

# Detailed Project Proposal

## Development of a Laravel and Bootstrap-Based Web System for Aircraft Scheduling, Passenger Eligibility Verification, and Invoicing

---

### 1. Project Title

#### Integrated Web System for Aircraft Scheduling, Passenger Eligibility Verification, and Invoicing Using Laravel and Bootstrap

---

### 2. Executive Summary

This proposal outlines the design and development of a secure, responsive, and modular **web-based system** built using **Laravel (PHP)** and **Bootstrap (CSS framework)** to automate aircraft scheduling, passenger verification, and invoicing. The system will ensure compliance with immigration policies by verifying passport and visa documentation and streamline airline operations through efficient scheduling, invoicing, and reporting functionalities.

---

### 3. Objectives

- Provide a **responsive dashboard interface** for managing flights, passengers, and documents.
  - Ensure **residency-based eligibility** verification:
    - Residents: Passport only.
    - Non-residents: Passport and Visa.
  - Automate **invoice generation** and **track payments** by passengers or aircraft operators.
  - Securely store and manage uploaded documents (passports, visas).
  - Offer a **role-based access system** for different users (admin, immigration officers, operators, handlers).
  - Deliver real-time alerts, reports, and analytics.
- 

### 4. Target Users and Roles

Role	Responsibilities
<b>Admin</b>	System setup, user creation, role assignment, full access
<b>Handler</b>	Schedule flights, assign aircraft details
<b>Operator</b>	Register passengers, upload documents

<b>Immigration Officer</b>	Verify documents and approve/flag passengers
<b>Passenger (optional)</b>	View personal information and payment details (if portal is open to them)

---

## **5. System Features**

### **A. Aircraft Management**

- Flight Number
- Aircraft Model & Manufacturer
- Capacity
- Assigned Handler (Name, Email, Phone)
- Operator (Name, Email, Phone)
- Scheduled Date and Time

### **B. Passenger Registration & Eligibility Verification**

- Full Name, Email, Phone Number
- Residency Status (Resident / Non-resident)
- Upload Passport (all)
- Upload Visa (only for non-residents)
- System checks required documents automatically
- Flag incomplete or missing documentation
- Approval workflow for immigration officers

### **C. Invoicing & Payment**

- Auto-generated Invoice Number
- Linked to Flight or Passenger
- Amount & Date
- Payment Date
- Payment Method (Credit Card, Mobile Money, Bank Transfer)
- Payment Status (Paid/Unpaid)
- Export Invoice as PDF

### **D. Notifications & Alerts**

- Email notifications for:
  - Missing documents
  - Upcoming flight deadlines
  - Payment reminders
- Dashboard alerts for admin and officers

### **E. Reports and Analytics**

- Flight manifests
- Document compliance reports
- Payment summaries
- Export as CSV or PDF

---

## 6. System Development Approach

### A. Technology Stack

Layer	Technology
Frontend	HTML5, CSS3, Bootstrap 5, JavaScript
Backend	Laravel 10 (PHP MVC Framework)
Database	MySQL (Relational DB)
Authentication	Laravel Jetstream (with roles and permissions)
Document Storage	Laravel Filesystem (local or AWS S3)
PDF Generation	Laravel DomPDF / Snappy
Payment Integration	Stripe, PayPal, or Local Mobile Money APIs
Hosting	Apache or NGINX with MySQL on AWS, DigitalOcean, or shared hosting

---

### B. Database Design

#### 1. Key Tables

Table	Description
users	Stores user information, roles
flights	Stores flight number, aircraft details, handler info
passengers	Stores passenger name, contact, residency status
documents	Stores passport and visa file references
invoices	Stores invoice number, amount, date, status
payments	Tracks payment date, amount, and method
roles	Defines system roles (admin, operator, etc.)
notifications	Stores alerts and reminders sent to users

#### 2. Relationships

- One flight has many passengers
- One invoice is linked to one flight or one passenger
- One user has one role
- One passenger has many documents

---

## **C. Laravel Modules & Components**

### **1. Authentication**

- Laravel Jetstream for secure login and role-based dashboard access

### **2. Aircraft Scheduling Module**

- Aircraft CRUD (Create, Read, Update, Delete)
- Date/time pickers for flight scheduling
- Handlers and Operators assignment forms

### **3. Passenger Module**

- Passenger registration form
- File upload (passport & visa)
- Conditional file requirement (Visa required only if non-resident)
- Auto-check logic with validation rules
- Document status flags

### **4. Eligibility Verification Module**

- Admin/Immigration panel for document review
- Approve, reject, or request more information
- Filter and sort passengers by document status

### **5. Invoice & Payment Module**

- Auto-generate invoice on passenger registration or flight scheduling
- Store payment details: amount, method, date
- Payment status management
- Generate downloadable PDF invoice

### **6. Notifications Module**

- Scheduled jobs for alerts (via Laravel Scheduler)
- Email notifications using Laravel Mail
- Dashboard alerts

### **7. Reporting Module**

- Filterable tables using Laravel Datatables
- Export reports to CSV and PDF

---

## 7. Development Phases & Timeline

Phase	Duration	Activities
<b>Phase 1:</b> Requirements Gathering	Week 1–2	Client meetings, final feature scope, UI sketches
<b>Phase 2:</b> UI/UX Design with Bootstrap	Week 3	Wireframing, page layout, dashboard structure
<b>Phase 3:</b> Backend Development (Laravel)	Week 4–7	Database models, routes, controllers, core logic
<b>Phase 4:</b> Document Upload & Verification	Week 8	File handling, eligibility logic, immigration flow
<b>Phase 5:</b> Invoicing & Payment Integration	Week 9	Invoice engine, payment API integration
<b>Phase 6:</b> Testing & Bug Fixing	Week 10	Manual testing, feature testing, QA checks
<b>Phase 7:</b> Deployment & Training	Week 11	Live server setup, documentation, stakeholder training
<b>Phase 8:</b> Maintenance & Feedback Loop	Week 12+	Continuous updates, support, feature improvements

---

## 8. Security Measures

- **CSRF and XSS protection** via Laravel security middleware
  - **Encrypted document storage** for passports and visas
  - **Role-based access control (RBAC)** using Laravel's Gate and Policies
  - **Secure API endpoints** for document and payment interactions
  - **Validation** for all forms and uploads
- 

## 9. Expected Deliverables

- Fully functional web system (hosted online)
  - Admin documentation and training manual
  - Source code (GitHub/private repo)
  - System design documents (ERD, workflows, use cases)
  - User guide for system roles
- 

## 11. Conclusion

This Laravel + Bootstrap-based web platform will streamline the entire process of aircraft scheduling, passenger eligibility verification, and financial transactions. It will ensure regulatory compliance, minimize manual errors, and improve operational transparency for aviation stakeholders.