



WEB & YAZILIM GELİŐTİRME SERİSİ · MODÜL 12

# Domain & Hosting

---

Siteyi dünyaya açmak: alan adı ve DNS, kayıt türleri, IP ve sunucular, hosting nedir, isteğın yolculuğı ve HTTPS; hosting türleri (paylaşımli, VPS, bulut, statik); SSL/TLS, CDN, performans, sunucu güvenliğı ve yedekleme; yayına alma, dağıtım, e-posta, izleme ve maliyetler. Özgün altyapı diyagramlarıyla.

# Bu Kitap Hakkında

Bu modül, yazdığın kodu dünyaya güvenle açmanın tüm zincirini öğretir: alan adı, DNS, hosting, SSL ve yayına alma. Önceki modüllerde uygulamalar geliştirdin; bu modül onları gerçek kullanıcılara ulaştıran altyapıyı kurar. Daha çok kavram ve operasyon odaklıdır; her halkayı (adres, ev, güvenli yol) net diyagramlarla açıklar. Dört seviye ve yirmi altı bölüm boyunca alan adının parçalarından ve kayıt kuruluşlarından, DNS çözümlemesine ve kayıt türlerine (A, CNAME, MX, TXT); IP ve sunuculardan, hosting türlerine (paylaşımlı, VPS, bulut, statik) ve seçim ölçütlerine; SSL/TLS el sıkışmasından, sertifika kurulumuna, CDN'e, performansa, sunucu güvenliğine ve yedeklemeye; bir siteyi yayına almaktan, dağıtım ortamlarına (geliştirme/staging/üretim), alan adı e-postasına, izlemeye ve maliyet yönetimine kadar uzanır.

Her bölümde konuyu görselleştiren özgün bir diyagram (DNS çözümleme akışı, DNS kayıt tabloları, hosting türleri karşılaştırması, istek yolculuğu ve dağıtım şemaları), pratik örnekler, 'pratikte ne olur?' kartı ve bir alıştırmaya yer alır. Güvenlik ve sorumlu işletim baştan sona vurgulanır: HTTPS'in zorunluluğu, sunucu güvenliği (güvenlik duvarı, güncelleme, SSH anahtarı, en az yetki), 3-2-1 yedekleme, parametrelili erişim, sırların gizlenmesi ve izlemede veri gizliliği (KVKK). Modül, FAZ 2'yi (sunucu tarafı ve altyapı) tamamlayan bir go-live kontrol listesiyle kapanır. Bu, on altı modüllük 'Web & Yazılım Geliştirme' serisinin on ikinci modülüdür. Bu seri eğitim amaçlıdır; örnek IP, kayıt ve fiyatlar temsilidir.

Web & Yazılım Geliştirme Serisi · Modül 12

# İçindekiler

## İNTERNET NASIL ÇALIŞIR?

---

- 01** Alan Adı (Domain) Nedir? 6
- 02** Alan Adı Almak ve Yönetmek 8
- 03** DNS Nedir? 10
- 04** DNS Kayıt Türleri 12
- 05** IP Adresleri ve Sunucular 14
- 06** Hosting Nedir? 16
- 07** Bir İsteğin Yolculuğu 18
- 08** HTTP ve HTTPS 20

## HOSTING TÜRLERİ VE KURULUM

---

- 09** Hosting Türleri 23
- 10** Paylaşımlı Hosting (Shared) 25
- 11** VPS ve Sunucu Yönetimi 27
- 12** Bulut Hosting (Cloud) 29
- 13** Statik Site Barındırma 31
- 14** Hosting Seçimi: Karşılaştırma 33

## GÜVENLİK VE PERFORMANS

---

- 15** SSL/TLS ve Sertifikalar 36
- 16** SSL Kurulumu ve Yönetimi 38
- 17** CDN ve İçerik Dağıtımı 40
- 18** Performans ve Önbellekleme 42
- 19** Sunucu Güvenliği 44
- 20** Yedekleme ve Felaket Kurtarma 46

## YAYINA ALMAK VE İŞLETMEK

---

- 21** Bir Siteyi Yayına Almak 49
- 22** Dağıtım Süreci (Deployment) 51
- 23** E-posta ve Alan Adı 53
- 24** İzleme ve Kesinti Yönetimi 55
- 25** Maliyetler ve Ölçeklenme 57
- 26** Bitirme: Yayına Alma Kontrol Listesi 59

★ Domain & Hosting Terimleri Sözlüğü 61

## SEVİYE 1

# İnternet Nasıl Çalışır?

Temeller: alan adı nedir, alan adı almak, DNS, kayıt türleri, IP ve sunucular, hosting nedir, bir isteğin yolculuğu, HTTP ve HTTPS.

**BÖLÜM 01**

# Alan Adı (Domain) Nedir?

Alan adı, internetteki adresinizdir: insanların sitenize ulaşmak için yazdığı isim (örneğin `ornek.com`). Arkada bir sayı (IP adresi) vardır ama kimse onu ezberlemez; alan adı, akılda kalıcı bir takma addır.

## Bir alan adının parçaları

- **Uzantı (TLD):** `.com` , `.org` , `.com.tr` — en sağdaki kısım.
- **Ana ad (SLD):** `ornek` — seçtiğin isim.
- **Alt alan adı:** `www` , `blog` — en soldaki kısım.



Şema 1.1 — Bir alan adının parçaları: alt alan adı . ana ad . uzantı.

### İPUCU

Alan adını, evinin **açık adresi** gibi düşün: kimse "şu enlem-boylamdaki ev" demez; "Atatürk Caddesi No:5" der. İnternette de gerçek adres bir IP'dir (örn. 93.184.x.x) ama insanlar `ornek.com` yazar. Uzantı seçimi anlam taşır: ticari için `.com` , Türkiye için `.com.tr` , dernek için `.org` yaygındır. Alan adı **benzersizdir** — bir kez birileri aldıysa, o ad onlarındır (kiralık olarak). Kısa, akılda kalıcı ve yazımı kolay bir ad seçmek, markan için değerlidir.

**Pratikte ne olur?****SONUÇ**

Birisi tarayıcıya `ornek.com` yazdığıında, bu isim arka planda bir IP adresine çevrilir ve o adresteki sunucuya ulaşılır. Kullanıcı sayıları hiç görmez; yalnızca akılda kalıcı adı yazar ve siten açılır. Alan adı, markanın internetteki kapısıdır.

**Alıştırma**

8 dk

Alan adını çöz:

- 1 Sevdiğin bir sitenin alan adını parçalarına ayır (alt ad / ana ad / uzantı).
- 2 Bir proje için üç olası alan adı ve uygun uzantı öner.
- 3 Neden insanlar IP yerine alan adı kullanır, kısaca açıkla.

**BÖLÜM 02**

# Alan Adı Almak ve Yönetmek

Alan adları, "kayıt kuruluşu" (registrar) denen şirketlerden yıllık olarak kiralanır. Aslında satın almazsın; belirli bir süre için kullanım hakkını alırsın, süre dolmadan yenilersin. Yenilemezsen ad serbest kalır.

## Kayıt süreci



Şema 2.1 — Alan adı yaşam döngüsü: ara → kaydet → yönet → yenile.

- **Registrar:** alan adı satan/yöneten yetkili şirket.
- **Yenileme:** genelde yıllık; otomatik yenileme açmak güvenlidir.
- **WHOIS gizliliği:** kişisel bilgilerini gizleme seçeneği.

### İPUCU

Birkaç pratik kural: **otomatik yenilemeyi aç** — unutup süresi dolan bir alan adını başkası kapabilir ve geri almak çok pahalı (hatta imkânsız) olabilir. Alan adının **kime ait olduğuna** dikkat et: mutlaka kendi hesabına/şirketine kayıtlı olsun, bir ajansın veya çalışanın hesabına değil — yoksa ileride kontrolü kaybedebilirsin. **WHOIS gizliliği** ile kişisel iletişim bilgilerini herkese açık olmaktan korursun. Alan adını bir firmadan, hosting'i başka bir firmadan almak tamamen mümkündür; bunlar ayrı hizmetlerdir.

**Pratikte ne olur?**

SONUÇ

Bir alan adını kaydettiğinde, o ad belirli bir süre (genelde 1 yıl) senin kontrolündedir; registrar'ın panelinden DNS ayarlarını ve iletişim bilgilerini yönetirsin. Süre dolmadan yenilersen ad sende kalır; yenilemezsen serbest kalır ve başkası alabilir. Yani alan adı bir mülk değil, yenilenen bir kira gibidir.

**Alıştırma**

10 dk

Yönetimi planla:

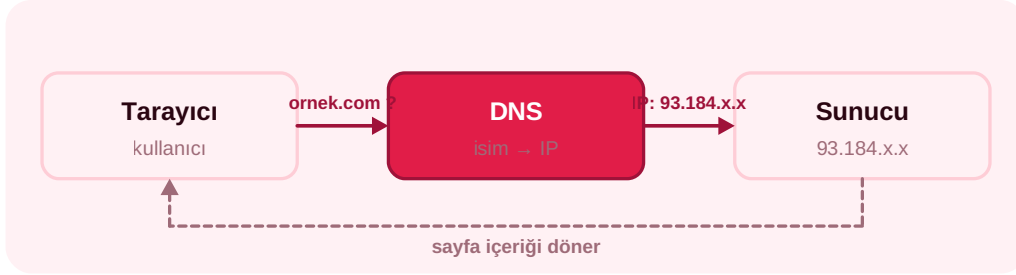
- 1 Otomatik yenilemenin neden önemli olduğunu kendi cümlele yaz.
- 2 Alan adının kimin hesabına kayıtlı olması gerektiğini açıkla.
- 3 WHOIS gizliliğinin ne işe yaradığını belirt.

## BÖLÜM 03

# DNS Nedir?

DNS (Alan Adı Sistemi), internetin telefon rehberidir: insanların yazdığı alan adlarını, makinelerin anladığı IP adreslerine çevirir. Sen `ornek.com` yazarsın, DNS onu `93.184.x.x` gibi bir adrese dönüştürür ve tarayıcı o sunucuya bağlanır.

## İsimden adrese



Şema 3.1 — DNS çözümüleme: alan adı → IP → sunucu → sayfa içeriği.

- **DNS:** alan adını IP adresine çeviren sistem.
- Tarayıcı önce DNS'e "bu adın IP'si ne?" diye sorar.
- Yanıt gelince asıl sunucuya bağlanır.

### İPUCU

DNS olmasaydı her siteye ulaşmak için IP adreslerini ezberlemek zorunda kalırdık — telefon numaralarını isim yerine ezberlemek gibi. DNS çözümüleme **çok hızlıdır** (milisaniyeler) ve sonuçlar bir süre **önbelleğe alınır**, böylece her seferinde baştan sorulmaz. Bir alan adının IP'sini değiştirdiğinde (örneğin yeni bir sunucuya taşındığında), bu değişikliğin tüm dünyaya yayılması zaman alır — buna **yayılma (propagation)** denir ve birkaç saati bulabilir. Bu yüzden site taşımaları genelde sabırla planlanır.

### Pratikte ne olur?

### SONUÇ

Tarayıcıya `ornek.com` yazıldığında, önce DNS'e sorulur: "bu adın IP'si ne?" DNS `93.184.x.x` yanıtını döner; tarayıcı bu adresteki sunucuya bağlanır ve sayfa içeriğini alır. Tüm bu işlem genelde göz açıp kapayıncaya kadar (milisaniyeler içinde) tamamlanır.

**Alıştırma**

10 dk

DNS'i kavra:

- 1 DNS'in görevini "telefon rehberi" benzetmesiyle açıkla.
- 2 Bir alan adı yazıldığında gerçekleşen adımları sırala.
- 3 "Yayılma (propagation)" ne demek, kendi cümlele yaz.

**BÖLÜM 04**

# DNS Kayıt Türleri

Bir alan adının nasıl yönlendirileceğini DNS kayıtları belirler. Farklı kayıt türleri farklı işler yapar: biri siteyi bir sunucuya, biri e-postayı bir posta sunucusuna yönlendirir, biri de bir doğrulama bilgisi taşır.

## En sık kullanılan kayıtlar

TÜR	AD	DEĞER / HEDEF
A	@	93.184.x.x
CNAME	www	ornek.com
MX	@	mail.ornek.com
TXT	@	site doğrulama

Şema 4.1 — DNS kayıtları: her tür, alan adının bir yönünü yönlendirir.

- **A:** alan adını bir IPv4 adresine bağlar (siteyi sunucuya).
- **AAAA:** A'nın IPv6 sürümü.
- **CNAME:** bir adı başka bir ada yönlendirir (www → ana ad).
- **MX:** e-postayı hangi sunucunun alacağını belirtir.
- **TXT:** serbest metin (doğrulama, güvenlik kayıtları).

### Örnek DNS kayıtları (metin gösterimi)

Tür	Ad	Değer	
A	@	93.184.x.x	# site bu sunucuda
CNAME	www	ornek.com	# www → ana ad
MX	@	mail.ornek.com	# e-posta sunucusu
TXT	@	"v=spf1 ..."	# e-posta güvenliği

### İPUCU

En çok kullanacağın kayıt **A kayıdır**: alan adını sunucunun IP'sine bağlar. **CNAME** ile `www.ornek.com` 'u `ornek.com` 'a yönlendirmek yaygındır (ikisi de aynı yere gitsin diye). **MX kayıtları** e-posta içindir — siten bir sunucuda, e-postan başka bir sağlayıcıda olabilir; MX bunu ayarlar. **TXT kayıtları** genelde "bu alan adı gerçekten bana ait" doğrulamaları ve e-posta güvenliği (SPF, DKIM) için kullanılır. Kayıtlardaki **TTL** değeri, bilginin ne kadar önbellekte tutulacağını söyler; taşımadan önce TTL'i düşürmek yayılmayı hızlandırır.

**Pratikte ne olur?****SONUÇ**

Bir A kaydı eklediğinde, alan adın artık o IP'deki sunucuya işaret eder — siten o sunucudan yüklenir. MX kaydı eklediğinde, o alan adına gelen e-postalar belirttiğin posta sunucusuna yönlendirilir. Her kayıt, alan adının farklı bir trafiğini doğru hedefe gönderir; doğru kurulduğunda her şey otomatik akar.

**Alıştırma**

12 dk

Kayıtları eşle:

- 1 A, CNAME, MX ve TXT kayıtlarının her birinin görevini bir cümleyle yaz.
- 2 www'yi ana ada yönlendirmek için hangi kaydı kullanırsın?
- 3 E-postayı yönlendirmek için hangi kayıt gerekir, belirt.

**BÖLÜM 05**

# IP Adresleri ve Sunucular

Her alan adının arkasında bir IP adresi, her IP adresinin arkasında bir sunucu vardır. Sunucu, sitenin dosyalarını barındıran ve 7/24 çalışan bir bilgisayardır. IP adresi ise o sunucunun internetteki sayısal kimliğidir.

## IP ve sunucu kavramı

- **IP adresi:** bir cihazın internetteki sayısal adresi.
- **IPv4:** 93.184.0.1 gibi; **IPv6:** daha uzun, yeni biçim.
- **Sunucu:** siteyi barındıran, sürekli açık bir bilgisayar.
- **Veri merkezi:** binlerce sunucunun bulunduğu tesis.



Şema 5.1 — Alan adının arkasında: IP, sunucu ve veri merkezi.

### İPUCU

Sunucu aslında özel bir bilgisayardır — ama senin masaüstünden farkı, **sürekli açık**, hızlı internet bağlantılı ve kontrollü bir ortamda (veri merkezinde) durmasıdır. Veri merkezleri yedek elektrik, soğutma ve güvenlik sağlar; bu yüzden siteyi kendi bilgisayarında değil, bir sunucuda barındırırsın (yoksa bilgisayarını kapattığında site kapanırdı). **IPv4** adresleri (93.184.0.1 gibi) tükenmek üzere olduğundan, daha uzun ve bol olan **IPv6** giderek yaygınlaşıyor. Bir sunucunun fiziksel yeri (ülke), ziyaretçilere yakınlığı nedeniyle hızı etkileyebilir.

### Pratikte ne olur?

### SONUÇ

Alan adın bir IP'ye, o IP bir sunucuya işaret eder; o sunucu, sitenin dosyalarını tutar ve ziyaretçi geldiğinde sayfayı gönderir. Sunucu bir veri merkezinde, sürekli açık ve bağlı durduğu için site gece gündüz erişilebilir kalır — senin cihazların kapalı olsa bile.

**Alıştırma**

10 dk

Sunucuyu anla:

- 1 Sunucu ile kendi bilgisayarın arasındaki temel farkı yaz.
- 2 Neden siteni kendi bilgisayarında barındırmazsın, açıkla.
- 3 IPv4 ile IPv6 arasındaki farkı kısaca belirt.

**BÖLÜM 06**

# Hosting Nedir?

Hosting (barındırma), sitenin dosyalarının yaşadığı yeri kiralamaktır. Bir hosting sağlayıcısından alan aldığında, dosyalarını onların sunucusuna koyarsın ve alan adını o sunucuya yönlendirirsin — böylece site dünyaya açılır.

## Hosting ne sağlar?



Şema 6.1 — Neden hosting? Site sürekli, hızlı ve güvenli erişilebilir olmalı.

- **Disk alanı:** dosyaların (HTML, görsel, veritabanı) için yer.
- **Bant genişliği:** ziyaretçilere veri taşıma kapasitesi.
- **Çalışma süresi (uptime):** sunucunun açık kalma oranı.

### İPUCU

Hosting ile alan adı **ayrı iki şeydir**: alan adı adresin (ornek.com), hosting ise o adreste duran evin (sunucu). İkisini aynı firmadan da, ayrı firmalardan da alabilirsin; alan adını DNS ile hosting sunucusuna yönlendirerek bağlarsın. Hosting seçerken bakılacak temel ölçütler: **çalışma süresi garantisi** (uptime, örn. %99.9), **hız**, **destek kalitesi** ve **yedekleme** politikası. İhtiyacın büyüdükçe hosting türünü yükseltebilirsin (sonraki seviyede türleri inceleyeceğiz). Başlangıçta küçük bir paket yeterlidir; gereğinden fazlasını baştan almaya gerek yok.

**Pratikte ne olur?**

SONUÇ

Bir hosting paketi aldıđında, sitenin dosyalarını sađlayıcının sunucusuna yklersin ve alan adını DNS ile o sunucuya ynlendirirsin. Artık biri alan adını yazınca, dosyaların o sunucudan yklenir ve site aılır — hem de senin cihazların kapalıyken bile, 7/24. Hosting, siteni "yayında" tutan altyapıdır.

**Alıřtırma**

10 dk

Hosting'i kavra:

- 1 Alan adı ile hosting arasındaki farkı kendi cmlenle yaz.
- 2 Bir hosting seerken bakacađın  lt listele.
- 3 Neden site kendi bilgisayarında deđil, hosting'te durmalı, aıkla.

**BÖLÜM 07**

# Bir İsteğin Yolculuğu

Bir kullanıcı tarayıcıya alan adını yazıp Enter'a bastığında, perde arkasında hızlı bir zincir çalışır: DNS çözümü, sunucuya bağlanma, yanıtın gelmesi ve sayfanın çizilmesi. Bu yolculuğu anlamak, sorun çıktığında nerede arayacağını öğretir.

## Adımlar



Şema 7.1 — Bir isteğin yolculuğu: çöz → bağlan → yanıtla → çiz.

### İPUCU

Bu zincir, Backend ve API modüllerinde gördüğün istek-yanıt döngüsünün tam hâlidir — ama bu kez DNS ve altyapıyı da içerir. Sorun giderirken her halka ayrı bir ipucu verir: site **hiç açılmıyorsa** önce DNS'i (alan adı doğru IP'ye gidiyor mu?) kontrol et; **açılıyor ama hata veriyorsa** sunucu/uygulama tarafına bak; **yavaşsa** sunucu performansını veya ağ mesafesini düşün. "Çalışmıyor" demek yerine zincirin hangi halkasında koptuğunu bulmak, çözümün yarısıdır. Geliştirici araçlarının "Network" sekmesi, bu yolculuğun her adımını sana gösterir.

**Pratikte ne olur?****SONUÇ**

Kullanıcı alan adını yazınca: tarayıcı DNS'ten IP'yi öğrenir, o sunucuya bir istek gönderir, sunucu sayfayı (HTML ve veriyi) hazırlayıp yanıt olarak döner, tarayıcı da bunu ekranda çizer. Hepsi genelde bir saniyenin altında olur. Bir adım aksarsa kullanıcı ya boş sayfa, ya hata, ya da yavaşlık yaşar — ve hangi adımın aksadığı, sorunun nerede olduğunu söyler.

**Alıştırma**

12 dk

Yolculuğu izle:

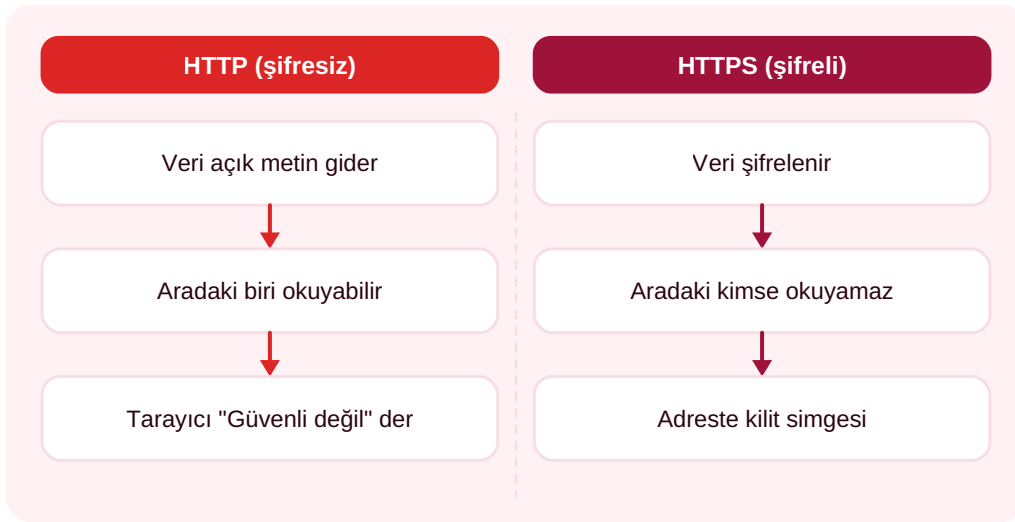
- 1 Bir alan adı yazıldığında gerçekleşen dört adımı sırala.
- 2 Site hiç açılmıyorsa önce neyi kontrol edersin, neden?
- 3 Bu zincirin Backend modülündeki istek-yanıt ile bağımlı yaz.

**BÖLÜM 08**

# HTTP ve HTTPS

Tarayıcı ile sunucu, HTTP adlı bir protokolle konuşur. HTTPS ise bunun şifreli ve güvenli sürümüdür. Bugün her site HTTPS kullanmalıdır — hem güvenlik hem güven için. Adres çubuğundaki kilit simgesi, bağlantının HTTPS olduğunu gösterir.

## HTTP vs HTTPS



Şema 8.1 — HTTPS, tarayıcı ile sunucu arasındaki veriyi şifreler.

- **HTTP:** veri açık (şifresiz) gider — risklidir.
- **HTTPS:** veri şifrelidir; araya giren okuyamaz.
- **SSL/TLS sertifikası:** HTTPS'i mümkün kılan teknoloji.

### İPUCU

HTTPS artık **standarttır, lüks değil**: tarayıcılar HTTP siteleri "Güvenli değil" diye işaretler, arama motorları HTTPS'i tercih eder ve kullanıcılar kilit simgesi olmayan sitelere güvenmez. HTTPS'i bir **SSL/TLS sertifikası** sağlar (Seviye 3'te ayrıntısına gireceğiz) ve bugün bu sertifikalar çoğu hosting'te **ücretsiz ve otomatik** kurulur (Let's Encrypt sayesinde). Özellikle form, giriş veya ödeme alan her sitede HTTPS **zorunludur** — yoksa kullanıcı şifreleri ve kişisel verileri açıkta gider. Kısacası: bir siteyi yayına alırken HTTPS'i mutlaka aç.

**Pratikte ne olur?****SONUÇ**

HTTP kullanan bir sitede, kullanıcının girdiği veriler (şifre, kart bilgisi) açık metin olarak gider ve ağdaki biri tarafından okunabilir. HTTPS'te ise bu veri şifrelenir; araya giren yalnızca anlamsız bir karışım görür. Sonuç: tarayıcı adres çubuğunda kilit simgesi gösterir ve kullanıcı sitene güvenle veri girer.

**Alıştırma**

10 dk

Güvenli bağlan:

- 1 HTTP ile HTTPS arasındaki temel farkı yaz.
- 2 Hangi tür siteler için HTTPS kesinlikle zorunludur, örnek ver.
- 3 Adres çubuğundaki kilit simgesi ne anlama gelir, açıkla.

**SEVİYE 2**

# Hosting Türleri ve Kurulum

Doğru evi seçmek: hosting türleri, paylaşımlı hosting, VPS ve sunucu yönetimi, bulut hosting, statik site barındırma ve hosting seçimi.

## BÖLÜM 09

# Hosting Türleri

Tek tip hosting yoktur; ihtiyaca göre farklı türler vardır. Bir blog ile milyonlarca ziyaretçili bir uygulama aynı altyapıyı kullanmaz. Dört ana tür arasındaki fark, üç şeyde toplanır: kontrol, maliyet ve ölçeklenme.

## Dört ana tür

- **Paylaşımlı:** bir sunucuyu birçok site paylaşır — ucuz, basit.
- **VPS:** sanal, ayrılmış bir bölüm — daha çok kontrol.
- **Bulut:** esnek, ölçeklenen kaynaklar — talebe göre büyür.
- **Statik:** sadece dosyalar — çok hızlı, çoğu zaman ücretsiz.

Paylaşımlı	VPS	Bulut
Kontrol: az	Kontrol: orta	Kontrol: yüksek
Maliyet: düşük	Maliyet: orta	Maliyet: değişken
Ölçek: sınırlı	Ölçek: iyi	Ölçek: esnek
Kim için: başlangıç	Kim için: büyüyen	Kim için: ölçeklenen

Şema 9.1 — Hosting türleri: kontrol, maliyet ve ölçeklenme dengesi.

### İPUCU

Doğru türü seçmenin sırrı, **bugünkü ihtiyacına** göre başlamak ve gerektiğinde yükseltmektir. Çoğu yeni site için **paylaşımlı hosting** fazlasıyla yeterlidir; trafiğin ve teknik ihtiyacın arttıkça **VPS'e**, oradan da **bulut'a** geçebilirsiniz. Sadece statik dosyalardan oluşan siteler (tek sayfa, dokümantasyon, blog) için **statik hosting** hem en hızlı hem en ucuz seçenektir. Baştan en pahalı/güçlü seçeneği almak yaygın bir hatadır — kullanmadığın kapasiteye para ödersin. "Önce küçük başla, ihtiyaç doğunca büyüt" en sağlıklı yoldur.

### Pratikte ne olur?

### SONUÇ

Bir blog açıyorsan paylaşımlı hosting saniyeler içinde yayına almanı sağlar; özel yazılım kurman gereken bir uygulama için VPS daha uygundur; ani trafik artışları yaşayan bir servis için bulut, talebe göre otomatik büyür. Doğru tür, sitenin bugün ve yarın ne kadar kaynağa ve kontrole ihtiyaç duyduğuna bağlıdır.

**Alıştırma**

12 dk

Tür eşle:

- 1 Dört hosting türünü ve birer kullanım örneğini yaz.
- 2 Kontrol, maliyet ve ölçeklenme açısından paylaşımlı ile bulutu karşılaştır.
- 3 Yeni bir blog için hangi türü seçerdin, neden?

**BÖLÜM 10**

# Paylaşımlı Hosting (Shared)

Paylaşımlı hosting, bir sunucunun birçok site arasında paylaştığı en yaygın ve ekonomik türdür. Maliyet bölüşüldüğü için ucuzdur; yönetimi sağlayıcı yaptığı için kolaydır. Karşılığında kontrol ve kaynak sınırlıdır.

## Nasıl çalışır?



Şema 10.1 — Paylaşımlı hosting: bir sunucu, birçok site, bölüşülen maliyet.

- **Avantaj:** ucuz, kurulum gerektirmez, panel (cPanel vb.) ile kolay.
- **Dezavantaj:** sınırlı kaynak, az kontrol, "komşu" etkisi.
- **Kim için:** bloglar, tanıtım siteleri, küçük işletmeler.

### İPUCU

Paylaşımlı hosting'in en bilinen riski "**gürültülü komşu**" etkisidir: aynı sunucudaki başka bir site ani bir trafik aldığı anda, ortak kaynaklar zorlanır ve senin siten de yavaşlayabilir. Yine de çoğu küçük-orta site için bu nadiren sorun olur. Yönetimi sağlayıcı üstlendiği için (güncelleme, güvenlik yamaları, yedekleme) **teknik bilgi gerektirmez** — bu, başlangıç için büyük bir avantajdır. Genelde bir **kontrol paneli** (cPanel, Plesk gibi) ile dosya yükleme, e-posta, veritabanı ve alan adı bağlama işlerini tıklayarak yaparsın. Trafiğin tutarlı biçimde artıyorsa, bir üst türe (VPS) geçme zamanı gelmiş demektir.

**Pratikte ne olur?****SONUÇ**

Paylaşımli bir paket aldıđında, sağlayıcının bir kontrol panelinden dosyalarını yükler, alan adını bağlar ve siten dakikalar içinde yayına girer. Sunucunun bakımı, güvenliđi ve güncellemeleri sana görünmez — hepsini sağlayıcı yapar. Karşılıđında CPU/bellek gibi kaynaklar diđer sitelerle paylaşıldıđı için yoğun trafikte sınırlara takılabilirsin.

**Alıştırma**

10 dk

Paylaşımliyı değerlendir:

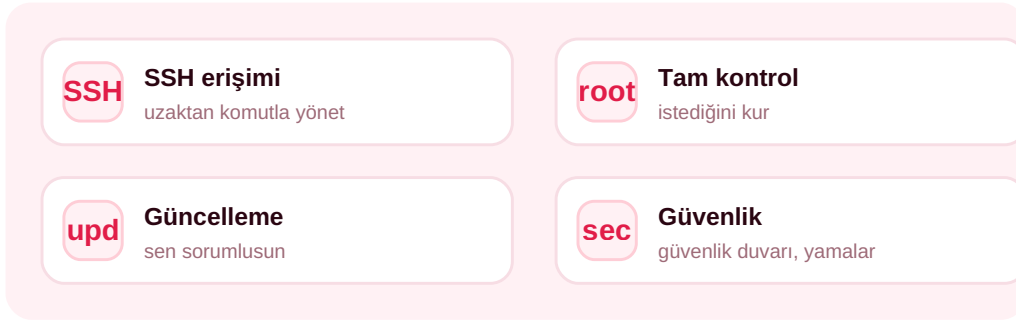
- 1 Paylaşımli hosting'in iki avantajını ve iki dezavantajını yaz.
- 2 "Gürültülü komşu" etkisini kendi cümlele açıkla.
- 3 Hangi tür siteler için paylaşımli hosting uygundur, örnek ver.

**BÖLÜM 11**

# VPS ve Sunucu Yönetimi

VPS (Sanal Özel Sunucu), bir fiziksel sunucunun yazılımla bölünmüş, sana ayrılmış bir parçasıdır. Paylaşımlıdan farkı: kaynakların garanti edilir ve sunucu üzerinde tam kontrolün (root erişimi) olur. Karşılığında yönetim sorumluluğu da sana geçer.

## VPS yönetim görevleri



Şema 11.1 — VPS: tam kontrol, ama yönetim sorumluluğu sende.

- **Garanti kaynak:** CPU/RAM sana ayrılır (paylaşılmaz).
- **Root erişimi:** istediğin yazılımı kurar, ayarlıyorsun.
- **SSH:** sunucuyu uzaktan, komut satırından yönetirsin.
- **Sorumluluk:** güncelleme, güvenlik, yedekleme sende.

### SSH ile sunucuya bağlanmak (terminal)

```
# Sunucuya güvenli bağlan
ssh kullanıcı@93.184.x.x

# Paketleri güncelle (örnek: Ubuntu)
sudo apt update && sudo apt upgrade
```

### İPUCU

VPS, "kendi sunucun" gibidir: **tam özgürlük** verir ama **tam sorumluluk** da getirir. Güvenlik yamalarını sen yüklersin, güvenlik duvarını sen yapılandırırısın, bir şey bozulursa sen düzeltirsin. Bu yüzden VPS, biraz teknik rahatlık ister — ya da "yönetilen VPS" (managed) olarak bu yükü sağlayıcıya bırakabilirsin. **SSH anahtarı** ile bağlanmak (parola yerine) çok daha güvenlidir; root ile doğrudan çalışmak yerine sınırlı yetkili bir kullanıcı kullanmak iyi bir alışkanlıktır (Modül kapsamındaki "en az yetki" ilkesi). VPS, paylaşımının darlığından kurtulmak ama bulutun karmaşası/maliyeti olmadan kontrol istemek için ideal orta yoldur.

**Pratikte ne olur?****SONUÇ**

Bir VPS aldığında, sana bir IP ve giriş bilgisi verilir; SSH ile bağlanıp sunucuyu sıfırdan kendin kurarsın (web sunucusu, veritabanı, uygulaman). Sana ayrılan kaynaklar garanti olduğu için komşu etkisi yaşamazsın. Ama güncelleme ve güvenlik senin işin: sunucuyu güncel ve korunaklı tutmak senin sorumluluğundur.

**Alıştırma**

12 dk

VPS'i kavra:

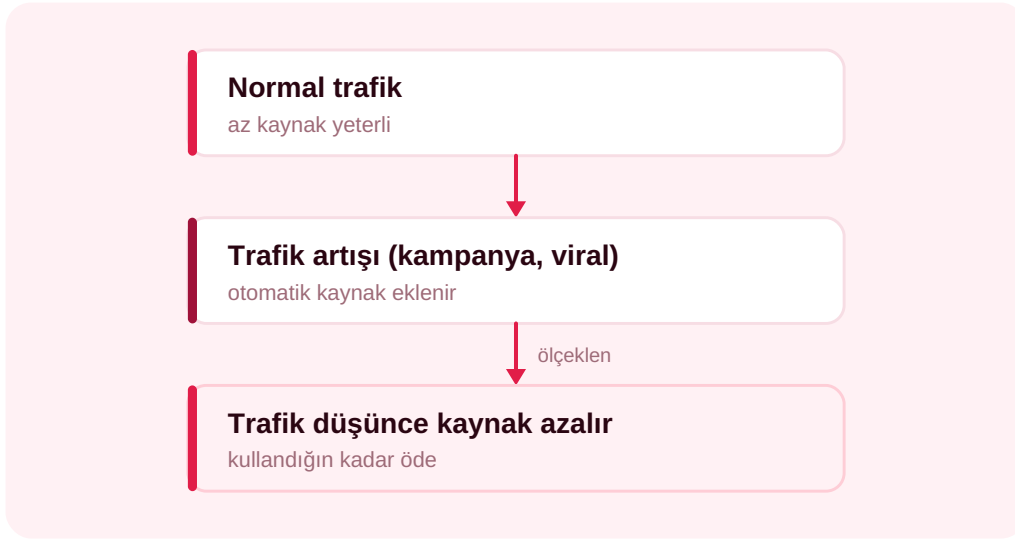
- 1 VPS ile paylaşımlı hosting arasındaki iki temel farkı yaz.
- 2 VPS'te hangi sorumluluklar sana geçer, listele.
- 3 Parola yerine SSH anahtarı kullanmanın avantajını açıkla.

**BÖLÜM 12**

# Bulut Hosting (Cloud)

Bulut hosting, kaynakları tek bir fiziksel sunucuya değil, birçok sunucudan oluşan esnek bir havuza dayandırır. En güçlü yanı ölçeklenebilirliktir: trafiğin arttığında kaynak otomatik eklenir, azaldığında geri çekilir — ve genelde kullandığın kadar ödersin.

## Esneklik ve ölçeklenme



Şema 12.1 — Bulut: kaynaklar talebe göre otomatik büyür ve küçülür.

- **Ölçeklenebilir:** talebe göre otomatik büyür/küçülür.
- **Dayanıklı:** bir sunucu çökse başka biri devralır.
- **Kullandığın kadar öde:** esnek faturalandırma.

### İPUCU

Bulutun en büyük avantajı **ani trafik dalgalarına** dayanmasıdır: bir kampanya, haber veya viral içerik sayfanı bir anda yüz binlerce kişiye taşıdığına, geleneksel bir sunucu çökerken bulut otomatik kaynak ekleyerek ayakta kalır. "Kullandığın kadar öde" modeli avantajlıdır ama dikkat ister: yanlış yapılandırma veya beklenmedik trafik **faturayı hızla şişirebilir** — bütçe uyarıları (alarm) kurmak önemlidir. Bulut çok güçlüdür ama aynı zamanda **en karmaşık** türdür; küçük bir site için fazla gelebilir. Büyüme potansiyeli yüksek veya değişken trafikli projeler için biçilmiş kaftandır.

**Pratikte ne olur?****SONUÇ**

Bulut hosting'te siten birden çok sunucuya yayılabilir; biri arızalansa diğeri devreye girer, böylece kesinti riski azalır. Trafiğin yükseldiğinde sistem otomatik kaynak ekler, sayfan yavaşlamaz veya çökmez; trafik düşünce kaynak (ve maliyet) geri iner. Sonuç: değişken yüklerle uyum sağlayan, dayanıklı bir altyapı — ama dikkatli bütçe yönetimiyle.

**Alıştırma**

10 dk

Bulut kavra:

- 1 Bulut hosting'in ölçeklenme avantajını bir senaryoyla açıkla.
- 2 "Kullandığın kadar öde" modelinin avantajı ve riski nedir, yaz.
- 3 Hangi tür projeler için bulut en uygundur, belirt.

**BÖLÜM 13**

# Statik Site Barındırma

Bir site yalnızca HTML, CSS, JS ve görsellerden oluşuyorsa (sunucu tarafında kod çalıştırmıyorsa), onu "statik" olarak barındırmak en hızlı ve genellikle en ucuz yoldur. Modern statik hosting, dosyalarını dünya geneline dağıtır ve şimşek hızında sunar.

## Statik + CDN



Şema 13.1 — Statik hosting: dosyalar CDN ile dağıtılır, hızlı sunulur.

- **Çok hızlı:** hazır dosyalar, üretim adımı yok.
- **Güvenli:** çalışan sunucu kodu olmadığı için saldırı yüzeyi küçük.
- **Ucuz:** birçok sağlayıcıda ücretsiz başlangıç katmanı.

### İPUCU

Statik hosting, Modül 3-4-5'te (HTML/CSS/JS) yaptığın türden siteler için idealdir; ayrıca modern "statik site üreteçleri" blog ve dokümantasyon gibi içerikleri de statik dosyalara dönüştürebilir. Avantajları çarpıcı: çalışan sunucu kodu olmadığı için **hacklenecek bir arka uç yoktur** (saldırı yüzeyi küçük), dosyalar önceden hazır olduğu için **çok hızlıdır** ve CDN sayesinde dünyanın her yerinden aynı hızla açılır. Veritabanı veya kullanıcı girişi gibi dinamik işler gerekiyorsa, bunları ayrı API'lerle (Modül 7) ekleyebilirsin — "statik ön yüz + API arka uç" çok yaygın, modern bir yaklaşımdır. Küçük ve orta projeler için önce statik seçeneği düşünmek çoğu zaman en akıllıcasıdır.

### Pratikte ne olur?

SONUÇ

Statik bir site yüklediğinde, dosyaların bir CDN üzerinden dünya geneline kopyalanır; bir ziyaretçi siteyi açtığında, içerik ona **coğrafi olarak en yakın** sunucudan gelir — bu yüzden çok hızlıdır. Sunucu tarafında kod çalışmadığı için hem güvenli hem ucuzdur. Sonuç: bakımı az, hızı yüksek, maliyeti düşük bir yayın yöntemi.

### Alıştırma

10 dk

Statığı değerlendir:

- 1 Statik hosting hangi tür siteler için uygundur, örnek ver.
- 2 Statik hosting'in hız ve güvenlik avantajlarını açıkla.
- 3 Dinamik bir özellik (örn. giriş) gerekirse ne yaparsın, yaz.

**BÖLÜM 14**

# Hosting Seçimi: Karşılaştırma

Doğru hosting'i seçmek için birkaç soruyu yanıtlaman yeterli: ne kadar trafik bekliyorsun, ne kadar teknik kontrol istiyorsun, bütçen ne? Bu sorular seni doğru türe yönlendirir. "En iyi" hosting yoktur; "senin projene en uygun" hosting vardır.

## Seçim soruları



Şema 14.1 — Hosting seçimi: birkaç soru seni doğru türe götürür.

### İPUCU

Karar verirken bakacağın somut ölçütler: **çalışma süresi (uptime)** garantisi (%99.9 iyi bir eşittir), **hız** (sunucu konumu ve teknolojisi), **destek** (7/24 mı, hangi kanaldan?), **yedekleme** politikası, **SSL** dahil mi (artık çoğu yerde ücretsiz olmalı), ve **büyüme yolu** (üst pakete kolay geçiş var mı?). Ucuz olana değil, **ihtiyacını karşılayana** odaklan — ama gereğinden fazlasını da alma. Sözleşme öncesi gerçek kullanıcı yorumlarına bakmak ve küçük bir pakette deneyip büyümek sağlıklı bir yaklaşımdır. Unutma: alan adı ve hosting ayrı alınabilir, bu da seni tek bir sağlayıcıya bağımlı olmaktan kurtarır.

**Pratikte ne olur?**

SONUÇ

Sorulara verdiğin yanıtlar seni doğru türe götürür: sadece dosyalardan oluşan bir site statik hosting'e, teknik kontrol istemeyen küçük bir site paylaşımlıya, kendi kurulumunu yapmak isteyen biri VPS'e, değişken/yüksek trafikli bir proje buluta uygundur. Doğru seçim, bugünkü ihtiyacını karşılayan ve yarın büyümene izin veren seçimdir.

**Alıştırma**

12 dk

Seçimini yap:

- 1 Bir proje hayal et; trafik, kontrol ve bütçe ihtiyacını yaz.
- 2 Bu ihtiyaçlara göre hangi hosting türünü seçerdin, gerekçelendir.
- 3 Hosting seçerken bakacağın dört ölçütü listele.

**SEVİYE 3**

# Güvenlik ve Performans

Hızlı ve güvenli site: SSL/TLS ve sertifikalar, SSL kurulumu, CDN, performans ve önbellekleme, sunucu güvenliği, yedekleme.

**BÖLÜM 15**

# SSL/TLS ve Sertifikalar

HTTPS'i mümkün kılan teknoloji SSL/TLS'tir. Bir SSL sertifikası iki iş yapar: tarayıcı ile sunucu arasındaki veriyi şifreler ve sitenin gerçekten iddia ettiği site olduğunu doğrular. Bağlantı kurulurken kısa bir "el sıkışma" (handshake) gerçekleşir.

## TLS el sıkışması



Şema 15.1 — TLS el sıkışması: selamla → sertifika → doğrula → şifrele.

- **Şifreleme:** veriyi aradaki gözlerden gizler.
- **Kimlik doğrulama:** sitenin gerçekliğini kanıtlar.
- **Sertifika otoritesi (CA):** sertifikayı veren güvenilir kurum.

### İPUCU

Sertifikayı, sitenin **kimlik kartı** gibi düşün: onu güvenilir bir kurum (Sertifika Otoritesi) imzalar ve tarayıcı bu imzayı tanır. El sıkışma sırasında tarayıcı, sertifikanın geçerli, süresi dolmamış ve doğru alan adı için olduğunu kontrol eder; bir sorun varsa kullanıcıya korkutucu bir uyarı gösterir. Sertifika türleri vardır (alan adı doğrulamalı DV, kuruluş doğrulamalı OV/EV) ama çoğu site için **ücretsiz DV sertifikaları** (Let's Encrypt) fazlasıyla yeterlidir. Önemli olan, her sitede HTTPS'in aktif ve sertifikanın **güncel** olmasıdır — süresi dolmuş bir sertifika, siteni anında "güvenli değil" durumuna düşürür.

**Pratikte ne olur?**

SONUÇ

Bir kullanıcı HTTPS sitene bağlandığında, saniyenin altında bir el sıkışma olur: tarayıcı bağlantı ister, sunucu sertifikasını gösterir, tarayıcı bunu doğrular ve ardından şifreli bir kanal kurulur. Bu noktadan sonra alışverişten tüm veri (şifreler, formlar) şifrelidir. Kullanıcı tüm bunları görmez; yalnızca adres çubuğundaki kilit simgesini fark eder.

**Alıştırma**

12 dk

SSL'i kavra:

- 1 Bir SSL sertifikasının yaptığı iki işi yaz.
- 2 TLS el sıkışmasının dört adımını sırala.
- 3 Süresi dolmuş bir sertifika ne soruna yol açar, açıkla.

**BÖLÜM 16**

# SSL Kurulumu ve Yönetimi

Eskiden SSL sertifikası almak ücretli ve zahmetliydi; bugün Let's Encrypt gibi ücretsiz otoriteler ve otomatik araçlar sayesinde dakikalar sürer. Çoğu modern hosting, HTTPS'i tek tıkla veya otomatik olarak kurar.

## Otomatik kurulum



Şema 16.1 — Modern SSL: doğrula → üret → kur → otomatik yenile.

### Otomatik sertifika (örnek komut)

```
# Certbot ile otomatik kurulum ve yenileme
sudo certbot --nginx -d ornek.com -d www.ornek.com

# Otomatik yenilemeyi test et
sudo certbot renew --dry-run
```

**İPUCU**

Let's Encrypt sertifikaları yalnızca **90 gün** geçerlidir — bu kısa süre kasıtlıdır ve **otomatik yenilemeyi** teşvik eder. Certbot gibi araçlar yenilemeyi arka planda kendiliğinden yapar; kurduktan sonra çoğu zaman bir daha düşünmezsin. Paylaşımlı hosting kullanıyorsan, HTTPS genelde panelde tek tıkla (hatta varsayılan olarak) açıktır. Kurduktan sonra iki şeyi yap: tüm HTTP trafiğini HTTPS'e **yönlendir** (kullanıcılar yanlışlıkla şifresiz bağlanmasın) ve sertifikanın **otomatik yenilendiğini doğrula**. Süresi dolan bir sertifika sık karşılaşılan ama tamamen önlenemez bir kesinti sebebidir; otomatik yenileme bu riski ortadan kaldırır.

**Pratikte ne olur?****SONUÇ**

Sertifika kurulum aracı önce alan adının gerçekten sana ait olduğunu doğrular, sonra ücretsiz bir sertifika üretip sunucuna kurar ve HTTPS'i etkinleştirir. En önemlisi: yenilemeyi otomatikleştirir, böylece sertifika süresi dolmadan kendiliğinden tazelenir. Sonuç: tarayıcıda kilit simgesi belirir ve siten kalıcı olarak güvenli kalır.

**Alıştırma**

10 dk

SSL'i yönet:

- 1 Modern SSL kurulumunun neden eskisinden kolay olduğunu yaz.
- 2 HTTP trafiğini HTTPS'e yönlendirmek neden gerekli, açıkla.
- 3 Otomatik yenilemenin önemini kendi cümlemlerle belirt.

## BÖLÜM 17

# CDN ve İçerik Dağıtımı

CDN (İçerik Dağıtım Ağı), sitenin içeriğini dünya geneline yayılmış birçok sunucuda kopyalayarak, her ziyaretçiye coğrafi olarak en yakın noktadan sunar. Sonuç: çok daha hızlı yükleme ve ani trafiğe karşı dayanıklılık.

## Yakından sunmak



Şema 17.1 — CDN: içeriği ziyaretçiye en yakın noktadan sunarak hızlandırır.

- **Hız:** içerik fiziksel olarak ziyaretçiye yakındır.
- **Dayanıklılık:** yük birçok sunucuya dağılır.
- **Koruma:** birçok CDN saldırı (DDoS) filtresi de sunar.

### İPUCU

Mesafe, web hızını gerçekten etkiler: içerik bir kıta öteden geliyorsa, ışık hızında bile olsa gecikme hissedilir. CDN bu sorunu, içeriğin kopyalarını dünya geneline (kenar sunuculara) dağıtarak çözer — bir ziyaretçi siteni açtığında, statik içerik (görseller, stil dosyaları, scriptler) ona en yakın noktadan gelir. Bu hem **hızı** artırır hem de asıl sunucunun **yükünü** azaltır. Birçok CDN ek olarak **DDoS koruması** ve önbellekleme de sağlar. Statik hosting genelde CDN'i baştan içerir; dinamik sitelerde ise en azından görsel/CSS/JS gibi statik varlıkları CDN'e koymak büyük bir hız kazancı sağlar. Modern web performansının temel araçlarından biridir.

**Pratikte ne olur?****SONUÇ**

CDN kullandığında, sitenin görselleri ve dosyaları dünya geneline kopyalanır; Ankara'daki bir ziyaretçi içeriği yakındaki bir sunucudan, Londra'daki ziyaretçi ise Londra'ya yakın bir sunucudan alır. Herkes için yükleme hızlanır, asıl sunucun rahatlar ve ani trafik dalgaları kolayca karşılanır. Kullanıcı yalnızca "site çok hızlı açıldı" der.

**Alıştırma**

10 dk

CDN'i kavra:

- 1 CDN'in hızı nasıl artırdığını "mesafe" kavramıyla açıkla.
- 2 Bir CDN'in hız dışında sağladığı iki faydayı yaz.
- 3 Hangi tür içerikler CDN'e konmaya en uygundur, belirt.

**BÖLÜM 18**

# Performans ve Önbellekleme

Hızlı bir site, daha mutlu kullanıcı ve daha iyi arama sıralaması demektir. Performansın en güçlü araçlarından biri önbelleklemedir (caching): bir kez üretilen veya yüklenen şeyi saklayıp tekrar tekrar kullanmak, her seferinde sıfırdan üretmemek.

## Önbellekleme mantığı



Şema 18.1 — Önbellekleme: bir kez üret, çok kez sun.

- **Tarayıcı önbelleği:** ziyaretçi dosyaları tekrar indirmez.
- **Sunucu önbelleği:** üretilen sayfa/sonuç saklanır.
- **Görsel optimizasyonu:** doğru boyut ve biçim büyük fark yaratır.

### İPUCU

Performansın birkaç temel kaldırıcı vardır: **önbellekleme** (aynı işi tekrarlamamak), **görsel optimizasyonu** (genelde sayfaların en ağır parçası görsellerdir — doğru boyut ve modern biçimler sayfayı çok hızlandırır), **dosya küçültme** (CSS/JS sıkıştırma) ve **CDN** (bir önceki bölüm). Önbellekleme güçlüdür ama bir tuzağı vardır: içeriği güncellediğinde, eski önbelleğin temizlenmesi (veya sürümlenmesi) gerekir — yoksa kullanıcılar eski sürümü görür. "Ölçmeden iyileştirme" hatasına düşme: önce sitenin neresinin yavaş olduğunu bir performans aracıyla **ölç**, sonra en çok fark yaratacak yeri iyileştir. Çoğu zaman birkaç basit değişiklik (görselleri küçültmek, önbelleği açmak) dramatik hızlanma sağlar.

**Pratikte ne olur?****SONUÇ**

Önbellek olmadan, her ziyaretçi için sayfa sıfırdan üretilir (veritabanı sorgulanır, şablon işlenir) — bu yavaştır ve sunucuyu yorar. Önbellekle, bir kez üretilen sonuç saklanır ve sonraki ziyaretçilere doğrudan sunulur; sayfa anında açılır, sunucu rahatlar. Kullanıcı için sonuç nettir: bekleme süresi düşer, site akıcı hisseder.

**Alıştırma**

10 dk

Hızı artır:

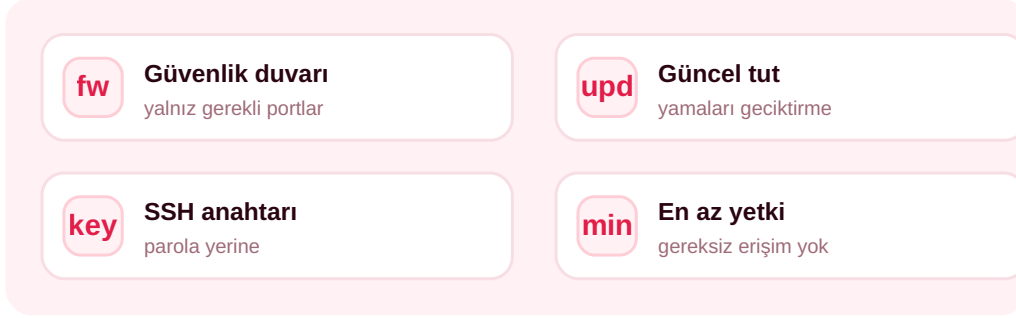
- 1 Önbelleklemenin performansı nasıl artırdığını açıkla.
- 2 Önbelleğin "eski içerik gösterme" tuzağını ve çözümünü yaz.
- 3 Görsel optimizasyonunun neden önemli olduğunu belirt.

## BÖLÜM 19

# Sunucu Güvenliđi

Bir sunucuyu yönetmek (özellikle VPS/bulutta), onu güvende tutma sorumluluđunu da getirir. İyi haber: birkaç temel önlem, saldırıların büyük çođunluđunu durdurur. Güvenlik bir ürün deđil, sürekli bir alışkanlıktır.

## Temel sunucu önlemleri



Şema 19.1 — Sunucu güvenliđinin dört temel önlemi.

- **Güvenlik duvarı:** yalnızca gereken portları (80/443/SSH) aç.
- **Güncellemeler:** işletim sistemi ve yazılımları güncel tut.
- **SSH anahtarı:** parola yerine anahtarla giriş yap.
- **En az yetki:** her kullanıcı/servise yalnız gerekeni ver.

### İPUCU

Bu ilkeler, tüm seri boyunca tekrarlanan güvenlik mantıđının altyapı tarafıdır: **saldırı yüzeyini küçült** (gereksiz port/servis kapalı), **bilinen açıkları kapat** (güncellemeler — saldırıların çođu yamanmış eski açıkları hedefler), **güçlü kimlik doğrulama** kullan (SSH anahtarı, parola deđil; mümkünse iki adımlı doğrulama) ve **en az yetki** ilkesini uygula (root ile sürekli çalışma, gereksiz erişim verme). Ek olarak: **fail2ban** gibi araçlarla otomatik saldırıları engelle, **günlükleri (log)** izle ve düzenli **yedek** al (sonraki bölüm). Yönetilen hosting kullanıyorsan bunların çođunu sağlayıcı üstlenir — bu da paylaşımlı/yönetilen hizmetlerin neden başlangıç için güvenli olduđunu gösterir.

### Pratikte ne olur?

### SONUÇ

Güvenli bir sunucu: yalnızca gerekli portları açık tutar, yazılımlarını güncel tutar, parola yerine SSH anahtarlarıyla giriş kabul eder ve her bileşene yalnızca ihtiyacı kadar yetki verir. Bu dört önlem, internetteki otomatik saldırıların büyük kısmını daha kapıda durdurur. Güvenlik tek seferlik bir kurulum deđil, sürdürülen bir disiplindir.

**Alıştırma**

12 dk

Sunucuyu koru:

- 1 Dört temel sunucu güvenlik önlemini ve neye karşı koruduđunu yaz.
- 2 Güncellemeleri geciktirmenin neden riskli olduđunu açıkla.
- 3 "En az yetki" ilkesini kendi cümlemlerle tanımla.

**BÖLÜM 20**

# Yedekleme ve Felaket Kurtarma

Hiçbir sistem hatasız değildir: disk bozulabilir, bir komut yanlışlıkla veriyi silebilir, bir saldırı dosyaları şifreleyebilir. Düzenli ve test edilmiş yedekler, bu felaketleri birer "geçici aksaklığa" dönüştürür. Yedeği olmayan veri, er ya da geç kaybolur.

## İyi bir yedek stratejisi



Şema 20.1 — Yedekleme: al → ayrı sakla → test et → geri yükle.

- **3-2-1 kuralı:** 3 kopya, 2 farklı ortam, 1 farklı konum.
- **Otomatik:** elle yedek unutulur; otomatikleştir.
- **Test et:** geri yüklenemeyen yedek, yedek değildir.

### İPUCU

Yedeklemenin altın kuralı **3-2-1**'dir: verinin en az **3** kopyası, **2** farklı ortamda, **1** tanesi farklı bir fiziksel konumda (örneğin başka bir bulut/şehir) bulunsun. Böylece tek bir arıza (disk, sunucu, hatta veri merkezi) tüm yedeklerini birden yok edemez. En sık yapılan hata, yedek almak ama **geri yüklemeyi hiç test etmemektir** — bozuk veya eksik bir yedeğin işe yaramadığını, en kötü anda (felaket anında) öğrenirsin. Bu yüzden yedekleri periyodik olarak geri yükleyip dene. Ayrıca **fidye yazılımına** karşı, en az bir yedek "değiştirilemez" (immutable) veya çevrimdışı olmalı. Yönetilen hosting genelde otomatik yedek sunar; yine de kendi bağımsız yedeğini almak en güvenlisidir.

**Pratikte ne olur?****SONUÇ**

İyi bir yedek stratejisi şöyle işler: veriyi düzenli ve otomatik olarak yedeklersin, kopyayı asıl sunucudan ayrı bir yerde saklarsın, geri yüklemenin gerçekten çalıştığını periyodik test edersin — ve bir felaket olduğunda (silinme, arıza, saldırı) hızla son sağlam yedeğe dönersin. Sonuç: veri kaybı bir kâbus olmaktan çıkar, yönetilebilir bir aksaklığa dönüşür.

**Alıştırma**

10 dk

Yedeğini kur:

- 1 3-2-1 yedekleme kuralını kendi cümlele açıkla.
- 2 Yedeği test etmenin neden şart olduğunu yaz.
- 3 Fidyeye yazılımına karşı yedeğin nasıl korunması gerektiğini belirt.

## SEVİYE 4

# Yayına Almak ve İşletmek

Yayında tutmak: bir siteyi yayına almak, dağıtım süreci, alan adı e-postası, izleme ve kesinti yönetimi, maliyetler ve yayına alma kontrol listesi.

**BÖLÜM 21**

# Bir Siteyi Yayına Almak

Tüm parçalar hazır: alan adın, hosting'in ve siten var. Şimdi sıra dosyaları sunucuya taşıyıp her şeyi birbirine bağlamakta. Yayına alma, "benim makinemde çalışıyor"u "herkes için yayında"ya dönüştürmektir.

## Yayına alma adımları



Şema 21.1 — Yayına alma: yükle → bağla → güvenli kıl → test et.

### Git ile dağıtım (örnek akış)

```
# Yerelde değişiklikleri gönder
git add .
git commit -m "Yayına hazır"
git push origin main

# Sunucu/servis push'u alıp otomatik yayına alır
```

**İPUCU**

Dosyaları taşımanın birkaç yolu var: **panel** üzerinden yükleme (paylaşımlı hosting'te en kolay), **FTP/SFTP** (klasik), veya **Git tabanlı dağıtım** (modern ve önerilen — kodu bir depoya gönderirsin, sunucu otomatik alır). Git tabanlı akış, "tek tıkla geri al" ve sürüm geçmişi avantajı sağlar. Yayına almadan önce mutlaka bir **kontrol listesi** uygula (son bölümde tam liste var): HTTPS açık mı, alan adı doğru bağlanmış mı, hata sayfaları çalışıyor mu, formlar gidiyor mu, gizli bilgiler (parolalar) koda gömülü değil mi? İlk yayına alma her zaman biraz heyecanlıdır; küçük adımlarla ilerle ve her adımı test et. DNS değişiklikleri yayılması (propagation) zaman alabileceğinden, "hemen görünmüyor" panik sebebi değildir.

**Pratikte ne olur?****SONUÇ**

Yayına alma şöyle ilerler: site dosyalarını sunucuya yüklersin (Git/FTP/panel), alan adını DNS ile sunucunun IP'sine bağlarsın, HTTPS'i etkinleştirirsin ve her şeyi test edersin. Bu adımlar tamamlandığında, biri alan adını yazdığında siten güvenli biçimde açılır. Artık projen yerel makinenden çıkıp dünyaya açılmıştır.

**Alıştırma**

12 dk

Yayına al:

- 1 Bir siteyi yayına almanın dört adımını sırala.
- 2 Dosya taşımanın üç yöntemini ve birinin avantajını yaz.
- 3 Yayına almadan önce kontrol edeceğin üç şeyi listele.

**BÖLÜM 22**

# Dağıtım Süreci (Deployment)

Profesyonel projeler doğrudan canlıya kod atmaz. Bunun yerine kod, ortamlardan geçer: geliştirme, test (staging) ve üretim (production). Bu ayırım, hataların kullanıcıya ulaşmadan yakalanmasını sağlar.

## Ortamlar



Şema 22.1 — Dağıtım: geliştirme → test → üretim, güvenli geçişlerle.

- **Geliştirme:** serbestçe denersin, kırarsın, düzeltirsin.
- **Staging:** canlının kopyası; son provayı burada yaparsın.
- **Production:** gerçek kullanıcıların gördüğü ortam.

### İPUCU

Ortam ayırımı, "doğrudan canlıda denemek" gibi tehlikeli bir alışkanlıktan korur: bir değişikliği önce **staging**'de (üretimin birebir kopyası) test edersin, sorun yoksa **production**'a alırsın. Modern ekipler bunu **CI/CD** (sürekli entegrasyon/dağıtım) ile otomatikleştirir: kodu gönderdiğinde testler otomatik çalışır, geçerse otomatik yayınlanır, geçmezse durur. Bu, insan hatasını azaltır ve güvenle sık sık yayına almayı mümkün kılar. **Geri alma (rollback)** planın olsun: bir şey ters giderse, hızla önceki sağlam sürüme dönebilmelisin (Git tabanlı dağıtımın büyük avantajı). Üretim ve test ortamlarının veritabanı/sırlarını **ayrı tut** — test verisiyle gerçek veriyi asla karıştırma.

**Pratikte ne olur?****SONUÇ**

Kod önce senin makinende (geliştirme) yazılır, sonra canlının kopyası olan staging ortamında test edilir, sorun yoksa gerçek kullanıcıların bulunduğu üretim ortamına alınır. CI/CD ile bu geçişler otomatik ve test güvenceli olur. Sonuç: kullanıcılar yalnızca denenmiş, çalışan sürümü görür; hatalar onlara ulaşmadan yakalanır.

**Alıştırma**

10 dk

Süreci kur:

- 1 Üç ortamı (geliştirme, staging, production) ve görevlerini yaz.
- 2 Doğrudan canlıda denemenin neden riskli olduğunu açıkla.
- 3 "Geri alma (rollback)" neden önemli, kendi cümlele belirt.

**BÖLÜM 23**

# E-posta ve Alan Adı

Kendi alan adıyla profesyonel bir e-posta adresi (siz@ornek.com) kullanmak, hem güven verir hem markanı güçlendirir. Bunu sağlayan, alan adının MX kayıtları ve bir e-posta sağlayıcısıdır. Site ile e-posta birbirinden bağımsız çalışabilir.

## Alan adı e-postası

TÜR	AD	DEĞER / HEDEF
MX	@	mail sağlayıcısı
TXT	@	SPF (gönderen doğrulama)
TXT	dkim	DKIM (imza)
CNAME	mail	sağlayıcı adresi

Şema 23.1 — E-posta DNS kayıtları: MX yönlendirir, TXT güvenliği sağlar.

- **MX kaydı:** e-postanı hangi sunucunun alacağını söyler.
- **SPF/DKIM (TXT):** e-postanın sahte olmadığını kanıtlar.
- **Sağlayıcı:** profesyonel e-posta hizmeti (site hosting'inden ayrı olabilir).

### İPUCU

Profesyonel e-posta ( siz@ornek.com ), ücretsiz bir adresten (örn. ...@gmail.com) çok daha güvenilir görünür ve markanı taşır. Kurulumu, bir e-posta sağlayıcısı seçip alan adının **MX kaydını** ona yönlendirmekle başlar. **SPF** ve **DKIM** (TXT kayıtları) çok önemlidir: bunlar, senin adına gönderilen e-postaların gerçekten senden geldiğini kanıtlar ve **e-postalarının spam'e düşmesini** engeller. Bu kayıtlar eksikse, gönderdiğin e-postalar alıcının spam kutusuna gidebilir. Site hosting'i ile e-posta hizmetinin ayrı olabileceğini unutma: siten bir yerde, e-postan başka bir sağlayıcıda durabilir — DNS bunları birbirine bağlar.

### Pratikte ne olur?

### SONUÇ

Alan adına MX kaydı eklediğinde, o adrese (siz@ornek.com) gelen e-postalar belirttiğin sağlayıcıya yönelir; SPF ve DKIM kayıtları ise gönderdiğin e-postaların gerçekliğini kanıtlayarak spam'e düşmesini önler. Sonuç: markanı taşıyan, güvenilir ve düzgün teslim edilen profesyonel bir e-posta altyapısı. Site ve e-posta ayrı sağlayıcılarda olsa bile sorunsuz çalışır.

**Alıştırma**

10 dk

E-postayı bağla:

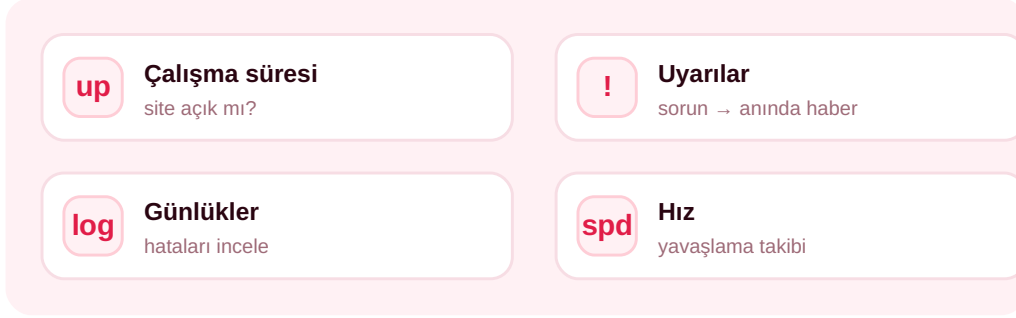
- 1 Profesyonel alan adı e-postasının avantajını yaz.
- 2 MX kaydının görevini kendi cümlele açıkla.
- 3 SPF/DKIM neden gerekli (spam ile ilişkisi), belirt.

## BÖLÜM 24

## İzleme ve Kesinti Yönetimi

Bir site yayına alınınca iş bitmez; yayında ve sağlıklı kalması gerekir. İzleme (monitoring), sitenin çalışıp çalışmadığını, hızını ve hatalarını sürekli gözler — ve bir sorun olduğunda, kullanıcılar fark etmeden seni uyarır.

## Neyi izlersin?



Şema 24.1 — İzleme: çalışma süresi, uyarılar, günlükler ve hız.

- **Uptime izleme:** site düşerse anında bildirim alırsın.
- **Günlükler (log):** hataların kaydı — sorun avının başlangıcı.
- **Durum sayfası:** kullanıcılara şeffaf bilgi.

## İPUCU

İzlemenin amacı, bir sorunu **kullanıcılardan önce** öğrenmektir. Bir **uptime izleme** servisi siteni belirli aralıklarla yoklar; yanıt vermezse sana anında (e-posta/SMS) haber verir — böylece gece yarısı çöken bir siteyi sabah müşteriden duymak yerine, anında müdahale edersin. **Günlükler (loglar)**, hatalar oluştuğunda ne olduğunu anlatır; düzenli incelemek, küçük sorunları büyümeden yakalamamı sağlar. Kesinti kaçınılmaz olduğunda, bir **durum sayfası** ile kullanıcıları şeffafça bilgilendirmek güveni korur. Önemli bir nokta: izleme verilerini **kişisel veri gizliliği** (KVKK) ilkelerine uygun toplamak gerekir — kullanıcıları gereğinden fazla izlememek hem etik hem yasal bir sorumluluktur.

## 🌐 Pratikte ne olur?

## SONUÇ

İzleme kurduğunda, bir servis siteni sürekli yoklar: çalışıyor mu, ne kadar hızlı, hata veriyor mu? Bir sorun belirdiğinde (site düştü, yavaşladı) sana anında haber verilir ve günlükler sorunun kaynağını gösterir. Sonuç: kesintiler kısalmı, sorunlar büyümeden çözülür ve kullanıcıların çoğu hiçbir aksaklık fark etmez.

**Alıştırma**

10 dk

Siteyi izle:

- 1 Uptime izlemenin neden önemli olduğunu bir senaryoyle açıkla.
- 2 Günlüklerin (log) ne işe yaradığını yaz.
- 3 İzleme yaparken hangi gizlilik ilkesine dikkat edilmeli, belirt.

**BÖLÜM 25**

# Maliyetler ve Ölçeklenme

Bir siteyi yayında tutmanın maliyeti birkaç kalemden oluşur: alan adı, hosting, e-posta ve (büyüdükçe) ek hizmetler. İyi haber: küçük başlayıp ihtiyaç doğdukça büyütebilirsin — baştan büyük harcamaya gerek yoktur.

## Maliyet kalemleri ve büyüme



Şema 25.1 — Maliyet: küçük başla, gerektiğe büyüt, sürekli optimize et.

- **Sabit kalemler:** alan adı (yıllık), hosting (aylık/yıllık).
- **Değişken kalemler:** bulut kaynakları, CDN, e-posta, depolama.
- **Gizli maliyetler:** yenileme zamları, aşım ücretleri, eklentiler.

### İPUCU

Bütçe planlarken birkaç noktaya dikkat: **giriş fiyatı ile yenileme fiyatı** farklı olabilir (alan adları ve hosting genelde ilk yıl indirimli, sonra zamlı yenilenir — toplam sahip olma maliyetine bak). Bulutun "kullandığın kadar öde" modeli esneklik ama **bütçe alarmı** kurmadan kullanmak risklidir; beklenmedik trafik veya yanlış ayar faturayı şişirebilir. Genel strateji: **küçük başla** (alan adı + paylaşımlı/statik hosting çoğu proje için aylık çok düşük maliyetlidir), **kullanımı izle** ve gerçekten ihtiyaç doğduğunda **yükselt**. Kullanmadığın kapasiteye ödeme yapmak en yaygın israftır. Maliyetleri periyodik gözden geçirip gereksiz hizmetleri kapatmak, sağlıklı bir alışkanlıktır.

**Pratikte ne olur?**

SONUÇ

Tipik bir başlangıç maliyeti düşüktür: bir alan adı (yıllık küçük bir ücret) ve bir paylaşımlı/statik hosting (aylık az bir tutar) çoğu projeyi yayına almaya yeter. Siten büyüdükçe — daha çok trafik, daha çok özellik — VPS, bulut veya CDN gibi kalemler eklenir ve maliyet ihtiyaca göre artar. Akıllı yaklaşım: küçük başla, kullanımı izle, yalnızca gerekeni ekle.

**Alıştırma**

10 dk

Bütçe planla:

- 1 Bir siteyi yayında tutmanın temel maliyet kalemlerini listele.
- 2 Giriş fiyatı ile yenileme fiyatı farkına neden dikkat etmeli, yaz.
- 3 "Küçük başla, gerektikçe büyüt" yaklaşımının avantajını açıkla.

**BÖLÜM 26**

# Bitirme: Yayına Alma Kontrol Listesi

Tüm modülü, bir siteyi güvenle yayına almak için kullanacağın pratik bir kontrol listesinde topluyoruz. Bu liste, "yayında ama eksik" ile "düzgün yayında" arasındaki farktır — her go-live öncesi tekrar tekrar kullanabilirsin.

## Go-live akışı



Şema 26.1 — Yayına alma kontrol listesi: dört aşamada güvenli go-live.

### Yayına alma kontrol listesi

- Alan adı doğru IP'ye işaret ediyor (A kaydı)
- www ve kök alan adı aynı yere gidiyor
- HTTPS açık, HTTP -> HTTPS yönlendirmesi var
- SSL sertifikası otomatik yenileniyor
- Sırlar/parolalar koda gömülü DEĞİL
- Veritabanı ve dosyalar yedekleniyor
- Hata sayfaları (404, 500) düzgün çalışıyor
- Formlar, e-posta ve ödeme test edildi
- Uptime izleme ve uyarılar aktif

**İPUCU**

Bu kontrol listesi, modülün tamamının özetidir; her yeni site yayına alırken baştan geç. Bu modülle birlikte FAZ 2'yi (sunucu tarafı: backend, veritabanı, diller ve şimdi altyapı) tamamladın — artık yalnızca kod yazmayı değil, o kodu **dünyaya güvenle açmayı** da biliyorsun. Önemli bir bütünsel kavrayış: bir site, birbirine bağlı bir **zincirdir** — alan adı, DNS, hosting, SSL, güvenlik ve izleme. Bir halka zayıfsa tüm sistem etkilenir. İyi haber: bu halkaların çoğu bir kez doğru kurulduğunda kendiliğinden çalışır; senin işin onları doğru kurmak ve göz kulak olmaktır. Sıradaki fazda mobil ve yayınlama gibi konularla bu temelin üstüne inşa etmeye devam edeceğiz.

**Pratikte ne olur?****SONUÇ**

Kontrol listesini baştan sona geçtiğinde, sitenin yalnızca "açık" değil, **doğru** yayında olduğundan emin olursun: alan adı doğru bağlı, HTTPS aktif, sırlar güvende, yedekler alınıyor, her şey test edilmiş ve izleniyor. Bu disiplin, yayına alma anının stresini azaltır ve sürpriz kesintileri önler. Artık bir fikri, dünyanın erişebileceği güvenli bir siteye dönüştürmenin tüm zincirini biliyorsun.

**Alıştırma**

20 dk

Kendi listeni hazırla:

- 1 Hayalî bir projeyi yayına alıyormuş gibi kontrol listesini doldur.
- 2 Her maddenin neden önemli olduğunu bir cümleyle yaz.
- 3 Alan adı-DNS-hosting-SSL zincirini kendi cümlelerinle özetle.
- 4 Yayına aldıktan sonra hangi izleme önlemlerini alacağını belirt.

**EK**

# Domain & Hosting Terimleri Sözlüğü

En sık kullanılan alan adı, DNS ve hosting terimleri. Bir başvuru kaynağı olarak saklayabilirsin.

Alan adı (domain)	İnternetteki adresin	TLD	.com, .org, .com.tr
Registrar	Alan adı satıcısı	DNS	İsim → IP çevirici
A kaydı	Alan adı → IP	CNAME	Ad → başka ad
MX kaydı	E-posta yönlendirme	TXT kaydı	Doğrulama / güvenlik
IP adresi	Sunucunun kimliği	Hosting	Sitenin barındığı yer
Uptime	Çalışma süresi oranı	HTTP / HTTPS	Protokol / şifreli protokol
SSL/TLS	HTTPS sertifikası	CDN	İçerik dağıtım ağı

## 🌐 Hepsi bir arada

**SONUÇ**

Bir site yayına almak için zincirin tüm halkaları birlikte çalışır: bir **alan adı** kiralarsın (registrar), onu **DNS kayıtlarıyla** (A, CNAME, MX) doğru yere yönlendirirsin, dosyalar bir **hosting** sunucusunda (IP adresiyle) durur, bağlantı **HTTPS** ile şifrelenir ve kullanıcı alan adını yazınca **istik** bu zinciri izleyerek sayfayı getirir. İleri seviyelerde hosting türlerini, SSL kurulumunu, performansı, güvenliği ve yayına alma sürecini ele alacağız — ama temel her zaman aynı: adres (domain) + ev (hosting) + güvenli yol (HTTPS).