

دليل المطورو (العربية)

1) لمحة عامة

تطبيق DeliveryApp هو تطبيق أحادي الطبقات (Layered Monolith) مبني على .NET 9 وإطار Entity Framework Core لقاعدة بيانات SQL Server. يوفر واجهات ويب OpenIddict (MVC/Razor) ومضيّف HTTP API، إضافة إلى مشاريع Blazor. يعتمد على Stripe للسداد وإرسال رسائل البريد عبر SendPulse والإشعارات الفورية عبر Firebase.

التقنيات الأساسية:

- .NET 9 / ASP.NET Core / ABP 9.x.
- (OpenIddict)
- EF Core + SQL Server
- واجهات: MVC/Razor + Blazor
- المصادقة: OpenIddict (OIDC/OAuth2) + تفويض بالأدوار
- التكاملات: Stripe, SendPulse, Firebase
- السجلات: Serilog (ملفات + وحدة التحكم)

2) بنية الحل

- الكيانات، الثوابت: DeliveryApp.Domain.Shared, DeliveryApp.Domain
- DeliveryApp.Application.Contracts, DeliveryApp.Application
- خدمات التطبيق وواجهاتها وDTOs
- EF Core: سياق قاعدة البيانات وتعيينات DeliveryApp.EntityFrameworkCore
- تطبيق MVC/Razor: DeliveryApp.Web
- REST: DeliveryApp.HttpApi + DeliveryApp.HttpApi.Host
- والمضيّف
- * .Blazor: DeliveryApp.Blazor
- الاختبارات: ضمن /test

3) نموذج المجال (مختصر)

الجذور المجمّعة (Aggregate Roots): Restaurant, MealCategory, Order, SpecialOffer, Advertisement, AdRequest, SystemSetting, ChatSession, ChatMessage وغيرها.

بيانات المدفوعات: PaymentTransaction, StripeCustomer, ConnectedAccount، بالإضافة إلى CODTransaction، FinancialTransaction، RestaurantPayout عند التسليم.

4) خدمات التطبيق

موجودة ضمن src/DeliveryApp.Application/Services مثل: RestaurantAppService, OrderAppService, MealCategoryAppService, StripePaymentService, CODService, AdminUserAppService،

DeliveryPerformanceService, FirebaseNotificationService,
.SendPulseEmailNotifier

5) الواجهات والتحكم بالوصول

- مسّيرات API ضمن DeliveryApp.HttpApi.Host و DeliveryApp.HttpApi
- مسّيرات الويب ضمن DeliveryApp.Web
- التفويض عبر الأدوار: admin, manager, restaurant_owner, delivery, customer . التفويض عبر الأدوار: [AllowAnonymous] مع استخدام customer حسب الحاجة.

6) الإعدادات والأسرار

- ملفات appsettings.*.json لـ كل مشروع (بيانات مختلفة)
- ملف env.example نموذجي: .env
- يجب عدم تخزين أسرار حقيقة داخل المستودع؛ استخدم متغيرات البيئة أو مدير الأسرار.

7) البناء والتغليف (محلي)

المطلبات: .NET 9, Node 20.

- تثبيت مكتبات الواجهة (عند الحاجة):

```
abp install-libs
```

- إنشاء/تحديث قاعدة البيانات:

```
cd src/DeliveryApp.DbMigrator  
(dotnet run)
```

- تشغيل تطبيق الويب:

```
cd src/DeliveryApp.Web  
(dotnet run)
```

- تشغيل مضيف الـ API:

```
cd src/DeliveryApp.HttpApi.Host  
(dotnet run)
```

8) الأمان

- OIDC لإصدار الرموز وعملاء OpenIddict
- شهادات توقيع/تشغیر في الإنتاج
- تفويض على مستوى مسارات المسّيرات عبر [Authorize(Roles = "...")]

(9) التكاملات

- معالجة الدفع والنوايا والرسائل والتوصيات وحسابات متصلة Stripe.
- إرسال البريد (باستخدام SendPulse : SendPulse.ClientId / ClientSecret)
- إرسال الإشعارات الفورية عبر Firebase : FirebaseNotificationService

(10) النشر

راجع ملفات: DEPLOYMENT_GUIDE.md, SMARTERASP_DEPLOYMENT_GUIDE.md, NETWORK_ACCESS_SETUP.md

(11) ملاحظات مهمة

- قم بتدوير جميع الأسرار الموجودة في appsettings.*.json ونقلها إلى متغيرات البيئة.
- فَعِّل HTTPS, CORS مناسب، وسجلات المراجعة.

لرؤية مخططات التدفق والمواصفات التفصيلية، راجع المستند: -
system-.specification.ar.md