Project Report: LogiCalc AI – An Intelligent Goods Transport Pricing System

---

1. Executive Summary

LogiCalc AI is an intelligent, AI-driven goods transportation platform designed to simplify and optimize shipping services. By integrating, **machine learning pricing engine**, the system ensures accurate fair, transparent pricing.The platform bridges the gap between customers and transport providers by leveraging **real-time data** on distance, fuel costs, and demand-supply ratios. Its scalability and adaptability make it suitable for city-wide and intercity logistics solutions.

2. Vision & Core Concept

The project aims to create a **seamless, transparent, and data-driven online logistics marketplace**. Unlike traditional systems where users manually input shipment details, LogiCalc AI introduces an **AI-powered volume calculator** that eliminates errors and fraud.

The core differentiators include:.

- **Dynamic pricing engine** factoring in real-time costs and surge conditions.

3.User Flow

1. **Input:** Customer enters pickup and delivery locations.
2. **Instant Quote:**
   .
    - Distance retrieved via **Google Distance Matrix API**.
    - Pricing engine calculates base cost + service charge + surge multiplier.
3. **Booking:** User reviews price breakdown and confirms booking.

---

4. System Architecture

The solution consists of three primary modules working in sync:n.

4.2 Dynamic Pricing Engine

The pricing engine computes the final cost with the formula:

`Final Price = (Base Transport Cost + Driver Service Charge) × Surge Multiplier`

- **A. Base Transport Cost:**
    - `Fuel Cost = (Distance_km ÷ Vehicle Efficiency) × Current Fuel Price`
    - Includes vehicle maintenance allowance.
- **B. Driver Service Charge:**
    - `Driver Charge = Distance_km × Service Rate per km`
    - Covers time, insurance, and margin.
- **C. Surge Pricing (Rule-Based):**
    - Demand-Supply ratio per zone.
    - Time-based peak multipliers.
    - Holiday multipliers.
    - Surge factor is mapped to tiers (1.0x, 1.5x, 2.0x, etc.) for simplicity.

4.3 Zonal Management System

- Cities divided into geographic zones (postal codes/suburbs).
- **Redis** database stores and updates:
    - Active demand (customer requests).
    - Available supply (drivers).

- Zone-level surge multipliers applied dynamically.

---

5. Data Flow – Single Price Request

1. User uploads photos + locations.
2. **Image Server:** Processes photos → ML model returns `volume_m3`.
3. **Mapping Server:** Fetches distance and duration via Google API.
4. **Pricing Engine:**
    - Retrieves fuel price (from live API).
    - Retrieves surge multiplier (from Redis).
    - Executes final price computation.
5. **Web/App Server:** Returns transparent breakdown:
   Base Cost, Fuel Cost, Surge Multiplier, Final Price.

---

6. Key Innovative Features

- **Computer Vision for Logistics:** Eliminates manual input, reducing fraud/errors.
- **Transparent Pricing:** Customers see real-time breakdown of costs.
- **Adaptive Surge Management:** Optimizes driver availability during peak hours.
- **Scalability:** Cloud-native design allows rapid expansion into new geographies.

---

7. Technology Stack

- **Frontend:** React Native (mobile apps), React.js (web).
- **Backend:** Python (Django / Flask).

- **Machine Learning:** TensorFlow / Keras.
- **Mapping Services:** Google Distance Matrix API.
- **Real-Time Database:** Redis.
- **Cloud Infrastructure:** AWS / GCP for storage and computation.
