✅ 1. Login & User Authentication System

(login.html, register.html, forgot_password.html, reset_password.html, logout.html)

The system includes a full secure user management module:

✓ User Registration (with admin approval)

Users sign up with name, email, username, and password.
Admins must approve new accounts before users can log in.

✓ User Login

JWT-based secure token authentication.

✓ Forgot + Reset Password

Email-based reset link (via Gmail SMTP).

✓ Logout

Clears JWT token and redirect to login.

This ensures only authorized users can access telematics data.

✅ 2. Global Telematics Dashboard

(HomePage.html)

This is the **main dashboard** after logging in.

🔹 Live Global Map (Leaflet.js)

Displays all machines/devices on a world map.
Each marker shows:

- Machine Name
- Location (lat/lon)
- Serial Number

🔹 Device List Panel

Shows all connected machines:

- Name / Model
- Online/Offline status
- Engine hours
- “More Info” button for details

🔹 Realtime Data via WebSocket

Your ESP32 sends live CAN/parameter data → Flask API → WebSocket → Dashboard updates instantly.

✅ 3. Device Detailed View

When a device is selected:

✓ Full machine information:

- Firmware version
- Configuration file name
- Max storage
- Free storage
- Last sync
- Last known location
- Machine type / model
- VIN
- Live parameters

✓ Detail Map

Shows exact last location of the machine.

✓ Vehicle Image

Placeholder image that can be replaced with real photos.

✅ 4. Live Parameter Monitoring

User can select which machine parameters to monitor:

- Engine RPM
- Temperature
- Pressure
- Fuel level
- Any CAN/MPL data
  (mapping depends on your CAN decode logic)

Dashboard automatically displays only chosen parameters.

✅ 5. Logger Data Storage Browser

(Under Device Detailed View → Logged Data)

This feature works like a cloud file explorer:

Folder View

Each folder = a date (example: 2025-11-12)

Shows:

- Folder name
- Number of log files

- Folder size

File View

Displays:

- Filenames
- Size
- Last modified
- Select multiple files

Download Options

User can download logs in format:

- CSV
- Excel
- Trace

This is perfect for analyzing CAN raw logs or re-playing logs.

✅ 6. Admin Panel

(admin.html)

A dedicated admin-only backend.

Features:

- View all users
- Approve pending new users
- Delete users (except admin)
- View extra user details (DOB, mobile, company)

Clear separation of user vs admin permissions.

✅ 7. Flask Backend (app.py)

Your Flask backend handles:

✓ User accounts & authentication

JWT tokens
Admin approval
Password reset email system

✓ Device management API

Stores device details:

- Name
- Serial

- Lat/Lon
- Firmware
- Storage
- Live parameters
- Log folders / files

✓ Receive Live CAN Data

/api/live-data endpoint receives live data from ESP32 and pushes updates to dashboard via WebSocket.

✓ File/Folder database models

Organized with SQLAlchemy ORM.

✓ Email system

Used for password reset.

✓ CORS support + SocketIO support

Allows cross-platform and real-time communication.

🎯 8. ESP32 Integration

Your ESP32 CAN Logger sends data to Flask using:

- REST API for uploads
- WebSocket for live data
- File uploads for completed log files

This dashboard is fully ready to integrate with your CAN logger.