

Teklif İsteği (Quote Request) Özelliği

Genel Bakış

Teklif İsteği özelliği, müşterilerin ürünler için özel fiyat teklifleri talep etmelerini sağlar. Bu özellik özellikle tarım ekipmanları gibi büyük alımlar için fiyat görüşmesi gerektiren e-ticaret platformları için tasarlanmıştır.

Özellikler

- ✓ Ürün detay sayfalarında “Teklif İste” butonu
- ✓ Modal içinde açılan, responsive form
- ✓ Zorunlu alan validasyonu
- ✓ E-posta format validasyonu
- ✓ Pozitif miktar kontrolü
- ✓ Yükleme durumu göstergesi
- ✓ Başarı/hata mesajları
- ✓ Mobil ve masaüstü uyumlu
- ✓ Türkçe dil desteği
- ✓ Erişilebilirlik (klavye navigasyonu, focus yönetimi)

Dosya Yapısı

src/		
types/		
quote.ts	#	TypeScript tip tanım-
lamaları		
lib/		
data/		
quotes.ts	#	API client fonksiyonları
modules/		
products/		
components/		
quote-request/		
index.tsx	#	QuoteRequestButton (ana b
ileşen)		
quote-request-form.tsx	#	Form bileşeni
quote-request-modal.tsx	#	Modal wrapper
product-actions/		
index.tsx	#	ProductActions (güncellen-
miş)		
mobile-actions.tsx	#	MobileActions (güncellen-
miş)		

Kurulum

1. Ortam Değişkenleri

`.env.local` dosyanızda aşağıdaki değişkenin ayarlandığından emin olun:

```
NEXT_PUBLIC_MEDUSA_BACKEND_URL=http://localhost:9000
```

Bu URL, backend API'nizin çalıştığı adresi göstermelidir.

2. Backend API Endpoint'i

Backend'inizde aşağıdaki endpoint'in hazır olduğundan emin olun:

Endpoint: `POST /store/quotes`

Request Body:

```
{
  "customer_name": "string",
  "customer_email": "string",
  "customer_phone": "string",
  "quantity": number,
  "customer_address": "string",
  "notes": "string (optional)",
  "product_id": "string",
  "product_title": "string"
}
```

Response (Success):

```
{
  "success": true,
  "message": "Teklif isteğiniz başarıyla alındı.",
  "quote": {
    "id": "string",
    "customer_name": "string",
    "customer_email": "string",
    "customer_phone": "string",
    "quantity": number,
    "customer_address": "string",
    "notes": "string",
    "product_id": "string",
    "product_title": "string",
    "status": "string",
    "created_at": "string"
  }
}
```

Response (Error):

```
{
  "success": false,
  "message": "Hata mesajı",
  "errors": {
    "field_name": ["error message"]
  }
}
```

Kullanım

Temel Kullanım

Teklif isteęi butonu otomatik olarak tüm ürün detay sayfalarında “Sepete Ekle” butonunun altında görünür.

Manuel Kullanım

Başka bir sayfada veya bileşende kullanmak isterseniz:

```
import QuoteRequestButton from "@modules/products/components/quote-request"

function MyComponent() {
  return (
    <QuoteRequestButton
      productId="prod_123"
      productTitle="Sera Ekipmanı XYZ"
      variant="secondary"
      className="custom-class"
    />
  )
}
```

Props

QuoteRequestButton

Prop	Tip	Zorunlu	Varsayılan	Açıklama
productId	string	✓	-	Ürün ID’si
productTitle	string	✓	-	Ürün başlığı
variant	“primary” “secondary”	✗	“secondary”	Buton stili
className	string	✗	“”	Ek CSS sınıfları

Form Alanları

Alan	Türkçe	Tip	Zorunlu	Validasyon
name	Ad	text	✓	Boş olamaz
email	E-posta	email	✓	Geçerli e-posta formatı
phone	Telefon	tel	✓	Boş olamaz
quantity	Miktar	number	✓	> 0
address	Adres	text	✓	Boş olamaz
notes	Notlar	textarea	✗	-

Validasyon Kuralları

- Ad:** En az 1 karakter olmalıdır
- E-posta:** Geçerli e-posta formatında olmalıdır (örn: user@example.com)
- Telefon:** Boş bırakılamaz
- Miktar:** Pozitif bir sayı olmalıdır (0'dan büyük)
- Adres:** Boş bırakılamaz
- Notlar:** Opsiyonel alan, herhangi bir kısıtlama yok

Durum Mesajları

Başarı Mesajı

Teklif isteğiniz başarıyla alındı. En kısa sürede size dönüş yapacağız.

Hata Mesajları

- Genel hata: "Bir hata oluştu. Lütfen tekrar deneyin."
- Validasyon hatası: İlgili alan için özel hata mesajı
- API hatası: Backend'den gelen hata mesajı

Styling

Form, mevcut Medusa UI ve Tailwind CSS stillerini kullanır. Özelleştirmek için:

- Form stilleri:** `quote-request-form.tsx` dosyasındaki Tailwind sınıflarını düzenleyin
- Buton stilleri:** `index.tsx` dosyasındaki `className` prop'unu kullanın
- Modal stilleri:** `@modules/common/components/modal` bileşenini özelleştirin

Responsive Tasarım

- **Masaüstü:** Form modal içinde, tam boyutlu inputlar
- **Mobil:**
 - Form modal içinde, touch-friendly büyük inputlar
 - “Teklif İste” butonu mobil aksiyonlar barında görünür
 - Sayfanın altında sticky bir bar ile kolay erişim

Erişilebilirlik

- ☒ Klavye navigasyonu (Tab, Enter, Escape)
- ☒ Focus yönetimi (modal açılınca form’a focus)
- ☒ ARIA etiketleri (required alanlar)
- ☒ Hata mesajları ekran okuyuculara uygun
- ☒ Semantik HTML kullanımı

Test

Form bileşeni şu test ID’leri ile işaretlenmiştir:

- `quote-request-button` : Teklif İste butonu
- `quote-request-modal` : Modal container

Örnek test:

```
// Jest + React Testing Library
const button = screen.getByTestId("quote-request-button")
fireEvent.click(button)
const modal = screen.getByTestId("quote-request-modal")
expect(modal).toBeInTheDocument()
```

Troubleshooting

Form gönderilemiyor

1. `NEXT_PUBLIC_MEDUSA_BACKEND_URL` değişkenini kontrol edin
2. Backend API’nin çalıştığından emin olun
3. CORS ayarlarını kontrol edin
4. Browser console’da hata mesajlarını kontrol edin

Buton görünmüyor

1. `product.id` ve `product.title` değerlerinin mevcut olduğundan emin olun
2. Import statement’ın doğru olduğundan emin olun

Validasyon çalışmıyor

1. Browser console’da JavaScript hatalarını kontrol edin
2. Form state’inin doğru güncellendiğini kontrol edin

API İletişimi

API çağrıları `src/lib/data/quotes.ts` dosyasında tanımlanmıştır:

```
import { submitQuoteRequest } from "@lib/data/quotes"

const result = await submitQuoteRequest({
  customer_name: "...",
  customer_email: "...",
  // ... diğer alanlar
})

if (result.success) {
  // Başarılı
} else {
  // Hata
}
```

Özelleştirme







Yeni Alan Ekleme

1. `src/types/quote.ts` dosyasına yeni alanı ekleyin
2. `quote-request-form.tsx` dosyasına input ekleyin
3. Validasyon kurallarını ekleyin
4. Backend API'yi güncelleyin

Farklı Stiller

```
<QuoteRequestButton
  productId={product.id}
  productTitle={product.title}
  variant="primary"
  className="bg-green-600 hover:bg-green-700"
/>
```

Güvenlik Notları

-  Form verileri client-side'da sanitize edilir
-  API çağrıları server-side'da yapılır
-  E-posta formatı doğrulanır
-  Backend'de ek güvenlik kontrolleri yapılmalıdır
-  Rate limiting uygulanmalıdır
-  SPAM koruması eklenmelidir

Gelecek Geliştirmeler

- ☐ ReCAPTCHA entegrasyonu
- ☐ Dosya ekleme özelliği
- ☐ Otomatik e-posta bildirimleri
- ☐ Admin panelinde teklif yönetimi

- [] Teklif durumu takibi
- [] Multi-language desteęi (şu an sadece Türkçe)

Lisans

Bu özellik Medusa IOX Storefront'un bir parçasıdır ve aynı lisans altındadır.

Destek

Sorularınız için:

- GitHub Issues: [github.com/your-repo/issues]
- Dokümantasyon: Bu dosya

Son Güncelleme: Ekim 2025