

# Local Scout – Neighbourhood Service Finder

## Project Description

Local Scout is a web application built with ASP.NET Core and MySQL that connects local users with service providers in their neighborhood. Users can search for services, view provider profiles, and book services securely, while service providers manage their offerings, track bookings, and receive payments. Using LocationIQ, the system shows results based on proximity, and SSLCommerz handles secure payments. The goal is to create a reliable, location-aware platform that simplifies finding trusted local professionals while providing a smooth experience for providers and admins.

## Requirement Analysis:

### User and System Requirements

#### 1. Users require secure access to accounts

- 1.1 Login and registration pages with email/phone and password
- 1.2 Password recovery/reset via email or OTP
- 1.3 Role-based access control (User, Service Provider, Admin)

#### 2. Users can search and book services

- 2.1 Search services using keywords and categories
- 2.2 Results sorted by proximity using LocationIQ
- 2.3 View detailed provider profiles (ratings, reviews, availability, pricing)
- 2.4 Book services and track status (Pending, Confirmed, Completed)
- 2.5 AJAX-powered “Load More” functionality for dynamic service browsing

#### 3. Service Providers manage listings and bookings

- 3.1 Register with additional info: company, license, service category, address
- 3.2 Create, update, and delete service listings
- 3.3 Approve and manage bookings
- 3.4 Track booking history and analytics (total jobs, ratings, earnings)
- 3.5 Receive payments via SSLCommerz

3.6 Request for new service category.

#### **4. Admins oversee operations and system integrity**

- 4.1 CRUD operations for all users and service providers
- 4.2 Approve or reject provider registrations
- 4.3 Generate reports for transactions, service usage, and user activity
- 4.4 Handle complaints, disputes, and review moderation
- 4.5 Manage, Approve/Reject service categories

### **Functional Requirements**

- 1. Role-based authentication and secure login
- 2. Location-based service discovery (search by category, keyword) and proximity sorting
- 3. Payment processing with SSLCommerz (cards, Bkash, Nagad, internet banking)
- 4. Booking lifecycle management (create, approve, cancel, complete)
- 5. Service provider dashboards with analytics
- 6. User dashboards with booking and payment tracking
- 7. Detailed transaction history dashboards for users and providers
- 8. Full public-facing profiles for users and service providers
- 9. Ratings and reviews for completed services
- 10. Notifications for booking and payment updates
- 11. Dynamic provider master profile pages
- 12. Timezone-aware date/timestamp handling

### **Non-Functional Requirements**

- 1. Security: Password hashing, SSL/TLS encryption, role-based access control
- 2. Performance: Fast service search and booking ( $\leq 3$  seconds), optimized queries
- 3. Usability: Responsive interface, clear error messages, and intuitive dashboards
- 4. Scalability: Support for additional cities, categories, and providers
- 5. Reliability: 99% uptime, UTC storage with site-wide timezone conversion
- 6. Maintainability: MVC architecture, modular code, configurable API keys

## Core Features & Functionality

1. Booking & Payment System: SSLCommerz integration, IPN confirmation, negotiable price workflow
2. Unified Experience & UI/UX: Fixed sidebar dashboards, active navigation, role-based redirects
3. Service Provider Portal: Master profile, dynamic layouts, trust stats, modern management pages
4. User Dashboard & Public Pages: AJAX-based search, profile management, detailed service pages

## Tech Stack & Tools

- ASP.NET Core – Backend and MVC framework
- MySQL – Primary relational database
- SSLCommerz – Payment gateway integration
- LocationIQ – Geocoding and distance calculations
- Cloudinary – Image storage and management

## Database Structure

