

خطة برمجة منصة MOLAK v2.0

ثنائي اللغة (AR/EN)

بدون MVC

PHP 8 / MySQL 8

خطة تنفيذ عملية من الفكرة إلى إطلاق ال MVP مع مراحل واضحة ومهام برمجية منظمة.

1. الهدف ونطاق خطة البرمجة

هذه الخطة تغطي التنفيذ البرمجي لمنصة MOLAK v2.0 وفق المعمارية المتفق عليها: **PHP 8 + MySQL 8 + (HTML/CSS/JS + Bootstrap + DataTables + Chart.js)** مع هيكل بسيط يعتمد على **كلاسات + صفحات** بدون أي إطار عمل (No MVC).

الخطة مقسمة إلى مراحل (Phases)، وكل مرحلة تحتوي على:

- **المهام الرئيسية (Tasks)** التي يجب تنفيذها.
- **المخرجات (Deliverables)** في شكل ملفات PHP / كلاسات / واجهات.
- **الاعتماديات** بين المهام لتفادي إعادة العمل.

2. الهيكل التقني العام (Technical Structure)

2.1 هيكل المجلدات

- **config/** – إعدادات الاتصال وقيم ثابتة (db.php, config.php).
- **includes/** – دوال مشتركة (auth.php, security.php, helpers.php, layout.php).
- **core/classes/** – كلاسات الدومين (User, Owner, Contract, Payment, ...).
- **lang/** – ملفات الترجمة (ar.php / en.php + t).
- **admin /owner /tenant /service-provider /marketplace/** – صفحات الواجهات لكل دور.
- **ajax/** – نقاط AJAX خفيفة (جلب بيانات، تحديث حالة).
- **api/** – نقاط REST إن لزم (Webhook, Payment Callback).

- **cron/** – مهام مجدولة (فواتير، تذكيرات، تقارير يومية).
- **templates/** – قوالب HTML/PDF (عقود، إيصالات، فواتير).
- **uploads/** – مرفقات محمية (IDs, عقود, صور وحدات).
- **logs/** – سجلات النظام والأخطاء.

2.2 طبقات التطبيق

- **Presentation Layer** – صفحات PHP في مجلدات (owner, tenant, ...).
- **Domain Layer** – كلاسات في core/classes/ (منطق الأعمال + SQL).
- **Infrastructure Layer** – الاتصال بقاعدة البيانات + التكاملات (Payment, Mail, SMS).
- **Support Layer** – الترجمة، الأمن، التنسيق العامة.

3. المراحل الرئيسية للبرمجة (Phases)

Phase 1 – البنية الأساسية (Infrastructure & Core Setup)

- إنشاء هيكل المجلدات الكامل وتهيئة ملف config.php و db.php.
- برمجة Database.php و BaseModel.php في core/classes/.
- إنشاء ملفات (CSRF/XSS) security.php و auth.php (دخول/جلسات/صلاحيات).
- إعداد نظام الترجمة ()t مع ar.php و en.php.
- إنشاء layout.php لتجميع الهيدر – الفوتر – القائمة لكل دور.

المخرجات: مشروع جاهز لاستقبال أي صفحة جديدة مع اتصال قاعدة بيانات آمن.

Phase 2 – طبقة الدومين (Domain Classes)

الكلاسات الأساسية التي يجب بناؤها:

MaintenanceRequest

Subscription

Invoice

Payment

Installment

Contract

Unit

Building

ServiceProvider

Tenant

Owner

User

AuditLog

Notification

Listing / Inquiry

RFQ / Quote / WorkOrder

- تصميم دوال CRUD أساسية لكل كلاس (create, update, delete, find, list).
- استخدام Prepared Statements في كل استعلام لتجنبّ SQL Injection.
- توحيد شكل الأخطاء (throw Exception أو return false مع رسالة واضحة).

المخرجات: طبقة منطق أعمال جاهزة لإعادة الاستخدام في أي صفحة.

Phase 3 – نظام الدخول والأمان (Auth + 2FA + Roles)

- برمجة صفحات login.php و logout.php و 2fa_verify.php.
- ربط تسجيل الدخول بجدول **users** مع **bcrypt** لكلمات المرور.
- تفعيل 2FA باستخدام **twofa_tokens** (Email أو SMS) بحسب الإعداد.
- إضافة middlewares:
 - require_login()
 - require_role(['owner', 'platform_admin', ...])
- تسجيل المحاولات في **login_attempts** لتطبيق Rate Limiting لاحقًا.

المخرجات: نظام دخول وصلاحيات آمن يميّز بين: Admin, Owner, Tenant, Provider.

Phase 4 – لوحة المالك (Owner Panel) – نسخة MVP

Dashboard 4.1

- إظهار إحصائيات: عدد المباني، الوحدات، العقود النشطة، الأقساط المتأخرة، طلبات الصيانة المفتوحة.
- رسم مخطط (Chart.js) لإيرادات الإيجار حسب الأشهر.
- جدول صغير لأقرب 5 أقساط مستحقة.

4.2 المباني والوحدات

- صفحات buildings_list.php و building_form.php.

- صفحات unit_form.php و units_list.php.
- استخدام DataTables مع فلترة بالحالة (vacant / occupied / vacant_soon).

4.3 المستأجرون و KYC

- قائمة المستأجرين المرتبطين بالمالك الحالي (lookup بـ owner_id).
- تغيير حالة (KYC (pending / approved / rejected).
- ربط الوثائق بجدول **attachments**.

4.4 العقود والأقساط والودائع

- إنشاء وتعديل العقود مع ربط (owner + unit + tenant).
- استدعاء **Contract->generateInstallments()** لإنشاء أقساط تلقائية.
- عرض حالة الوديعة وتواريخ الدفع/الاسترجاع.
- إمكانية تسجيل دفعة Offline للقسط (installments + payments تحديث).

4.5 الصيانة (RFQ → عروض → Work Orders)

- عرض طلبات الصيانة الخاصة بوحدة المالك (maintenance_requests).
- إنشاء RFQ جديد ودعوة مزود خدمة (rfqs + rfq_invites).
- استقبال عروض (quotes) واختيار عرض واحد لتحويله إلى Work Order.
- بعد التنفيذ، إنشاء دفعة Work Order مع احتساب عمولة 30% لصالح المالك/المنصة.

4.6 الاشتراكات وإعدادات المالك

- عرض الباقة الحالية من **subscriptions** و **subscription_plans**.
- عرض عدد الوحدات المستخدمة مقابل الحد المسموح + أي Overage.
- تفعيل/إلغاء **online installments** (قبول خصم 3.1%).
- إدارة إعدادات اللغة والإشعارات من **owner_settings**.

- Dashboard بسيط يعرض أهم بيانات العقد الحالي.
- صفحات:
 - tenant/contracts.php – قائمة العقود.
 - tenant/contract_view.php – تفاصيل العقد + تحميل النسخة PDF.
 - tenant/installments.php – الأقساط مع زر "Pay Online".
 - tenant/maintenance_requests.php + maintenance_request_form.php.
 - tenant/payments_history.php – كل المدفوعات.
- رفع تقييم بعد إغلاق طلب الصيانة (إضافة سجل في **reviews**).

Phase 6 – لوحة مزود الخدمة (Service Provider Panel)

- Dashboard مع ملخص RFQs المفتوحة و Work Orders الحالية.
- قائمة RFQs المدعو إليها من جدول **rfq_invites**.
- صفحة لتقديم عروض الأسعار (quotes) على RFQ معين.
- قائمة أوامر العمل (work_order_view + work_orders_list).
- عرض محفوظة مزود الخدمة من **provider_payouts** وحالة الصرف.

Phase 7 – لوحة إدارة المنصة (Platform Admin)

- Dashboard إجمالي:
- عدد الملاك/المستأجرين/المزودين.
- قيمة المدفوعات الإجمالية.
- دخل المنصة من رسوم 3.1% + أي عمولات أخرى.
- إدارة الملاك ومزودي الخدمة (اعتماد/تعليق حساب).

- إدارة الباقات (CRUD على subscription_plans).
- تقارير مالية تشغيلية (تجميع من payments + invoices).

Phase 8 – التكاملات والمهام المجدولة (Integrations & Cron)

- كلاس **PaymentGateway** + صفحات Webhook (استقبال رد بوابة الدفع).
- كلاس (SMTP) **MailService** لإرسال رسائل العقود/الفواتير.
- كلاس **SmsService** (اختياري) للإشعارات الحرجة.
- مهام Cron:
 - cron/daily_billing.php/ – إنشاء فواتير الاشتراكات الدورية.
 - cron/overdue_installments.php/ – توسيم الأقساط المتأخرة وإرسال تذكيرات.
 - cron/contracts_expiry.php/ – تنبيه قبل انتهاء العقود.
 - cron/daily_summary.php/ – ملخص يومي للمنصة.

Phase 9 – الترجمة، الـ UI النهائي، وتقوية الأمان

- تمرير على كل الصفحات واستبدال النصوص الثابتة بـ t('key').
- ضبط RTL/ LTR تلقائيًا (dir="rtl" للعربية، "ltr" للإنجليزية).
- تفعيل CSRF Token في جميع POST Forms.
- مراجعة جميع الاستعلامات والتأكد من استخدام Prepared Statements.
- توحيد تصميم الواجهة استخدام Bootstrap 5 + DataTables + Chart.js.

4. الجدول الزمني المقترح (Timeline & Milestones)

الأسبوع	المراحل	المخرجات الأساسية
الأسبوع 1-2	Phase 1 + Phase 2 + بداية Phase 3	إعداد الهيكل، الاتصال بقاعدة البيانات، الكلاسات الأساسية (User/Owner/Building/Contract)، نظام الدخول الأولي.
الأسبوع 3-4	Phase 4 (Owner Panel – MVP)	لوحة المالك كاملة: المباني، الوحدات، المستأجرون، العقود، الأقساط، الصيانة الأساسية، الاشتراكات الأساسية.
الأسبوع 5	Phase 5 + Phase 6	لوحة المستأجر + لوحة مزود الخدمة (إدارة RFQs و Work Orders + التقييمات).
الأسبوع 6	Phase 7 (Platform Admin)	لوحة إدارة المنصة، إدارة الباقات، مراقبة السوق، تقارير إجمالية مبسطة.
الأسبوع 7	Phase 8 + Phase 9	تكامل بوابة الدفع + SMTP + SMS (اختياري) + مهام Cron + إكمال الترجمة وال UI والأمان.
الأسبوع 8	Stabilization + UAT	اختبارات وظيفية وأمنية وأداء، إصلاح العيوب، تجربة مع مالك واحد (Pilot)، قرار Go/No-Go لإطلاق MVP.

Milestone رئيسية:

- M1 – نهاية الأسبوع 2: البنية الأساسية + الكلاسات جاهزة + تسجيل الدخول يعمل.
- M2 – نهاية الأسبوع 4: لوحة المالك صالحة للتشغيل الفعلي (Real Owner).
- M3 – نهاية الأسبوع 6: المنظومة كاملة لكل الأدوار (Owner/Tenant/Provider/Admin).
- M4 – نهاية الأسبوع 8: جاهزية لإطلاق MVP مع تكامل الدفع والمهام المجدولة.

5. معايير Go / No-Go لإطلاق MVP

5.1 شروط Go (الإطلاق)

- يمكن لمالك واحد على الأقل إدارة:
 - المباني + الوحدات + المستأجرين.
 - إنشاء عقود مع أقساط تلقائية.
 - قبول دفعات أقساط أونلاين مع تطبيق خصم 3.1%.
 - إدارة طلبات صيانة كاملة حتى صرف مستحقات مزود الخدمة (30%).

5.2 أسباب No-Go (التأجيل)

- تعطل في دورة الدفع (Installment أو WorkOrder) أو عدم تطابق الأرقام (gross/net/fees).
- ثغرات تؤدي لتجاوز الصلاحيات (Tenant يرى بيانات مالك آخر... إلخ).
- عدم استقرار كبير في الأداء (صفحات لا تفتح، Timeouts متكررة مع بيانات تجريبية كبيرة).
- عدم اكتمال مسار أساسي من المسارات المتفق عليها للم MVP.

- لوحات المستأجر والمزود تعمل في السيناريوهات الأساسية (عرض عقد، دفع، طلب صيانة، RFQ/Quote/WO).
- لوحة إدارة المنصة تعرض إحصائيات عامة ولا يوجد أخطاء حرجية.
- اختبارات الأمان الأساسية ناجحة (SQLi/XSS/CSRF/Rate Limiting تسجيل الدخول).