

CMS Üzerinden İş Kuralları & Low-Code Konfig Planlama Şablonu — Yazılım / Low-Code (v1.0)

Asset Amaç: Bu şablon; iş kurallarını “panel ayarı mı, kod mu?” karar matrisiyle net bir şekilde sınıflandırarak low-code esneklik sınırlarını belirlemenizi sağlar. RBAC (Rol Bazlı Erişim Kontrolü), loglama ve versiyonlama standartlarıyla panelden yönetilen ayarların güvenli, izlenebilir ve gerektiğinde geri alınabilir olmasını hedefler; otel ve B2B senaryolarına özel örnek konfigürasyon setleri sunar.

Kim Kullanır?: Ürün Sahipleri (Product Owner), Pazarlama Liderleri, Teknik Liderler (Tech Lead), Bilgi Güvenliği Sorumluları ve Ajans Proje Yöneticileri (PM).

Nasıl Kullanılır?

1. CMS üzerinde yönetilmesi istenen tüm iş kurallarının envanterini çıkarın ve her birine risk/karmaşıklık puanı verin.
2. Panelde kalması uygun görülen konfigürasyon alanlarını veri tipleriyle tanımlayıp gerekli validasyon (doğrulama) kurallarını ekleyin.
3. RBAC yetkilendirmesi, audit log (denetim kaydı) ve versiyon kontrolü ile entegre çalışan yayınlama ile geri alma (rollback) prosedürünü yazın.

TEMPLATE — CMS İş Kuralları & Low-Code Yönetim Matrisi

1) İş Kuralı Envanteri

CMS paneline taşınması düşünülen iş kuralının yapısal özellikleri:

- **Kural Adı:** _____
- **Kuralın Amacı:** _____
- **Değişim Sıklığı:** ☐ Sık (Günlük/Haftalık) | ☐ Orta (Aylık/Dönemsel) | ☐ Nadir (Yıllık)
- **Teknik / Operasyonel Risk Seviyesi:** ☐ Düşük | ☐ Orta | ☐ Yüksek
- **Finansal Etkisi Var mı? (Fiyat/Kupon/İndirim vb.):** ☐ Evet | ☐ Hayır
- **KVKK / Kişisel Veri Teması Mevcut Mu?:** ☐ Evet | ☐ Hayır

2) Karar Matrisi: Panel mi, Kod mu?

Kuralın karmaşıklık ve risk analizine göre teknik olarak konumlandırılacağı yer:

- ☐ **Panel (Tam Konfigürasyon):** Kural tamamen CMS arayüzündeki parametrelerle (açma/kapama, metin değiştirme vb.) yönetilir. Yazılımcı müdahalesi gerektirmez.
- ☐ **Kod (Hardcoded Mantık):** Yüksek riskli, finansal hesaplama içeren veya core mimariyi etkileyen kurallar doğrudan kaynak kodda geliştirilir ve CI/CD süreçlerinden geçer.
- ☐ **Hibrit Yapı (Panel Parametresi + Kod Mantığı):** Mantıksal iskelet kodda saklanır; ancak limitler, tarihler veya hedef segmentler gibi değişkenler panelden dinamik olarak beslenir.

3) Panel Konfigürasyon Alanları & Veri Tipleri

Pazarlama ve operasyon ekiplerinin yönetim paneli üzerinden değiştirebileceği izinli veri alanları:

- **toggle_is_active** (Boolean - True/False): Kuralın veya kampanya modülünün anlık olarak aktif/pasif edilmesi.
- **start_date / end_date** (DateTime): Kuralın geçerli olacağı zaman aralığının (başlangıç ve bitiş saati dahil) planlanması.
- **message_title / message_body** (String/Text): Kullanıcıya gösterilecek dinamik uyarı, teklif veya bilgilendirme metinleri.
- **cta_text / cta_url** (String/URL): Eylem butonunun metni ve yönleneceği güvenli URL adresi.
- **target_segments** (Array/Select): (Varsayım: Kişiselleştirilmiş deneyim için) Kuralın hangi kullanıcı segmentlerine (Örn: Sadık Müşteri, B2B Lead vb.) uygulanacağı seçimi.
- **limit_count** (Integer): Kampanya veya kural bazında uygulanacak maksimum kullanım veya gösterim sınırı sayısı.

4) Governance (Yönetişim & Güvenlik Standartları)

Yetkisiz değişiklikleri engellemek ve sistem kararlılığını korumak için zorunlu güvenlik katmanları:

- **RBAC Yetki Seviyesi (Kim Değiştirebilir?):** _____ (Örn: Sadece Growth_Lead veya Admin)
- **İş Onay Akışı (Workflow):** ☐ Onay Gerekmez | ☐ Tekli Onay (Yönetici Onayı) | ☐ Çift Onay (IT + İş Birimi Ortak Onayı)
- **Audit Log (Denetim İzleri):** Sistemi kimin, hangi IP'den, ne zaman değiştirdiği ve veri alanının **eski_deger** $\rightarrow$ **yeni_deger** fark analizi (diff) zorunlu olarak loglanır.
- **Versiyon Kontrolü (Rollback):** Panelden yapılan her kayıt yeni bir sürüm numarası alır. Hatalı bir konfigürasyonda tek tıkla eski sürüme geri dönebilir (**rollback candidate**).

5) Test, Validasyon & Canlıya Alım (Publish Gate)

Hatalı veri girişlerinin canlı sistemi kilitlemesini önleyecek son filtreler:

- **Staging Test Zorunluluğu:** ☐ Evet (Değişikliklerin önce test ortamında simüle edilmesi ve doğrulanması zorunludur)
- **Publish Gate (Yayına Alma Bariyeri):** ☐ Otomatik Validasyon (Regex/Tip Kontrolü) | ☐ Manuel Checklist Onayı
- **Kritik Alert Mekanizması:** Finansal veya yüksek riskli alanlarda yapılan değişiklikler anında Slack/E-posta kanalları üzerinden IT ve Güvenlik ekiplerine otomatik uyarı (alert) olarak düşer.

6) Nasıl Uygulanır? (5 Altın Governance Kuralı)

1. **Finansal Etkisi Olan Her Kurala Çift Onay Mekanizması Kurun:** CMS paneli üzerinden tanımlanan indirim oranları, otel oda fiyat çarpanları veya B2B kupon kodları gibi doğrudan ciroya etki eden konfigürasyonlarda "Yapan" ve "Onaylayan" rollerini kesin olarak ayırın.
2. **Panel Giriş Alanlarına Katı Veri Tipi Doğrulaması (Validation) Ekleyin:** Kullanıcının sayı girmesi gereken bir limite (Örn: **limit_count**) metin girmesini veya

URL alanına geçersiz formatta link yazmasını engellemek için backend seviyesinde tip kontrollerini ve sınır değer limitlerini (min/max) zorunlu kılın.

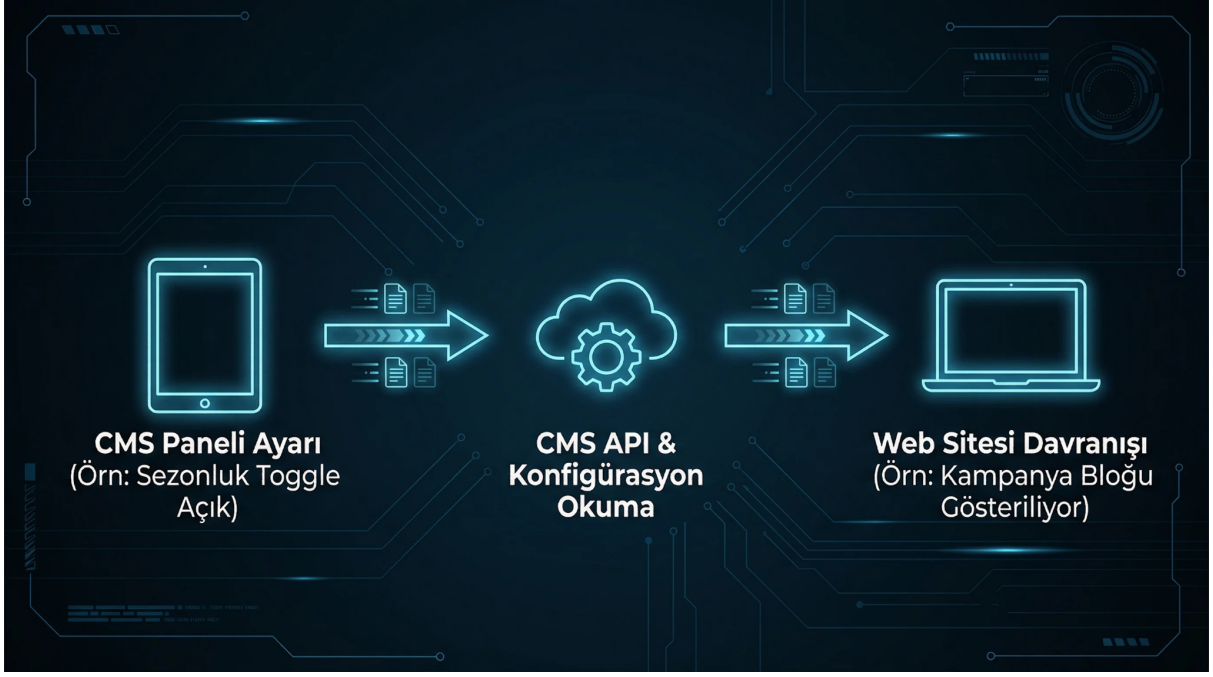
3. **Her Değişikliği Geri Alınabilir (Rollback) Kurgulayın:** Panel mimarisini tasarlarken mevcut konfigürasyonun üzerine yazmak yerine, veritabanında satırları versiyonlayarak saklayın. Canlı ortamda bir sorun yaşandığında sistemin kararlı çalışan son konfigürasyon sürümüne saniyeler içinde dönebilmesini garantileyin.
4. **Kritik Değişiklikler İçin Anlık Bildirim Hattı Oluşturun:** Yüksek riskli bir iş kuralı panelden güncellendiği veya yayına alındığı anda, bu değişikliğin detaylarını (değiştiren, eski ve yeni değerler) içeren otomatik bir bilgi notunu yazılım ve güvenlik ekiplerinin ortak izleme kanalına (Örn: Slack webhook) iletin.
5. **Yılda Bir Kez "Konfig Temizliği" Yapın (365 Gün Döngüsü):** Zamanında geçici bir kampanya veya özel bir B2B senaryosu için panele eklenmiş ancak artık kullanılmayan atıl iş kurallarını ve konfigürasyon değişkenlerini tespit etmek için yılda bir kez teknik temizlik envanteri gerçekleştirin.

Deliverables (Teslim Edilecek Çıktılar)

- **Konfigürasyon Sözlüğü (v1.0):** CMS panelindeki tüm dinamik alanların isimlerini, veri tiplerini, alabilecekleri sınır değerleri ve validasyon kurallarını listeleyen teknik rehber.
- **Risk Matrisi + Sınır Dokümanı:** Hangi iş kurallarının panelden low-code olarak yönetilebileceğini, hangilerinin ise kesinlikle kod seviyesinde kalması gerektiğini belirleyen stratejik karar sınırı dokümanı.
- **RBAC / Log / Rollback Prosedürü:** Yetkilendirme matrislerini, audit log veri şemasını ve kriz anında konfigürasyonların nasıl eski sürüme geri çekileceğini açıklayan operasyonel kılavuz.
- **365 Gün Gözden Geçirme Planı:** Paneldeki low-code modüllerin güvenlik, performans ve kullanım ömürlerinin her yıl düzenli olarak denetlenmesini sağlayan sürdürülebilirlik takvimi.

[Checklist / Framework Card]

Ürün sahipleri ve yazılım mimarlarının CMS üzerindeki low-code esneklik sınırlarını ve yönetim (governance) kurallarını planlamaları için tasarlanmış, 1200x1200px kare formatında yüksek çözünürlüklü bir kontrol ve çerçeve kartı. Derin grafit gri ve endüstriyel antrasit tonlarındaki teknolojik arka plan üzerinde, canlı neon turuncu ve elektrik moru detaylarla işlenmiş panel konfigürasyon bölmeleri yer alıyor. Kartın üzerinde; "Kural Karmaşıklık Puanlaması", "RBAC Yetki Dağılımı", "Audit Log Doğrulaması" ve "Rollback Sürüm Güvencesi" gibi kritik governance adımları, yanlarında parlayan neon yeşili onay ikonlarıyla son derece temiz, scannable ve kurumsal bir checklist düzeninde sunuluyor.



[Diagram / Flow]

CMS panelinden değiştirilen bir iş kuralının veya konfigürasyon parametresinin (Örn: kampanya tarihleri veya segment limitleri), uygulamanın çalışma zamanı (runtime) davranışını nasıl dinamik olarak etkilediğini modelleyen, 1920x1080px Full HD çözünürlükte teknik bir mimari akış diyagramı. Minimalist gece siyahı bir zemin üzerinde, en soldaki "CMS Kullanıcı Arayüzü / Konfig Girişi" düğümünden çıkan mavi veri hatları, ortadaki "RBAC & Validasyon Kapısı"ndan geçerek "Veritabanı Konfigürasyon Versiyon Matrisi"ne ulaşıyor. Buradan aşağıya süzülen entegre hat, uygulamanın core kod yapısındaki iş mantığına (business logic) bağlanarak parametrelerin canlı sistem davranışını mor ve altın sarısı oklarla nasıl anlık değiştirdiğini "clarity at a glance" felsefesiyle görselleştiriyor.