HireLink - Academic Project Documentation

Table of Contents

HireLink - Local Service Provider Platform

Academic Project Documentation

UIT,ADOOR,Academic Year 2025-2026

A Full-Stack Web Application for Connecting Customers with Local Service Providers

---

Submitted in partial fulfillment of the requirements for the BCA,

Graduate Academic Project

Project Team: HireLink Development Team

Submission Date: January 2026

Table of Contents

---

1. Executive Summary

HireLink is a comprehensive full-stack web application designed to bridge the gap between customers seeking home services and verified local service providers. The platform facilitates seamless booking, tracking, and management of services such as electrical work, plumbing, carpentry, cleaning, painting, and more.

Key Highlights

Aspect	Description
Platform Type	Web-based Service Marketplace
Target Users	Customers, Service Providers, Administrators
Primary Services	Electrical, Plumbing, Carpentry, Cleaning, Painting, AC & Appliances
Technology Stack	React.js, Spring Boot, MySQL, Docker
Architecture	Microservices-ready Monolith

Project Objectives

1. Create a user-friendly platform for booking home services
2. Implement a robust verification system for service providers
3. Enable real-time booking management and tracking
4. Provide transparent pricing and rating mechanisms
5. Ensure secure authentication and data protection

---

2. Introduction

2.1 Problem Statement

The home services industry faces significant challenges in connecting customers with reliable service providers. Traditional methods of finding electricians, plumbers, or cleaners often result in:

- **Lack of transparency** in pricing and service quality
- **Difficulty in verifying** provider credentials
- **Inefficient scheduling** and booking processes
- **No standardized feedback** mechanisms
- **Trust deficit** between customers and providers

2.2 Proposed Solution

HireLink addresses these challenges by providing a centralized digital platform that:

1. **Verifies Service Providers** through KYC documentation and background checks

2. **Standardizes Pricing** with transparent cost structures
3. **Enables Easy Booking** with intuitive scheduling interfaces
4. **Implements Rating Systems** for quality assurance
5. **Provides Real-time Tracking** of service status

2.3 Scope of the Project

In Scope

- User registration and authentication (Customers, Providers, Admins)
- Service category management
- Booking creation, tracking, and management
- Provider profile and service management
- Review and rating system
- Location-based service discovery
- Multi-role dashboard interfaces

Out of Scope (Future Enhancements)

- Payment gateway integration
- Real-time chat/messaging
- Mobile native applications
- AI-based service recommendations

2.4 Target Audience

User Type	Description
Customers	Individuals seeking home services for residential needs
Service Providers	Skilled professionals offering home services
Administrators	Platform managers overseeing operations and verification

---

3. Software Requirements Specification (SRS)

3.1 Functional Requirements

3.1.1 User Management Module

ID	Requirement	Priority
FR-001	Users shall be able to register with email/phone and password	High
FR-002	Users shall be able to login using credentials	High
FR-003	System shall support OTP-based	High

ID	Requirement	Priority
	verification	
FR-004	Users shall be able to update their profile information	Medium
FR-005	Users shall be able to manage multiple addresses	Medium
FR-006	System shall support role-based access (Customer, Provider, Admin)	High

3.1.2 Service Provider Module

ID	Requirement	Priority
FR-007	Providers shall be able to create and manage service listings	High
FR-008	Providers shall submit KYC documents for verification	High
FR-009	Providers shall set availability schedules	Medium
FR-010	Providers shall define service areas by pincode/radius	Medium
FR-011	Providers shall accept/reject booking requests	High
FR-012	System shall track provider statistics (ratings, bookings, earnings)	High

3.1.3 Service Category Module

ID	Requirement	Priority
FR-013	System shall support hierarchical service categories	High
FR-014	Categories shall have configurable pricing structures	Medium
FR-015	Featured categories shall be highlighted on homepage	Low
FR-016	Categories shall support multiple pricing units (per hour, per sqft, fixed)	Medium

3.1.4 Booking Module

ID	Requirement	Priority
FR-017	Customers shall be able to create service bookings	High
FR-018	System shall generate unique booking numbers	High

ID	Requirement	Priority
FR-019	Bookings shall support multiple status states	High
FR-020	Customers shall track booking status in real-time	High
FR-021	System shall prevent duplicate active bookings	Medium
FR-022	Bookings shall support cancellation with reason tracking	Medium

3.1.5 Review and Rating Module

ID	Requirement	Priority
FR-023	Customers shall rate completed services	High
FR-024	System shall support multi-dimensional ratings	Medium
FR-025	Reviews shall be visible on provider profiles	High
FR-026	System shall calculate and display average ratings	High

3.1.6 Search and Discovery Module

ID	Requirement	Priority
FR-027	Users shall search services by keyword	High
FR-028	Users shall filter by category, rating, and price	Medium
FR-029	System shall support location-based provider discovery	High
FR-030	Featured providers shall be highlighted	Low

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

ID	Requirement	Metric
NFR-001	Page load time shall be under 3 seconds	< 3s
NFR-002	API response time shall be under 500ms	< 500ms
NFR-003	System shall support 1000 concurrent users	1000 users
NFR-004	Database queries shall execute under 100ms	< 100ms

3.2.2 Security Requirements

ID	Requirement	Implementation
NFR-005	Passwords shall be encrypted using bcrypt	BCrypt hashing
NFR-006	API endpoints shall be secured with JWT tokens	JWT Authentication

ID	Requirement	Implementation
NFR-007	Sensitive data shall be encrypted at rest	AES Encryption
NFR-008	System shall implement role-based access control	Spring Security RBAC

3.2.3 Usability Requirements

ID	Requirement
NFR-009	UI shall be responsive across devices (mobile, tablet, desktop)
NFR-010	System shall provide clear error messages
NFR-011	Navigation shall be intuitive with maximum 3 clicks to any feature

3.2.4 Reliability Requirements

ID	Requirement	Target
NFR-012	System uptime shall be 99.5%	99.5%
NFR-013	Data backup frequency	Daily
NFR-014	Maximum planned downtime	4 hours/month

3.3 Use Case Specifications

Use Case UC-001: User Registration

Field	Description
Use Case ID	UC-001
Name	User Registration
Actors	Customer, Service Provider
Precondition	User has valid email/phone number
Main Flow	1. User navigates to registration page2. User selects account type (Customer/Provider)3. User enters required details4. System validates input5. System sends OTP verification6. User verifies OTP7. Account is created
Postcondition	User account is created and verified
Alternative Flow	If validation fails, display error and allow retry

Use Case UC-002: Book a Service

Field	Description
Use Case ID	UC-002
Name	Book a Service
Actors	Customer
Precondition	Customer is logged in
Main Flow	1. Customer browses/searches services2. Customer

Field	Description
	selects a service3. Customer views service details and provider profile4. Customer clicks “Book Now”5. Customer selects date and time6. Customer enters service address7. Customer describes the issue8. System creates booking with PENDING status9. Provider receives notification
Postcondition	Booking is created with unique booking number

Use Case UC-003: Manage Booking (Provider)

Field	Description
Use Case ID	UC-003
Name	Manage Booking
Actors	Service Provider
Precondition	Provider is logged in, has pending bookings
Main Flow	1. Provider views pending bookings2. Provider reviews booking details3. Provider accepts or rejects booking4. If accepted, status changes to ACCEPTED5. Provider can update status to IN_PROGRESS6. Provider completes work and updates to COMPLETED7. Customer is notified at each status change
Postcondition	Booking status is updated

---

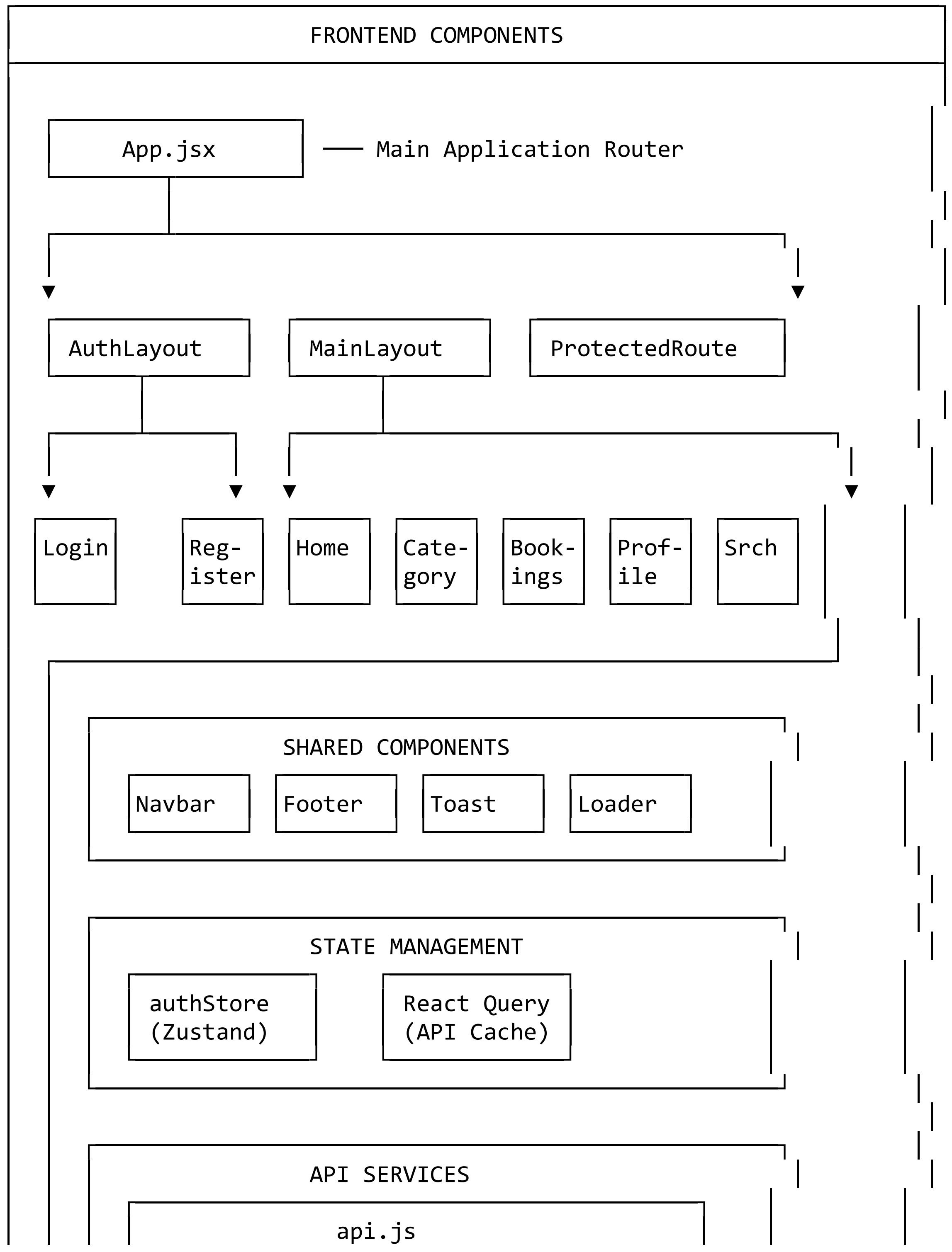
4. System Architecture

4.1 High-Level Architecture

```
┌──────────────────────────────────────────────────────────────────────┐
│                           CLIENT LAYER                                │
├──────────────────────────────────────────────────────────────────────┤
│                                                                     │
│  ┌──────────────┐  ┌──────────────┐  ┌──────────────┐               │
│  │   Customer   │  │   Provider   │  │    Admin     │               │
│  │   Browser    │  │   Browser    │  │   Browser    │               │
│  └──────┬───────┘  └──────┬───────┘  └──────┬───────┘                │
│         │                 │                 │                        │
│         └─────────────────┼─────────────────┘                       │
│                           ▼                                         │
│  ┌──────────────────────────────────────────────────────────────┐  │   │
│  │              REACT.JS FRONTEND APPLICATION                  │ │   │
│  │  ┌─────────┐  ┌─────────┐│ ┌─────────┐ ┌─────────┐        │   │
│  │  │  Vite   │  │TailwindCSS│ │ React   │ │ │ Zustand │ │      │   │
│  │  │ Bundle  │  │  Styles │ │ │ Query   │ │ State   │         │   │
│  │  └─────────┘  └─────────┘  └─────────┘  └─────────┘         │   │
│  └──────────────────────────────────────────────────────────────┘    │
│                                                                     │
```

```
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼ HTTPS / REST API
┌─────────────────────────────────────────────────────────────────────┐
│                        API GATEWAY LAYER                            │
├─────────────────────────────────────────────────────────────────────┤
│  ┌───────────────────────────────────────────────────────────────┐  │
│  │                   NGINX REVERSE PROXY                         │  │
│  │           (Load Balancing, SSL Termination)                   │  │
│  └───────────────────────────────────────────────────────────────┘  │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│                     APPLICATION LAYER                               │
├─────────────────────────────────────────────────────────────────────┤
│  ┌───────────────────────────────────────────────────────────────┐  │
│  │              SPRING BOOT 3.2 BACKEND                          │  │
│  │  ┌─────────────────────────────────────────────────────────┐  │  │
│  │  │                 CONTROLLER LAYER                        │  │  │
│  │  │  AuthController │ BookingController │ ServiceController│  │  │
│  │  │  UserController │ ProviderController│ CategoryController│ │  │
│  │  └─────────────────────────────────────────────────────────┘  │  │
│  │  ┌─────────────────────────────────────────────────────────┐  │  │
│  │  │                  SERVICE LAYER                          │  │  │
│  │  │  AuthService  │ BookingService  │ ProviderService       │  │  │
│  │  │  UserService  │ EmailService    │ LocationService       │  │  │
│  │  └─────────────────────────────────────────────────────────┘  │  │
│  │  ┌─────────────────────────────────────────────────────────┐  │  │
│  │  │                 SECURITY LAYER                          │  │  │
│  │  │  JWT Authentication │ BCrypt Encryption │ RBAC          │  │  │
│  │  └─────────────────────────────────────────────────────────┘  │  │
│  │  ┌─────────────────────────────────────────────────────────┐  │  │
│  │  │                REPOSITORY LAYER                         │  │  │
│  │  │  UserRepository │ BookingRepository │ ServiceRepository│  │  │
│  │  └─────────────────────────────────────────────────────────┘  │  │
│  └───────────────────────────────────────────────────────────────┘  │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼ JPA / Hibernate
┌─────────────────────────────────────────────────────────────────────┐
│                        DATA LAYER                                   │
├─────────────────────────────────────────────────────────────────────┤
│  ┌───────────────────────────────────────────────────────────────┐  │
│  │                    MySQL 8.0 DATABASE                        │  │
│  │  ┌─────────┐ ┌─────────┐ ┌─────────┐ ┌─────────┐             │  │
│  │  │  users  │ │bookings │ │services │ │ reviews │             │  │
│  │  └─────────┘ └─────────┘ └─────────┘ └─────────┘             │  │
│  │  ┌─────────────┐ ┌───────────────┐ ┌─────────────────┐       │  │
│  │  │  providers  │ │  categories   │ │  user_addresses │       │  │
```

4.2 Component Architecture



```
│  │  │  │  authAPI │ bookingsAPI │ categoriesAPI  │    │         │
│  │  │  │  servicesAPI │ providersAPI             │    │         │
│  │  │  └─────────────────────────────────────────┘    │         │
│  │  └───────────────────────────────────────────────────┘         │
└──┴─────────────────────────────────────────────────────────────────┘
```

4.3 Technology Stack

Layer	Technology	Version	Purpose
Frontend Framework	React.js	18.x	UI Component Library
Build Tool	Vite	5.x	Fast Development & Build
Styling	TailwindCSS	3.x	Utility-First CSS
State Management	Zustand	4.x	Lightweight State Store
API Client	React Query	5.x	Data Fetching & Caching
Icons	Heroicons	2.x	SVG Icon Library
Date Handling	date-fns	2.x	Date Manipulation
Backend Framework	Spring Boot	3.2.x	REST API Framework
Language	Java	17	Backend Programming
Security	Spring Security	6.x	Authentication & Authorization
JWT	jjwt	0.12.x	Token Generation
ORM	Hibernate/JPA	6.x	Object-Relational Mapping
Database	MySQL	8.0	Relational Database
Build Tool	Maven	3.9.x	Dependency Management
Containerization	Docker	24.x	Application Containers
Orchestration	Docker Compose	2.x	Multi-Container Management
Web Server	Nginx	1.25.x	Reverse Proxy & Static Files

5. Data Flow Diagrams (DFD)

5.1 Context Diagram (Level 0 DFD)

```
                                          ┌─────────────────────────────────┐
                                          │                                 │
        User Registration                 │                                 │    Service
Listings
        Login Credentials ───►│                                 │◄─── Provide
r Details
        Service Search                    │                                 │   KYC Doc
uments
        Booking Requests                  │          HIRELINK SYSTEM        │
        Reviews & Ratings                 │                                 │
                                          │      (Local Service Provider    │
        ◄─── Authentication               │           Platform)             │───► Bookin
g Notifications
             Search Results               │                                 │    Status
 Updates
             Booking Status               │                                 │    Earnin
gs Reports
             Service Details              │                                 │
                                          │                                 │
                                         └─────────────────────────────────┘
    ┌──────────────┐                                                          ┌───
───────────┐
    │   CUSTOMER   │                            │                             │
PROVIDER   │
    │              │                            │                             │
           │
    └──────────────┘                            │                             └───
───────────┘
                                                │
                                                ▼
                                        ┌──────────────┐
                                        │ ADMINISTRATOR│
                                        │              │
                                        │  - User Mgmt │
                                        │  - KYC Verify│
                                        │  - Reports   │
                                        └──────────────┘
```

5.2 Level 1 DFD - Main Processes

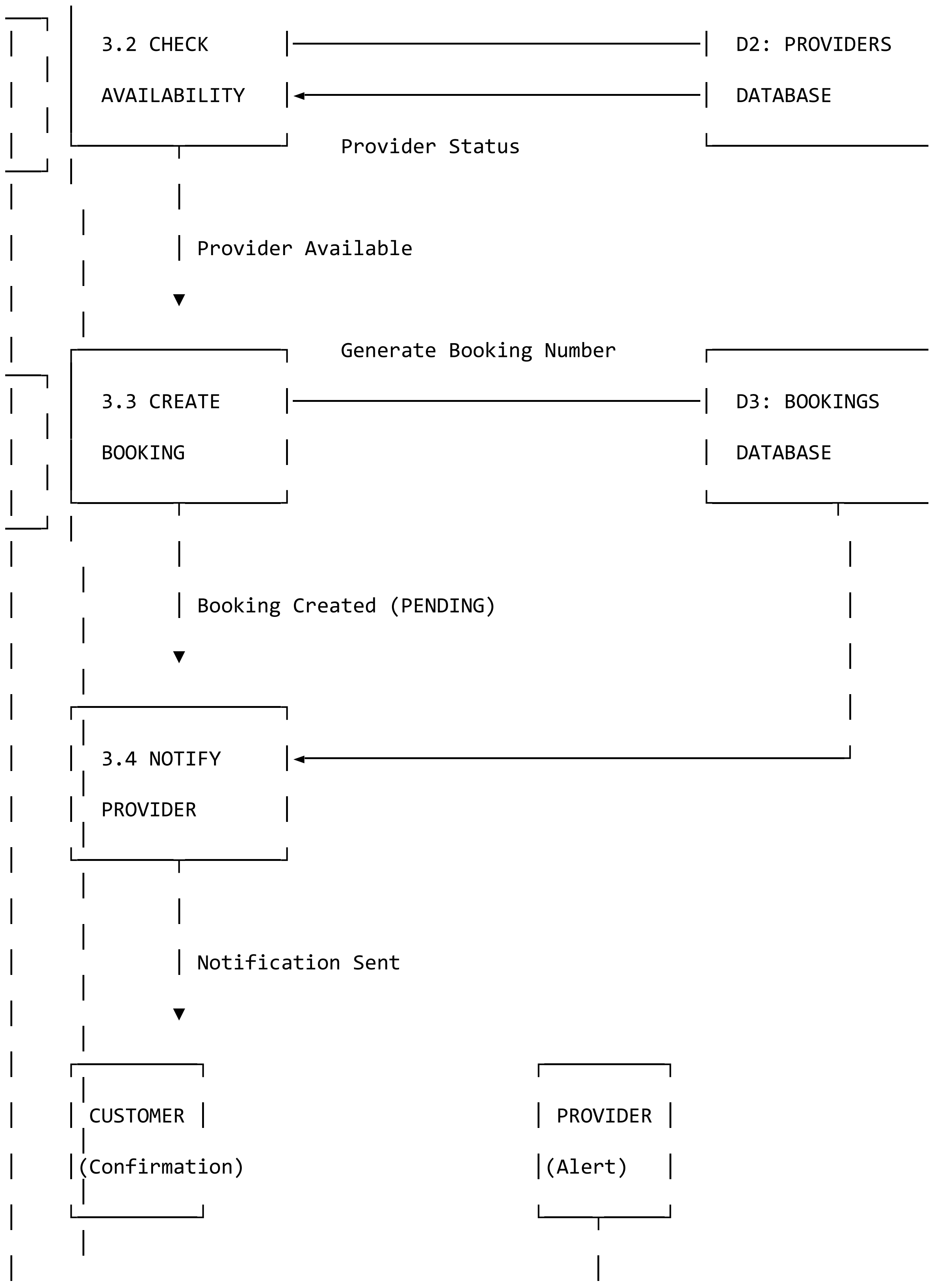
```
┌───────────────────────────────────────────────────────────────────────────
──────┐
│
      │
│    ┌──────────┐
      │
│    │ CUSTOMER │
      │
```

```
│    └────┬─────┘
      │
│         │
      │
│         │ Registration Data, Login Credentials
      │
│         ▼
      │
│    ┌─────────────────┐
│    │   1.0 USER      │ ──────────────────────────────────────────┐
│    │   MANAGEMENT    │                                            │
      │
│    └─────────┬───────┘                                            │
      │
│              │                                                     │
       │
│              │ User Profile, Auth Token                            │
       │
│              ▼                                                     │
       │
│    ┌─────────────────┐      Search Query      ┌───────────▼────────
    │
┐    │   2.0 SERVICE   │◄───────────────────────│   D1: USERS
│   │
│    │   DISCOVERY     │                        │   DATABASE
│   │
┘    └─────────┬───────┘                        └───────────┬────────
    │
│              │                                             │
     │
│              │ Service List, Provider Details              │
     │
│              ▼                                             │
     │
│    ┌─────────────────┐      Booking Request   ┌───────────▼────────
    │
┐    │   3.0 BOOKING   │◄───────────────────────│   D2: SERVICES
│   │
│    │   MANAGEMENT    │────────────────────────│   DATABASE
│   │
┘    └─────────┬───────┘      Booking Created   └───────────┬────────
    │
│              │                                             │
     │
│              │ Booking Confirmation, Status Updates        │
     │
│              ▼                                             │
     │
```

```
│    ┌─────────────────┐        Review Data          ┌───────────▼────────
┐   │                 │                             │
│   │  4.0 REVIEW     │─────────────────────────────│   D3: BOOKINGS
│   │                 │                             │
│   │  SYSTEM         │                             │   DATABASE
│   │                 │                             │
┘   │└─────────┬───────┘                             └───────────┬────────
│    │          │                                                │
│    │          │ Review Confirmation                            │
│    │          ▼                                                │
│    │┌──────────┐                                   ┌───────────▼────────
┐   │                                                │
│   ││ CUSTOMER │                                    │   D4: REVIEWS
│   │└──────────┘                                    │   DATABASE
│   │
┘   │                                                └────────────────────
│     │
│    ┌──────────┐
│    ││         │
│    │ PROVIDER │
│    ││         │
│    └────┬─────┘
│     │    │
│     │    │ Service Listings, KYC Documents
│     │    ▼
│     │
│    ┌─────────────────┐
│    │  5.0 PROVIDER   │
│    │  MANAGEMENT     │
│    └────────┬────────┘
│    │         │
│     │        │ Verification Status, Booking Notifications
│     │        ▼
│     │
│        ┌──────────┐
│        │
│        │ PROVIDER │
│        │
│        └──────────┘
│        │
│        │
└──────────────────────────────────────────────────────────────────────────────
─────────┘
```

5.3 Level 2 DFD - Booking Management Process

```
┌──────────────────────────────────────────────────────────────────────────────
─────────┐
│                         3.0 BOOKING MANAGEMENT (Expanded)
│        │
├──────────────────────────────────────────────────────────────────────────────
─────────┤
│
│        │
│        │
│        ┌──────────┐
│        │
│        │ CUSTOMER │
│        │
│        └──────────┘
│        │   │
│        │   │
│            │ Service ID, Schedule, Address, Issue Details
│        │
│            ▼
│        │
│        ┌─────────────────┐
│
│          3.1 VALIDATE    │
│
│          BOOKING DATA    │
│
│        └─────────────────┘
│        │       │
│                │
│        │
│                │ Validated Data
│        │
│                ▼
│        │
│        ┌─────────────────┐    Check Provider Availability    ┌──────────────────
```


3.2 CHECK AVAILABILITY
D2: PROVIDERS DATABASE
Provider Status
Provider Available
Generate Booking Number
3.3 CREATE BOOKING
D3: BOOKINGS DATABASE
Booking Created (PENDING)
3.4 NOTIFY PROVIDER
Notification Sent
CUSTOMER (Confirmation)
PROVIDER (Alert)

```
          │
│         │                                      │ Accept/Reject Response
│         │                                      ▼
│         │                                 ┌──────────────┐
│         │                                 │  3.5 UPDATE   │
│         │                                 │ BOOKING STATUS│
│         │                                 └──────┬───────┘
│         │                                        │
│         │                      ┌─────────────────┼─────────────────┐
│         │                      │                 │                 │
│         │                      ▼                 ▼                 ▼
│         │                  ┌────────┐        ┌────────┐        ┌─────────
─┐      │                  │ACCEPTED│        │REJECTED│        │IN_PROGRE
│ SS│     │                  └────────┘        └────────┘        └─────────
│         │
─┘      │                                                            │
│         │                                                            ▼
│         │                                                        ┌─────────
─┐      │                                                        │COMPLETED
│ │     │                                                        └─────────
─┘      │
│         │
└──────────────────────────────────────────────────────────────────────────
─────────┘
```

---

6. Entity-Relationship Diagrams

6.1 Complete ER Diagram

```
┌──────────────────────────────────────────────────────────────────────────
──────────────┐
```

```
│
            │
│                               HIRELINK DATABASE ER DIAGRAM
            │
│
            │
├──────────────────────────────────────────────────────────────────────────
────────────┤
│
            │
│    ┌────────────────────────┐            ┌────────────────────────┐
           │
│    │       USERS            │            │  USER_ADDRESSES        │
           │
│    ├────────────────────────┤            ├────────────────────────┤
           │
│    │ PK user_id             │──────────┤ │ PK address_id          │
           │
│    │    name                │    1:N    ││ FK user_id             │
           │
│    │    email (UK)          │           ││    address_type        │
           │
│    │    phone (UK)          │           ││    address_line1       │
           │
│    │    password_hash       │           ││    city                │
           │
│    │    profile_image_url   │           ││    state               │
           │
│    │    date_of_birth       │           ││    pincode             │
           │
│    │    gender              │           ││    latitude            │
           │
│    │    user_type           │           ││    longitude           │
           │
│    │    account_status      │           ││    is_default          │
           │
│    │    is_email_verified   │           │└────────────────────────┘
           │
│    │    is_phone_verified   │
           │
│    │    preferred_language  │
           │
│    │    created_at          │
           │
│    │    updated_at          │
           │
│    └────────────┬───────────┘
           │
│                 │
            │
```

```
│                │ 1:1
            │
│                ▼
            │
│    ┌────────────────────────┐            ┌────────────────────────┐
           │
│    │  SERVICE_PROVIDERS     │            │  SERVICE_CATEGORIES    │
           │
│    ├────────────────────────┤            ├────────────────────────┤
           │
│    │ PK provider_id         │            │ PK category_id         │
           │
│    │ FK user_id (UK)        │            │    category_name (UK) │◄───────┐
           │
│    │    business_name       │            │    category_slug (UK) │        │ Se
lf-Ref    │
│    │    business_description│            │    category_description│        │
(Parent)  │
│    │    tagline             │            │    category_icon       │        │
           │
│    │    experience_years    │            │ FK parent_category_id │────────┘
           │
│    │    specializations     │            │    category_level      │
           │
│    │    base_latitude       │            │    display_order       │
           │
│    │    base_longitude      │            │    min_base_price      │
           │
│    │    service_radius_km   │            │    max_base_price      │
           │
│    │    aadhaar_encrypted   │            │    price_unit          │
           │
│    │    pan_encrypted       │            │    is_active           │
           │
│    │    kyc_status          │            │    is_featured         │
           │
│    │    average_rating      │            │    created_at          │
           │
│    │    total_bookings      │            │    updated_at          │
           │
│    │    completed_bookings  │            └───────────┬────────────┘
           │
│    │    completion_rate     │                        │
            │
│    │    total_earnings      │                        │ 1:N
           │
│    │    is_available        │                        │
            │
│    │    availability_status│                         ▼
            │
```

```
|    |    is_featured          |           ┌────────────────────────┐
           |
|    |    created_at           |           |      SERVICES          |
           |
|    |    updated_at           |           ├────────────────────────┤
           |
|    └─────────────────────────┘    1:N    | PK service_id          |
           |
|                |─────────────────────────| FK provider_id          |
            |
|                |                         | FK category_id          |
            |
|                |                         |    service_name         |
            |
|                |                         |    service_description|
            |
|                |                         |    base_price           |
            |
|                |                         |    price_type           |
            |
|                |                         |    estimated_duration |
            |
|                |                         |    advance_booking_hrs|
            |
|                |                         |    is_active            |
            |
|                |                         |    is_featured          |
            |
|                |                         |    times_booked         |
            |
|                |                         |    average_rating       |
            |
|                |                         |    created_at           |
            |
|                |                         |    updated_at           |
            |
|                |                         └────────────────────────┘
            |
|                |                                     |
             |
|                | 1:N                                 | 1:N
            |
|                |                                     |
             |
|                ▼                                     ▼
             |
|    ┌────────────────────────────────────────────────────────────────────────
──┐        |
|    |                              BOOKINGS
   |        |
│    ├────────────────────────────────────────────────────────────────────────
──┤        │
│    │ PK booking_id
   │        │
│    │    booking_number (UK)
   │        │
│    │ FK user_id               ◄──────────────────────────────────────────────
─────── USERS│
│    │ FK provider_id           ◄──────────────────────────────────────────────
─── PROVIDERS│
│    │ FK service_id            ◄──────────────────────────────────────────────
─── SERVICES │
│    │    scheduled_date           │    urgency_level            │
          │
│    │    scheduled_time           │    estimated_amount         │
          │
│    │    scheduled_end_time       │    material_cost            │
          │
│    │    actual_start_time        │    labor_cost               │
          │
│    │    actual_end_time          │    final_amount             │
          │
│    │    service_address          │    booking_status           │
          │
│    │    service_pincode          │    cancelled_by             │
          │
│    │    service_latitude         │    cancellation_reason      │
          │
│    │    service_longitude        │    work_summary             │
          │
│    │    issue_title              │    user_rating              │
          │
│    │    issue_description        │    created_at               │
          │
│    │    issue_images (JSON)      │    updated_at               │
          │
│    └──────────────────────────────────────────────┬─────────────────────────
──┘        │
│                                                    │
          │
│                                                    │ 1:1
          │
│                                                    ▼
          │
│    ┌────────────────────────────────────────────────────────────────────────
──┐        │
│    │                             REVIEWS
   │        │
│    ├────────────────────────────────────────────────────────────────────────
──┤        │
```

```
│    │ PK review_id
   │        │
│    │ FK booking_id (UK)
   │        │
│    │ FK reviewer_id                ◄─────────────────────────────────────────
 USERS     │
│    │ FK reviewee_provider_id       ◄─────────────────────────────────────────
 PROVIDERS │
│    │    overall_rating            │    professionalism_rating │
            │
│    │    quality_rating            │    value_for_money_rating │
            │
│    │    punctuality_rating        │    review_title           │
            │
│    │    review_text               │    review_images (JSON)   │
            │
│    │    provider_response         │    moderation_status      │
            │
│    │    is_visible                │    helpful_count          │
            │
│    │    created_at                │    updated_at             │
            │
│    └────────────────────────────────────────────────────────────────────────
──┘        │
│
            │
│    ┌───────────────────────┐
           │
│    │   OTP_VERIFICATIONS   │
           │
│    ├───────────────────────┤
           │
│    │ PK otp_id             │
           │
│    │    identifier         │    (email/phone)
           │
│    │    identifier_type    │    (EMAIL/PHONE)
           │
│    │    otp_hash           │
           │
│    │    purpose            │    (REGISTRATION/LOGIN/PASSWORD_RESET)
           │
│    │    attempts           │
           │
│    │    expires_at         │
           │
│    │    is_verified        │
           │
│    │    created_at         │
           │
```

```
│    └────────────────────────┘
            │
│
             │
└────────────────────────────────────────────────────────────────────────────────
─────────────┘

LEGEND:
─────────────────
PK = Primary Key
FK = Foreign Key
UK = Unique Key
1:1 = One-to-One Relationship
1:N = One-to-Many Relationship
```

6.2 Relationship Summary

Parent Entity	Child Entity	Relationship	Cardinality
Users	UserAddresses	Has	1:N
Users	ServiceProviders	Is	1:1
Users	Bookings	Makes	1:N
Users	Reviews	Writes	1:N
ServiceProviders	Services	Offers	1:N
ServiceProviders	Bookings	Receives	1:N
ServiceProviders	Reviews	Receives	1:N
ServiceCategories	Services	Contains	1:N
ServiceCategories	ServiceCategories	HasSubcategory	1:N (Self-Ref)
Services	Bookings	BookedFor	1:N
Bookings	Reviews	HasReview	1:1

---

7. Database Design

7.1 Schema Overview

```
-- Database: hirelink_db
-- Character Set: utf8mb4
-- Collation: utf8mb4_unicode_ci

-- Total Tables: 8 Core Tables
-- 1. users
-- 2. user_addresses
-- 3. service_providers
-- 4. service_categories
-- 5. services
-- 6. bookings
-- 7. reviews
-- 8. otp_verifications
```

7.2 Data Dictionary

Table: users

Column	Data Type	Constraints	Description
user_id	BIGINT	PK, AUTO_INCREMENT	Unique user identifier
name	VARCHAR(100)	NOT NULL	User's full name
email	VARCHAR(150)	UNIQUE	Email address
phone	VARCHAR(15)	UNIQUE	Phone number
password_hash	VARCHAR(255)		BCrypt hashed password
auth_provider	ENUM	DEFAULT 'LOCAL'	LOCAL, GOOGLE
profile_image_url	VARCHAR(500)		Profile picture URL
date_of_birth	DATE		User's birth date
gender	ENUM		MALE, FEMALE, OTHER
user_type	ENUM	NOT NULL, DEFAULT 'CUSTOMER'	CUSTOMER, PROVIDER, ADMIN, SUPER_ADMIN
account_status	ENUM	DEFAULT 'ACTIVE'	ACTIVE, INACTIVE, SUSPENDED, BANNED
is_email_verified	BOOLEAN	DEFAULT FALSE	Email verification status
is_phone_verified	BOOLEAN	DEFAULT FALSE	Phone verification status
preferred_language	ENUM	DEFAULT 'EN'	Language preference code
failed_login_attempts	INT	DEFAULT 0	Failed login counter
locked_until	TIMESTAMP		Account lock expiry
last_login_at	TIMESTAMP		Last successful login
created_at	TIMESTAMP	NOT NULL	Record creation time
updated_at	TIMESTAMP		Last update time

Table: service_providers

Column	Data Type	Constraints	Description
provider_id	BIGINT	PK, AUTO_INCREMENT	Unique provider identifier

Column	Data Type	Constraints	Description
user_id	BIGINT	FK, UNIQUE, NOT NULL	Reference to users table
business_name	VARCHAR(200)		Business/brand name
business_description	TEXT		Detailed description
tagline	VARCHAR(255)		Short marketing tagline
experience_years	INT	DEFAULT 0	Years of experience
specializations	JSON		Array of specialties
base_latitude	DECIMAL(10,8)		Business location lat
base_longitude	DECIMAL(11,8)		Business location long
base_pincode	VARCHAR(6)		Primary service pincode
service_radius_km	INT	DEFAULT 10	Service coverage radius
aadhaar_encrypted	VARCHAR(500)		Encrypted Aadhaar number
pan_encrypted	VARCHAR(500)		Encrypted PAN number
kyc_status	ENUM	DEFAULT 'NOT_SUBMITTED'	KYC verification status
average_rating	DECIMAL(3,2)	DEFAULT 0.00	Average service rating
total_bookings	INT	DEFAULT 0	Total booking count
completed_bookings	INT	DEFAULT 0	Completed booking count
cancelled_bookings	INT	DEFAULT 0	Cancelled booking count
completion_rate	DECIMAL(5,2)	DEFAULT 0.00	Booking completion percentage
total_earnings	DECIMAL(12,2)	DEFAULT 0.00	Cumulative earnings
is_available	BOOLEAN	DEFAULT TRUE	Current availability
availability_status	ENUM	DEFAULT 'OFFLINE'	ONLINE, OFFLINE, BUSY, ON_BREAK
is_featured	BOOLEAN	DEFAULT FALSE	Featured provider flag

Column	Data Type	Constraints	Description
created_at	TIMESTAMP	NOT NULL	Record creation time
updated_at	TIMESTAMP		Last update time

Table: bookings

Column	Data Type	Constraints	Description
booking_id	BIGINT	PK, AUTO_INCREMENT	Unique booking identifier
booking_number	VARCHAR(20)	UNIQUE, NOT NULL	Human-readable booking ID
user_id	BIGINT	FK, NOT NULL	Customer reference
provider_id	BIGINT	FK, NOT NULL	Provider reference
service_id	BIGINT	FK, NOT NULL	Service reference
scheduled_date	DATE	NOT NULL	Scheduled service date
scheduled_time	TIME	NOT NULL	Scheduled start time
scheduled_end_time	TIME		Expected end time
actual_start_time	TIMESTAMP		Actual work start time
actual_end_time	TIMESTAMP		Actual work completion time
service_address	TEXT	NOT NULL	Service location address
service_pincode	VARCHAR(6)	NOT NULL	Service location pincode
service_latitude	DECIMAL(10,8)		Service location latitude
service_longitude	DECIMAL(11,8)		Service location longitude
issue_title	VARCHAR(255)		Brief issue summary
issue_description	TEXT		Detailed issue description
issue_images	JSON		Array of issue image URLs
urgency_level	ENUM	DEFAULT 'MEDIUM'	LOW, MEDIUM, HIGH, EMERGENCY
estimated_a	DECIMAL(10,2)		Initial cost estimate

Column	Data Type	Constraints	Description
mount			
material_cost	DECIMAL(10,2)	DEFAULT 0.00	Materials cost
labor_cost	DECIMAL(10,2)	DEFAULT 0.00	Labor charges
travel_charge	DECIMAL(10,2)	DEFAULT 0.00	Travel expenses
discount_amount	DECIMAL(10,2)	DEFAULT 0.00	Applied discounts
final_amount	DECIMAL(10,2)		Final billing amount
booking_status	ENUM	DEFAULT 'PENDING'	Current booking status
cancelled_by	ENUM		USER, PROVIDER, ADMIN, SYSTEM
cancellation_reason	TEXT		Reason for cancellation
work_summary	TEXT		Provider's work completion notes
user_rating	DECIMAL(3,2)		Customer's rating
created_at	TIMESTAMP	NOT NULL	Booking creation time
updated_at	TIMESTAMP		Last status update time

7.3 Booking Status State Machine

```
┌─────────────────────────────────────────────────────────────────
─┐
│                      BOOKING STATUS TRANSITIONS
  │
├─────────────────────────────────────────────────────────────────
─┤
│
  │
│
  │
│                            ┌──────────┐
  │
│                            │ PENDING  │
  │
│                            └────┬─────┘
  │
│                   ┌─────────────┼─────────────┐
  │
│                   │             │             │
  │
│                   ▼             ▼             ▼
  │
│             ┌─────────┐   ┌─────────┐   ┌──────────┐
│              │ ACCEPTED │  │ REJECTED │  │ CANCELLED │
│              └────┬─────┘  └──────────┘  └───────────┘
│                   │                             ▲
│                   ▼                             │
│              ┌──────────┐                       │
│              │CONFIRMED │───────────────────────┤
│              └────┬─────┘                       │
│                   │                             │
│                   ▼                             │
│           ┌─────────────┐                       │
│           │ IN_PROGRESS │───────────────────────┤
│           └──────┬──────┘                       │
│          ┌───────┼───────┐                      │
│          │       │       │                      │
│          ▼       │       ▼                      │
│    ┌──────────┐  │  ┌──────────┐               │
│    │  PAUSED  │──┘  │COMPLETED │               │
│    └──────────┘     └────┬─────┘               │
│                          │                      │
│                          ▼                      │
│                    ┌──────────┐                 │
│                    │ DISPUTED │─────────────────┤
│                    └────┬─────┘                 │
│                         │                       │
│                         ▼                       │
│                    ┌──────────┐                  │
 │
│                    │ REFUNDED │                  │
 │
│                    └──────────┘                  │
 │
│
 │
└──────────────────────────────────────────────────────────────────────────
─┘
```

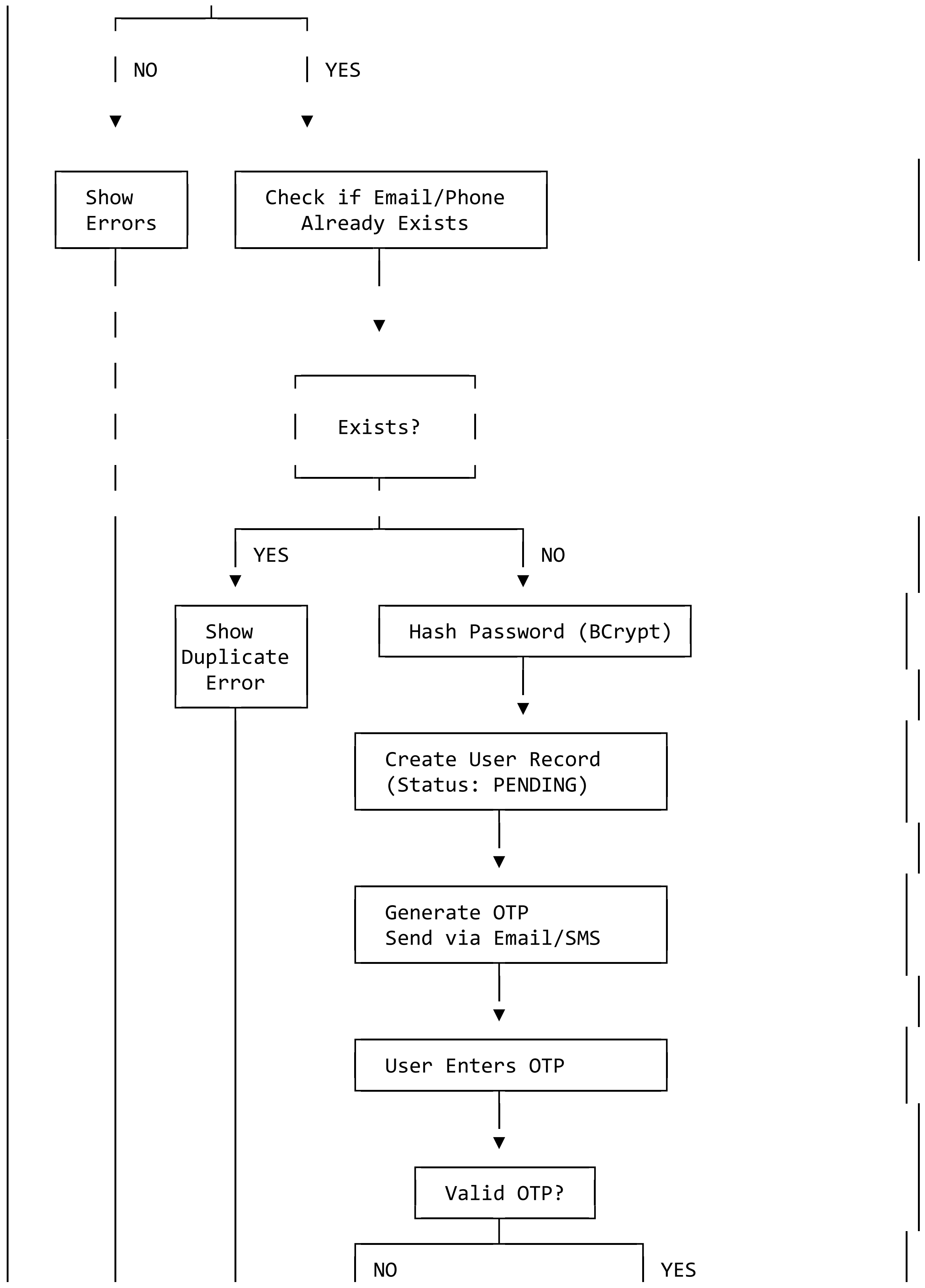
```
STATUS DEFINITIONS:
─────────────────────
PENDING      : Initial state, awaiting provider response
ACCEPTED     : Provider accepted the booking
REJECTED     : Provider declined the booking
CONFIRMED    : Booking confirmed by both parties
IN_PROGRESS  : Service is currently being performed
PAUSED       : Work temporarily paused
COMPLETED    : Service successfully completed
CANCELLED    : Booking cancelled by any party
DISPUTED     : Customer raised a dispute
REFUNDED     : Payment refunded after dispute
```

---

8. Workflow Diagrams

8.1 User Registration Workflow

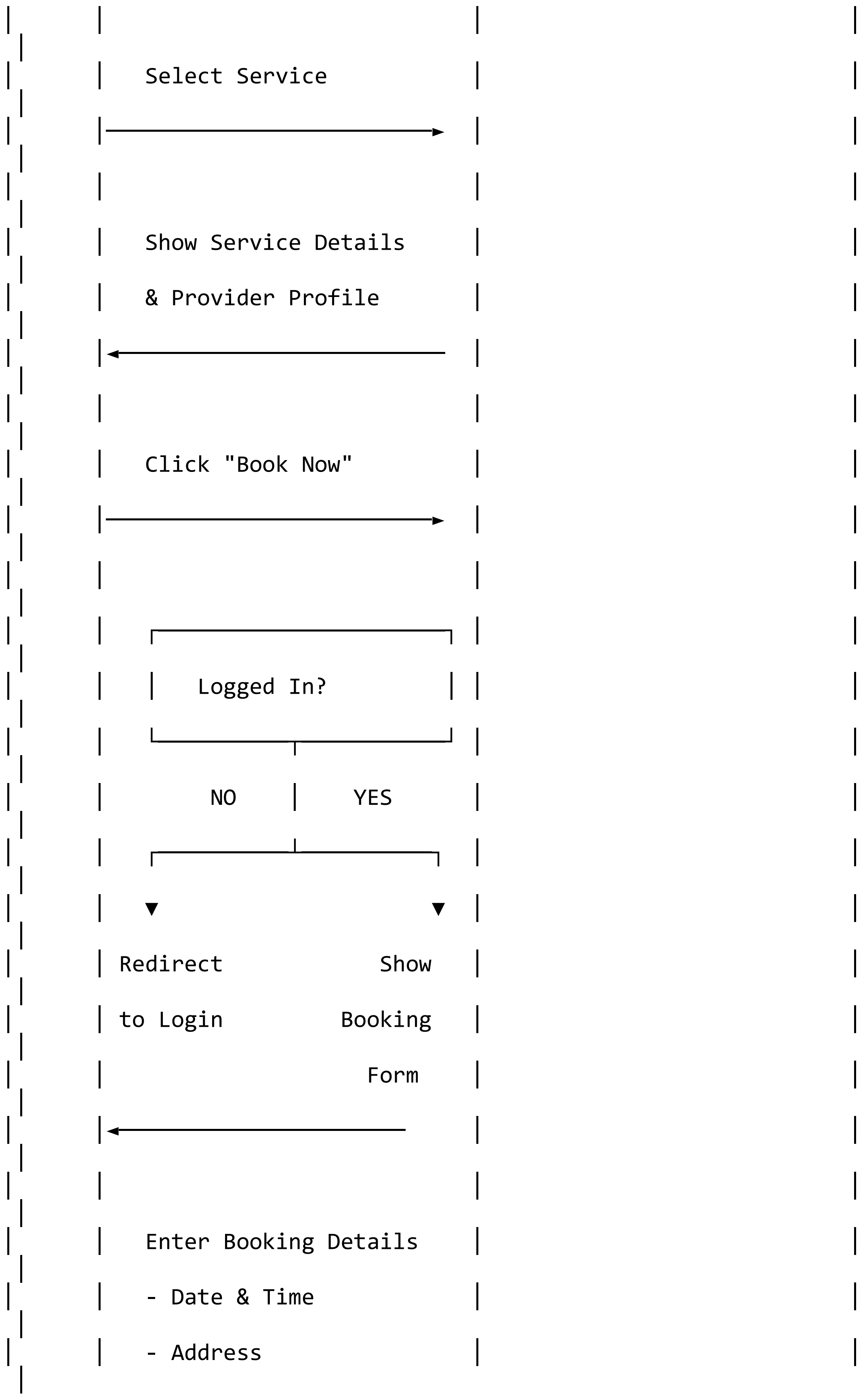
```
┌──────────────────────────────────────────────────────────────────────────
─┐
│                          USER REGISTRATION WORKFLOW
  │
├──────────────────────────────────────────────────────────────────────────
─┤
│
 │
│    ┌─────────┐
 │
│    │  START  │
 │
│    └────┬────┘
 │
│         │
 │
│         ▼
 │
│    ┌─────────────────────────┐
│    │  User visits Register  │
│    │        Page            │
│    └───────────┬────────────┘
│                │
 │               ▼
│
 │
│    ┌────────────────────────┐
│    │  Select Account Type   │
│    │  (Customer/Provider)   │
│    └───────────┬────────────┘
│                │
 │               ▼
│
 │
│    ┌────────────────────────┐
│    │  Enter Registration    │
│    │  Details               │
│    │  - Name                │
│    │  - Email/Phone         │
│    │  - Password            │
│    └───────────┬────────────┘
│                │
 │               ▼
│
 │
│         ┌──────────────┐
│         │  Validate    │
│         │   Input?     │
│         └──────┬───────┘
│                │
 │
```


NO
YES
Show Errors
Check if Email/Phone Already Exists
Exists?
YES
NO
Show Duplicate Error
Hash Password (BCrypt)
Create User Record (Status: PENDING)
Generate OTP Send via Email/SMS
User Enters OTP
Valid OTP?
NO
YES

```
│         │         │        ▼                    ▼                       │
│         │         │  ┌──────────────┐    ┌─────────────────┐            │
│         │         │  │ Retry/Resend │    │ Verify Account  │            │
│         │         │  │     OTP      │    │(Status: ACTIVE) │            │
│         └─────────┴──┴──────────────┘    └─────────────────┘            │
│                                                   │                       │
│                                                   ▼                       │
│                                          ┌─────────────────┐            │
│                                          │ Generate JWT    │            │
│                                          │ Login User      │            │
│                                          └─────────────────┘            │
│                                                   │                       │
│                                                   ▼                       │
│                                          ┌─────────────────┐            │
│                                          │   Redirect to   │            │
│                                          │     Dashboard   │            │
│                                          └─────────────────┘            │
│                                                   │                       │
│                                                   ▼                       │
│                                              ┌─────────┐                 │
│                                              │   END   │                 │
│                                              └─────────┘                 │
└─────────────────────────────────────────────────────────────────────────
┘
```

8.2 Service Booking Workflow

```
┌─────────────────────────────────────────────────────────────────────────
┐
│                      SERVICE BOOKING WORKFLOW
│
├─────────────────────────────────────────────────────────────────────────
┤
│
│   CUSTOMER                    SYSTEM                      PROVIDER
│
│       │                          │                            │
│       │   Browse Categories      │                            │
│       │─────────────────────────►│                            │
│       │                          │                            │
│       │   Display Categories     │                            │
│       │◄─────────────────────────│                            │
│
```

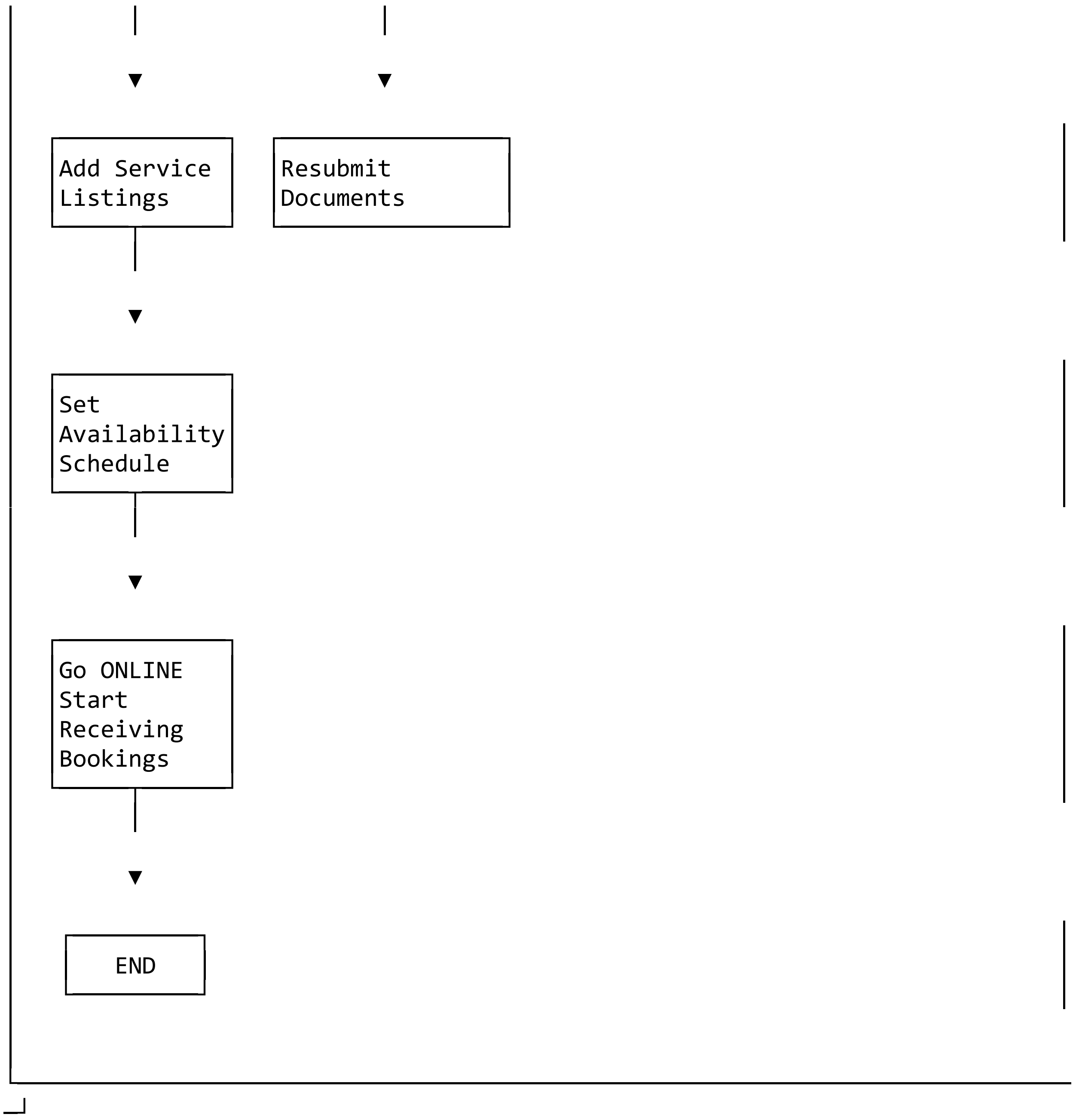

Select Service
Show Service Details
& Provider Profile
Click "Book Now"
Logged In?
NO
YES
Redirect
to Login
Show
Booking
Form
Enter Booking Details
- Date & Time
- Address

```
|      |   - Issue Description        |                             |
 |
|      |──────────────────────────►   |                             |
 |
|      |                              |                             |
 |
|      |                    Validate & Create                       |
 |
|      |                    Booking (PENDING)                       |
 |
|      |                              |                             |
 |
|      |                    Generate Booking#                       |
 |
|      |                    (HL2026XXXXXXX)                         |
 |
|      |                              |                             |
 |
|      |   Booking Confirmation       |   Notification Email/Push   |
 |
|      |◄──────────────────────────   |──────────────────────────►  |
 |
|      |                              |                             |
 |
|      |                              |         View Booking        |
 |
|      |                              |◄──────────────────────────  |
 |
|      |                              |                             |
 |
|      |                              |    Accept/Reject Decision   |
 |
|      |                              |◄──────────────────────────  |
 |
|      |                              |                             |
 |
|      |                   ┌───────────────────┐                    |
 |
|      |                   |    Decision?      |                    |
 |
|      |                   └─────────┬─────────┘                    |
 |
|      |            ACCEPT           |         REJECT               |
 |
|      |         ┌───────────────────┴───────────────────┐          |
 |
|      |         ▼                                       ▼          |
 |
|      |   Update Status:                      Update Status:       |
 |
│      │     ACCEPTED                         REJECTED        │
 │
│      │         │                                │           │
 │
│      │   Notify Customer                  Notify Customer   │
 │
│      │◄────────┘                        │◄──────────────    │
 │
│      │                                  │                   │
 │
│      │                                  │   Return to       │
 │
│      │                                  │   Search          │
 │
│      │                                  │                   │
 │
│      │   On Scheduled Day               │                   │
 │
│      │──────────────────────────►       │                   │
 │
│      │                       Provider arrives               │
 │
│      │                                  │──────────────────►│
 │
│      │                                  │                   │
 │
│      │                                  │  Start Work       │
 │
│      │                       Update: IN_PROGRESS            │
 │
│      │◄─────────────────────────────────┼───────────────────│
 │
│      │                                  │                   │
 │
│      │                                  │  Complete Work    │
 │
│      │                       Update: COMPLETED              │
 │
│      │◄─────────────────────────────────┼───────────────────│
 │
│      │                                  │                   │
 │
│      │   Leave Review & Rating          │                   │
 │
│      │──────────────────────────►       │                   │
 │
│      │                                  │                   │
 │
│      │                 Update Provider Stats                │
 │
```

```
│      │                   (Rating, Reviews)                  │
 │
│      │                          │                           │
 │
│      │   Confirmation           │                           │
 │
│      │◄─────────────────────────│                           │
 │
│      │                          │                           │
 │
│     END                        END                         END
 │
│
 │
└──────────────────────────────────────────────────────────────────────
─┘
```

8.3 Provider Onboarding Workflow

```
┌──────────────────────────────────────────────────────────────────────
─┐
│                      PROVIDER ONBOARDING WORKFLOW
  │
├──────────────────────────────────────────────────────────────────────
─┤
│
 │
│
 │
│    ┌────────┐
 │
│    │ START  │
 │
│    └───┬────┘
 │
│        │
 │
│        ▼
 │
│    ┌─────────────────────┐
│    │ User Registers as   │
│    │ Service Provider    │
│    └─────────┬───────────┘
│              │
 │
│              ▼
 │
│    ┌─────────────────────┐
│    │  Account Verified        │
│    │  (OTP Verification)      │
│    └────────────┬─────────────┘
│                 │
 │
│                 ▼
 │
│    ┌──────────────────────────┐
│    │  Complete Business       │
│    │  Profile                 │
│    │  - Business Name         │
│    │  - Description           │
│    │  - Experience            │
│    │  - Specializations       │
│    └────────────┬─────────────┘
│                 │
 │
│                 ▼
 │
│    ┌──────────────────────────┐
│    │  Set Service Location    │
│    │  - Base Address          │
│    │  - Pincode               │
│    │  - Service Radius        │
│    └────────────┬─────────────┘
│                 │
 │
│                 ▼
 │
│    ┌──────────────────────────┐
│    │  Upload KYC Documents    │
│    │   - Aadhaar Card         │
│    │   - PAN Card             │
│    │   - Address Proof        │
│    └────────────┬─────────────┘
│                 │
 │                ▼
│
 │
│    ┌──────────────────────────┐
│    │  KYC Status: PENDING     │
│    └────────────┬─────────────┘
│                 │
 │                ▼
│
 │
│    ┌──────────────────────────┐
│    │      ADMIN REVIEW        │
│    │   - Verify Documents     │
│    │   - Background Check     │
│    └────────────┬─────────────┘
│                 │
 │          ┌─────┴───────┐
│           │             │
│     APPROVED        REJECTED
│           │             │
│           ▼             ▼
│   ┌────────────┐  ┌────────────────┐
│   │KYC Status: │  │KYC Status:     │
│   │ VERIFIED   │  │ REJECTED       │
│   └─────┬──────┘  │(with reason)   │                                      │
│         │         └──────┬─────────┘                                      │
```



9. Technical Specifications

9.1 Frontend Technical Specifications

9.1.1 Project Structure

```
frontend/
├── src/
│   ├── components/          # Reusable UI components
│   │   └── layout/
│   │       ├── MainLayout.jsx
│   │       ├── AuthLayout.jsx
│   │       ├── Navbar.jsx
│   │       └── Footer.jsx
│   ├── pages/               # Page components (routes)
│   │   ├── auth/
│   │   │   ├── Login.jsx
│   │   │   └── Register.jsx
│   │   ├── Home.jsx
│   │   ├── Categories.jsx
│   │   ├── CategoryServices.jsx
│   │   ├── ServiceDetail.jsx
│   │   ├── ProviderProfile.jsx
│   │   ├── Bookings.jsx
│   │   ├── BookingDetail.jsx
│   │   ├── BookService.jsx
│   │   ├── Profile.jsx
│   │   ├── SearchResults.jsx
│   │   └── NotFound.jsx
│   ├── services/            # API service layer
│   │   └── api.js
│   ├── context/             # State management
│   │   └── authStore.js
│   ├── styles/              # Styling configuration
│   │   └── theme.js
│   ├── App.jsx              # Main application component
│   ├── main.jsx             # Application entry point
│   └── index.css            # Global styles (Tailwind)
├── public/                  # Static assets
├── index.html
├── package.json
├── vite.config.js
├── tailwind.config.js
└── postcss.config.js
```

9.1.2 Key Dependencies

Package	Version	Purpose
react	^18.2.0	UI Component Library
react-dom	^18.2.0	React DOM Rendering
react-router-dom	^6.x	Client-side Routing
@tanstack/react-query	^5.x	Data Fetching & Caching
zustand	^4.x	State Management
axios	^1.x	HTTP Client
date-fns	^2.x	Date Manipulation
@heroicons/react	^2.x	Icon Library
tailwindcss	^3.x	CSS Framework

9.1.3 State Management (Zustand Store)

```
// authStore.js - Authentication State Structure
{
  user: {
    userId: Number,
    name: String,
    email: String,
    phone: String,
    userType: 'CUSTOMER' | 'PROVIDER' | 'ADMIN' | 'SUPER_ADMIN',
    profileImageUrl: String,
    isEmailVerified: Boolean,
    isPhoneVerified: Boolean
  },
  token: String,           // JWT Access Token
  isAuthenticated: Boolean,
  isLoading: Boolean,

  // Actions
  login: (credentials) => Promise,
  register: (userData) => Promise,
  logout: () => void,
  updateProfile: (data) => Promise,
  refreshToken: () => Promise
}
```

9.2 Backend Technical Specifications

9.2.1 Project Structure

```
backend/
├── src/main/java/com/hirelink/
│   ├── HireLinkApplication.java     # Main application entry
│   ├── config/
│   │   └── SecurityConfig.java       # Security configuration
│   ├── controller/                   # REST API controllers
│   │   ├── AuthController.java
│   │   ├── UserController.java
│   │   ├── BookingController.java
│   │   ├── CategoryController.java
│   │   ├── ProviderController.java
│   │   └── ServiceController.java
│   ├── dto/                          # Data Transfer Objects
│   │   ├── ApiResponse.java
│   │   ├── AuthDTO.java
│   │   ├── BookingDTO.java
│   │   ├── CategoryDTO.java
│   │   ├── ProviderDTO.java
│   │   └── ServiceDTO.java
│   ├── entity/                       # JPA Entities
│   │   ├── User.java
│   │   ├── UserAddress.java
│       │   ├── ServiceProvider.java
│       │   ├── ServiceCategory.java
│       │   ├── Service.java
│       │   ├── Booking.java
│       │   ├── Review.java
│       │   └── OtpVerification.java
│       ├── exception/                  # Exception handling
│       │   ├── GlobalExceptionHandler.java
│       │   ├── ResourceNotFoundException.java
│       │   ├── BadRequestException.java
│       │   └── UnauthorizedException.java
│       ├── repository/                 # Data repositories
│       │   ├── UserRepository.java
│       │   ├── BookingRepository.java
│       │   ├── ServiceRepository.java
│       │   └── ...
│       ├── security/                   # Security components
│       │   ├── JwtService.java
│       │   ├── JwtAuthenticationFilter.java
│       │   ├── CustomUserDetails.java
│       │   └── UserDetailsServiceImpl.java
│       └── service/                    # Business logic
│           ├── AuthService.java
│           ├── BookingService.java
│           ├── ProviderService.java
│           ├── CategoryService.java
│           ├── EmailService.java
│           ├── SmsService.java
│           └── LocationService.java
├── src/main/resources/
│   ├── application.properties          # Main configuration
│   └── application-local.properties    # Local environment config
└── pom.xml                             # Maven dependencies
```

9.2.2 Key Dependencies (pom.xml)

Dependency	Version	Purpose
spring-boot-starter-web	3.2.x	REST API Framework
spring-boot-starter-data-jpa	3.2.x	Data Access Layer
spring-boot-starter-security	3.2.x	Authentication & Authorization
spring-boot-starter-validation	3.2.x	Input Validation
mysql-connector-j	8.x	MySQL Database Driver
jjwt-api	0.12.x	JWT Token Generation
lombok	1.18.x	Boilerplate Reduction
springdoc-openapi	2.x	API Documentation (Swagger)

9.2.3 Application Configuration

```
# application.properties
server.port=8080

# Database Configuration
spring.datasource.url=jdbc:mysql://localhost:3306/hirelink_db
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.jpa.hibernate.ddl-auto=validate
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true

# JWT Configuration
jwt.secret=<base64-encoded-secret-key>
jwt.access-token-expiry=86400000    # 24 hours
jwt.refresh-token-expiry=604800000  # 7 days

# File Upload
spring.servlet.multipart.max-file-size=10MB
spring.servlet.multipart.max-request-size=10MB

# CORS Configuration
cors.allowed-origins=http://localhost:3000
```

---

10. API Documentation

10.1 API Overview

Base URL	Version	Format
`/api`	v1	JSON

10.2 Authentication Endpoints

POST /api/auth/register

Register a new user account.

Request Body:

```
{
  "name": "John Doe",
  "email": "john@example.com",
  "phone": "9876543210",
  "password": "securePassword123",
  "userType": "CUSTOMER"
}
```

Response (201 Created):

```json
{
  "success": true,
  "message": "Registration successful. Please verify your account.",
  "data": {
    "userId": 1,
    "name": "John Doe",
    "email": "john@example.com",
    "userType": "CUSTOMER",
    "accountStatus": "PENDING_VERIFICATION"
  }
}
```

POST /api/auth/login

Authenticate user and receive JWT token.

Request Body:

```json
{
  "identifier": "john@example.com",
  "password": "securePassword123"
}
```

Response (200 OK):

```json
{
  "success": true,
  "message": "Login successful",
  "data": {
    "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "tokenType": "Bearer",
    "expiresIn": 86400,
    "user": {
      "userId": 1,
      "name": "John Doe",
      "email": "john@example.com",
      "userType": "CUSTOMER"
    }
  }
}
```

10.3 Category Endpoints

GET /api/categories

Retrieve all service categories.

Response (200 OK):

```json
{
  "success": true,
  "data": [
    {
      "categoryId": 1,
      "categoryName": "Electrical",
      "categorySlug": "electrical",
      "categoryDescription": "All electrical repair and installation services
",
      "categoryIcon": "BoltIcon",
      "minBasePrice": 200,
      "maxBasePrice": 5000,
      "priceUnit": "PER_VISIT",
      "isActive": true,
      "isFeatured": true,
      "serviceCount": 15
    }
  ]
}
```

10.4 Service Endpoints

GET /api/services/{id}

Get service details by ID.

Response (200 OK):

```
{
  "success": true,
  "data": {
    "serviceId": 1,
    "serviceName": "Complete House Wiring",
    "serviceDescription": "Full house wiring with quality copper wires",
    "basePrice": 5000.00,
    "priceType": "STARTING_FROM",
    "estimatedDurationMinutes": 480,
    "advanceBookingHours": 24,
    "isActive": true,
    "isFeatured": true,
    "timesBooked": 45,
    "averageRating": 4.80,
    "provider": {
      "providerId": 1,
      "businessName": "Ramesh Electrical Works",
      "averageRating": 4.75
    },
    "category": {
      "categoryId": 1,
      "categoryName": "Electrical"
    }
  }
}
```

10.5 Booking Endpoints

POST /api/bookings

Create a new service booking.

Headers:

```
Authorization: Bearer <access_token>
```

Request Body:

```json
{
  "serviceId": 1,
  "providerId": 1,
  "scheduledDate": "2026-02-15",
  "scheduledTime": "10:00:00",
  "serviceAddress": "42, Shanti Nagar, Near City Mall",
  "servicePincode": "560001",
  "serviceLatitude": 12.9716,
  "serviceLongitude": 77.5946,
  "issueTitle": "Ceiling Fan Not Working",
  "issueDescription": "The fan in living room stopped working",
  "urgencyLevel": "MEDIUM"
}
```

Response (201 Created):

```json
{
  "success": true,
  "message": "Booking created successfully",
  "data": {
    "bookingId": 1,
    "bookingNumber": "HL2026021500001",
    "bookingStatus": "PENDING",
    "estimatedAmount": 300.00,
    "scheduledDate": "2026-02-15",
    "scheduledTime": "10:00:00",
    "createdAt": "2026-01-27T10:30:00"
  }
}
```

PUT /api/bookings/{id}/status

Update booking status.

Request Body:

```json
{
  "status": "ACCEPTED",
  "notes": "Will arrive at scheduled time"
}
```

Response (200 OK):

```
{
  "success": true,
  "message": "Booking status updated",
  "data": {
    "bookingId": 1,
    "bookingNumber": "HL2026021500001",
    "bookingStatus": "ACCEPTED",
    "providerResponseAt": "2026-01-27T11:00:00"
  }
}
```

10.6 Provider Endpoints

GET /api/providers/{id}

Get provider profile details.

Response (200 OK):

```
{
  "success": true,
  "data": {
    "providerId": 1,
    "businessName": "Ramesh Electrical Works",
    "businessDescription": "Expert electrical services with 15 years experien
ce",
    "tagline": "Quality electrical work guaranteed",
    "experienceYears": 15,
    "specializations": ["House Wiring", "Fan Installation", "Electrical Repai
rs"],
    "basePincode": "560058",
    "serviceRadiusKm": 15,
    "kycStatus": "VERIFIED",
    "averageRating": 4.75,
    "totalBookings": 180,
    "completedBookings": 165,
    "completionRate": 91.67,
    "isAvailable": true,
    "availabilityStatus": "ONLINE",
    "services": [
      {
        "serviceId": 1,
        "serviceName": "Complete House Wiring",
        "basePrice": 5000.00
      }
    ],
    "reviews": [
      {
        "reviewId": 1,
        "overallRating": 5.0,
        "reviewTitle": "Excellent Service!",
        "reviewerName": "Priya S."
      }
    ]
  }
}
```

10.7 API Error Responses

HTTP Code	Error Type	Description
400	Bad Request	Invalid input or validation failure
401	Unauthorized	Missing or invalid authentication
403	Forbidden	Insufficient permissions
404	Not Found	Resource does not exist
409	Conflict	Resource conflict (e.g., duplicate)
500	Internal Server Error	Server-side error

Error Response Format:

```
{
  "success": false,
  "message": "Detailed error message",
  "errors": [
    {
      "field": "email",
      "message": "Email address is already registered"
    }
  ],
  "timestamp": "2026-01-27T10:30:00",
  "path": "/api/auth/register"
}
```

---

11. User Interface Design

11.1 Design Principles

1. **Responsive Design**: Mobile-first approach with breakpoints for tablet and desktop
2. **Consistent Visual Language**: Unified color palette, typography, and spacing
3. **Accessibility**: WCAG 2.1 compliant with proper contrast ratios
4. **Intuitive Navigation**: Maximum 3 clicks to reach any feature
5. **Feedback & Loading States**: Clear visual feedback for all user actions

11.2 Color Palette

Color Name	Hex Code	Usage

Color Name	Hex Code	Usage
Primary-600	#4F46E5	Primary actions, links
Primary-700	#4338CA	Primary hover states
Accent-500	#F59E0B	CTA buttons, highlights
Success-500	#22C55E	Success states
Warning-500	#EAB308	Warning messages
Error-500	#EF4444	Error states
Gray-900	#111827	Primary text
Gray-500	#6B7280	Secondary text
Gray-50	#F9FAFB	Background

11.3 Page Layout Structure

```
┌─────────────────────────────────────────────────────────────────────┐
│                          NAVIGATION BAR                              │
│  ┌──────┐  ┌──────────────────────────┐  ┌────────┐  ┌──────────┐  │
│  │ Logo │  │        Search Bar        │  │Location│  │User Menu │  │
│  └──────┘  └──────────────────────────┘  └────────┘  └──────────┘  │
├─────────────────────────────────────────────────────────────────────┤
│                                                                     │
│                         MAIN CONTENT                                 │
│                                                                     │
│  ┌─────────────────────────────────────────────────────────────┐  │
│  │                                                               │ │
│  │                    PAGE-SPECIFIC CONTENT                      │ │
│  │                                                               │ │
│  └─────────────────────────────────────────────────────────────┘  │
│                                                                     │
├─────────────────────────────────────────────────────────────────────┤
│                            FOOTER                                    │
│  ┌───────────┐  ┌───────────┐  ┌───────────┐  ┌───────────┐       │
│  │ About Us  │  │ Services  │  │  Support  │  │   Legal   │       │
│  └───────────┘  └───────────┘  └───────────┘  └───────────┘       │
│                        © 2026 HireLink                               │
└─────────────────────────────────────────────────────────────────────┘
```

11.4 Key Page Wireframes

Home Page

```
┌───────────────────────────────────────────────────────────────┐
│                        [NAVIGATION BAR]                         │
├───────────────────────────────────────────────────────────────┤
│ ╔═══════════════════════════════════════════════════════════╗ │
│ ║                 HERO SECTION WITH GRADIENT                    ║ │
│ ║                                                               ║ │
│ ║   Expert Home Services at Your Doorstep                       ║ │
│ ║                                                                   ║ │
│ ║    ┌─────────────────────────────────────────────────┐           ║ │
│ ║    │ 🔍 What service do you need?        [Search]    │           ║ │
│ ║    └─────────────────────────────────────────────────┘           ║ │
│ ║                                                                   ║ │
│ ║     500+ Providers     10K+ Bookings     4.8★ Rating              ║ │
│ ╚═══════════════════════════════════════════════════════════════════╝ │
│                                                                       │
│  POPULAR SERVICES                                                     │
│  ┌──────────┐  ┌──────────┐  ┌──────────┐  ┌──────────┐              │
│  │ ⚡        │  │ 🔧        │  │ ✨        │  │ 🏠        │              │
│  │Electrical│  │ Plumbing │  │ Cleaning │  │Carpentry │              │
│  └──────────┘  └──────────┘  └──────────┘  └──────────┘              │
│                                                                       │
│  WHY CHOOSE HIRELINK?                                                 │
│  ┌─────────────────────────────────────────────────────────┐         │
│  │ ✓ Verified Professionals   ✓ On-Time Service            │         │
│  │ ✓ Transparent Pricing      ✓ Quality Assurance          │         │
│  └─────────────────────────────────────────────────────────┘         │
│                                                                       │
│  TOP RATED PROVIDERS                                                  │
│  ┌────────────┐  ┌────────────┐  ┌────────────┐  ┌────────────┐      │
│  │ Provider 1 │  │ Provider 2 │  │ Provider 3 │  │ Provider 4 │      │
│  │ ★★★★★      │  │ ★★★★☆      │  │ ★★★★★      │  │ ★★★★☆      │      │
│  └────────────┘  └────────────┘  └────────────┘  └────────────┘      │
│                                                                       │
│                       [FOOTER]                                        │
└───────────────────────────────────────────────────────────────────────┘
```

Booking Form Page

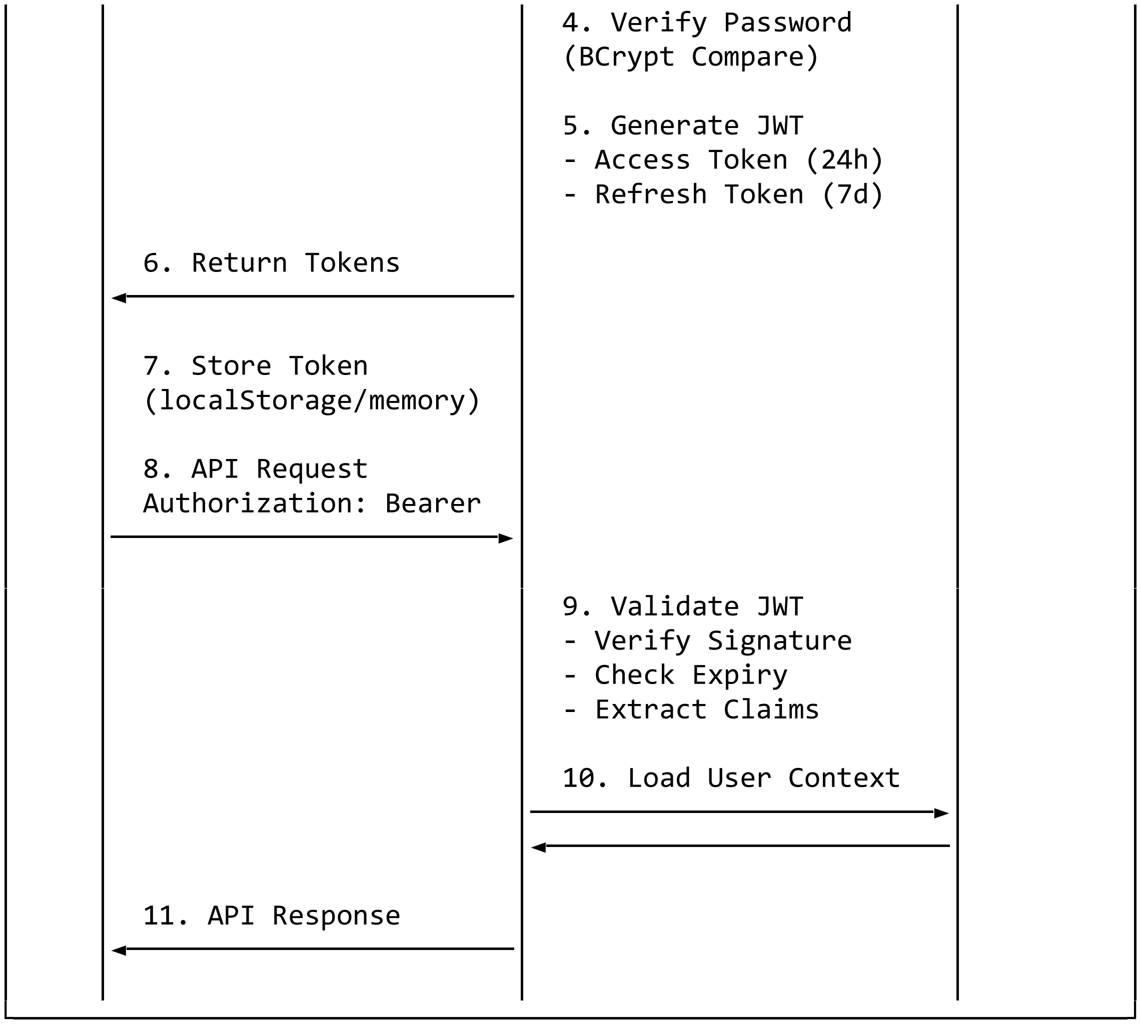
```
┌───────────────────────────────────────────────────────────────────────┐
│                       [NAVIGATION BAR]                                │
├───────────────────────────────────────────────────────────────────────┤
│                                                                       │
│  ← Back to Service Details                                            │
│                                                                       │
│  Book Service: [Service Name]                                         │
│  ─────────────────────────────────────────────────────────────        │
│                                                                       │
│  ┌─────────────────────────┐  ┌──────────────────────────────┐       │
│  │   SERVICE SUMMARY       │  │      BOOKING FORM            │       │
│  │                         │  │                              │       │
│  │ Provider: XYZ Works     │  │ 📅 Select Date:              │       │
│  │ Rating: ★★★★★ (4.8)     │  │   ┌────────────────────┐     │       │
│  │ Price: ₹300             │  │   │ [Date Picker]      │     │       │
│  │                         │  │   └────────────────────┘     │       │
│  │                         │  │                              │       │
│  │  ┌────────────────┐     │  │ ⏰ Select Time:              │       │
│  │  │ Provider Image │     │  │                              │       │
│  │  └────────────────┘     │  │ ┌────────────────────┐       │       │
│  │                         │  │                              │       │
│  │                         │  │ │  [Time Picker]     │       │ │
│  └─────────────────────────┘  │ └────────────────────┘       │ │
│                               │                               │ │
│                               │ 📍 Service Address:           │ │
│                               │ ┌────────────────────┐       │ │
│                               │ │  [Address Input]   │       │ │
│                               │ └────────────────────┘       │ │
│                               │                               │ │
│                               │ 📝 Describe Issue:            │ │
│                               │ ┌────────────────────┐       │ │
│                               │ │  [Text Area]       │       │ │
│                               │ └────────────────────┘       │ │
│                               │                               │ │
│                               │ ⚠ Urgency Level:              │ │
│                               │ ○ Low   ● Medium   ○ High     │ │
│                               │                               │ │
│                               │ ┌────────────────────┐       │ │
│                               │ │  CONFIRM BOOKING   │       │ │
│                               │ └────────────────────┘       │ │
│                               └───────────────────────────────┘ │
│                                                                   │
│                     [FOOTER]                                      │
└───────────────────────────────────────────────────────────────────┘
```

---

12. Security Implementation

12.1 Authentication Architecture

```
┌─────────────────────────────────────────────────────────────────────┐
│                        AUTHENTICATION FLOW                          │
├─────────────────────────────────────────────────────────────────────┤
│                                                                     │
│   CLIENT                    SERVER                    DATABASE      │
│     │                         │                          │          │
│     │  1. Login Request       │                          │          │
│     │  (email/phone,password) │                          │          │
│     │────────────────────────▶│                          │          │
│     │                         │                          │          │
│     │                         │  2. Fetch User           │          │
│     │                         │─────────────────────────▶│          │
│     │                         │                          │          │
│     │                         │  3. User Record          │          │
│     │                         │◀─────────────────────────│          │
│     │                         │                          │          │
```



12.2 JWT Token Structure

```
// JWT Header
{
  "alg": "HS256",
  "typ": "JWT"
}

// JWT Payload (Claims)
{
  "sub": "1",                             // User ID
  "email": "user@example.com",
  "userType": "CUSTOMER",
  "iat": 1706345400,                      // Issued At
  "exp": 1706431800                       // Expiration
}

// JWT Signature
HMACSHA256(
  base64UrlEncode(header) + "." + base64UrlEncode(payload),
  SECRET_KEY
)
```

12.3 Password Security

Aspect	Implementation
Algorithm	BCrypt
Cost Factor	12 rounds
Salt	Auto-generated per password
Storage	Hash only (never plaintext)

12.4 Security Headers

```
// CORS Configuration
@Configuration
public class SecurityConfig {

    @Bean
    public CorsConfigurationSource corsConfigurationSource() {
        CorsConfiguration configuration = new CorsConfiguration();
        configuration.setAllowedOrigins(Arrays.asList("http://localhost:3000
"));
        configuration.setAllowedMethods(Arrays.asList("GET", "POST", "PUT", "
DELETE"));
        configuration.setAllowedHeaders(Arrays.asList("*"));
        configuration.setAllowCredentials(true);
        return new UrlBasedCorsConfigurationSource();
    }
}
```

12.5 Input Validation

Validation Type	Implementation
Email Format	Jakarta Bean Validation (@Email)
Phone Format	Regex Pattern Validation
Password Strength	Min 8 chars, mixed case, numbers
SQL Injection	JPA Parameterized Queries
XSS Prevention	React's automatic escaping

12.6 Role-Based Access Control (RBAC)

Role	Permissions
CUSTOMER	View services, Create bookings, Write reviews, Manage own profile
PROVIDER	All CUSTOMER permissions + Manage services,

Role	Permissions
	Accept/reject bookings
ADMIN	All PROVIDER permissions + User management, KYC verification
SUPER_ADMIN	All permissions + System settings, Admin management

13. Testing Strategy

13.1 Testing Levels

```
┌──────────────────────────────────────────────────────────────────────┐
│                        TESTING PYRAMID                               │
├──────────────────────────────────────────────────────────────────────┤
│                                                                      │
│                           /\                                         │
│                          /  \                                        │
│                         / E2E\        ← End-to-End Tests             │
│                        / Tests\         (Cypress/Playwright)         │
│                       /────────\        5% of tests                  │
│                      /Integration\    ← Integration Tests            │
│                     /   Tests     \     (Spring Boot Test)           │
│                    /───────────────\    15% of tests                 │
│                   /   Unit Tests    \ ← Unit Tests                   │
│                  /                   \  (JUnit 5, Jest)              │
│                 /─────────────────────\ 80% of tests                 │
│                /───────────────────────\                             │
│                                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

13.2 Test Categories

13.2.1 Unit Tests

Component	Framework	Coverage Target
Backend Services	JUnit 5, Mockito	80%
Backend Controllers	MockMvc	75%
Frontend Components	Jest, React Testing Library	70%
Frontend Utilities	Jest	90%

13.2.2 Integration Tests

Test Type	Description
API Integration	Test complete request-response cycle
Database Integration	Test repository operations with test DB
Authentication Flow	Test login, registration, token refresh

13.2.3 Sample Test Cases

Booking Service Unit Test:

```
@ExtendWith(MockitoExtension.class)
class BookingServiceTest {

    @Mock
    private BookingRepository bookingRepository;

    @InjectMocks
    private BookingService bookingService;

    @Test
    void createBooking_Success() {
        // Given
        CreateBookingRequest request = createValidBookingRequest();

        // When
        BookingResponse response = bookingService.createBooking(1L, request);

        // Then
        assertNotNull(response);
        assertEquals("PENDING", response.getBookingStatus());
        verify(bookingRepository, times(1)).save(any(Booking.class));
    }

    @Test
    void createBooking_DuplicateActive_ThrowsException() {
        // Given
        when(bookingRepository.existsByUserUserIdAndServiceServiceIdAndBookin
gStatusIn(
            anyLong(), anyLong(), anyList()
        )).thenReturn(true);

        // When/Then
        assertThrows(BadRequestException.class, () ->
            bookingService.createBooking(1L, createValidBookingRequest())
        );
    }
}
```

13.3 Test Data Management

Environment	Database	Data Strategy
Development	Local MySQL	Seed data from init.sql
Testing	H2 In-Memory	Fresh data per test class
Staging	MySQL (Cloud)	Production-like sample data

14. Deployment Architecture

14.1 Docker Containerization

```yaml
# docker-compose.yml
version: '3.8'

services:
  frontend:
    build: ./frontend
    ports:
      - "80:80"
    depends_on:
      - backend
    networks:
      - hirelink-network

  backend:
    build: ./backend
    ports:
      - "8080:8080"
    environment:
      - SPRING_DATASOURCE_URL=jdbc:mysql://db:3306/hirelink_db
      - SPRING_DATASOURCE_USERNAME=hirelink
      - SPRING_DATASOURCE_PASSWORD=password
    depends_on:
      - db
    networks:
      - hirelink-network

  db:
    image: mysql:8.0
    ports:
      - "3306:3306"
    environment:
      - MYSQL_DATABASE=hirelink_db
      - MYSQL_USER=hirelink
      - MYSQL_PASSWORD=password
      - MYSQL_ROOT_PASSWORD=rootpassword
    volumes:
      - mysql-data:/var/lib/mysql
      - ./database/init.sql:/docker-entrypoint-initdb.d/init.sql
    networks:
      - hirelink-network

volumes:
  mysql-data:
```

```yaml
networks:
  hirelink-network:
    driver: bridge
```

14.2 Deployment Diagram

```
┌──────────────────────────────────────────────────────────────────────┐
│                        DEPLOYMENT ARCHITECTURE                       │
├──────────────────────────────────────────────────────────────────────┤
│                                                                      │
│   ┌─────────────┐                                                    │
│   │    USERS    │                                                    │
│   │  (Browser)  │                                                    │
│   └──────┬──────┘                                                    │
│          │ HTTPS (Port 443)                                          │
│          ▼                                                           │
│   ┌───────────────────────────────────────────────────┐             │
│   │               LOAD BALANCER / CDN                 │             │
│   │            (Cloudflare / AWS ALB)                 │             │
│   └──────────────────────────┬────────────────────────┘             │
│                              │                                       │
│          ┌───────────────────┼───────────────────┐                   │
│          ▼                   ▼                   ▼                   │
│   ┌─────────────┐     ┌─────────────┐     ┌─────────────┐           │
│   │  FRONTEND   │     │  FRONTEND   │     │  FRONTEND   │           │
│   │  CONTAINER  │     │  CONTAINER  │     │  CONTAINER  │           │
│   │  (Nginx)    │     │  (Nginx)    │     │  (Nginx)    │           │
│   │  Port: 80   │     │  Port: 80   │     │  Port: 80   │           │
│   └─────────────┘     └─────────────┘     └─────────────┘           │
│                              │ /api/* requests                       │
│                              ▼                                       │
│   ┌───────────────────────────────────────────────────┐             │
│   │               API GATEWAY / NGINX                 │             │
│   │             (Reverse Proxy, SSL)                  │             │
│   └──────────────────────────┬────────────────────────┘             │
│                              │                                       │
│          ┌───────────────────┼───────────────────┐                   │
│          ▼                   ▼                   ▼                   │
│   ┌─────────────┐     ┌─────────────┐     ┌─────────────┐           │
│   │  BACKEND    │     │  BACKEND    │     │  BACKEND    │           │
│   │  CONTAINER  │     │  CONTAINER  │     │  CONTAINER  │           │
│   │(Spring Boot)│     │(Spring Boot)│     │(Spring Boot)│           │
│   │  Port: 8080 │     │  Port: 8080 │     │  Port: 8080 │           │
│   └─────────────┘     └─────────────┘     └─────────────┘           │
│                              │                                       │
│                         │ JDBC Connection                      │
│                         ▼                                      │
│  ┌──────────────────────────────────────────────────┐          │
│  │                   MySQL 8.0                       │          │
│  │             (Primary - Read/Write)                │          │
│  │                   Port: 3306                      │          │
│  └──────────────────────┬───────────────────────────┘          │
│                         │                                      │
│                         │ Replication                          │
│                         ▼                                      │
│  ┌──────────────────────────────────────────────────┐          │
│  │                   MySQL 8.0                       │          │
│  │               (Replica - Read Only)               │          │
│  └──────────────────────────────────────────────────┘          │
│                                                                │
└────────────────────────────────────────────────────────────────┘
```

14.3 Environment Configuration

Environment	URL	Purpose
Development	localhost:3000 / localhost:8080	Local development
Staging	staging.hirelink.in	Pre-production testing
Production	www.hirelink.in	Live application

---

15. Conclusion

15.1 Project Summary

HireLink successfully addresses the challenges in the home services industry by providing a comprehensive digital platform that connects customers with verified local service providers. The application implements:

- **Robust Authentication**: Secure JWT-based authentication with role-based access control
- **Efficient Booking System**: Complete booking lifecycle management with status tracking
- **Provider Verification**: KYC-based verification ensuring trust and quality
- **User-Friendly Interface**: Modern, responsive design with intuitive navigation
- **Scalable Architecture**: Containerized deployment ready for horizontal scaling

15.2 Key Achievements

Objective	Status	Implementation
User Registration & Authentication	✅ Complete	JWT + BCrypt + OTP

Objective	Status	Implementation
Service Category Management	✅ Complete	Hierarchical categories
Booking System	✅ Complete	Full lifecycle management
Provider Management	✅ Complete	KYC verification, profiles
Review System	✅ Complete	Multi-dimensional ratings
Location-Based Discovery	✅ Complete	Geolocation integration
Responsive UI	✅ Complete	Mobile-first design

15.3 Future Enhancements

1. **Payment Gateway Integration**: Razorpay/Stripe for online payments
2. **Real-Time Notifications**: WebSocket-based instant notifications
3. **Mobile Applications:** Native iOS and Android apps
4. **AI Recommendations**: Machine learning for service suggestions
5. **Multi-Language Support**: Full localization for regional languages
6. **Analytics Dashboard**: Business intelligence for providers and admins

15.4 Lessons Learned

1. **Technology Selection**: Spring Boot + React combination provides excellent developer productivity
2. **State Management**: Zustand offers simpler state management compared to Redux
3. **Database Design**: Proper normalization early prevents scalability issues
4. **Security First**: Implementing security from the start is easier than retrofitting
5. **Containerization**: Docker significantly simplifies deployment and environment consistency

---

16. References

1. Spring Boot Documentation (2025). https://docs.spring.io/spring-boot/
2. React Documentation (2025). https://react.dev/
3. MySQL 8.0 Reference Manual. https://dev.mysql.com/doc/refman/8.0/en/
4. JWT.io - JSON Web Tokens Introduction. https://jwt.io/introduction
5. TailwindCSS Documentation. https://tailwindcss.com/docs
6. Docker Documentation. https://docs.docker.com/
7. OWASP Security Guidelines. https://owasp.org/
8. IEEE Software Requirements Specification Standard (IEEE 830)

---

17. Appendices

Appendix A: Glossary

Term	Definition
JWT	JSON Web Token - A compact, URL-safe means of representing claims
KYC	Know Your Customer - Identity verification process
OTP	One-Time Password - Temporary code for verification
API	Application Programming Interface
CRUD	Create, Read, Update, Delete operations
ORM	Object-Relational Mapping
REST	Representational State Transfer
SPA	Single Page Application

Appendix B: Demo Accounts

Role	Email	Password
Customer	priya.sharma@email.com	password123
Provider	ramesh.electrician@email.com	password123
Admin	admin@hirelink.in	password123

Appendix C: API Endpoints Summary

Method	Endpoint	Description
POST	/api/auth/register	User registration
POST	/api/auth/login	User login
GET	/api/categories	List all categories
GET	/api/services/{id}	Get service details
GET	/api/services/search	Search services
POST	/api/bookings	Create booking
GET	/api/bookings/my-bookings	Get user’s bookings
PUT	/api/bookings/{id}/status	Update booking status
GET	/api/providers/{id}	Get provider profile
POST	/api/bookings/{id}/review	Add review

Document Version: 1.0
Last Updated: January 27, 2026
Prepared By: HireLink Development Team
Project Support: ENSATE

This document is submitted as part of the academic project requirements for the Graduate Academic Project at UIT,ADOOR, Academic Year 2025-2026.