

AI

AI DANIřMAN

ÜCRETSİZ SÜRÜM

Otomasyon için Yapay Zekâ

Kurmak kolay, ayakta tutmak zor. Otomasyonların çoğu teknik sebeplerden değil, hata senaryosu ve sahibi olmadığı için ölüyor.

AI Danışman – Ücretsiz Eğitim Serisi

Bu belgeyi dilediğin gibi paylaşabilirsin. Satılamaz, değiştirilemez.

Eylül 2026 · aidanisman.pages.dev · t.me/aidanisman

01 Otomasyonlar neden ölür

02 Hangi iş otomasyona uygun

03 Üç katman: tetik, iş, çıkış

04 Yapay zekâ nereye girer

05 Hata senaryosu — asıl iş

06 Bildirim ve sahiplik

07 Araç seçimi

08 İlk otomasyon: yedi gün

01 Otomasyonlar neden ölür

Otomasyon kurmak, on yıl öncesine göre kıyaslanamayacak kadar kolay. Buna rağmen kurulan otomasyonların çoğu birkaç ay içinde çalışmaz hâle geliyor. Sebep neredeyse hiç "kurulum yanlış" değil.

Ölüm sebebi	Nasıl olur	Ne kadar sürede
Sessiz durma	Bir bağlantı kopar, akış durur, kimse fark etmez	Günler — bazen aylar
Girdi değişimi	Form alanı eklenir, tablo sütunu adı değişir	İlk değişiklikte
Yetki bitmesi	Token süresi dolar, şifre değişir	3–12 ay
Sahipsizlik	Kuran kişi ayrılır, kimse nasıl çalıştığını bilmez	İlk arızada
Sessiz yanlış	Akış çalışıyor ama yanlış sonuç üretiyor	Fark edilene kadar

Çıkarım: Listenin tamamı *kurulumla* değil, **kurulumdan sonrasıyla** ilgili. Otomasyonun zor kısmı kurmak değil; bu ders o yüzden otomasyon kurmayı değil, **ayakta kalan otomasyon kurmayı** anlatıyor.

En tehlikelisi: sessiz yanlış

Duran otomasyon er geç fark ediliyor. **Çalışıp yanlış sonuç üreten** otomasyon fark edilmiyor — ve arada geçen sürede yanlış veriyi kaydediyor, yanlış mesajı gönderiyor, yanlış raporu üretiyor. Yapay zekâ içeren otomasyonlarda bu risk özellikle yüksek, çünkü model hata verip durmuyor: *inandırıcı ve yanlış* bir çıktı üretip akışı devam ettiriyor.

02 Hangi iş otomasyona uygun

Yanlış işi otomatikleştirmek, otomasyon kurmamaktan pahalı. Beş ölçüt:

Ölçüt	Soru	Neden
Tekrar	Ayda en az 10 kez oluyor mu	Altındaki işlerde bakım maliyeti kazancı yiyor
Kural	Adımları yazıya dökabiliyor musun	Yazamadığın işi otomatikleştiremezsin
Girdi kararlılığı	Girdi hep aynı biçimde mi geliyor	Değişken girdi, sürekli bakım demek
Hata maliyeti	Yanlış giderse ne olur	Yüksekse insan onayı şart
Sahip	Bozulunca kim bakacak	Cevabı yoksa kurma

KURAL: ÖNCE ELLE YAP

Bir işi **en az on kez elle yapmadan** otomatikleştirme. Elle yaparken fark ettiğin istisnalar, otomasyonun yarısını oluşturuyor. Elle yapmadan kurulan otomasyonlar, ilk istisnada kırılıyor — çünkü istisnanın varlığı bile bilinmiyordu.

İyi ve kötü adaylar

İyi aday	Kötü aday
Gelen taleplerin sınıflandırılıp yönlendirilmesi	Müşteriye denetimsiz otomatik yanıt
Belge ve formlardan yapılandırılmış veri çıkarma	Fiyat ve teklif kararı verme
Tekrar eden raporun ilk taslağının hazırlanması	Raporun onaysız gönderilmesi
Toplantı kaydından not ve madde çıkarma	Kişi hakkında değerlendirme üretme
İçerik türevlerinin hazırlanması	İçeriğin onaysız yayımlanması

Kötü adayların ortak yanı: **çıktı, insan onayı olmadan dışarı çıkıyor**. Otomasyonun sınırı burada.

03 Üç katman: tetik, iş, çıkış

Her otomasyon üç katmandan oluşuyor. Sorunlar da hangi katmanda olduğuna göre farklı çözülüyor.

Katman	Ne yapar	Tipik arıza
1. Tetik	Akışı ne başlatır: yeni form, yeni e-posta, saat, yeni satır	Tetik gelmiyor veya iki kez geliyor
2. İş	Veriyi al, dönüştür, karar ver	Girdi biçimi değişmiş, mantık kırılmış
3. Çıkış	Sonucu bir yere yaz veya gönder	Yetki bitmiş, hedef değişmiş

Bir kural: her katman ayrı denenmeli

Otomasyon bozulduğunda "çalışmıyor" demek işe yaramıyor. Üç katmanı ayrı ayrı test edebiliyorsan arıza bulmak dakikalar, edemiyorsan saatler alıyor. Kurarken katmanları birbirinden ayrı çalışabilir biçimde kur.

İKİ KEZ TETİKLENME

En sinsi arıza tipi. Aynı olay iki kez işlenirse iki kayıt, iki mesaj, iki fatura oluşuyor. Çözüm basit ve neredeyse hiç uygulanmıyor: her olayın **benzersiz bir kimliği** olsun ve akış, o kimliği daha önce işleyip işlemediğine baksın. Kurulumda beş dakika, sonrasında saatlerce hata ayıklamadan tasarruf.

04 Yapay zekâ nereye girer

Yapay zekâ otomasyonun tamamı değil, **ikinci katmanın bir parçası**. Doğru yerleştirildiğinde otomatikleştirilemeyen işleri otomatikleştirilebilir hâle getiriyor; yanlış yerleştirildiğinde sistemin en kırılgan parçası oluyor.

Girer	Girmez
Serbest metni yapılandırılmış veriye çevirme	Aritmetik ve hesaplama
Sınıflandırma ve etiketleme	Kesin kuralla çözülebilen dallanma
Özetleme ve madde çıkarma	Kayıt arama ve eşleştirme
Metin taslağı üretme	Onaysız gönderim
Duygu/aciliyet tespiti	Para, hukuk veya kişi hakkında karar

Çıkarım: "Bu adım basit bir kuralla çözülüyor mu?" diye sor. Çözülüyorsa kuralı kullan. Kural, modelden *ucuz, hızlı ve öngörülebilir*. Yapay zekâyı yalnızca kuralın yetmediği yerde kullan — genelde bu, işin belirsiz ve serbest metin içeren tek adımıdır.

Modeli otomasyonda kullanırken üç kural

- Çıktı biçimini zorla.** "JSON döndür, şu alanlar olsun, başka bir şey yazma." Serbest metin çıktısı, sonraki adımı kırıyor.
- Bilmiyorum diyebilsin.** Emin olamadığında uydurmak yerine belirsiz döndürmesini iste ve o durumu insana yönlendir. Bu tek madde, sessiz yanlışların çoğunu engelliyor.
- Çıktıyı doğrula.** Beklenen alanlar var mı, değerler beklenen kümede mi? Yoksa akışı durdur — devam ettirme.

OTOMASYON İÇİ MODEL ÇAĞRISI – İSKELET

Görev: Aşağıdaki [gelen talep metnini] sınıflandır.

Yalnızca şu JSON'u döndür, başka hiçbir şey yazma:

```
{
  "kategori": "fatura" | "teknik" | "satis" | "diger" | "belirsiz",
  "aciliyet": "yuksek" | "normal" | "dusuk",
  "gerekce": "en fazla 15 kelime"
}
```

Kurallar:

- Kategoriden emin değilsen "belirsiz" yaz. Tahmin etme.
- Metinde olmayan bilgiyi çıkarma.
- Kişi adı veya iletişim bilgisi döndürme.

Metin: [icerik]

05 Hata senaryosu — asıl iş

Bu bölüm dersin en önemli kısmı. Otomasyonun başarısını belirleyen şey, iyi gittiğinde ne yaptığı değil — kötü gittiğinde ne yaptığı.

Dört soru

#	Soru	Cevabı yoksa
1	Ne zaman durur? Hangi durumda devam etmemeli	Yanlış veriyle çalışmaya devam eder
2	Kim haber alır? Hata olunca bildirim nereye gider	Sessizce ölür
3	Ne kaydedilir? Sonradan bakınca ne olduğu anlaşılır mı	Arıza bulunamaz
4	Elle nasıl yapılır? Otomasyon çalışmazken iş nasıl yürür	İş tamamen durur

Dördüncü soru en çok atlanan. Otomasyon bir işi *hızlandırmalı*, o işin tek yapılma yolu hâline gelmemeli. Bozulduğunda işin elle yürüyebilmesi gerekiyor — ve bunu bilen birinin olması.

Üç hata sınıfı, üç farklı davranış

Hata sınıfı	Örnek	Doğru davranış
Geçici	Ağ hatası, servis kısa süre yanıt vermiyor	Bekle ve tekrar dene — artan aralıklarla, sınırlı sayıda
Veri	Beklenen alan yok, biçim bozuk	Durdur, kaydet, bildir. Tekrar deneme — aynı sonucu verir
Yetki	Token süresi dolmuş, erişim kesilmiş	Durdur ve hemen bildir — kendi kendine düzelmez

SONSUZ TEKRAR TUZAĞI

Veri hatasında tekrar denemek, aynı hatayı sonsuz üretiyor — bazı araçlarda bu ücretsiz kotayı bir gecede bitiriyor, e-posta gönderiyorsa yüzlerce mesaj atıyor. Kural: **yalnızca geçici hatalarda tekrar dene, sayıyı sınırla.**

06 Bildirim ve sahiplik

Bildirim: iki tür

Tür	Ne zaman	Nereye
Hata bildirimi	Akış durduğunda, anında	Sahibinin gerçekten baktığı yere — Telegram, e-posta değil
Yaşam sinyali	Günlük veya haftalık özet	Aynı yer

İkincisi sık atlanıyor ama kritik: **hata bildirimi de gelmiyorsa**, otomasyon hiç çalışmıyor olabilir. "Bugün 14 talep işlendi" gibi tek satırlık bir günlük özet, sessiz ölümü fark etmenin en ucuz yolu.

Bildirimde bulunması gerekenler

- Hangi otomasyon
- Ne zaman
- Hangi adımda
- Hata mesajının kendisi
- Etkilenen kayıt (varsa kimliğiyle)

"Bir hata oluřtu" diyen bildirim, bildirim deęil. Bu beř bilgi olmadan sahibin otomasyonu aıp bařtan arařtırması gerekiyor.

Sahiplik

SAHİPSİZ OTOMASYON KURULMAZ

Her otomasyonun tek bir sahibi olmalı ve sahip üç řeyi bilmeli: **ne yaptığını, nasıl durdurulacağını, bozulduğunda işin elle nasıl yürüyeceğini**. Sahibi olmayan otomasyon, kuran kişinin ayrıldığı gün ölü sayılır — alıřmaya devam etse bile, kimse dokunamadığı için ilk deęişiklikte kırılır ve kimse tamir edemez.

Her otomasyon için tek sayfa

Sahibin bıraktığı belge en az řunları içermeli: ne işe yarıyor · neyle tetikleniyor · hangi servislere baęlı · hangi hesap/yetkiyle alıřıyor · nasıl durdurulur · bozulunca iş elle nasıl yapılır. Altı satır, ve bir yıl sonra bu otomasyonu kurtaran řey bu.

07 Ara seçimi

Ara, otomasyonun en az önemli kararı — ama en ok tartıřılanı. Ü aile var:

Aile	Mantığı	Uygun olduęu durum
Görsel akıř araları Zapier, Make	Kutuları baęlarsın, altyapı onlarda	Hızlı bařlangı, az sayıda akıř, teknik bilgi az
Kendi sunucunda alıřanlar n8n gibi	Aynı mantık, ama senin ortamında	ok sayıda alıřtırma, veri kontrolü, maliyet
Kod betik + zamanlayıcı	Doęrudan yazarsın	Karmařık mantık, sürüm takibi, tam kontrol

Fiyatlandırma modelini anlamak, fiyatı bilmekten önemli

Araların ücretlendirmesi genelde **alıřtırma sayısı** veya **adım sayısı** üzerinden işliyor. Bu ayırım, sonsuz tekrar hatasında faturanın büyüklüğünü belirliyor. Ayrıca ücretsiz katmanların sınırları sık deęişiyor — karar vermeden önce sağlayıcının güncel sayfasından kontrol et.

Çıkarım: İlk otomasyonunu *elindeki* arala kur. Ara arařtırmasına harcanan hafta, kurulup alıřan tek bir otomasyondan daha az öğretiyor. Ara deęiřtirmek kolay; hangi işin otomasyona uygun olduğunu öğrenmek

zor.

08 İlk otomasyon: yedi gün

Gün	İş
1	İşi seç. Beş ölçütten geçsin. En sıkıcı ve en sık tekrarlayanı seç — heyecan verici olanı değil.
2	İşi elle yap ve her adımı yaz . İstisnaları ayrı bir listeye not et.
3	Üç katmanı belirle: tetik ne, iş ne, çıkış nereye. Kâğıda çiz.
4	Kur — ama çıkışı kendine yönlendirerek . Gerçek hedefe henüz yazma.
5	Hata senaryosunu ekle: dört sorunun cevabını kur. Bildirimi test et — bilerek bozup bildirimin geldiğini gör.
6	Gerçek çıkışa bağla. İlk hafta çıktıları elle kontrol et.
7	Tek sayfalık belgeyi yaz. Sahibini belirle (muhtemelen sen). Yaşam sinyali bildirimini kur.

Beşinci gün, çoğu kişinin atladığı ve otomasyonun ölüp ölmeyeceğini belirleyen gün. Bildirimi **bilerek bozarak test etmek**, kurulumun tamamından daha değerli — çünkü test edilmemiş bildirim, çalışmayan bildirimdir.

Buradan sonrası

Otomasyon seçim puanlaması, hata senaryosu şablonu, model çağrısı desenleri ve çıktı doğrulama, bildirim metinleri, tek sayfalık devir belgesi, izleme paneli ve on hazır otomasyon tarifi 'nda.

t.me/aidanisman • aidanisman.pages.dev

Bu belgede araç isimleri örnek olarak geçmektedir; hiçbirisiyle ticari ilişkimiz yoktur. Ücretsiz katman sınırları ve fiyatlandırma modelleri sık değişmektedir — karar vermeden önce sağlayıcının güncel sayfasını kontrol edin. Belge eğitim amaçlıdır.