, designed specifically for the Bangladesh market. The platform uses a **lean core + extensible plugin architecture** allowing vendors to enable only the features they need, resulting in superior performance and flexibility.

### **Key Architectural Changes in v3.0**

* **Lean Core**: Only authentication, products, cart, checkout, and orders in core
* **30+ Plugins**: All additional features as optional plugins
* **Performance Gain**: 60% faster load times with selective plugin loading
* **Flexible Deployment**: Start with core, add plugins based on demand
* **Revenue Model**: Tiered subscriptions with different plugin access

### **Core System (Months 1-3)**

* User authentication & basic profiles
* Product management (CRUD)
* Shopping cart
* Basic checkout
* Order management
* Essential payment (SSLCommerz)
* Basic vendor dashboard

### **Plugin Ecosystem (Months 4-12)**

* **Payment Plugins**: bKash, COD, Nagad
* **Shipping Plugins**: Pathao, RedX, Sundarban
* **Feature Plugins**: Reviews, Disputes, Analytics, Fraud Prevention
* **Marketing Plugins**: Email, SMS, Push notifications
* **Localization Plugins**: Bengali language, regional settings

### **Target Metrics**

* **Core Launch**: 3 months
* **Initial Plugins**: 10 plugins by month 6
* **Full Platform**: 30+ plugins by month 12
* **Performance**: <1 second core load (mobile 3G)
* **Plugin Adoption**: 60% vendors using 5+ plugins

## **2. OVERVIEW**

### **2.1 Purpose & Goals**

**Purpose:** Vendora Bangladesh aims to become the leading multivendor marketplace by offering a **lightweight, fast core** with optional plugins, allowing vendors to customize their experience without performance penalties.

**Unique Value Proposition:**

* **Fastest marketplace** in Bangladesh (sub-1 second core load)
* **Pay for what you use** plugin model
* **Zero bloat** - unused features don't load
* **Vendor choice** - enable only needed features
* **Progressive enhancement** - start simple, grow complex

**Goals:**

* Launch lean core in 3 months
* Support 30+ plugins by year end
* Maintain <1 second load time for core
* Enable vendor-specific plugin configurations
* Create plugin marketplace for developers

### **2.2 Market Context & Competitive Analysis**

**Why Plugin Architecture Wins:**

**Competitors' Monolithic Problems:**

* **Daraz**: Slow 4-5 second loads due to feature bloat
* **Chaldal**: All features loaded even if unused
* **Facebook Marketplace**: No customization options

**Vendora's Plugin Advantage:**

* Start selling with core in <1 second
* Add complexity only when needed
* Lower hosting costs for small vendors
* Premium plugins for advanced sellers
* Custom plugin development possible

### **2.3 Success Criteria**

**Core Metrics:**

* Core system load time <1 second
* Core stability >99.99% uptime
* Zero critical bugs in core

**Plugin Metrics:**

* Average vendor uses 5-7 plugins
* Plugin activation <10 seconds
* Plugin failure doesn't affect core
* 90% plugin satisfaction rate

## **3. CORE OBJECTIVES**

### **3.1 Minimal Viable Core**

* **Principle**: Core contains ONLY what's needed to buy and sell
* **Size Target**: Core <5MB, loads <1 second on 3G
* **Dependencies**: Minimal external libraries in core

### **3.2 Plugin-First Development**

* **New features**: Always plugins, never core additions
* **Backwards compatibility**: Core API never breaks
* **Plugin isolation**: One plugin failure doesn't affect others

### **3.3 Vendor Empowerment**

* **Self-service**: Vendors activate/deactivate plugins
* **Instant changes**: No restart required
* **Cost control**: Pay only for active plugins

## **4. ARCHITECTURAL DESIGN**

### **4.1 Core System Components**

**Core Database Tables (Always Present):**

sql

- users (authentication)

- vendors (basic profile)

- products (essential fields only)

- categories (system-defined)

- cart\_items (session-based)

- orders (basic order data)

- order\_items (products in order)

- media (images/files)

- settings (core configuration)

**Core API Endpoints:**

/auth/\* - Authentication

/products/\* - Product CRUD

/cart/\* - Cart operations

/checkout - Basic checkout

/orders/\* - Order management

**Core Services:**

* AuthenticationService
* ProductService
* CartService
* OrderService
* PaymentService (base class)
* MediaService

### **4.2 Plugin System Architecture**

**Plugin Structure:**

/plugins/

/bkash-payment/

/config/

plugin.json # Metadata

/src/

BkashPlugin.php # Main class

/Controllers/

/Models/

/Views/

/assets/

/js/

/css/

/database/

/migrations/

/tests/

**Plugin Lifecycle:**

1. **Discovery**: System scans /plugins directory
2. **Registration**: Plugin metadata loaded
3. **Activation**: Vendor/admin enables plugin
4. **Initialization**: Plugin boots, registers hooks
5. **Execution**: Plugin responds to events
6. **Deactivation**: Cleanup and disable

**Plugin Metadata (plugin.json):**

json

{

"name": "bkash-payment",

"version": "1.0.0",

"displayName": "bKash Payment Gateway",

"description": "Accept payments via bKash",

"author": "Vendora Team",

"requires": {

"core": ">=1.0.0",

"php": ">=8.1",

"plugins": ["payment-base"]

},

"provides": ["payment-method"],

"hooks": [

"checkout.payment\_methods",

"order.payment\_process"

],

"settings": true,

"price": {

"type": "subscription",

"amount": 500,

"currency": "BDT",

"period": "monthly"

}

}

### **4.3 Hook System & Events**

**Core Hooks (Extension Points):**

php

*// Filter Hooks (modify data)*

apply\_filters('product.price', $price, $product);

apply\_filters('cart.items', $items);

apply\_filters('checkout.fields', $fields);

*// Action Hooks (trigger events)*

do\_action('order.created', $order);

do\_action('payment.completed', $payment);

do\_action('user.registered', $user);

**Plugin Hook Implementation:**

php

class BkashPlugin extends Plugin {

public function boot() {

*// Add bKash to payment methods*

add\_filter('checkout.payment\_methods', [$this, 'addBkashMethod']);

*// Process bKash payments*

add\_action('payment.process.bkash', [$this, 'processPayment']);

*// Add settings page*

add\_action('admin.menu', [$this, 'addSettingsPage']);

}

}

## **5. USER ROLES & PERMISSIONS**

### **Core Roles (Always Available)**

**Super Admin**

* Manage core system settings
* Activate/deactivate global plugins
* Override vendor plugin settings
* Access all core features

**Vendor**

* Manage own products
* View own orders
* Activate/deactivate own plugins
* Configure plugin settings

**Customer**

* Browse products
* Add to cart
* Checkout
* View orders

### **Plugin-Added Roles**

**Moderator** (via moderation-plugin)

* Content approval capabilities

**Support Agent** (via support-plugin)

* Customer service features

**Store Manager** (via team-plugin)

* Vendor delegation features

## **6. CORE MODULES (Essential - Always Loaded)**

### **6.1 Authentication Core**

**Size:** ~500KB **Load Time:** <100ms

**Features:**

* Email/password registration
* Login/logout
* Password reset
* Session management
* Basic 2FA

**Excluded (Plugin Territory):**

* Social login → social-auth-plugin
* Phone OTP → phone-verify-plugin
* Advanced 2FA → security-plugin

### **6.2 Vendor Core**

**Size:** ~300KB **Load Time:** <80ms

**Features:**

* Basic vendor profile (name, email, slug)
* Vendor status (active/inactive)
* Basic dashboard

**Excluded (Plugin Territory):**

* KYC verification → kyc-plugin
* Trust scores → reputation-plugin
* Subscription tiers → subscription-plugin
* Analytics → analytics-plugin

### **6.3 Product Core**

**Size:** ~800KB **Load Time:** <150ms

**Features:**

* Product CRUD
* Basic fields (title, description, price, images)
* Simple inventory tracking
* Categories

**Excluded (Plugin Territory):**

* Product variants → variants-plugin
* Digital products → digital-products-plugin
* Services/bookings → booking-plugin
* Reviews → review-plugin
* Import/export → import-export-plugin

### **6.4 Cart Core**

**Size:** ~200KB **Load Time:** <50ms

**Features:**

* Add/remove items
* Update quantities
* Session persistence
* Basic validation

**Excluded (Plugin Territory):**

* Saved carts → wishlist-plugin
* Abandoned cart recovery → marketing-plugin
* Coupons → discount-plugin

### **6.5 Checkout Core**

**Size:** ~400KB **Load Time:** <100ms

**Features:**

* Guest checkout
* Address collection
* Order summary
* Basic payment (SSLCommerz only)

**Excluded (Plugin Territory):**

* Multiple addresses → address-book-plugin
* Gift messages → gift-plugin
* Shipping calculation → shipping-plugins
* Tax calculation → tax-plugin

### **6.6 Order Core**

**Size:** ~350KB **Load Time:** <80ms

**Features:**

* Order creation
* Basic status tracking
* Order history
* Email confirmation

**Excluded (Plugin Territory):**

* Refunds → refund-plugin
* Returns → return-plugin
* Disputes → dispute-plugin
* Invoices → invoice-plugin
* Tracking → shipment-tracking-plugin

## **7. PLUGIN MODULES (Optional - Load on Demand)**

### **7.1 Payment Plugins**

**bkash-payment-plugin**

* Size: 150KB
* Adds bKash payment method
* Handles bKash API integration
* Subscription: FREE with Basic plan

**cod-payment-plugin**

* Size: 100KB
* Adds Cash on Delivery
* Phone verification for COD
* COD limits and fees
* Subscription: FREE

**nagad-payment-plugin**

* Size: 150KB
* Adds Nagad payment
* Subscription: ৳200/month

### **7.2 Shipping Plugins**

**pathao-shipping-plugin**

* Size: 200KB
* Pathao API integration
* Label generation
* Tracking updates
* Subscription: ৳300/month

**redx-shipping-plugin**

* Size: 200KB
* RedX integration
* Subscription: ৳300/month

**shipping-zones-plugin**

* Size: 100KB
* Bangladesh zone configuration
* Rate calculation
* Subscription: FREE with Pro plan

### **7.3 Feature Plugins**

**review-rating-plugin**

* Size: 300KB
* Product reviews
* Vendor ratings
* Review moderation
* Subscription: ৳500/month

**dispute-resolution-plugin**

* Size: 400KB
* Dispute workflow
* Mediation system
* Auto-escalation
* Subscription: ৳1000/month

**analytics-dashboard-plugin**

* Size: 500KB
* Sales analytics
* Traffic reports
* Product performance
* Subscription: ৳800/month

**fraud-prevention-plugin**

* Size: 350KB
* Risk scoring
* Velocity checks
* Blacklist management
* Subscription: ৳1500/month

### **7.4 Marketing Plugins**

**email-marketing-plugin**

* Size: 400KB
* Campaign management
* Automated emails
* Templates
* Subscription: ৳1000/month

**sms-notifications-plugin**

* Size: 200KB
* Order SMS
* Marketing SMS
* Subscription: ৳500/month

**discount-coupon-plugin**

* Size: 250KB
* Coupon creation
* Bulk discounts
* Flash sales
* Subscription: ৳600/month

### **7.5 Vendor Plugins**

**subscription-tiers-plugin**

* Size: 300KB
* Tier management
* Billing cycles
* Feature limits
* Subscription: CORE REQUIREMENT

**kyc-verification-plugin**

* Size: 400KB
* Document upload
* Verification workflow
* Subscription: CORE REQUIREMENT

**multi-store-plugin**

* Size: 350KB
* Multiple stores per vendor
* Store switching
* Subscription: ৳2000/month

### **7.6 Localization Plugins**

**bengali-language-plugin**

* Size: 200KB
* Full Bengali translation
* Number formatting
* Date formatting
* Subscription: FREE

**multi-currency-plugin**

* Size: 300KB
* USD support
* INR support
* Auto-conversion
* Subscription: ৳1000/month

## **8. PLUGIN DEVELOPMENT GUIDELINES**

### **8.1 Plugin Standards**

**Naming Convention:**

* Folder: kebab-case-plugin
* Class: PascalCasePlugin
* Hooks: plugin\_name.hook\_name

**Required Files:**

plugin.json # Metadata (required)

PluginNamePlugin.php # Main class (required)

README.md # Documentation (required)

LICENSE # License file (required)

changelog.md # Version history (required)

### **8.2 Plugin API**

**Available Core APIs:**

php

*// Product API*

$products = PluginAPI::getProducts($filters);

PluginAPI::updateProduct($id, $data);

*// Order API*

$order = PluginAPI::getOrder($id);

PluginAPI::updateOrderStatus($id, $status);

*// User API*

$user = PluginAPI::getCurrentUser();

PluginAPI::getUserMeta($userId, $key);

*// Settings API*

PluginAPI::getSetting($key);

PluginAPI::saveSetting($key, $value);

*// Database API*

PluginAPI::createTable($tableName, $schema);

PluginAPI::dropTable($tableName);

### **8.3 Plugin Restrictions**

**Plugins CANNOT:**

* Modify core database tables directly
* Override core functionality
* Access other plugins' private data
* Slow page load >500ms
* Use deprecated PHP functions
* Include external tracking without consent

**Plugins MUST:**

* Handle activation/deactivation cleanly
* Provide uninstall cleanup
* Document all settings
* Include error handling
* Be compatible with latest core
* Follow PSR-4 autoloading

### **8.4 Plugin Testing**

**Required Tests:**

php

class PluginTest extends TestCase {

public function test\_activation() { }

public function test\_deactivation() { }

public function test\_hooks\_registered() { }

public function test\_settings\_saved() { }

public function test\_uninstall\_cleanup() { }

}

## **9. TECHNOLOGY STACK**

### **Core Technologies**

* **Framework:** Laravel 11 (minimal installation)
* **Frontend:** Inertia.js + Vue 3 (core components only)
* **Database:** MySQL 8.0
* **Cache:** Redis (optional, defaults to file)
* **Queue:** Database queue (upgradable to Redis)

### **Plugin System**

* **Plugin Loader:** Custom PSR-4 autoloader
* **Hook System:** WordPress-style actions/filters
* **Event Bus:** Laravel Events
* **Plugin Storage:** Filesystem + database metadata

### **Development Tools**

* **Plugin CLI:** php artisan plugin:create
* **Plugin Testing:** PHPUnit + custom assertions
* **Plugin Packager:** php artisan plugin:package

## **10. BANGLADESH MARKET SPECIFICS**

### **Core Inclusions**

* BDT currency (core)
* Bangladesh address format (core)
* Mobile number validation (core)

### **Plugin Extensions**

* bKash payment (plugin)
* Bengali language (plugin)
* Local couriers (plugins)
* COD verification (plugin)

## **11. SECURITY & COMPLIANCE**

### **Core Security**

* CSRF protection
* XSS prevention
* SQL injection prevention
* Password hashing (bcrypt)
* Rate limiting

### **Plugin Security**

* Sandboxed execution
* Permission system
* Code review required
* Automated security scanning
* Vulnerable plugin alerts

## **12. PHASE PLAN (Revised for Core + Plugin)**

### **PHASE 1: Core Development (Months 1-3)**

**Goal:** Launch minimal viable marketplace

**Deliverables:**

* ✅ Authentication system
* ✅ Vendor management (basic)
* ✅ Product CRUD
* ✅ Cart & checkout
* ✅ Order management
* ✅ SSLCommerz payment
* ✅ Plugin architecture

**Success Criteria:**

* Core loads <1 second
* Can complete purchase
* Plugin system works

### **PHASE 2: Essential Plugins (Months 4-5)**

**Goal:** Add Bangladesh-specific features

**Deliverables:**

* ✅ bkash-payment-plugin
* ✅ cod-payment-plugin
* ✅ bengali-language-plugin
* ✅ pathao-shipping-plugin
* ✅ kyc-verification-plugin
* ✅ subscription-tiers-plugin

**Success Criteria:**

* 80% vendors use bKash plugin
* COD orders processing
* Bengali UI available

### **PHASE 3: Growth Plugins (Months 6-7)**

**Goal:** Enhance marketplace capabilities

**Deliverables:**

* ✅ review-rating-plugin
* ✅ analytics-dashboard-plugin
* ✅ email-marketing-plugin
* ✅ sms-notifications-plugin
* ✅ dispute-resolution-plugin

**Success Criteria:**

* 60% products have reviews
* Vendors using analytics
* Dispute resolution <72 hours

### **PHASE 4: Advanced Plugins (Months 8-9)**

**Goal:** Premium features for scaling

**Deliverables:**

* ✅ fraud-prevention-plugin
* ✅ multi-store-plugin
* ✅ import-export-plugin
* ✅ advanced-seo-plugin
* ✅ affiliate-plugin

**Success Criteria:**

* Fraud detection accuracy >90%
* Large vendors using multi-store
* SEO traffic increased 50%

### **PHASE 5: Optimization (Months 10-11)**

**Goal:** Performance and polish

**Tasks:**

* Plugin performance optimization
* CDN integration
* Caching strategies
* A/B testing framework
* Plugin marketplace beta

**Success Criteria:**

* All plugins <500ms load
* 99.9% uptime achieved
* Plugin conflicts resolved

### **PHASE 6: Ecosystem (Month 12)**

**Goal:** Open plugin development

**Deliverables:**

* Plugin marketplace
* Developer documentation
* Plugin SDK
* Revenue sharing model
* Community plugins

**Success Criteria:**

* 10+ third-party plugins
* Developer community active
* Plugin revenue stream live

## **13. SUCCESS METRICS**

### **Core Metrics**

* **Core Load Time:** <1 second (mobile 3G)
* **Core Size:** <5MB
* **Core Uptime:** >99.99%
* **Core Bugs:** 0 critical, <5 minor

### **Plugin Metrics**

* **Average Plugins/Vendor:** 5-7 plugins
* **Plugin Activation Success:** >99%
* **Plugin Load Time:** <500ms each
* **Plugin Satisfaction:** >90%

### **Business Metrics**

* **Plugin Revenue:** ৳2M/month by month 12
* **Premium Plugins:** 40% adoption
* **Custom Plugin Requests:** 20/month
* **Third-party Developers:** 50+ active

### **Performance Metrics**

* **With 5 Plugins:** <2 second total load
* **With 10 Plugins:** <3 second total load
* **API Response:** <200ms (p95)
* **Plugin Isolation:** 100% (no conflicts)

## **APPENDICES**

### **APPENDIX A: Core Database Schema**

sql

*-- Minimal core tables only*

CREATE TABLE users (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

email VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

role ENUM('customer', 'vendor', 'admin') DEFAULT 'customer',

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

INDEX idx\_email (email)

);

CREATE TABLE vendors (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

user\_id BIGINT UNSIGNED NOT NULL,

slug VARCHAR(255) UNIQUE NOT NULL,

business\_name VARCHAR(255) NOT NULL,

status ENUM('active', 'inactive') DEFAULT 'active',

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id),

INDEX idx\_slug (slug)

);

CREATE TABLE products (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

vendor\_id BIGINT UNSIGNED NOT NULL,

title VARCHAR(500) NOT NULL,

slug VARCHAR(500) UNIQUE NOT NULL,

description TEXT,

price DECIMAL(12,2) NOT NULL,

stock INT DEFAULT 0,

status ENUM('active', 'inactive') DEFAULT 'active',

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (vendor\_id) REFERENCES vendors(id),

INDEX idx\_slug (slug),

FULLTEXT idx\_search (title, description)

);

CREATE TABLE orders (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

order\_number VARCHAR(50) UNIQUE NOT NULL,

customer\_id BIGINT UNSIGNED NOT NULL,

vendor\_id BIGINT UNSIGNED NOT NULL,

total DECIMAL(12,2) NOT NULL,

status VARCHAR(50) DEFAULT 'pending',

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (customer\_id) REFERENCES users(id),

FOREIGN KEY (vendor\_id) REFERENCES vendors(id),

INDEX idx\_order\_number (order\_number)

);

*-- Plugin tables created dynamically*

CREATE TABLE plugins (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100) UNIQUE NOT NULL,

version VARCHAR(20) NOT NULL,

status ENUM('active', 'inactive') DEFAULT 'inactive',

settings JSON,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE plugin\_activations (

id BIGINT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

plugin\_id BIGINT UNSIGNED NOT NULL,

vendor\_id BIGINT UNSIGNED NULL,

activated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (plugin\_id) REFERENCES plugins(id),

FOREIGN KEY (vendor\_id) REFERENCES vendors(id),

UNIQUE KEY unique\_activation (plugin\_id, vendor\_id)

);

### **APPENDIX B: Plugin Development Example**

php

<?php

*// File: /plugins/bkash-payment/src/BkashPaymentPlugin.php*

namespace Plugins\BkashPayment;

use Vendora\Core\Plugin;

use Vendora\Core\Interfaces\PaymentMethodInterface;

class BkashPaymentPlugin extends Plugin implements PaymentMethodInterface

{

public function boot()

{

*// Register payment method*

add\_filter('checkout.payment\_methods', [$this, 'registerPaymentMethod']);

*// Handle payment processing*

add\_action('payment.process.bkash', [$this, 'processPayment']);

*// Add admin settings*

add\_action('admin.menu', [$this, 'addSettingsMenu']);

*// Register webhook endpoint*

add\_action('init', [$this, 'registerWebhookEndpoint']);

}

public function registerPaymentMethod($methods)

{

$methods['bkash'] = [

'id' => 'bkash',

'title' => \_\_('bKash', 'bkash-payment'),

'description' => \_\_('Pay with bKash mobile banking', 'bkash-payment'),

'icon' => $this->getAssetUrl('images/bkash-logo.png'),

'supports' => ['products', 'refunds']

];

return $methods;

}

public function processPayment($order)

{

$bkashApi = new BkashAPI($this->getSettings());

try {

$payment = $bkashApi->createPayment([

'amount' => $order->total,

'reference' => $order->order\_number,

'callback' => route('bkash.callback')

]);

return [

'success' => true,

'redirect' => $payment->payment\_url

];

} catch (BkashException $e) {

return [

'success' => false,

'message' => $e->getMessage()

];

}

}

public function activate()

{

*// Run on plugin activation*

$this->createDatabaseTables();

$this->setDefaultSettings();

}

public function deactivate()

{

*// Clean up on deactivation*

$this->clearCache();

}

public function uninstall()

{

*// Complete removal*

$this->dropDatabaseTables();

$this->deleteSettings();

}

}

### **APPENDIX C: Plugin Pricing Model**

#### **Free Plugins (Included in All Plans)**

* Bengali language
* COD payment
* Basic shipping zones
* Email notifications

#### **Basic Plan Plugins (৳1,500/month)**

* bKash payment
* Pathao shipping
* SMS notifications
* Review system

#### **Pro Plan Plugins (৳5,000/month)**

* All Basic plugins
* Analytics dashboard
* Email marketing
* Discount/coupons
* Dispute resolution
* Import/export

#### **Enterprise Plugins (Custom Pricing)**

* Fraud prevention
* Multi-store
* Advanced analytics
* API access
* Custom plugins

### **APPENDIX D: Plugin Performance Budget**

| **Plugin Type** | **Max Load Time** | **Max Memory** | **Max Queries** |
| --- | --- | --- | --- |
| Payment | 300ms | 10MB | 5 |
| Shipping | 400ms | 8MB | 8 |
| Analytics | 500ms | 20MB | 15 |
| Marketing | 200ms | 5MB | 3 |
| UI Enhancement | 100ms | 2MB | 1 |

### **APPENDIX E: Migration Path from v2.0 to v3.0**

**Phase 1: Core Extraction**

1. Identify core features
2. Remove non-essential code
3. Create core-only branch
4. Test core functionality

**Phase 2: Plugin Creation**

1. Extract features as plugins
2. Create plugin interfaces
3. Implement hook system
4. Test plugin activation

**Phase 3: Data Migration**

1. Backup existing database
2. Create plugin tables
3. Migrate settings to plugins
4. Verify data integrity

**Phase 4: Deployment**

1. Deploy core first
2. Activate essential plugins
3. Enable plugins per vendor
4. Monitor performance

## **CONCLUSION**

PRD v3.0 represents a fundamental architectural shift from monolithic to modular. This plugin-based approach delivers:

✅ **60% faster load times** through selective loading ✅ **40% lower hosting costs** for basic vendors  
 ✅ **Infinite scalability** through plugin ecosystem ✅ **Faster time-to-market** (3 months for core vs 6 months monolithic) ✅ **Revenue opportunities** through plugin subscriptions ✅ **Future-proof architecture** that grows with needs

### **Next Steps**

1. **Week 1:** Approve architectural change
2. **Week 2:** Set up plugin development environment
3. **Week 3:** Begin core extraction
4. **Month 2:** Start core development
5. **Month 3:** Launch core + essential plugins

### **Key Success Factors**

* **Core discipline:** Resist adding features to core
* **Plugin quality:** Enforce strict standards
* **Vendor education:** Teach plugin benefits
* **Performance monitoring:** Track load times
* **Community building:** Encourage third-party plugins