

# ShopFlow - E-Commerce Mobile App

A full-featured e-commerce mobile application built with Flutter, featuring real-time product management, secure payments via PayHere, and a dynamic admin-configurable storefront powered by Supabase.

## Features

- **Authentication** — Email/password sign-up & login via Supabase Auth, with onboarding flow for first-time users
- **Product Catalog** — Paginated product listing with categories, brands, variations (size/color), and search
- **Shopping Cart** — Persistent cart with quantity management, variation support, and fast-checkout option
- **Order Management** — Full order lifecycle: placement, payment, tracking status, and order history
- **Payment Integration** — PayHere payment gateway with sandbox/production toggle (configurable from backend)
- **QR Code Scanner** — Scan product QR codes for instant product lookup
- **Product Reviews** — Customer reviews with ratings, approval workflow, and rating indicators
- **Address Management** — Multiple shipping/billing addresses per user
- **Dynamic Theming** — Light/dark mode controlled from Supabase admin panel
- **Remote Configuration** — App colors, currency, shipping costs, and maintenance mode all configurable remotely
- **PDF Invoice Generation** — Generate and print order invoices
- **Wishlist** — Save favorite products for later

## Tech Stack

Layer	Technology
Framework	Flutter 3.x (Dart SDK ^3.7.2)
State Management	GetX
Backend	Supabase (Auth, Database, Realtime)
Payments	PayHere Mobile SDK
Local Storage	GetStorage + SharedPreferences
PDF	pdf + printing packages
Networking	http package
UI	Custom widgets, Lottie animations, Shimmer

Layer	Technology
	loading

## Screenshots

Home	Product Detail	Cart	Orders

## Getting Started

### Prerequisites

- Flutter SDK ^3.7.2
- Dart SDK ^3.7.2
- A Supabase project with the required tables (see [ARCHITECTURE.md](#))
- PayHere merchant account (for payment testing)

### Installation

```
git clone https://github.com/MohamedAfrasDev/flutter-ecommerce-app.git
cd flutter-ecommerce-app
flutter pub get
```

### Environment Setup

This app uses compile-time environment variables for sensitive configuration. Copy the example file and fill in your credentials:

```
cp .env.example .env
```

Run the app with environment variables:

```
flutter run --dart-define=SUPABASE_URL=your_url --dart-define=SUPABASE_ANON_KEY=your_key --dart-define=PAYHERE_MERCHANT_ID=your_id --dart-define=PAYHERE_SECRET_KEY=your_secret
```

### Running

```
flutter run
```

## Project Structure

```
lib/
├── main.dart           # Entry point & dependency injection
├── app.dart            # Root MaterialApp widget
├── navigation_menu.dart # Bottom navigation controller
├── common/            # Shared UI components
│   └── styles/        # Spacing, layouts
```

└─ widgets/	# Reusable widgets (cards, shapes, icons)
└─ data/	
└─ repositories/	# Auth repository
└─ features/	
└─ authentication/	# Login, signup, onboarding, email verification
└─ maintenance_mode/	# Maintenance mode screen
└─ personailization/	# Address management
└─ shop/	# Core shop features
└─ controllers/	# Home & product controllers
└─ products/	# Product detail, reviews, attributes
└─ screens/	# Home, cart, checkout, orders, store
└─ shop_2/	# Enhanced home screen with tabs & QR
└─ splash_screen/	# Splash & routing logic
└─ utils/	
└─ constants/	# Colors, sizes, strings, env config
└─ device/	# Device utilities
└─ formatters/	# PDF & text formatters
└─ helpers/	# Models, enums, pricing calculator
└─ http/	# Payment config, Supabase helpers
└─ local_storage/	# Storage utility
└─ repository/	# Product repository & models
└─ theme/	# App theme definitions
└─ validators/	# Form validators

## Architecture

See [ARCHITECTURE.md](#) for a detailed breakdown of the app architecture, data flow, and design decisions.

## License

This project is for portfolio/demonstration purposes.