Software Requirements Specification (SRS) for SteelFabPro

Document Version: 1.0
Date: June 21, 2025
Prepared By: Gouresh Madye

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) details the functional and nonfunctional requirements for SteelFabPro, a multi-role web and mobile application designed to streamline operations in steel fabrication mills. The primary goal of SteelFabPro is to digitize and automate manual workflows, particularly concerning inventory management, project tracking, and communication, thereby improving efficiency, transparency, and collaboration between clients, manufacturers, and administrators.

1.2 Scope

SteelFabPro will provide a centralized platform for:

- **Clients:** To upload project requirements, provide guidelines, communicate with manufacturers, track project progress, and manage payments.
- **Manufacturers:** To accept client requests, manage inventory, track fabrication tasks, submit weekly progress reports (including digitizing handwritten notes), send payment requests, and communicate with clients.
- **Administrators:** To oversee all activities, manage users, and access comprehensive reports.

This document outlines the features to be developed, the user roles involved, the system architecture, and quality attributes, forming the basis for design, development, and testing activities.

1.3 Definitions, Acronyms, and Abbreviations

- **SRS:** Software Requirements Specification
- **MVP:** Minimum Viable Product
- **UI:** User Interface
- **UX:** User Experience
- **API:** Application Programming Interface
- **DB:** Database
- **OEM:** Original Equipment Manufacturer
- **QA:** Quality Assurance
- **PoC:** Proof of Concept

- **OCR:** Optical Character Recognition
- **KPI:** Key Performance Indicator
- **PDF:** Portable Document Format
- **PNG:** Portable Network Graphics
- **JPEG/JPG:** Joint Photographic Experts Group
- **REST:** Representational State Transfer (for APIs)
- **SQL:** Structured Query Language
- **CRUD:** Create, Read, Update, Delete

1.4 References

- Previous discussions and feature lists provided by the client.
- System Design Diagram: https://drive.google.com/file/d/1odN4Xz1qcAbzsvKXH1s3kTfFNtZvvS4X/view?usp=sharing
- Similar Projects/Research Section (from previous report).

2. Overall Description

2.1 Product Perspective

SteelFabPro is a standalone, cloud-based application that integrates various functionalities that are currently handled manually or through disparate, non-integrated tools within steel fabrication mills. It will serve as the primary digital interface for managing fabrication projects from inception to completion, bringing clients, manufacturers, and administrators onto a single platform.

2.2 Product Functions (High-Level Summary)

- **User & Role Management:** Secure authentication and authorization for different user types.
- **Project Lifecycle Management:** From client requirements upload to project acceptance and completion tracking.
- **Inventory & Material Management:** Real-time tracking of raw materials and finished goods.
- **Communication & Collaboration:** In-app chat, notifications, and structured reporting.
- **Financial Tracking:** Payment requests and confirmations.
- **Reporting & Analytics:** Overviews and detailed insights for administrators.

2.3 User Classes and Characteristics

User Role	Characteristics	Key Responsibilities in App
Client	Architects, contractors, builders, or other entities	Upload project requirements (specs,

	requiring steel fabrication. May not be tech-savvy, requires intuitive interface.	drawings). Provide guidelines. Chat with Manufacturer. Receive & confirm payment requests. View project progress via weekly reports.
Manufacturer	Steel fabrication mill owners, project managers, supervisors, inventory managers. May include shop-floor personnel using mobile devices.	Review & accept client requirements. Manage material inventory (input, deduct). Log fabrication tasks. Submit weekly progress reports with images. Scan/process supervisor handwritten notes into tasks. Send payment requests. Chat with Client.
Admin	System administrators, company owners, high-level management. Requires comprehensive oversight and reporting.	User management (create/edit roles). Monitor all ongoing projects. View overall inventory status. Access aggregated reports and analytics. Set system configurations.

2.4 Operating Environment

- **Client Devices:** Web browsers (Chrome, Firefox, Safari, Edge) on desktop/laptop. Mobile devices (smartphones, tablets) running Android or iOS (responsive web app initially, dedicated mobile apps for future phases).
- **Backend Hosting:** Cloud-based platform (e.g., AWS, Azure, Google Cloud).
- **Database:** PostgreSQL.

2.5 General Constraints

- **Performance:** Must handle concurrent users (up to 100-200 initially, scalable to thousands).
- **Security:** Adherence to industry best practices for data encryption, access control, and vulnerability prevention.
- **Usability:** Intuitive and easy-to-use interface, especially for shop-floor personnel.

- **Data Accuracy:** High emphasis on accurate inventory and project data.
- **Scalability:** Designed with a microservices architecture to allow for future expansion and increased load.
- **Maintainability:** Codebase should be clean, modular, and well-documented.
- **Technology Stack:** Adherence to Java/Spring Boot for backend, React.js/JavaScript for frontend, PostgreSQL for database.

3. Functional Requirements

This section details the specific functionalities for each user role.

3.1. User & Role Management (Core System)

- **FR1.1 User Registration:**
  - **FR1.1.1:** The system shall allow new users to register with unique email addresses.
  - **FR1.1.2:** The system shall require password creation with minimum complexity rules.
  - **FR1.1.3:** The system shall assign a default 'Client' role upon initial registration, requiring Admin approval for 'Manufacturer' or 'Admin' roles.
- **FR1.2 User Authentication:**
  - **FR1.2.1:** The system shall allow users to log in with their registered credentials (email/password).
  - **FR1.2.2:** The system shall implement secure password hashing and salting.
  - **FR1.2.3:** The system shall provide a "Forgot Password" functionality.
- **FR1.3 Role-Based Access Control (RBAC):**
  - **FR1.3.1:** The system shall enforce distinct access permissions and UI views based on assigned roles (Client, Manufacturer, Admin).
  - **FR1.3.2:** The system shall restrict unauthorized access to features not pertaining to the user's role.
- **FR1.4 User Profile Management:**
  - **FR1.4.1:** Users shall be able to view and edit their profile information (e.g., name, contact details).

3.2. Client Functional Requirements

- **FR2.1 Project Requirements Upload:**
  - **FR2.1.1:** The client shall be able to create a new project by providing a project name, description, and key details.
  - **FR2.1.2:** The client shall be able to upload multiple files (PDFs, JPEGs, PNGs, CAD files - initially just image/PDF support) as part of the project requirements.
  - **FR2.1.3:** The system shall store uploaded documents securely and associate them with the specific project.
  - **FR2.1.4:** The system shall notify relevant Manufacturers of new project requirements.

- **FR2.2 Project Progress Tracking:**
  - **FR2.2.1:** The client shall be able to view a list of all their active and completed projects.
  - **FR2.2.2:** For each project, the client shall see its current status (e.g., "Requirements Uploaded," "Accepted by Manufacturer," "In Progress," "Payment Requested," "Completed").
  - **FR2.2.3:** The client shall be able to view weekly progress reports submitted by the assigned Manufacturer, including summary text and attached images.
- **FR2.3 Client-Manufacturer Communication (Chat):**
  - **FR2.3.1:** The client shall be able to initiate and participate in a real-time chat with the assigned Manufacturer for a specific project.
  - **FR2.3.2:** The chat interface shall support text messages. (Future: file attachments).
  - **FR2.3.3:** The system shall provide in-app notifications for new chat messages.
- **FR2.4 Payment Management:**
  - **FR2.4.1:** The client shall receive notifications for payment requests from the Manufacturer for their projects.
  - **FR2.4.2:** The client shall be able to view details of the payment request (amount, description).
  - **FR2.4.3:** The client shall be able to mark a payment request as "Payment Done" (a simple tick/confirmation button).
  - **FR2.4.4:** The system shall notify the Manufacturer once payment is confirmed by the client.
- **FR2.5 Project Guidelines/Documentation Upload:**
  - **FR2.5.1:** After a project is assigned, the client shall be able to upload additional documentation or guidelines specific to the fabrication process.
  - **FR2.5.2:** The system shall associate these guidelines with the project and make them accessible to the Manufacturer.

3.3. Manufacturer Functional Requirements

- **FR3.1 Project Review and Acceptance:**
  - **FR3.1.1:** The Manufacturer shall receive notifications for new client project requirements.
  - **FR3.1.2:** The Manufacturer shall be able to view client project requirements, including uploaded documents.
  - **FR3.1.3:** Based on review, the Manufacturer shall be able to "Accept" or "Decline" a project request.
  - **FR3.1.4:** Upon acceptance, the system shall update the project status and notify the client.
- **FR3.2 Weekly Progress Reporting:**
  - **FR3.2.1:** The Manufacturer shall be able to create and submit weekly progress reports for active projects.
  - **FR3.2.2:** Each report shall include a text summary of work done.

    - **FR3.2.3:** The Manufacturer shall be able to upload multiple images (JPEG/PNG) to accompany the report, showing work progress.
    - **FR3.2.4:** The system shall send reminders to the Manufacturer to submit weekly reports by a configurable deadline.
    - **FR3.2.5:** The Manufacturer shall be able to customize report formats, including uploading their company logo.
    - **FR3.2.6:** Submitted reports shall be accessible to the client for viewing.
- **FR3.3 Inventory Management:**
    - **FR3.3.1:** The Manufacturer shall be able to input new materials into inventory with their initial quantity (e.g., weight in kg, length in meters).
    - **FR3.3.2:** The Manufacturer shall be able to record which part of the fabrication (e.g., a specific task or sub-assembly) is using a specific amount of material.
    - **FR3.3.3:** As each task or unit of work is completed/logged, the system shall automatically deduct the "required material" for that task from the "total material" in inventory.
    - **FR3.3.4:** The system shall display the remaining material for each type in real-time.
    - **FR3.3.5:** The system shall provide alerts for low stock levels of materials.
    - **FR3.3.6:** The Manufacturer shall be able to view a historical log of material movements (in/out).
    - **FR3.3.7:** The system shall allow for recording of wastage material separately.
- **FR3.4 Task Assignment from Handwritten Notes:**
    - **FR3.4.1:** The Manufacturer (or supervisor) shall be able to upload an image of a handwritten note/report.
    - **FR3.4.2:** The system shall display the uploaded image.
    - **FR3.4.3 (MVP):** The Manufacturer shall manually transcribe key information from the note into structured fields (e.g., "Task Description," "Assigned To," "Due Date").
    - **FR3.4.4 (Future - OCR Integration):** The system shall attempt to perform OCR on the handwritten note image to extract text, which can then be used to pre-fill task fields, requiring human verification and correction.
    - **FR3.4.5:** The system shall convert these processed notes into trackable tasks/assignments associated with a project.
- **FR3.5 Material Procurement Request:**
    - **FR3.5.1:** The Manufacturer shall be able to generate a request for the required amount of materials for a project, including main material, wastage allowance, and testing material.
    - **FR3.5.2:** This request shall be visible internally (e.g., for procurement department). (Future: Integrates with external supplier modules).
- **FR3.6 Manufacturer-Client Communication (Chat):**
    - **FR3.6.1:** The Manufacturer shall be able to initiate and participate in a real-time chat with the assigned client for a specific project.
    - **FR3.6.2:** The chat interface shall support text messages. (Future: file attachments).
    - **FR3.6.3:** The system shall provide in-app notifications for new chat messages.

- **FR3.7 Payment Request:**
  - **FR3.7.1:** The Manufacturer shall be able to send payment requests to the client for completed milestones or the entire job.
  - **FR3.7.2:** The payment request shall include the project name, amount, and a brief description.
  - **FR3.7.3:** The system shall display the status of payment requests (e.g., "Sent," "Confirmed by Client," "Overdue").

3.4. Administrator Functional Requirements

- **FR4.1 User Management:**
  - **FR4.1.1:** The Admin shall be able to create, edit, and delete user accounts.
  - **FR4.1.2:** The Admin shall be able to assign and change user roles (Client, Manufacturer, Admin).
  - **FR4.1.3:** The Admin shall be able to activate or deactivate user accounts.
- **FR4.2 Project Oversight:**
  - **FR4.2.1:** The Admin shall have a comprehensive overview dashboard displaying all ongoing and completed projects across all clients and manufacturers.
  - **FR4.2.2:** The Admin shall be able to view detailed information for any project, including requirements, reports, material usage, and communication logs.
- **FR4.3 Inventory Overview:**
  - **FR4.3.1:** The Admin shall be able to view the aggregated inventory status across all manufacturers (if a multi-manufacturer setup is implemented, or just the inventory of the main mill).
  - **FR4.3.2:** The Admin shall be able to view material usage trends over time.
- **FR4.4 Reporting and Analytics:**
  - **FR4.4.1:** The Admin shall be able to generate reports on key performance indicators (KPIs) such as:
    - Project completion rates.
    - Material consumption trends.
    - Manufacturer performance (e.g., report submission adherence).
    - Payment status summaries.
  - **FR4.4.2:** Reports shall be viewable within the app and exportable (e.g., to CSV or PDF).
- **FR4.5 System Configuration:**
  - **FR4.5.1:** The Admin shall be able to configure system-wide settings (e.g., notification thresholds, report submission frequency reminders).
  - **FR4.5.2:** The Admin shall manage templates for manufacturer reports (e.g., fields, placeholder for logos).

4. Non-Functional Requirements

4.1. Performance Requirements

- **Response Time:**
  - UI load time: Max 3 seconds for initial page load.
  - API response time: Max 500ms for data retrieval operations, 1 second for create/update operations under normal load.
- **Throughput:** Support at least 100 concurrent active users without significant performance degradation.
- **Scalability:** The system architecture must be designed to scale horizontally to accommodate increased user load and data volume without major re-architecture. Microservices, load balancing, and database read replicas are key.
- **Data Volume:** The system shall efficiently manage large volumes of project data, material transactions, and chat messages over several years.

4.2. Security Requirements

- **Authentication:** Implement JWT (JSON Web Tokens) or OAuth2 for secure API authentication.
- **Authorization:** Strict role-based access control (RBAC) to ensure users can only access features and data permitted by their role.
- **Data Encryption:**
  - Data in transit (client-server communication) must be encrypted using HTTPS/TLS.
  - Sensitive data at rest (database, cloud storage) must be encrypted.
- **Input Validation:** All user inputs must be rigorously validated on both client and server sides to prevent SQL injection, XSS, and other common vulnerabilities.
- **Audit Trails:** Log all critical system actions (e.g., user login attempts, project status changes, material deductions, payment confirmations) for traceability and security auditing.
- **Session Management:** Secure session management to prevent session hijacking.
- **Vulnerability Management:** Regular security audits and penetration testing.

4.3. Reliability Requirements

- **Availability:** The system shall be available 99.9% of the time (excluding planned maintenance windows).
- **Data Integrity:** All data transactions (e.g., inventory deduction, task completion) must maintain data consistency and integrity, adhering to ACID properties.
- **Error Handling & Logging:** Robust error handling mechanisms with comprehensive logging to facilitate quick debugging and issue resolution.
- **Backup & Recovery:** Daily automated backups of the database and cloud storage, with a clear recovery plan in case of disaster.

4.4. Usability Requirements

- **Intuitive UI/UX:** The user interface for all roles shall be intuitive, easy to navigate, and require minimal training, especially for Manufacturer's shop floor personnel.
- **Consistency:** Consistent design language, navigation patterns, and terminology across the entire application.
- **Feedback:** Provide clear and immediate feedback to users for their actions (e.g., success messages, error notifications, loading indicators).
- **Accessibility:** (Future consideration) Adherence to WCAG guidelines for accessibility where applicable.
- **Responsiveness:** The UI shall be responsive and adapt well to various screen sizes (desktop, tablet, mobile).

4.5. Maintainability Requirements

- **Modularity:** The microservices architecture promotes modularity, allowing independent development, testing, and deployment of services.
- **Code Quality:** Adherence to coding standards, best practices, and thorough code reviews.
- **Documentation:** Comprehensive technical documentation for APIs, database schema, and deployment procedures.
- **Testability:** High unit test coverage and robust integration/end-to-end test suites.

4.6. Portability Requirements

- **Cloud Agnostic (High-Level):** While initially targeting AWS, the architecture should avoid deep vendor lock-in where possible, allowing for migration to other cloud providers if needed.
- **Browser Compatibility:** Compatible with the latest stable versions of major web browsers (Chrome, Firefox, Safari, Edge).

5. Data Model (High-Level Entities)

This section provides a high-level overview of key data entities and their relationships.

- **Users:** (User ID, Name, Email, Password Hash, Role (Client/Manufacturer/Admin), Contact Info, Status)
- **Projects:** (Project ID, Project Name, Description, Client ID, Manufacturer ID, Status, Start Date, End Date, Budget)
- **Requirements:** (Requirement ID, Project ID, Description, File Paths, Upload Date)
- **Weekly Reports:** (Report ID, Project ID, Manufacturer ID, Week Number, Summary, Image Paths, Submission Date)
- **Materials:** (Material ID, Material Name, Unit of Measure (kg, meter), Initial Quantity, Current Quantity, Low Stock Threshold)
- **Inventory Transactions:** (Transaction ID, Material ID, Project ID (nullable), Quantity

Change, Type (In/Out/Wastage), Date/Time, User ID)
- **Fabrication Tasks:** (Task ID, Project ID, Description, Material Consumed (reference to Materials/Quantity), Status, Assigned To, Due Date, Completion Date, Derived From Note ID (nullable))
- **Handwritten Notes:** (Note ID, Project ID, Image Path, Upload Date, Transcribed Text (if OCR'd), Processed Status)
- **Chat Messages:** (Message ID, Project ID, Sender ID, Receiver ID, Message Content, Timestamp)
- **Payment Requests:** (Payment ID, Project ID, Manufacturer ID, Client ID, Amount, Description, Status (Sent/Confirmed/Overdue), Request Date, Confirmation Date)
- **Company Profiles (for Manufacturer):** (Company ID, Manufacturer ID, Company Name, Logo Path, Report Template Settings)

6.1. UI Wireframes / Mockups (Refer to design phase)

- *Placeholder for links to wireframes once created.*