# Module 8: Shared OAuth + App Infrastructure

## ✅ Module Title: Shared OAuth + App Infrastructure (Multi-Tenant)

## 🎯 Purpose:

This module ensures that **all Facebook, Instagram, WhatsApp, Google, and Shopify/WooCommerce integrations** are powered by **a central set of platform-level apps**. This avoids each brand/customer having to create their own developer app and allows the SaaS owner (Lavishta) to control, monitor, and secure all integrations from a single point.

## 🧩 What It Enables:

* One-time authentication and permissions via platform-managed apps.
* No need for individual customers to go through Facebook App Review.
* Faster onboarding via click-to-connect.
* Central audit logs, token refresh system, and rate limit handling.
* OAuth token management per brand, scoped to their data only.

## 🔐 Platforms Supported:

* **Facebook/Instagram (Meta Graph API + Webhooks)**
* **WhatsApp Business Cloud API**
* **Google (Gmail, Google Trends, Google Analytics, Search Console)**
* **WooCommerce REST API**
* **Shopify Admin API**

## 🏗️ System Architecture:

* Platform-level App (e.g., 1 Facebook App, 1 Google Project)
* OAuth2 Flow per brand (stored in brand\_tokens table)
* Access Token + Refresh Token securely encrypted
* Webhooks (e.g., Meta Post/Comment Events) routed to tenant-aware handler
* Internal Auth Proxy to wrap calls and tag them with brand context

## 🧰 Database Tables:

* brands: holds info like brand\_id, domain, status
* brand\_integrations: brand\_id, platform, client\_id, token, refresh\_token, scopes, connected\_on, expires\_at
* webhook\_logs: brand ID, event type, payload, processed time

## 🧪 Example Integration Flow (Facebook):

1. User clicks **“Connect Facebook Page”** in their brand panel
2. Redirects to Meta OAuth URL with platform app ID
3. After user accepts permissions, Facebook redirects to /oauth/facebook/callback
4. SaaS backend exchanges code for access\_token
5. Store token in brand\_integrations with scope = pages\_messaging, pages\_read\_engagement
6. Register Webhook with callback = https://platform/api/meta/webhook and brand context
7. Incoming webhooks now contain Page ID → map to brand\_id

## 🔐 Security Controls:

* AES256 encryption for tokens
* Scoped access per brand
* Tokens auto-refreshed via CRON every 12h
* Superadmin can manually revoke or re-auth
* All API calls to Meta/Google/Woo/etc are wrapped by a brand-aware proxy with rate limiters

## 🧑‍💼 Superadmin Controls:

* Enable/Disable any integration per brand
* View logs of connection attempts, token expiry, webhook failures
* Re-auth token manually
* Limit API usage per brand to avoid abuse

## 🚦 Error Handling:

* Token expired → auto-refresh or show “Re-auth Required” warning
* Invalid scope → show in brand panel with fix instruction
* Connection failure → log and alert superadmin

## 🧠 AI-Powered Onboarding:

* Setup Wizard (per brand): Auto guides users to connect Meta, WhatsApp, Gmail, Woo/Shopify
* Tips: Explains what each permission does and how it will be used
* Flags missing scopes (e.g., user skips “Instagram Access”)

## 🌐 Admin API Endpoints:

* POST /oauth/:platform/start
* GET /oauth/:platform/callback
* GET /integrations/:brand\_id
* POST /integrations/revoke
* POST /integrations/refresh

## ⚙️ Tech Stack:

* **Backend**: Node.js (Express) or Python (FastAPI)
* **DB**: PostgreSQL
* **Encryption**: AES256 + Vault secrets
* **Queue**: Redis (for refresh retries)
* **Frontend**: React-based Panel with real-time status widgets

## ✅ Result:

This module makes it extremely easy for brands to connect 3rd-party platforms securely with 1-click. The SaaS superadmin has complete control, audit visibility, and modular enable/disable capabilities.

✅ Fully isolated per brand ✅ Secure token management ✅ Platform apps managed once centrally ✅ Reduces onboarding time to <1 min

**Next Module: Inbox Reply Module (FB, IG, WhatsApp unified)**?