

Music Rental System



Dosen Pengampu :

JEFRI SUNUPURWA ASRI , S.Kom., M.Kom.

Disusun Oleh :

Fernanda - 20230801118

Business Requirements Document (BRD) & Software Requirements Specification (SRS)

1. Executive Summary

Music Rental System adalah platform digital yang memungkinkan pelanggan untuk menyewa alat musik secara online dengan sistem manajemen yang terintegrasi. Sistem ini dikembangkan untuk mempermudah proses rental alat musik dengan memberikan kemudahan bagi pelanggan dan efisiensi operasional bagi penyedia layanan.

2. Business Objectives

2.1 Primary Goals

- Digitalisasi Proses Rental: Mengubah proses rental manual menjadi sistem digital yang efisien.
- Customer Experience: Memberikan pengalaman pelanggan yang seamless dari pencarian hingga pengembalian.
- Operational Efficiency: Meningkatkan efisiensi operasional dengan otomatisasi proses administrasi.
- Data Management: Centralized data management untuk customer, inventory, dan transaksi.

2.2 Business Benefits

- Pengurangan waktu proses booking hingga 80%.
- Peningkatan akurasi data inventory dan customer.
- Real-time tracking untuk status pesanan.
- Automated billing dan invoice generation.
- Improved customer satisfaction melalui self-service portal.

3. Stakeholder Analysis

3.1 Primary Stakeholders

- End Customers: Individu/grup yang ingin menyewa alat musik.
- Business Owner: Pemilik toko rental alat musik.
- Admin Staff: Staff yang mengelola operasional harian.
- Customer Service: Tim yang menangani customer support.

3.2 Secondary Stakeholders

- Delivery Partners: Untuk layanan antar jemput.
- Financial Department: Untuk proses payment dan accounting.
- IT Support: Untuk maintenance dan technical support.

4. Business Process Requirements

4.1 Customer Journey

1. Browse Catalog
2. Select Instrument
3. Book Rental
4. Payment
5. Delivery/Pickup
6. Usage Period
7. Return

4.2 Admin Operations

1. Inventory Management
2. Order Processing
3. Customer Management
4. Payment Tracking
5. Delivery Coordination
6. Return Processing

5. Success Metrics

5.1 Key Performance Indicators (KPIs)

- Conversion Rate: Minimum 15% dari visitor menjadi customer.
- Order Processing Time: Maksimal 30 menit dari booking ke konfirmasi.
- Customer Satisfaction: Rating minimal 4.5 dari 5.0.
- System Uptime: 99.5% availability.
- Revenue Growth: Target 25% increase dalam 6 bulan.

6. System Architecture

6.1 Technology Stack

- Backend Framework: Laravel 12
- Admin Panel: Filament v3
- Database: MySQL
- Frontend: Blade Templates + Alpine.js
- API: RESTful API with OpenAPI 3.0
- Authentication: Laravel Sanctum
- File Storage: Local/Cloud Storage
- Deployment: Docker dan Docker Compose

6.2 System Components

- Frontend Application: Customer-facing web interface
- Admin Panel: Business management interface
- API Layer: RESTful endpoints untuk mobile/external integration
- Database Layer: Data persistence dan management
- Storage System: File dan media management

7. Functional Requirements

7.1 User Management Module

Customer Registration & Authentication

- Sistem harus memungkinkan customer untuk registrasi dengan email dan password.
- Sistem harus memvalidasi uniqueness email dan format yang benar.
- Sistem harus menyimpan informasi customer: first_name, last_name, email, phone, city, postal_code, id_card_number, address.
- Sistem harus mendukung login/logout functionality.

Admin User Management

- Sistem harus mendukung role-based access control (RBAC) dengan roles: super_admin, admin, staff.
- Sistem harus memiliki user management interface di admin panel.

- Sistem harus menyimpan avatar dan profile information untuk admin users.

7.2 Instrument Management Module

Inventory Management

- Sistem harus menyimpan data alat musik: name, description, category, daily_rate, security_deposit, image.
- Sistem harus mengelola availability status: is_available (boolean) dan quantity_available (integer).
- Sistem harus mendukung kategorisasi alat musik.
- Sistem harus mengupdate availability secara real-time saat terjadi booking.

Catalog Display

- Sistem harus menampilkan catalog alat musik di frontend dengan filtering dan search.
- Sistem harus menampilkan detail alat musik dengan gambar, spesifikasi, dan harga.
- Sistem harus menampilkan availability status pada setiap item.

7.3 Rental Order Management Module

Booking Process

- Sistem harus memungkinkan customer untuk melakukan booking dengan data: instrument_id, rental_start_date, rental_end_date, quantity, customer_info, delivery_method.
- Sistem harus memvalidasi availability sebelum konfirmasi booking.
- Sistem harus menghitung total cost: subtotal, tax (10%), security_deposit, total_amount.
- Sistem harus generate unique order_number dengan format "ORD-YYYY-NNNN".
- Sistem harus menyimpan delivery_method: pickup atau delivery.

Order Processing

- Sistem harus memiliki status tracking: pending, confirmed, delivered, in_use, returned, completed, cancelled.
- Sistem harus memiliki payment_status: pending, paid, partial, refunded.
- Sistem harus auto-assign user_id untuk admin yang memproses order.
- Sistem harus menyimpan notes untuk informasi tambahan.

Order Management (Admin)

- Admin harus dapat melihat, edit, dan update status semua orders.
- Admin harus dapat melakukan filtering orders berdasarkan status, tanggal, customer.
- Admin harus dapat melakukan bulk actions untuk multiple orders.
- Sistem harus otomatis calculate subtotal dan total amount.

7.4 API Integration Module

RESTful API Endpoints

- Sistem harus menyediakan API endpoint POST /api/booking/create-order untuk create rental order.
- Sistem harus menyediakan API endpoint GET /api/booking/order/{orderNumber} untuk get order details.
- Sistem harus menyediakan comprehensive OpenAPI documentation dengan L5-Swagger.
- API harus mengembalikan standardized JSON response format.

API Security & Validation

- API harus memvalidasi semua required fields sebelum processing.
- API harus mengembalikan appropriate HTTP status codes dan error messages.
- API harus menggunakan CSRF protection untuk security.

8. Non-Functional Requirements

8.1 Performance Requirements

- Page load time harus kurang dari 3 detik.
- API response time harus kurang dari 2 detik.
- Sistem harus support minimum 100 concurrent users.
- Database query optimization untuk large datasets.

8.2 Security Requirements

- Password harus di-hash menggunakan bcrypt algorithm.
- Sistem harus menggunakan CSRF protection.

- Input validation dan sanitization untuk semua user inputs.
- Role-based access control untuk admin features.

8.3 Usability Requirements

- User interface harus responsive untuk desktop dan mobile.
- Admin panel harus user-friendly dengan intuitive navigation.
- Error messages harus informatif dan user-friendly.
- Sistem harus memiliki search dan filtering capabilities.

8.4 Reliability Requirements

- System availability 99.5% uptime.
- Automated backup system untuk data protection.
- Error logging dan monitoring system.
- Graceful error handling tanpa system crash.

9. Database Schema

9.1 Core Tables

users

- id (bigint, primary key)
- avatar_url (string, nullable)
- name (string)
- email (string, unique)
- password (string)
- roles (relationship)

customers

- id (bigint, primary key)
- first_name (string)

- last_name (string)
- email (string, unique)
- phone (string)
- city (string)
- postal_code (string)
- id_card_number (string)
- address (text)
- created_at, updated_at (timestamps)

instruments

- id (bigint, primary key)
- name (string)
- description (text)
- category (string)
- daily_rate (decimal)
- security_deposit (decimal)
- is_available (boolean)
- quantity_available (integer)
- image (string, nullable)
- created_at, updated_at (timestamps)

rental_orders

- id (bigint, primary key)
- order_number (string, unique)
- customer_id (foreign key)
- user_id (foreign key, nullable)
- rental_start_date (date)
- rental_end_date (date)
- subtotal (decimal)
- tax_amount (decimal)

- security_deposit (decimal)
- total_amount (decimal)
- outstanding_amount (decimal)
- status (enum: pending, confirmed, delivered, in_use, returned, completed, cancelled)
- payment_status (enum: pending, paid, partial, refunded)
- delivery_method (enum: pickup, delivery)
- notes (text, nullable)
- created_at, updated_at (timestamps)

rental_order_items

- id (bigint, primary key)
- rental_order_id (foreign key)
- instrument_id (foreign key)
- quantity (integer)
- unit_price (decimal)
- total_price (decimal)
- created_at, updated_at (timestamps)

9.2 Permission System Tables

- roles: User roles management
- permissions: System permissions
- model_has_roles: User-role assignments
- model_has_permissions: Direct user permissions
- role_has_permissions: Role-permission assignments

10. API Documentation

10.1 Endpoint Overview

- POST /api/booking/create-order - Create new rental order
- GET /api/booking/order/{orderNumber} - Get order details

- GET /api/health - API health check

10.2 API Response Format

success: true atau false

message: Response message

data: Response data object

errors: Validation errors (jika ada)

11. User Interface Requirements

11.1 Frontend Components

- Homepage: Hero section dengan featured instruments
- Catalog Page: Grid/list view dengan filter dan search
- Instrument Detail: Detailed view dengan booking form
- Booking Form: Multi-step form dengan validation
- Booking Success: Confirmation page dengan order details
- Track Order: Order status tracking

11.2 Admin Panel Components

- Dashboard: Overview statistics dan recent activities
- Order Management: CRUD operations untuk rental orders
- Instrument Management: Inventory management interface
- Customer Management: Customer data management
- User Management: Admin user management
- Role & Permissions: Access control management

12. Integration Requirements

12.1 Internal Integrations

- Frontend dan API integration untuk real-time updates

- Admin Panel dan Database untuk data management
- Notification system untuk order updates
- File upload system untuk instrument images

12.2 External Integration Possibilities

- Payment gateway integration (pengembangan selanjutnya)
- SMS/Email notification service
- Google Maps integration untuk delivery tracking
- Inventory sync dengan external POS systems

13. Deployment & Environment

13.1 Development Environment

- PHP 8.2+
- Laravel 12
- MySQL
- Docker dan Docker Compose
- Node.js untuk asset compilation

13.2 Production Requirements

- Web Server: Nginx
- SSL Certificate untuk HTTPS
- Automated backup system
- Monitoring dan logging tools
- CDN untuk static assets

14. Testing Requirements

14.1 Testing Scope

- Unit Testing: Individual component testing

- Integration Testing: API endpoint testing
- Functional Testing: User journey testing
- Performance Testing: Load testing dengan concurrent users
- Security Testing: Vulnerability assessment

14.2 Test Coverage

- Minimum 80% code coverage untuk critical functions
- API endpoint testing dengan Postman atau PHPUnit
- Frontend functionality testing
- Database integrity testing

15. Maintenance & Support

15.1 Regular Maintenance Tasks

- Database optimization dan cleanup
- Security updates dan patches
- Performance monitoring dan tuning
- Backup verification dan restoration testing

15.2 Support Documentation

- User manual untuk end customers
- Admin guide untuk business operations
- API documentation untuk developers
- Troubleshooting guide untuk technical issues

16. Implementation Summary

16.1 Completed Features

- User Authentication & Role Management
- Customer Registration & Management

- Instrument Catalog & Management
- Rental Order Creation & Processing
- Admin Panel dengan CRUD Operations
- Frontend Booking System
- API Endpoints dengan OpenAPI Documentation
- Database Schema dan Relationships
- Order Status Tracking
- Automatic Pricing Calculation

16.2 Technical Achievements

- Laravel 12 dengan Filament v3 implementation
- MySQL database dengan proper relationships
- RESTful API dengan comprehensive documentation
- Responsive frontend dengan Blade templates
- Role-based access control (RBAC)
- Real-time inventory management
- Docker containerization
- L5-Swagger API documentation

16.3 System Capabilities

- Multi-user Support: Customer dan admin dengan akses berbeda
- Real-time Operations: Instant availability updates dan order processing
- Scalable Architecture: Modular design untuk pengembangan lebih lanjut
- API-First Approach: Siap untuk integrasi mobile app
- Comprehensive Logging: Activity tracking untuk audit trails
- Data Integrity: Foreign key constraints dan validation rules

17. Conclusion

Music Rental System telah berhasil dikembangkan sesuai dengan business requirements dan technical specifications. Sistem ini menyediakan solusi lengkap untuk bisnis rental alat musik dengan:

1. End-to-End Customer Journey: Dari discovery hingga booking completion
2. Powerful Admin Tools: Comprehensive management capabilities
3. API Integration: Siap untuk pengembangan mobile app selanjutnya
4. Scalable Architecture: Dapat dikembangkan untuk fitur tambahan
5. Security & Performance: Dibangun dengan best practices Laravel

Sistem ini siap untuk deployment dan dapat langsung digunakan untuk operasional bisnis rental alat musik dengan kapasitas untuk pengembangan dan integrasi di masa depan.

Document Version: 1.0.0

Last Updated: 22 Juli 2025

Author: Fernanda

Status: Production Not Ready Yet