Functionality Requirement Document (FRD)

Project Name: Client Accel
Version: v1.0.0 (MVP)
Release Date: 15 July 2025
Prepared For: Internal Stakeholders, Product Team, Development Team, QA Team
Created By: Venkatesh P
Reviewed By:
Approval Status: Draft
Document Version: 1.0
Document Status: Active
Confidentiality Level: Internal Use Only

V-ACCEL
AI DYNAMICS PRIVATE LIMITED

Table of Contents

1. **Abstract**
2. **Introduction**
3. **System Overview**
4. **Module Overview**
   - 4.1. Proposal Management
   - 4.2. Client Management
   - 4.3. Project Management
   - 4.4. Payment Management
   - 4.5. Team management
   - 4.6. User Management
5. **Feature Specification**
6. **User Roles and Permissions**
7. **User Interface Requirements**
   - 7.1. Layout and Responsive
   - 7.2. Navigation Flow
   - 7.3. Form Design and Interactions
   - 7.4. Compatibility Testing
   - 7.5. Regression and Smoke Tests
8. **Input Validation and Error Handling**
9. **System Behaviour and Edge Case Handling**
10. **Performance Benchmarks**
11. **Integration Points**
12. **Security Considerations**
13. **Testing Guide**
    - 13.1. Testing Scenarios
    - 13.2. Functional Testing
    - 13.3. UI Testing
    - 13.4. Performance Testing
    - 13.5. Compatibility Testing
14. **Appendices**
    - 14.1. Terminology

14.2. System Diagram
14.3. Revision History

1. Abstract

This document outlines the complete functionality and expected behavior of Client Accel, version 1.0.0, which constitutes the Minimum Viable Product (MVP) release. Client Accel is a multi-tenant, web-based client management processes. The system is built using modern web technologies including React, Node.js, and MongoDB, providing a scalable and secure platform for managing client across multiple organizations.

2.Introduction

2.1 Purpose

The purpose of this Functionality Requirement Document (FRD) is to define the complete scope of functional behaviors, user interactions, and system responses for the Client Accel v1.0.0 release. This document ensures that all implemented features meet the intended business requirements and are verifiable through structured testing.

2.2 Scope

This document covers:

- Client Management functionalities
- User authentication and authorization
- Payment management
- Project Management

The document exclusively focuses on features and specifications shipped in version 1.0.0 of the application. Features planned for future releases are briefly referenced but not described in detail.

2.3 Intended Audience

This document is intended for:

- **Product Managers** – to validate alignment with the MVP vision
- **Software Engineers** – to implement features and integrations accordingly
- **Quality Assurance Engineers** – to write functional, UI, and regression test cases
- **UI/UX Designers** – to maintain design consistency and behaviour expectations
- **Project Stakeholders** – to track progress against defined deliverables

3. System Overview

3.1 System Description

Client Accel is a cloud-based web application to manage their clients efficiently. The system supports multiple organizations (tenants) with isolated data and configurations, while maintaining a unified codebase and infrastructure.

3.2 Key Functional Modules

1. User Management
2. Organization Management
3. Client Management
4. Project Management
5. Payment Management

3.3 Target Users

The application supports multiple user personas with distinct access levels:

Role	Description
Super Admin	System-wide access and configuration
Tenant Admin	Organization-level management
Manager	Project and Team Management
Member	Project Information
Viewer	Project information

3.4 Architecture & Platform

- **Frontend:** React.js with TypeScript, Material UI, Tailwind CSS
- **Backend:** Node.js with Express.js, TypeScript
- **Database:** MongoDB
- **Authentication:** JWT-based
- **Build Tool:** Vite

4. Modules Overview

4.1 User Management

4.1.1 Feature: User Authentication

- JWT-based authentication

- Role-based access control
- Password encryption using bcrypt
- Multi-tenant isolation

4.2 Proposal Management

4.2.1 Feature: Proposal Creating

- Proposal creation
- Document collection
- Client Details

4.3.2 Feature: Proposal Information

- Proposal details
- Payment Info
- Client Details
- SPOC Details

4.3 Client Management

4.4.1 Feature: Client Application

- **Client summary**
- General summary
- Meeting Tracking
- Approval workflow
-

4.5 Project & Milestone Management

4.5.1 Feature: Project Lifecycle

- Project Creation
- Team management
- Milestone breakdown with deadlines
- Payment Splits

4.6 Team & User Management

4.6.1 Feature: Role Bases Access

- User Management (only for super-admin)
- Invite Team members for project
- Assign RBAC for users

5. User Roles and Permissions

Role	Permissions
Super Admin	Create and manage the organization and users
Tenant Admin	Create the team and manage the organization
Manager	View and Manage Projects , Proposals and clients
Member	View access only the assigned proposals, clients and projects
Viewer	View access only the proposals, clients and projects

6. User Interface Requirements

6.1 Layout and Responsiveness

- Material UI components
- Tailwind CSS for custom styling

6.2 Navigation Flow

- Intuitive navigation structure
- Quick access menu
- Role-based menu items

6.3 Form Design

- Consistent form layouts
- Real-time validation
- Error handling
- Success feedback

6.3 Table Design

- Consistent form layouts
- Pagination for large datasets
- Action Button Inline
- Role Based visibility for sensitive columns

7. Technical Specifications

7.1 System Requirements

- Node.js v18+
- MongoDB v6.x+
- Modern web browsers
- Minimum 4GB RAM for development
-

7.2 Performance Benchmarks

- Page load time < 2s
- API response time < 500ms

- Real-time updates < 1s
- Concurrent users support: 1000+

8. Security Considerations

- JWT-based authentication
- Role-based access control
- Data encryption
- Rate limiting
- Input validation
- XSS protection
- CSRF protection

9. Testing Guide

9.1 Test Scenarios

- Unit testing
- Integration testing
- End-to-end testing
- Performance testing
- Security testing

9.2 Test Cases

- Layout responsiveness
- Button tap/click feedback
- Long-press vs right-click equivalency
- Scroll behavior and fixed headers