

App Summary

Repo-based snapshot of the Laravel app in `C:\laragon\www\himaPlus_backend-main`.

What It Is

A Laravel 10 application with web, admin-panel, and ``/api`` routes for a marketplace-style workflow around products, cart, orders, quotations, maintenance requests, offers, ratings, chat, and payments.

The codebase supports both browser views and JWT-protected API clients, with email/SMS verification, admin management, and provider-specific flows.

Who It's For

Primary persona: an authenticated customer/buyer who browses products or providers, adds items to cart, places direct orders or service requests, reviews offers, pays online, and tracks updates. The repo also includes provider (``user_type = 2``) and admin roles.

What It Does

- Registers and authenticates users with verification codes, password reset, JWT auth, and account/profile updates.
- Lists products, categories, countries, providers, featured items, and offers; supports search, filtering, sorting, and ratings.
- Manages cart and favorites, then converts grouped cart items into provider-specific orders.
- Supports three order flows: direct product orders, quotation requests, and maintenance requests with file uploads.
- Lets providers submit offers and lets users accept/reject them, with timeline updates across order states.
- Includes in-app chat, notifications, email jobs, and Firebase push messaging for order and chat events.
- Provides online payment integration through Paymob, plus coupon, VAT, and order total handling.

How It Works

- Routing: ``routes/web.php`` serves user/vendor browser flows, ``routes/admin.php`` serves the admin panel, and ``routes/api.php`` exposes public and authenticated API endpoints.
- Application layer: controllers under ``App\Http\Controllers\Api``, ``Front``, and ``Admin`` orchestrate requests; some admin logic is delegated to repository classes under ``app/Http/Repositories/Eloquent/Admin``.
- Domain/data: Eloquent models include ``User``, ``Product``, ``Order``, ``OrderItem``, ``Offer``, ``Cart``, ``Favorite``, ``Rating``, ``Conversation``, ``Message``, ``Notification``, ``Category``, ``Country``, and ``Info``, backed by migrations in ``database/migrations``.
- Supporting services: ``PaymobService`` handles payment URLs/webhooks, ``FairbaseService`` sends Firebase push notifications and chat payloads, ``ForJawalyService`` is configured for SMS, and queued mail/SMS jobs deliver account and order messages.
- Flow: client request -> route middleware (``limitReq``, auth, role checks) -> controller validation/business logic -> Eloquent/repository persistence -> notifications/jobs/payment callbacks -> JSON or Blade response.

How To Run

- Install PHP dependencies: ``composer install``.
- Create local config from the template if needed: ``.env.example`` -> ``.env``, then set MySQL and Paymob values used by the repo.
- Generate the app key: ``php artisan key:generate``.
- Create the schema: ``php artisan migrate``.
- Start the app: ``php artisan serve`` for Laravel; optionally run ``npm install`` and ``npm run dev`` for Vite assets.

Not Found In Repo

No project-specific README or deployment guide was found. The exact product branding, official business description, seed data steps, and production runbook are not documented in this repo.