GLOBALVISTahr – Professional Technical Documentation

Version 1.0
Technologies: MERN Stack + Tailwind CSS

This document contains the complete technical overview, architecture, backend structure, frontend implementation, API specifications, deployment workflow, and future scalability roadmap for the GlobalVistaHR platform.

TABLE OF CONTENTS

1. PROJECT OVERVIEW

GlobalVistaHR is a modern recruitment consultancy website built to streamline hiring workflows for both clients and job seekers. The platform enables companies to submit hiring requirements, while candidates can upload their resumes directly. The website also includes service details, interactive animations, industry-focused pages, a blog section, and contact capabilities.

2. SYSTEM ARCHITECTURE

The system follows a client–server architecture where the frontend communicates with the backend via REST APIs. The backend manages data processing, file uploads, and email notifications, while MongoDB serves as the centralized database. Cloudinary is used for resume storage.
Frontend → React + Tailwind CSS
Backend → Node.js + Express
Database → MongoDB Atlas
File Storage → Cloudinary
Email Service → Brevo / Nodemailer

3. FRONTEND DOCUMENTATION

The frontend is built using React.js with Tailwind CSS for styling. React Router manages navigation and Axios handles API communication. The UI includes advanced animations built using Framer Motion and Lottie.
Main Pages:
Home, About, Services, Industries, Clients, Candidates, Insights, Contact

4. BACKEND DOCUMENTATION

The backend uses Express.js to expose REST APIs that handle form submissions, resume uploads, and inquiry processing. Multer handles file uploads and Cloudinary stores the resumes. All data is stored in MongoDB using Mongoose models.

5. DATABASE STRUCTURE

Candidate: name, email, phone, resumeUrl, createdAt
HireRequest: companyName, name, email, phone, requirements, createdAt
Message: name, email, message, createdAt

6. API DOCUMENTATION

POST /api/candidates/upload – Upload candidate resume
POST /api/hire – Submit hire request
POST /api/contact – Send contact message

7. FILE UPLOAD HANDLING

The file upload process uses Multer to capture the file and then uploads it to Cloudinary. A secure URL is generated and stored in MongoDB.

8. EMAIL NOTIFICATION SYSTEM

The system sends admin email notifications for every new CV upload, hire request, or message. Brevo or Nodemailer is used for SMTP handling.

9. DEPLOYMENT INSTRUCTIONS

Frontend → Vercel (Free Hosting)
Backend → Render (Free Hosting)
Database → MongoDB Atlas (Free Cluster)
Files → Cloudinary (Free Tier)

10. SECURITY MEASURES

- CORS enabled
- File type restrictions
- Input sanitization
- MongoDB injection protection
- Recommended: Helmet.js for advanced security

11. FUTURE ENHANCEMENTS

- Full ATS System
- Admin Dashboard
- Job Posting Module
- Role-Based Authentication
- Analytics Dashboard