أنشئ لوحة تحكم (Admin Dashboard) كاملة وشاملة لمشروع **"المُنفذ"** وهو عبارة عن منصة خدمات عمل حر تجمع بين خدمات منزلية مثل السباكة والكهرباء والتنظيف، وخدمات طلبات توصيل من مطاعم ومحلات.

هتحتاج الfiles دي في معرفة البيانات وال apis اللي هتستخدمها interfaces/index.ts lib/apis.ts

## 📦 الموديلات الموجودة:

* User: المستخدمين سواء عاديين أو مقدمين خدمات (Workers) أو سواقين (DeliveryDrivers).
* Category: تصنيفات الخدمات.
* Service: الخدمات المتاحة.
* ServiceParameter: تفاصيل إضافية للخدمة (زي الحجم، الوقت، الخ).
* Worker: بيانات مقدم الخدمة.
* WorkExperience: خبرات العامل.
* Review: التقييمات.
* DeliveryDriver: بيانات السواقين.
* Order: الطلبات.
* Wallet: المحفظة المالية لكل مستخدم.
* Transaction: المعاملات المالية.
* Store: المحلات (لجزء المطاعم والمحلات).
* StoreLocation: مواقع المحلات.
* StoreCategory: تصنيفات المحلات.
* Product: المنتجات المعروضة داخل المحلات.
* StoreOffer: العروض على المنتجات.
* Discount: الخصومات.
* Coupon: الكوبونات.
* GiftCard: بطاقات الهدايا.
* Reward: النقاط والهدايا التحفيزية.
* UserLocation: مواقع المستخدمين.
* StoreWorkingHours: أوقات عمل المحلات.

## 🧰 التعاملات المطلوبة في الـ Dashboard:

### 👤 User Management:

* عرض كل المستخدمين.
* تصفية حسب النوع (عميل - مقدم خدمة - سائق).
* تعديل بيانات المستخدم.
* إيقاف/تفعيل الحساب.
* عرض سجل الطلبات لكل مستخدم.
* متابعة رصيد المحفظة.
* إضافة يدوي لمستخدم جديد.
* البحث الذكي عن المستخدمين.
* متابعة تقييمات المستخدم.

### 📂 Category Management:

* عرض التصنيفات.
* إضافة/تعديل/حذف تصنيف.
* رفع صورة لكل تصنيف.
* ترتيب التصنيفات.

### 🛠️ Service Management:

* عرض الخدمات داخل كل تصنيف.
* إضافة/تعديل/حذف خدمة.
* ضبط تفاصيل الخدمة (ServiceParameter).
* إسناد الخدمة لفئة مقدمين الخدمات (Workers).

### 🧑‍🔧 Worker Management:

* عرض مقدمي الخدمات.
* مراجعة خبراتهم (WorkExperience).
* تعديل بيانات مقدم الخدمة.
* إيقاف/تفعيل الحساب.
* ربط مقدم الخدمة بخدمات معينة.
* متابعة تقييماته (Reviews).

### ⭐ Review Management:

* عرض التقييمات.
* تصفية حسب الخدمة أو العامل.
* حذف التقييمات المسيئة.

### 🚚 Delivery Driver Management:

* عرض السواقين.
* متابعة تقييمهم.
* ضبط حالات السائق (متاح - مشغول - موقوف).
* تعديل بيانات السواق.

### 🛒 Order Management:

* عرض كل الطلبات.
* تصفية حسب (الحالة - العميل - العامل - السائق).
* تعديل حالة الطلب.
* إلغاء الطلب.
* عرض تفاصيل الطلب (خدمة - وقت - موقع - الدفع).

### 💰 Wallet & Transaction Management:

* متابعة أرصدة المحافظ.
* عرض سجل المعاملات.
* إضافة/خصم رصيد يدوي.
* البحث حسب (المستخدم - التاريخ - نوع المعاملة).

### 🏪 Store Management:

* عرض المحلات.
* إضافة/تعديل/حذف محل.
* ضبط الموقع (StoreLocation).
* ربط المحل بتصنيف (StoreCategory).
* متابعة أوقات العمل (StoreWorkingHours).
* رفع قائمة المنتجات (Products).
* ضبط عروض (StoreOffer).
* إدارة خصومات (Discount).
* تفعيل كوبونات (Coupon).
* إدارة بطاقات الهدايا (GiftCard).

### 🎁 Reward Management:

* ضبط أنظمة النقاط.
* إضافة مكافآت جديدة.
* متابعة استبدال النقاط.

### 📍 UserLocation Management:

* متابعة عناوين المستخدمين.
* تعديل/حذف العناوين.
* ضبط عنوان افتراضي.

## 📊 إضافات:

* **Dashboard Overview Page** فيها:
  + عدد المستخدمين، العمال، السواقين.
  + إجمالي الطلبات.
  + الطلبات المفتوحة.
  + الطلبات المكتملة.
  + الأرباح اليومية/الأسبوعية/الشهرية.
  + تقييم عام للخدمات.
* **فلترة وتصدير للبيانات (Export to Excel).**
* **صلاحيات مستخدمين الـ Dashboard (Admins Roles & Permissions).**

## 🎨 المطلوب في التصميم:

### 🎨 التصميم والـ Layout المطلوب:

#### ✅ RightSidebar:

* ثابت في الوضع الطبيعي (Desktop).
* يحتوي على **أيقونات + أسماء الروتنج**.
* في الشاشات المتوسطة (md) يظهر **الأيقونات فقط** ويختفي النص.
* في الشاشات الصغيرة جدًا (xs - أقل من 400px) يختفي الـ RightSidebar تمامًا ويظهر بداله **Drawer** جانبي.

#### ✅ Topbar:

* يحتوي على **Logo** + **اسم الموقع** في البداية.
* على اليمين صورة البروفايل (Avatar).
* عند الضغط على صورة البروفايل يظهر **منيو بيانات الحساب** (اسم المستخدم - الإيميل - زر تسجيل خروج).

## 🎨 ****التصميم والتقنيات المستخدمة:****

* **الإطار الأساسي:** Next.js 15 مع **App Router**.
* **المكتبات الأساسية:**
  + **UI & Styling:** @mui/material, @emotion/react, @emotion/styled, tailwindcss, tailwind-merge, tailwindcss-animate.
  + **التحقق من البيانات:** zod.
  + **إدارة الحالة:** zustand.
  + **التعامل مع النماذج:** react-hook-form, @hookform/resolvers.
  + **إدارة الطلبات الشبكية:** @tanstack/react-query, axios.
  + **الأمان والتوثيق:** next-auth, jose, jsonwebtoken, js-cookie.
  + **الأيقونات:** @tabler/icons-react, lucide-react.
  + **التنقل والـ SEO:** react-helmet, next-intl.
  + **التعامل مع الخرائط:** @react-google-maps/api.
  + **الإشعارات والتفاعلات:** react-toastify, framer-motion.
* **🎨 الألوان الرسمية للوحة التحكم (Tailwind Theme Colors):** primary: "#0068FF", "primary-light": "#5A9CFF", "primary-dark": "#0041B3",

## 📡 API تعاملات:

* كل صفحة لازم تكون مربوطة بـ API كامل (GET, POST, PUT, DELETE).
* دعم Pagination وSearch في كل الصفحات.
* التعامل مع الرسائل التنبيهية (Notifications) في حالة الفشل أو النجاح.

## 🎥 أنيميشن خفيف:

* Transitions بسيطة أثناء التنقل.
* Loading Indicators عند استدعاء الـ API.

### 📣 التعامل مع الإشعارات:

* استخدام **socket.io** لتلقي إشعارات الـ Real-time من السيرفر.
* دعم إشعارات الجهاز (Browser Notification API) مع السماح للمستخدم بقبول/رفض الإشعارات.
* عند وصول إشعار جديد يتم:
  1. إظهار **toast** باستخدام react-toastify.
  2. إذا قبل المستخدم إشعارات الجهاز يتم إظهار **notification** من النظام نفسه.

## ****إضافات أساسية في اللوحة الرئيسية (Dashboard Overview Page):****

* عدد المستخدمين، مقدمي الخدمات، السائقين.
* إجمالي الطلبات، الطلبات المفتوحة، الطلبات المكتملة.
* الأرباح اليومية/الأسبوعية/الشهرية.
* تقييم عام للخدمات.

**✍️ مميزات أخرى:**

* **فلترة وتصفيه متقدمة** للبيانات.
* **إمكانية التصدير لملفات Excel**.
* **إدارة صلاحيات المشرفين (Admin Roles & Permissions)**.
* **أنيميشن خفيف باستخدام framer-motion**.
* **تحميل البيانات مع مؤشرات Loading عند استدعاء الـ API**