



## Project Specification Document: SkillHub - Freelance Marketplace Platform

---

### 1. Project Overview

**Project Name:** SkillHub

**Type:** Full-stack Web Application

**Framework:** Laravel 11 (PHP)

**Frontend:** Laravel Blade / Livewire or Inertia.js + Vue 3

**Database:** MySQL / PostgreSQL

**Deployment:** Docker / Laravel Forge / Laravel Vapor

**Version Control:** Git + GitHub

**Objective:**

To build a freelance marketplace where clients can post projects, and freelancers can bid, get hired, deliver work, and get paid. This application will support a scalable, secure, and modular architecture suitable for production.

---

### 2. User Roles

1. **Admin**

2. Manage users (clients and freelancers)
3. Manage categories, tags, and reported content
4. View analytics (platform usage, transactions)

5. **Client**

6. Create and manage projects
7. Browse freelancers and proposals
8. Hire freelancers and communicate
9. Leave reviews and ratings

10. **Freelancer**

11. Create and manage profile and portfolio
  12. Browse projects
  13. Submit proposals
  14. Deliver work and get paid
  15. Respond to messages and reviews
-

### **3. Functional Requirements**

#### **3.1 Authentication & Authorization**

- Register/Login (email verification)
- Role-based access control (Spatie Laravel Permissions)
- Password reset, profile settings

#### **3.2 User Profile Management**

- Profile image, bio, location, experience
- Skills and tags
- Portfolio section (images, descriptions)

#### **3.3 Project Management**

- Clients can post jobs with:
  - Title, Description, Budget, Deadline, Category
  - Attached files (optional)
- Project listing with filters (category, budget, deadline)

#### **3.4 Proposal & Hiring System**

- Freelancers submit:
  - Cover letter, bid amount, estimated time
- Clients can:
  - Review proposals, shortlist candidates, hire
- Statuses: Open > In Progress > Submitted > Approved > Completed

#### **3.5 Messaging System**

- Real-time messaging (Laravel Echo + Pusher or WebSockets)
- File attachments, typing indicator

#### **3.6 Payments (Stripe Integration)**

- Client pays when hiring (escrow model)
- Upon project approval, freelancer receives payment
- Admin manages disputes and commissions

#### **3.7 Reviews and Ratings**

- Both sides rate each other post-project
- Public profile rating visible to other users

#### **3.8 Admin Panel**

- Dashboard (users, revenue, disputes, growth metrics)
- CRUD for categories, tags, reported content

- Manual override on disputes and blocked users
- 

#### 4. Non-Functional Requirements

- **Security:** CSRF, XSS protection, validation, role checks
  - **Performance:** Caching with Redis, queues for email/notifications
  - **Scalability:** Modular structure, service providers, Horizon for queues
  - **Responsiveness:** Mobile-first design with Tailwind CSS
  - **Localization:** Multi-language support (EN, optionally others)
- 

#### 5. Technical Stack

- **Backend:** Laravel 11
  - **Frontend:** Blade + Livewire OR Inertia.js + Vue 3
  - **Database:** MySQL or PostgreSQL
  - **Search:** Laravel Scout + MeiliSearch
  - **File Storage:** Laravel Filesystem (Local, S3-ready)
  - **Real-time:** Pusher / WebSockets / Laravel Echo
  - **CI/CD:** GitHub Actions for tests + deployment
  - **Environment:** Laravel Sail (Docker), Laravel Forge for production
- 

#### 6. Project Phases & Milestones

1. **Phase 1 - Planning & Setup**
2. Project scaffolding
3. Auth system with role management
4. **Phase 2 - Core Features**
5. Project CRUD
6. Proposal system
7. Messaging system
8. **Phase 3 - Payments & Reviews**
9. Stripe payment integration
10. Review and rating module
11. **Phase 4 - Admin Dashboard**
12. Analytics, user moderation, category management

### 13. Phase 5 - Deployment & Optimization

- 14. Final QA
- 15. CI/CD setup
- 16. Server configuration

---

**7. Deliverables** - Complete Laravel codebase (GitHub) - ER Diagram and database migration files - REST API documentation (Postman collection or Swagger) - Admin credentials and test user accounts - Deployment-ready Docker file or Laravel Forge link

---

**8. Optional Features (Future Work)** - Mobile App (Flutter or React Native) - AI-generated proposals using OpenAI API - SMS/email notification system - Gamified freelancer achievements

---

**Prepared By:** [Your Team Name]

**Date:** [Insert Date]