

Fasheone Mobile API - Kullanım Kılavuzu

Bu doküman, mobil uygulama geliştiricileri için Fasheone API'sinin nasıl kullanılacağını detaylı olarak açıklar.

İçindekiler

- [Hızlı Başlangıç](#)
- [Kimlik Doğrulama](#)
- [Kullanıcı İşlemleri](#)
- [AI Görsel Oluşturma](#)
- [Ödeme Entegrasyonu](#)
- [Hata Yönetimi](#)
- [SDK Örnekleri](#)

Hızlı Başlangıç

Base URL

Production: <https://api.fasheone.com/api/v1>
Development: <http://localhost:3002/api/v1>

Health Check

```
curl https://api.fasheone.com/api/v1/health
```

Yanıt:

```
{  
  "status": "OK",  
  "version": "1.0.0",  
  "services": {  
    "supabase": true,  
    "gemini": true  
  }  
}
```

Kimlik Doğrulama

1. Kayıt Olma

```
curl -X POST https://api.fasheone.com/api/v1/auth/signup \
-H "Content-Type: application/json" \
-d '{
  "email": "user@example.com",
  "password": "guvenli123",
  "fullName": "Ahmet Yılmaz"
}'
```

Yanıt:

```
{
  "success": true,
  "user": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "email": "user@example.com"
  }
}
```

Yeni kullanıcılara otomatik olarak **5 ücretsiz kredi** verilir.

2. Giriş Yapma

```
curl -X POST https://api.fasheone.com/api/v1/auth/login \
-H "Content-Type: application/json" \
-d '{
  "email": "user@example.com",
  "password": "guvenli123"
}'
```

Yanıt:

```
{
  "success": true,
  "session": {
    "access_token": "eyJhbGciOiJIUzI1NiIs... ",
    "refresh_token": "v1.MjNiNzI4... ",
    "expires_at": 1706000000
  },
  "user": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "email": "user@example.com",
    "fullName": "Ahmet Yılmaz"
  },
  "profile": {
    "credits": 5,
    "subscriptionTier": "free",
    "isAdmin": false
  }
}
```

⚠ access_token değerini saklayın! Tüm korumalı isteklerde kullanacaksınız.

3. Token Kullanımı

Korumalı endpointlere istek yaparken Authorization header'ı ekleyin:

```
curl -X GET https://api.fasheone.com/api/v1/user/profile \
-H "Authorization: Bearer eyJhbGciOiJIUzI1NiIs..."
```

4. Token Yenileme

Access token süresi dolduğunda (1 saat) refresh token ile yenileyin:

```
curl -X POST https://api.fasheone.com/api/v1/auth/refresh \
-H "Content-Type: application/json" \
-d '{
  "refreshToken": "v1.MjNiNzI4..."
}'
```

Kullanıcı İşlemleri

Profil Bilgisi Alma

```
curl -X GET https://api.fasheone.com/api/v1/user/profile \
-H "Authorization: Bearer TOKEN"
```

Yanıt:

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "email": "user@example.com",
  "fullName": "Ahmet Yılmaz",
  "credits": 45,
  "subscriptionTier": "pro",
  "isAdmin": false
}
```

Üretim Geçmişi

```
curl -X GET "https://api.fasheone.com/api/v1/user/generations?limit=10&offset=0" \
-H "Authorization: Bearer TOKEN"
```

Yanıt:

```
{
  "generations": [
    {
      "id": "uuid",
      "type": "product_to_model",
      "credits_used": 1,
      "output_image_url": "data:image/png;base64,...",
      "created_at": "2024-01-20T10:30:00Z"
    }
  ],
  "total": 25,
  "limit": 10,
  "offset": 0
}
```

AI Görsel Oluşturma

Kredi Maliyetleri

İşlem	Kredi
Eskiz → Ürün	1
Ürün → Canlı Model	1
Video Oluşturma	3
Teknik Çizim	1
Pixshop Düzenleme	1
Fotomatik Transform	1
Fotomatik Describe	1
AdGenius Görsel	1
AdGenius Video	2
Kolaj	2

1. Eskiz → Ürün Dönüşümü

Moda eskizinizi fotorealistik ürün görseline dönüştürün.

```
curl -X POST https://api.fasheone.com/api/v1/generation/sketch-to-product \
-H "Authorization: Bearer TOKEN" \
-F "image=@eskiz.png" \
-F "color=Kırmızı"
```

Yanıt:

```
{
  "success": true,
  "image": "...",
  "creditsRemaining": 44
}
```

image alanındaki base64 veriyi doğrudan olarak kullanabilirsiniz.

2. Ürün → Canlı Model

Ürün görselinizi bir model üzerinde gösterin.

```
curl -X POST https://api.fasheone.com/api/v1/generation/product-to-model \
-H "Authorization: Bearer TOKEN" \
-F "image=@urun.jpg" \
-F "gender=Female" \
-F "ethnicity=Türk" \
-F "bodyType=Standart" \
-F "pose=Ayakta" \
-F "location=Stüdyo" \
-F "lighting=Doğal" \
-F "aspectRatio=3:4"
```

Tüm Parametreler:

Parametre	Değerler	Varsayılan
clothingType	Üst, Alt, Elbise, Dış Giyim, Alt & Üst, Takım Elbise	-
gender	Female, Male	Female
ethnicity	Farklı, Türk, Avrupalı, İskandinav, Akdeniz, Asyalı, Afrikalı, Latin, Orta Doğulu	Farklı
ageRange	Child, Teen, Adult, Elderly	Adult
bodyType	Standart, İnce, Kıvrımlı, Atletik, Büyük Beden, Battal Beden	Standart
pose	Rastgele, Ayakta, Yürürken, Eller Belde, Otururken	Rastgele
hairColor	Doğal, Sarı, Kumral, Siyah, Kızıl, Gri, Pastel Pembe	Doğal
hairStyle	Doğal, Uzun Düz, Uzun Dalgalı, Kısa Küt, Kısa Pixie, Topuz, At Kuyruğu, Kıvrıkcık	Doğal
location	Podyum, Stüdyo, Sokak, Doğal Mekan, Lüks Mağaza	Stüdyo
lighting	Doğal, Stüdyo, Gün Batımı, Dramatik, Neon	Doğal
cameraAngle	Normal, Alt Açı, Üst Açı, Geniş Açı, Portre	Normal
aspectRatio	3:4, 9:16, 4:5, 1:1, 16:9	3:4
customPrompt	Özel metin	-

3. Video Oluşturma

Statik görselden video oluşturun.

```
curl -X POST https://api.fasheone.com/api/v1/generation/video \
-H "Authorization: Bearer TOKEN" \
-F "image=@model.png" \
-F "prompt=Model yavaşça yürüyor" \
-F "duration=5" \
-F "aspectRatio=9:16"
```

4. Teknik Çizim

Ürün fotoğrafından profesyonel teknik çizim oluşturun.

```
curl -X POST https://api.fasheone.com/api/v1/generation/tech-sketch \  
-H "Authorization: Bearer TOKEN" \  
-F "image=@urun.jpg" \  
-F "style=blackwhite" # veya "colored"
```

5. Pixshop - Fotoğraf Düzenleme

AI ile görseli düzenleyin.

```
curl -X POST https://api.fasheone.com/api/v1/generation/pixshop/edit \  
-H "Authorization: Bearer TOKEN" \  
-F "image=@foto.jpg" \  
-F "prompt=Arka planı mavi yap"
```

6. Pixshop - Arka Plan Kaldırma

```
curl -X POST https://api.fasheone.com/api/v1/generation/pixshop/remove-bg \  
-H "Authorization: Bearer TOKEN" \  
-F "image=@foto.jpg"
```

7. Fotomatik - Görüntü Dönüştürme

```
curl -X POST https://api.fasheone.com/api/v1/generation/fotomatik/transform \  
-H "Authorization: Bearer TOKEN" \  
-F "image=@foto.jpg" \  
-F "prompt=Anime stiline dönüştür" \  
-F "aspectRatio=1:1"
```

8. Fotomatik - Prompt Üretme

Görselden Midjourney ve Stable Diffusion promptları üretin.

```
curl -X POST https://api.fasheone.com/api/v1/generation/fotomatik/describe \  
-H "Authorization: Bearer TOKEN" \  
-F "image=@foto.jpg"
```

Yanıt:

```
{
  "success": true,
  "prompts": {
    "tr": "Profesyonel moda fotoğrafı, kadın model...",
    "en": "Professional fashion photography, female model...",
    "midjourney": "fashion photography, female model wearing elegant dress --ar 3:4 --stylize 500",
    "stableDiffusion": {
      "positive": "masterpiece, best quality, fashion photography...",
      "negative": "low quality, blurry, distorted...",
      "params": "Steps: 25, CFG: 7, Sampler: DPM++ 2M Karras"
    },
    "tips": [
      "Daha iyi sonuç için 85mm lens kullanın",
      "Golden hour ışığı deneyin",
      "Kontrastı artırın"
    ]
  }
}
```

9. AdGenius - Ürün Analizi (ÜCRETSİZ)

```
curl -X POST https://api.fasheone.com/api/v1/generation/adgenius/analyze \
-H "Authorization: Bearer TOKEN" \
-F "image=@urun.jpg"
```

Yanıt:

```
{
  "success": true,
  "analysis": {
    "productType": "Elbise",
    "name": "Zarif Yazlık Maxi Elbise",
    "description": "Çiçek desenli, V yaka, uzun yazlık elbise",
    "features": ["Çiçek deseni", "V yaka", "Bel korsajlı"],
    "targetAudience": "25-45 yaş kadın, romantik stil seven",
    "suggestedStyles": ["Lifestyle", "Beach", "Romantic"],
    "colors": ["Beyaz", "Pembe", "Yeşil"]
  }
}
```

10. AdGenius - Reklam Görseli

```
curl -X POST https://api.fasheone.com/api/v1/generation/adgenius/image \
-H "Authorization: Bearer TOKEN" \
-F "image=@urun.jpg" \
-F "style=Lifestyle" \
-F "prompt=Sahil kenarında romantik görünüm" \
-F "aspectRatio=1:1"
```

11. Kolaj Oluřturma

2-6 gorseli birleřtirin.

```
curl -X POST https://api.fasheone.com/api/v1/generation/collage \
-H "Authorization: Bearer TOKEN" \
-F "images=@gorsel1.jpg" \
-F "images=@gorsel2.jpg" \
-F "images=@gorsel3.jpg" \
-F "prompt=Grid layout, minimalist" \
-F "aspectRatio=16:9"
```

Ödeme Entegrasyonu

Kredi Paketlerini Görüntüleme

```
curl https://api.fasheone.com/api/v1/payment/packages
```

Yanıt:

```
{
  "packages": [
    { "id": "small", "credits": 50, "priceTL": 250 },
    { "id": "medium", "credits": 100, "priceTL": 500 },
    { "id": "large", "credits": 200, "priceTL": 1000 }
  ],
  "subscriptions": [
    { "id": "starter", "name": "Starter", "credits": 100, "priceTL": 500, "monthly": true },
    { "id": "pro", "name": "Pro", "credits": 500, "priceTL": 2250, "monthly": true },
    { "id": "premium", "name": "Premium", "credits": 1000, "priceTL": 4000, "monthly": true }
  ]
}
```

Stripe ile Ödeme Başlatma

```
curl -X POST https://api.fasheone.com/api/v1/payment/create-intent \
-H "Authorization: Bearer TOKEN" \
-H "Content-Type: application/json" \
-d '{
  "packageId": "medium",
  "amountTL": 500,
  "credits": 100
}'
```

Yanıt:


```
{
  "clientSecret": "pi_3NkFe2Hk..._secret...",
  "amountEUR": "14.12",
  "rate": 35.42
}
```

clientSecret değerini Stripe SDK ile kullanarak ödeme formunu gösterin.

⚠ Hata Yönetimi

Hata Formatı

Tüm hatalar aşağıdaki formatta döner:

```
{
  "error": "Hata açıklaması"
}
```

HTTP Status Kodları

Kod	Açıklama
200	Başarılı
400	Geçersiz istek (eksik parametre, yetersiz kredi)
401	Yetkilendirme hatası (geçersiz/süresi dolmuş token)
403	Erişim reddedildi (admin yetkisi gerekli)
500	Sunucu hatası

Yaygın Hatalar

```
// Yetersiz kredi
{ "error": "Insufficient credits" }

// Geçersiz token
{ "error": "Invalid or expired token" }

// Görsel yok
{ "error": "No image generated" }

// API yapılandırma hatası
{ "error": "Gemini API not configured" }
```

SDK Örnekleri

React Native (Axios)

```

import axios from 'axios';
import AsyncStorage from '@react-native-async-storage/async-storage';

const API_BASE = 'https://api.fasheone.com/api/v1';

// API Client
const api = axios.create({
  baseURL: API_BASE,
  timeout: 60000, // AI işlemleri için 60 saniye
});

// Token interceptor
api.interceptors.request.use(async (config) => {
  const token = await AsyncStorage.getItem('accessToken');
  if (token) {
    config.headers.Authorization = `Bearer ${token}`;
  }
  return config;
});

// Giriş
export const login = async (email, password) => {
  const res = await api.post('/auth/login', { email, password });
  await AsyncStorage.setItem('accessToken', res.data.session.access_token);
  await AsyncStorage.setItem('refreshToken', res.data.session.refresh_token);
  return res.data;
};

// Canlı Model Oluşturma
export const generateLiveModel = async (imageUri, options) => {
  const formData = new FormData();
  formData.append('image', {
    uri: imageUri,
    type: 'image/jpeg',
    name: 'product.jpg',
  });

  Object.keys(options).forEach(key => {
    formData.append(key, options[key]);
  });

  const res = await api.post('/generation/product-to-model', formData, {
    headers: { 'Content-Type': 'multipart/form-data' },
  });

  return res.data.image; // Base64 data URL
};

```

Flutter (Dio)

```

import 'package:dio/dio.dart';
import 'package:shared_preferences/shared_preferences.dart';

class FasheoneAPI {
  static const String baseUrl = 'https://api.fasheone.com/api/v1';
  late Dio _dio;

  FasheoneAPI() {
    _dio = Dio(BaseOptions(
      baseUrl: baseUrl,
      connectTimeout: Duration(seconds: 60),
    ));

    _dio.interceptors.add(InterceptorsWrapper(
      onRequest: (options, handler) async {
        final prefs = await SharedPreferences.getInstance();
        final token = prefs.getString('accessToken');
        if (token != null) {
          options.headers['Authorization'] = 'Bearer $token';
        }
        return handler.next(options);
      },
    ));
  }

  // Giriş
  Future<Map<String, dynamic>> login(String email, String password) async {
    final response = await _dio.post('/auth/login', data: {
      'email': email,
      'password': password,
    });

    final prefs = await SharedPreferences.getInstance();
    await prefs.setString('accessToken', response.data['session']['access_token']);

    return response.data;
  }

  // Canlı Model Oluşturma
  Future<String> generateLiveModel(String imagePath, Map<String, dynamic> options) async {
    final formData = FormData.fromMap({
      'image': await MultipartFile.fromFile(imagePath),
      ...options,
    });

    final response = await _dio.post('/generation/product-to-model', data: formData);
    return response.data['image']; // Base64 data URL
  }
}

```

Swift (iOS Native)

import Foundation

```
class FasheoneAPI {
    static let shared = FasheoneAPI()
    private let baseURL = "https://api.fasheone.com/api/v1"

    var accessToken: String? {
        get { UserDefaults.standard.string(forKey: "accessToken") }
        set { UserDefaults.standard.set(newValue, forKey: "accessToken") }
    }

    // Login
    func login(email: String, password: String) async throws -> LoginResponse {
        let url = URL(string: "\baseURL/auth/login")!
        var request = URLRequest(url: url)
        request.httpMethod = "POST"
        request.setValue("application/json", forHTTPHeaderField: "Content-Type")
        request.httpBody = try JSONEncoder().encode(["email": email, "password": password])

        let (data, _) = try await URLSession.shared.data(for: request)
        let response = try JSONDecoder().decode(LoginResponse.self, from: data)

        self.accessToken = response.session.accessToken
        return response
    }

    // Generate Live Model
    func generateLiveModel(imageData: Data, options: [String: String]) async throws -> String {
        let url = URL(string: "\baseURL/generation/product-to-model")!
        var request = URLRequest(url: url)
        request.httpMethod = "POST"
        request.setValue("Bearer \accessToken ?? """, forHTTPHeaderField: "Authorization")

        let boundary = UUID().uuidString
        request.setValue("multipart/form-data; boundary=\boundary", forHTTPHeaderField: "Content-Type")

        var body = Data()
        // Add image
        body.append("--\boundary\r\n".data(using: .utf8)!)
        body.append("Content-Disposition: form-data; name=\"image\"; filename=\"image.jpg\"\r\n".data(using:
        .utf8)!)
        body.append("Content-Type: image/jpeg\r\n\r\n".data(using: .utf8)!)
        body.append(imageData)
        body.append("\r\n".data(using: .utf8)!)

        // Add other fields
        for (key, value) in options {
            body.append("--\boundary\r\n".data(using: .utf8)!)
            body.append("Content-Disposition: form-data; name=\"\key\"\r\n\r\n".data(using: .utf8)!)
            body.append("\value\r\n".data(using: .utf8)!)
        }
    }
}
```

```
body.append("--\n(boundary)--\n".data(using: .utf8!))
request.httpBody = body

let (data, _) = try await URLSession.shared.data(for: request)
let response = try JSONDecoder().decode(GenerationResponse.self, from: data)
return response.image
}
```

Rate Limiting

Kullanıcı Tipi	Limit
Anonim	60 istek/dakika
Authenticated	300 istek/dakika
Admin	1000 istek/dakika

Destek

- **Email:** support@fasheone.com
- **Web:** <https://fasheone.com>
- **API Durumu:** <https://status.fasheone.com>

Versiyon Geçmişi

v1.0.0 (2024-01-21)

- İlk sürüm
- Tüm AI generation endpointleri
- Stripe ödeme entegrasyonu
- Admin dashboard API