

Section	Details
1. Introduction	<p>Purpose: This document specifies the software requirements for Sho8la, a hyper-local freelance marketplace. It defines the functionalities, constraints, and interactions required to connect Egyptian student developers with local business clients.</p> <p>Scope: The software covers the complete workflow of a freelance project: user registration, identity verification, job posting, proposal submission, real-time communication, and secure payment processing via a local escrow system.</p> <p>Overview: The system acts as a trusted intermediary platform. It provides a web interface for Clients to find affordable talent and for Freelancers (Students) to find work and receive payments safely.</p>
2. General Description	<p>Functions:</p> <ul style="list-style-type: none"> • Account Management: User registration, login, and role selection (Client vs. Freelancer). • Verification System: Manual review of University IDs to verify student status. • Job Marketplace: Facilities for posting jobs, browsing listings, and filtering by category/budget. • Bidding System: Submission and review of project proposals. • Escrow Payments: Secure deposit collection and fund release upon work approval. • Communication: Integrated real-time chat for project collaboration. <p>User Community:</p>

	<ul style="list-style-type: none"> • Clients: Local SME owners or individuals requiring technical services. • Freelancers: Computer Science students seeking practical work experience. • Administrators: Platform moderators managing verifications and payouts.
3. Functional Requirements	<p>FR1 (Authentication): The system shall allow users to create accounts and log in securely. It must distinguish between Client, Freelancer, and Admin roles.</p> <p>FR2 (Identity Verification): The system shall allow Freelancers to upload ID documents. Admins must be able to view these documents and approve/reject the verification status.</p> <p>FR3 (Job Management): The system shall allow Clients to post detailed job descriptions with budgets. Freelancers must be able to browse these jobs and filter them by criteria.</p> <p>FR4 (Proposals): The system shall allow Freelancers to submit bids (price/timeline) for specific jobs. Clients must be able to review these bids and "Accept" a candidate.</p> <p>FR5 (Financial Transactions): The system shall interface with an external payment gateway to process deposits. Funds must be held in an internal "Escrow" state until the Client approves the work.</p> <p>FR6 (Messaging): The system shall provide a text-based chat interface between the Client and Freelancer once a proposal is active. It must filter prohibited contact information.</p>

	<p>FR7 (Withdrawals): The system shall allow Freelancers to request payout of their earnings to local wallets. Admins must be able to approve these requests.</p>
<p>4. User Interface Requirements</p>	<p>Software Interfaces:</p> <ul style="list-style-type: none"> • The system must interface with an external Payment Gateway Service (e.g., Paymob) for processing transactions. • The system must interface with a File Storage Service for saving user-uploaded images/IDs. <p>Examples:</p> <ul style="list-style-type: none"> • Dashboard: A central hub showing active jobs, wallet balance, and notifications. • Job Board: A searchable list view of available projects. • Chat Window: A messaging interface for real-time conversation.
<p>5. Performance Requirements</p>	<p>Response Time:</p> <ul style="list-style-type: none"> • General page loads and interactions should complete within acceptable web standards (e.g., < 2 seconds). • Real-time chat messages should be delivered with minimal latency. <p>Throughput: The system shall support multiple concurrent users (Clients posting jobs and Freelancers bidding) simultaneously without service interruption.</p> <p>Scalability: The system architecture must be designed to handle a growing database of users and job history over time.</p>
<p>6. Non-Functional Attributes</p>	<p>Usability: The interface must be intuitive for non-technical Clients to post jobs easily. It should support local language/cultural context (e.g., Arabic/Franco text).</p>

	<p>Reliability: The system must ensure data consistency, particularly for financial records (Escrow states) and user reputation.</p> <p>Security:</p> <ul style="list-style-type: none"> • All sensitive user data (passwords, IDs) must be stored securely. • Payment processing must adhere to security standards, ensuring no sensitive financial data is stored directly on the platform servers.
7. Schedule and Budget	<p>Timeline: The project development is divided into phases: Frontend Design, Backend Implementation, Database Integration, and Final Testing.</p> <p>Deployment: The application will be deployed and hosted on Render.</p> <p>Cost Estimate: The total monetary cost for the project development and initial deployment is 0 (Zero Cost), utilizing free tiers of hosting and development tools.</p>
8. Appendices	<p>Glossary:</p> <ul style="list-style-type: none"> • Escrow: A financial arrangement where the system holds funds on behalf of two parties until a transaction is completed. • Client: The party posting the job and paying for the service. • Freelancer: The student party performing the work. • SME: Small and Medium-sized Enterprises.