Software Requirements Specification (SRS)

Functional Requirements

- **User Registration & Authentication:** Customers (individual or business), drivers, and admins each have secure accounts (sign up/in by email or phone). Passwords should be stored safely (hashed).
- **User Profile Management:** Customers can view/edit their profile (name, address, contact). Business users can enter company details. Drivers have a profile with availability. Admins can manage all accounts.
- **Service Browsing & Search:** Display all cleaning services. Provide search/filter functionality (e.g. filter by garment type, price range, provider location or rating).
- **Order Management:** Customers can create new orders by selecting service(s), specifying pickup/drop locations and times. They can review, confirm, modify or cancel an order. Drivers can view and accept delivery requests. Admins can view/assign any order.
- **Scheduling:** Allow users to pick pickup and drop-off dates/times when placing an order. The system should check for available drivers in the area.
- **Pricing & Quotes:** Automatically calculate total price based on selected services and quantities. Show itemized cost before confirming. Optionally support discount codes or dynamic pricing.
- **Payment Processing:** Integrate with payment gateway(s) to accept credit card, Fawry, PayPal, etc., and update order payment status in the system. Also support in-app wallet balance: customers can pre-fund and pay from their wallet.
- **Order Tracking & Notifications:** Update order status (e.g. “Requested”, “Picked up”, “In Cleaning”, “Out for Delivery”, “Completed”) and notify the user via SMS/email/app notifications at each step. Customers and admins should be able to track progress.
- **Rating and Feedback:** After completion, prompt customers to rate the service and provide feedback. Store these reviews for display.
- **Administration:** Admin users can add/edit services, manage users, assign or reassign orders, and view analytics (e.g. total orders, revenue, driver performance).
- **Reporting/Analytics:** Generate reports on orders, usage patterns, and finances for business insight.

Non-Functional Requirements

- **Performance:** The system should respond quickly (e.g. page loads < 2 seconds) even with many concurrent users. Batch operations (like reading/writing JSON) must not block the UI.
- **Scalability:** Design should allow scaling (e.g. to more cities or countries) by easily adding new service providers and expanding server capacity.
- **Availability:** The service must be highly available (24/7 uptime) to serve users anytime. Use reliable hosting and database backups.

- **Security:** Protect all user data and transactions. Use HTTPS for data in transit, encrypt sensitive information, and follow payment security standards (PCI DSS). The system must be robust against cyber-attacks and fraud.
- **Usability:** The web interface must be intuitive and accessible (responsive design for mobile, multilingual support, and accessibility for disabled users).
- **Reliability:** Ensure data integrity (e.g. no lost orders), with proper error handling and recovery.
- **Maintainability:** Code should follow good OOP practices so it can be easily updated (e.g. adding new features or payment methods without major rewrites).

AI-Powered Features

We can enhance the platform with AI in several ways:

- **AI Chatbot / Voice Assistant:** Implement a 24/7 AI chatbot to handle common customer queries (e.g. “When will my order arrive?”) and even schedule pickups. A voice-based assistant could take orders or answer FAQs by phone.
- **Personalized Recommendations:** Use AI to analyze a user’s order history and suggest services or reminders. For example, if a customer often dry-cleans suits monthly, the system can automatically send reminders or loyalty coupons. AI can also upsell additional services (“Would you like ironing with that?”).
- **Smart Pricing/Forecasting:** Apply AI to predict demand (e.g. peak days/times) and optimize staffing. Demand forecasting keeps enough drivers available and can adjust promotions. AI can also dynamically price or suggest discounts based on load and customer profile.
- **Image Recognition:** Let users upload a photo of a garment or stain. AI can identify fabric type or stain category and provide an instant quote or care advice. This reduces errors in service selection.
- **Route Optimization:** Use AI algorithms to assign drivers to orders and compute optimal delivery routes, minimizing travel time and fuel. This improves efficiency as orders scale.
- **Sentiment Analysis:** Automatically monitor customer feedback (reviews or social media) using AI to gauge satisfaction. If negative sentiment spikes, the system can alert managers to take action.
- **Maintenance and Inventory:** On the provider side, AI can predict when equipment needs servicing to prevent downtime, manage inventory of supplies, and schedule maintenance tasks automatically. By using QR codes on items that redirect to the owner’s account, warehouse management becomes faster and more organized.

Integrations

1. Authentication & User Management

- **Auth0 / Firebase Authentication** → For secure email/phone login, password hashing, and account management.
- **Twilio Verify** → For OTP-based phone verification.
- **JWT/OAuth2** → For secure token-based authentication.

2. Location & Maps

- **Google Maps Platform / Mapbox**
  - Location autocomplete for pickup/drop-off addresses
  - Geocoding (convert address ↔ coordinates)
  - Map display for drivers & customers
  - Distance/time estimation for pricing & scheduling

3. Scheduling

- **Calendly API / Custom Scheduler** → Provide time-slot availability and conflict-free scheduling.
- **Google Calendar API (optional)** → Allow customers to add pickups/deliveries to their personal calendar.

4. Pricing & Quotes

- **Custom Pricing Engine (internal)** → You'll implement this logic yourself, but you may use:
  - **TaxJar / Avalara API (if applicable)** → For calculating taxes dynamically.

5. Payment Processing

- **Stripe / PayPal / Fawry API**
  - Card payments
  - Wallet top-up and balance tracking
  - Refunds and order cancellations
- **PCI-DSS compliance tools** (Stripe handles this if you use them)

6. Notifications

- **Twilio / Vonage** → For SMS notifications at order status updates
- **SendGrid / AWS SES / Mailgun** → For email notifications

- **Firebase Cloud Messaging (FCM)** → For push notifications in web/app

7. Order Tracking

- **Google Maps Directions API** or **OSRM (Open Source Routing Machine)** → For driver route tracking & live ETA updates.
- **WebSockets / SignalR** → For real-time updates to customers on order progress.

8. AI-Powered Features

- **Dialogflow / Rasa / OpenAI API** → For chatbot and voice assistant
- **Recommendation Engine (Custom ML Model)** → Or use AWS Personalize / Google Recommendations AI
- **Dynamic Pricing Engine** → Could be built with AWS SageMaker or a lightweight ML model
- **Image Recognition** → AWS Rekognition, Google Vision API, or custom TensorFlow model
- **Route Optimization** → Google Maps Distance Matrix API + clustering algorithms, or OR-Tools
- **Sentiment Analysis** → OpenAI API or AWS Comprehend

9. Admin & Reporting

- **Metabase / Power BI / Tableau (optional)** → For analytics dashboards
- **ElasticSearch + Kibana** → For search and analytics on user data

10. Maintenance & Inventory

- **QR Code Generator API** → For creating codes that link to item profiles
- **IoT Integrations (Optional)** → For predictive maintenance (if you scale up equipment monitoring)