HAFTALIK YAPAY ZEKA ÖZETİ

#4

Bu haftanın en önemli 6 gelişmesi

Kod asistanını video stüdyosuna çeviren bir araç, arayüzlerdeki yapay zeka görüntüsünü silen bir skill, 49 agent'lık bir oyun stüdyosu, her bot'a kendi bilgisayarını veren bir ekip, 1 milyon token bağlamlı yeni bir açık model ve herhangi bir siteyi yeniden kurduran bir araç. Hepsi tek yerde, sade ve doğrulanmış şekilde.

6	OpenMontage	5	Unslop
4	Claude Code Game Studios	3	OpenMausBot
2	Kimi K3.1 Pro	1	Craft Anything

Ön koşullar ve erişim

Bu haftaki araçların hepsi ücretsiz ve açık kaynak. Asıl fark, arkasında hangi agent'ın çalıştığı ve hangi servislerin ücretli olduğu. Denemeye başlamadan önce sana uygun olanı bu tablodan seç.

Gelişme	Ne gerekiyor?	Nerede çalışır?
OpenMontage	Ücretsiz (AGPLv3). API anahtarı olmadan da çalışıyor, bulut görsel ve video üreticileri ücretli.	Python 3.10+, FFmpeg, Node.js ve bir kod asistanı (Claude Code, Cursor vb.).
Unslop	Ücretsiz, CC BY 4.0 lisanslı.	Claude Code, Codex veya Google Antigravity.
Claude Code Game Studios	Ücretsiz, MIT lisanslı. Claude Code kullanımı kendi planından düşer.	Git, Claude Code, Python 3, Bash ve Chrome ya da Edge.
OpenMausBot	Uygulama ücretsiz (Apache 2.0). Bulut bilgisayar ve bağlı servisler ayrı ücretli olabilir.	macOS, Windows ve Ubuntu 24.04. Giriş yapılmış Claude Code, Codex ya da Grok CLI.
Kimi K3.1 Pro	Ağırlıklar açık, ticari kullanım serbest (çok büyük ürünlere şart var). Ollama bulutunda ücretli.	API ya da Ollama bulutu. Kendi sunucunda çalıştırmak veri merkezi düzeyinde donanım ister.
Craft Anything	Ücretsiz, MIT veya Apache 2.0. Her build kendi Claude Code kullanımından düşer.	Windows, macOS, Linux ve FreeBSD. Giriş yapılmış Claude Code.

Ücretsiz araçlar da ücretli servislere bağlanabiliyor. Bir modeli ya da üreticiyi API üzerinden kullanacaksan güncel fiyatı kendi hesabında kontrol et.

Terimler

- **AI agent (ajan):** Sadece cevap vermeyen, araç kullanıp adım adım iş yapan yapay zeka.
- **Skill:** Bir agent'a belirli bir işi nasıl yapacağını anlatan, tekrar kullanılabilir talimat paketi.
- **Hook:** Belirli bir anda (commit, dosya yazma vb.) kendiliğinden çalışan kontrol betiği.
- **Açık ağırlık:** Modelin kendisi indirilebilir. Kendi sunucunda çalıştırabilir, üzerine eğitim yapabilirsin.
- **MoE (uzman karışımı):** Her token'da modelin sadece bir kısmının çalıştığı mimari. Toplam parametre büyük, çalışan kısım küçük.
- **Context (bağlam) penceresi:** Modelin tek seferde okuyabildiği metin miktarı. Token cinsinden ölçülür.

6 numara

OpenMontage: kod asistanını video stüdyosuna çevir

Claude Code, Cursor, Copilot, Windsurf ve Codex ile çalışan açık kaynak bir video üretim sistemi. Videoyu tarif ediyorsun; araştırmayı, senaryoyu, kurguyu ve render'ı agent yapıyor.

Ne işe yarıyor?

Ne istediğini düz bir cümleyle yazıyorsun, örneğin "nöral ağların nasıl öğrendiğini anlatan 60 saniyelik animasyonlu bir video". Agent önce internette araştırma yapıyor, senaryoyu yazıyor, görselleri ve sesi hazırlıyor, sonra Remotion ve FFmpeg ile videoyu birleştiriyor. Animasyonlu anlatım, konuşan kafa, belgesel kurgusu, podcast'ten kısa video ve dublaj gibi hazır üretim hatları var.

Öne çıkanlar

- **Onay noktaları:** Her yaratıcı kararda senin onayını bekliyor.
- **Anahtarsız başlangıç:** Ücretli API anahtarı olmadan da video çıkarabiliyorsun: Piper ile seslendirme, Archive.org, NASA ve Wikimedia görüntüleri, Remotion ve FFmpeg ile render.
- **Kendini kontrol ediyor:** Render'dan sonra kareleri, ses seviyesini ve altyazıları kendisi denetliyor.
- **Canlı storyboard:** Backlot adlı yerel panoda üretimi adım adım izleyebiliyorsun.

Nasıl kurulur?

```
git clone https://github.com/calesthio/OpenMontage.git
cd OpenMontage
make setup
```

Sonra klasörü Claude Code ya da Cursor'da aç ve videonu tarif et. Windows'ta `make` yoksa README'deki PowerShell kurulum satırını kullan.

Ön koşullar

Gereksinim	Detay
Ücret	Ücretsiz, AGPLv3 lisanslı. API anahtarlarının hepsi isteğe bağlı. Bulut görsel ve video üreticileri ücretli; README'deki örnek videonun toplam maliyeti 1,33 dolar.
Yazılım	Python 3.10+, FFmpeg ve Node.js (22.x, 24.x ya da 26.x serisinin güncel sürümü).
GPU	Sadece videoyu kendi bilgisayarında ürettirmek istersen gerekli.

Dene: github.com/calesthio/OpenMontage

5 numara

Unslop: arayüzlerdeki yapay zeka görüntüsünü temizleyen skill

Claude Code, Codex ve Google Antigravity için ücretsiz bir skill. Yapay zekayla yapılmış gibi duran ekranları önce teşhis ediyor, sonra adım adım düzeltiyor.

Neden önemli?

Yapay zekaya arayüz yaptırınca tanıdık bir görüntü çıkıyor: mor gradyanlar, parlayan kenarlar, gradyanlı yazılar, emojili etiketler, ünlemli metinler ve dengesiz boşluklar. Unslop bu kalıpları tanıyor ve ekranı bir tasarımcının elinden çıkmış gibi yeniden düzenliyor. Skill'i tasarımcı Damir Sokolovsky hazırladı.

Öne çıkanlar

- **Önce teşhis:** Ekranı inceliyor, beş satırlık bir yön öneriyor ve "direkt yap" demedikçe senin cevabını bekliyor.
- **Dokuz geçiş:** Boşluk, renk, ikonlar, metin, yazı tipi, yerleşim, detaylar, görseller ve hareket sırayla elden geçiyor.
- **Her yerde çalışıyor:** Next.js, React, Vue, Svelte, Tailwind ya da düz HTML/CSS. Canlı bir site adresi de verebilirsin.
- **Gereksiz yere bozmuyor:** Ekran zaten iyiyse bunu söylüyor ve sadece küçük düzeltmeler öneriyor.

Nasıl kurulur?

Claude Code için terminalde şu iki satırı çalıştır (Windows'ta Git Bash ile):

```
git clone https://github.com/heydamir/unslop
mkdir -p ~/.claude/skills && cp -r unslop/skills/unslop ~/.claude/skills/
```

Sonra Claude Code'da `/unslop` yazıp hangi ekranı düzelteceğini söyle, örneğin "/unslop bu dashboard". Codex'te aynı işi `$unslop` ile yapıyorsun.

Ön koşullar

Gereksinim	Detay
Ücret	Ücretsiz, CC BY 4.0 lisanslı. Paylaşırken ya da değiştirirken kaynak göstermen gerekiyor.
Nerede	Claude Code, Codex ve Google Antigravity.
İsteğe bağlı	Canlı site görüntüsü için Playwright, kısa tanıtım videosu için ffmpeg.

> Unslop uzun yazı temizleyicisi değil: düzelttiği metinler başlık, buton ve etiket gibi arayüz metinleri. Bazı işlerde kendi Node betiklerini çalıştırıyor, kurmadan önce göz at.

Dene: github.com/heydamir/unslop

4 numara

Claude Code Game Studios: 49 agent'lık oyun stüdyosu

Claude Code'u bir oyun stüdyosuna çeviren ücretsiz, açık kaynak bir şablon. Oyununu tarif ediyorsun, tarayıcıda oynanabilir ilk sürümü kuruyor.

Nasıl çalışıyor?

Repoyu klonlayıp Claude Code'u o klasörde açıyorsun. `/prototype` ile oyun fikrini yazınca Claude Code birkaç soru soruyor, tek sayfalık bir oyun özeti çıkarıyor ve tarayıcıda oynanabilir bir prototip kuruyor. Sonra istediğini değiştirtiyor, paylaşıyor ya da onu büyük bir web projesine dönüştürüyorsun. Daha ağır "stüdyo modunda" ise 49 agent'lık ekip, tasarım dokümanları ve sprint'ler devreye giriyor.

Öne çıkanlar

- **Stüdyo hiyerarşisi:** En üstte 3 direktör (yaratıcı, teknik, yapımcı), altında 8 departman lideri, onların altında programcı, tasarımcı, ses ve test uzmanları.
- **Hazır iş akışları:** 75 skill, 15 hook ve 14 kural. Hook'lar commit'leri, asset'leri ve tasarım dokümanlarını kendiliğinden kontrol ediyor.
- **Oyunu kendisi açıyor:** Prototipi tarayıcıda çalıştırıp ekran görüntüleriyle kontrol ediyor.
- **Motor seçimi:** Varsayılan web. Godot 4, Unity ve Unreal Engine 5 için de uzman agent'lar var.

Nasıl kurulur?

```
git clone https://github.com/Donchitos/Claude-Code-Game-Studios.git my-game
cd my-game
claude
```

Claude Code açılınca `/prototype` yaz ve oyun fikrini ekle, örneğin "/prototype nadir balıkların sadece gün batımında oltaya geldiği bir balık tutma oyunu". Claude Code'u alt klasörde değil projenin kök klasöründe başlat; yoksa hook'lar ve izin kuralları çalışmıyor.

Ön koşullar

Gereksinim	Detay
Ücret	Ücretsiz, MIT lisanslı. Claude Code kullanımı kendi planından düşer.
Yazılım	Git, Claude Code, Python 3 ve Bash (Windows'ta Git for Windows ile geliyor). Tarayıcı kontrolleri için Chrome, Edge ya da Chromium.
İsteğe bağlı	Oyunu web projesine büyütmek için Node.js.

> Geliştirici Godot, Unity ve Unreal yollarının henüz gerçek bir oyuncuyla test edilmediğini söylüyor. Varsayılan "minimal" mod bilerek seçilmiş: ağır modda ilk kod satırından önce 30 doküman çıkıyor.

Dene: github.com/Donchitos/Claude-Code-Game-Studios

3 numara

OpenMausBot: her bot'un kendi bilgisayarı olan açık kaynak ekip

Sohbet uygulaması gibi çalışan açık kaynak bir AI bot kadrosu. Her bot'un kendi kişiliği, modeli, bilgisayarı ve bağlı uygulamaları var.

Nasıl çalışıyor?

Kenar çubuğundaki her bot aslında bir agent: arka planda senin bilgisayarında kurulu Claude Code, Codex ya da Grok CLI, mevcut girişlerinle çalışıyor. Her bot'a bir bulut Linux masaüstü, izole bir yerel sanal makine ya da açıkça izin verirsen kendi bilgisayarını verebiliyorsun. Botları grup sohbetlerinde toplayıp ortak bir klasörde birlikte çalıştırabiliyorsun. Proje kendini Grok Bot'un açık kaynak alternatifi olarak tanıtıyor.

Öne çıkanlar

- **Önce soruyor:** Komut çalıştırma ve dosya değiştirme gibi işlemler sohbette İzin ver / Reddet kartı olarak önüne geliyor.
- **Bot başına model:** Her bot için ayrı model seçip konuşmanın ortasında değiştirebiliyorsun.
- **Canlı ekran:** Bot'un bilgisayarını canlı izleyip istediğin an kontrolü alabiliyorsun.
- **500+ uygulama:** Gmail, Slack, GitHub, Notion gibi uygulamaları bir kez bağlıyorsun, botlar onları araç olarak kullanıyor.

Nasıl kurulur?

En kolayı GitHub'daki Releases bölümünden kurulum dosyasını indirmek: macOS (.dmg), Windows (.exe) ve Ubuntu 24.04 (.deb). Node.js kuruluysa npm ile de başlatabilirsin:

```
npm install -g openmausbot
openmausbot
```

Başlamadan önce Claude Code, Codex ya da Grok CLI'dan en az birinin kurulu ve giriş yapılmış olması gerekiyor.

Ön koşullar

Gereksinim	Detay
Ücret	Uygulama ücretsiz (Apache 2.0). Bulut bilgisayarlar, bağlı uygulamalar ve ses servisleri kendi hesapları ve fiyatlarıyla geliyor.
Platform	macOS, Windows x64 ve Ubuntu 24.04 (Ubuntu desteği beta).
Agent	Kurulu ve giriş yapılmış claude, codex ya da grok CLI.

> Geliştirici projeyi "erken ama gerçek" diye tanımlıyor. Kripto para ya da token ile bağlantısı olmadığını da özellikle belirtiyor.

Dene: github.com/milind-soni/OpenMausBot

2 numara

Kimi K3.1 Pro: 1 milyon token bağlamlı yeni açık model

Moonshot AI, K3.1 serisinin ilk modelinin ağırlıklarını Hugging Face'te yayınladı. Metin ve görsel anlıyor; kodlama ve agent işlerinde en güçlü kapalı modellerle yarışıyor.

Ne oldu?

Kimi K3.1 Pro, Kimi K3 üzerine eğitilmiş 2,8 trilyon parametreli bir MoE model. Her token'da bunun sadece 104 milyarı çalışıyor. 1 milyon token bağlam penceresi var, yani çok büyük bir kod tabanını ya da uzun belgeleri tek seferde okuyabiliyor.

Öne çıkanlar

- **Açık ağırlık:** Modelin kendisi indirilebiliyor. Kendi lisansıyla geliyor ve sıradan ticari kullanım serbest.
- **Lisans şartı:** Çok büyük ürünler (aylık 100 milyon kullanıcı ya da 20 milyon dolar gelir üstü) arayüzde "Kimi K3.1-Pro" adını göstermek zorunda.
- **Düşünme ayarı:** Düşünme seviyesini low, high ya da max olarak seçebiliyorsun.

Moonshot'un paylaştığı test sonuçları

Test	Kimi K3.1 Pro	Claude Opus 5.5
DeepSWE v1.1	**75.5**	74.2
HLE-Full (araçlarla)	**70.0**	67.7
Terminal-Bench 4.0	65.7	**66.4**
OSWorld V2.1	79.6	**81.8**

Kısacası: kodlama ve araç kullanımında Claude Opus 5.5 ile başa baş. DeepSWE ve HLE'de önde, Terminal-Bench ve OSWorld'de az farkla geride.

Nasıl denersin?

Kendi bilgisayarında çalıştırmak gerçekçi değil, model veri merkezi düzeyinde donanım istiyor. En kolay yol Ollama'nın bulut sürümü:

```
ollama run kimi-k3.1-pro:cloud
```

Bu sürüm Ollama'nın sunucularında çalışıyor ve ücretli: 1 milyon girdi token'ı 2 dolar, çıktı 10 dolar. Moonshot'un kendi API'sinde (platform.kimi.ai) model adı `kimi-k3.1-pro`; OpenAI ve Anthropic uyumlu.

> **Agent Arena sıralaması:** Arena'nın kendi paylaşımında açık modeller arasında 1. sırada Kimi K3 Max görünüyordu. K3.1 Pro'nun yerini birincil kaynakta doğrulayamadık; yukarıdaki skorlar Moonshot'un kendi testleri.

Dene: huggingface.co/moonshotai/Kimi-K3.1-Pro · ollama.com/library/kimi-k3.1-pro

1 numara

Craft Anything: herhangi bir siteyi, uygulamayı ya da oyunu yeniden kurdur

Bir hedef gösteriyorsun, ne istediğini söylüyorsun; Craft Anything onu inceliyor, Claude Code'a yeni bir proje olarak kurduruyor ve sonucu orijinalle karşılaştırıyor.

Nasıl çalışıyor?

- **1. İnceliyor:** Orijinali her ekran genişliğinde açıyor, nasıl çalıştığına bakıyor.
- **2. Kuruyor:** Agent, istediğin dil, framework ya da oyun motoruyla yeni proje kuruyor.
- **3. Karşılaştırıyor:** Yeni sürümü orijinalle karşılaştırıp henüz tutmayan yerleri listeliyor.
- **4. Hazır:** Sonucu açıp deniyorsun ya da değişiklik isteyip geliştirtiyorsun.

Öne çıkanlar

- **Ne olursa:** Web siteleri, masaüstü ve telefon uygulamaları, oyunlar, hatta dosyalar ve veritabanları.
- **İki mod:** Varsayılan "clean room" modu orijinalin içini okumuyor, sadece dışarıdan inceliyor. "Deep" mod kodu ve dosyaları da okuyor ve yalnız kişisel kullanım için.
- **Claude Code öncelikli:** Senin kendi, değiştirilmemiş Claude Code'unu kendi hesabınla çalıştırıyor. Copilot CLI, Grok ve başka agent'lar için de bağlantılar var.

Nasıl kurulur?

Hazır kurulum dosyaları GitHub'daki Releases bölümünde: Windows (.msi), macOS (.dmg) ve Linux. Kaynak koddan kurmak istersen Rust 1.97 ya da üstü gerekiyor:

```
git clone https://github.com/storytold/craft-anything && cd craft-anything
cargo run --release -p craft-anything # masaüstü uygulaması
cargo install --path apps/craft-anything-cli # komut satırı aracı
craft-anything-cli doctor # kurulumu kontrol eder
```

Ön koşullar

Gereksinim	Detay
Ücret	Ücretsiz, MIT veya Apache 2.0. Her build kendi Claude Code planından düşer, kullanım limitlerin geçerli.
Platform	Windows, macOS, Linux ve FreeBSD.
Agent	Kurulu ve giriş yapılmış Claude Code (`claude auth login`).

> Proje erken alfa aşamasında. Geliştiriciler aracı yalnız sahibi olduğun ya da incelemeye iznin olan yazılımlarda kullanmanı istiyor. DRM, giriş ve anti-cheat korumalarını aşmıyor; logo, fotoğraf ve metinleri izin olmadan kopyalamıyor.

Dene: github.com/storytold/craft-anything

Bu haftanın büyük resmi

Altı gelişmeyi yan yana koyunca üç net eğilim görünüyor.

1. Kod asistanı her işin stüdyosu oluyor

OpenMontage video, Claude Code Game Studios oyun, Craft Anything uygulama üretimini aynı kod asistanının içine taşıyor. Unslop gibi skill'ler de çıkan işin "yapay zeka yapmış" gibi görünmesini engelliyor.

2. Agent'lar kendi bilgisayarına kavuşuyor, onay sende kalıyor

OpenMausBot'ta her bot'un kendi bilgisayarı var ama riskli işlemler onay kartıyla önüne geliyor. OpenMontage her yaratıcı kararda, Unslop ise düzeltmeye başlamadan önce senin cevabını bekliyor.

3. Açık modeller kapalılarla başa baş

Kimi K3.1 Pro kendi testlerinde Claude Opus 5.5 ile yarışıyor. Yine de 2,8 trilyon parametreyle evde çalışmıyor; açıklık şimdilik daha çok fiyat ve sağlayıcı seçme özgürlüğü getiriyor.