

# Kolay Adresleme Sistemi Referans Dokümanı

Haziran 2020

Sürüm 1.1

*Hizmete Özel*

<b>1. TANIMLAR VE KISALTMALAR</b>	<b>3</b>
<b>2. AMAÇ VE KAPSAM</b>	<b>4</b>
2.1. Kolay Adres İle Ödeme Akışı (Alıcı Sorgulama)	4
2.2. Kolay Adres Örnekleri	5
<b>3. TEMEL PRENSİPLER</b>	<b>6</b>
<b>4. KOLAS SERVİSLERİ</b>	<b>8</b>
4.1. Kolay Adres Ekleme	8
4.2. Kolay Adres Güncelleme	8
4.3. Kolay Adres Silme	9
4.4. Kolay Adres Sorgulama	9
4.5. Alıcı Sorgulama (Ödeme Akışı Öncesi)	10
4.6. Taşınan Adres Sorgulama	10
<b>5. TAKVİM VE GEREKSİNİMLER</b>	<b>11</b>
5.1. Proje Takvimi	11
5.2. Katılımcı Gereksinimleri	12
5.2.1. Network (VPN Bulutu) Gereksinimleri	12
5.2.2. İş Sürekliliği ve Yedeklilik Gereksinimleri	12
5.2.3. Güvenlik Gereksinimleri	13
5.2.4. Uygulama Gereksinimleri	13
<b>6. ENTEGRASYON ADIMLARI</b>	<b>14</b>
6.1. Servis Erişimi	14
6.1.1. Test Ortamı :	14
6.1.2. Üretim Ortamı	14
6.1.3. İş Sürekliliği Yedekleme Merkezi Ortamı	14
6.2. Servis Veri Yapıları	15
6.3. Servis Metodları	17
6.3.1. Token Oluşturma – Authorization Servisi	17
6.3.2. Kolay Adres Ekleme Servisi	18
6.3.3. Kolay Adres Güncelleme Servisi	19
6.3.4. Kolay Adres Silme Servisi	20
6.3.5. Kolay Adres Sorgulama Servisi	20
6.3.6. Alıcı Sorgulama Servisi	21
6.3.7. Taşınan Adres Sorgulama Servisi	22
6.4. Servis Hata Yönetimi	23

## 1. Tanımlar ve Kısaltmalar

<b>Alıcı (Müşteri):</b>	Ödeme işlemine konu fonun ulaşması istenen gerçek veya tüzel kişidir.
<b>Alıcı Katılımcı:</b>	FAST veya PÖS Sistemi'nden gönderilen bir mesajın iletildiği katılımcıdır.
<b>Anahtar Alanlar:</b>	Kayıtların tekiliğini sağlamak üzere kullanılan "Müşteri Kimlik Numarası", "Kolay Adres Tipi" ve "Kolay Adres Değeri" bilgilerinden oluşan veri setidir.
<b>BKM:</b>	Bankalararası Kart Merkezi A.Ş.
<b>Kolay Adres:</b>	Ödeme ve fon transferlerinde hesap numarası ile ad soyad yerine kullanılan ve sistemde belirli bir hesap ile eşleştirilmiş adres bilgisidir. Dünya örneklerinde kolay adres olarak telefon numarası, e-posta adresi, kimlik numarası, pasaport numarası gibi adresler veya sosyal medya hesabı, dijital kimlik numarası, biyometrik veri ya da tamamen kullanıcı tarafından verilen bir rumuz kullanılabilmektedir.
<b>EFT:</b>	TCMB Ödeme Sistemleri Elektronik Fon Transfer Sistemi
<b>FAST:</b>	Fonların Anlık ve Sürekli Transferi
<b>Gönderen (Müşteri):</b>	Kendi ödeme hesabından veya ödeme hesabı bulunmaksızın ödeme emri veren gerçek veya tüzel kişidir.
<b>Gönderen Katılımcı:</b>	FAST veya PÖS Sistemi'ne fon transfer mesajı gönderen katılımcıdır.
<b>Hesap:</b>	Kolay adres bilgisinin eşleştirildiği IBAN bilgisidir.
<b>Katılımcı:</b>	KOLAS'a üye olan, sisteme doğrudan işlem gönderme veya KOLAS'tan gönderilen mesajı alma yetkisine sahip olan ve sistem kurallarına uymakla yükümlü tüzel kişilerdir.
<b>KOLAS:</b>	Kolay Adresleme Sistemi
<b>Müşteri:</b>	KOLAS'a iletilecek kolay adres bilgisinin ve servis mesajının kaynağı olan bireysel veya tüzel kişilerdir.
<b>Müşteri Kimlik Numarası:</b>	Katılımcının müşterisini ayırt etmek için kullandığı T.C. vatandaşları için TCKN, T.C.'nde yerleşik şirketler için VKN, yabancı uyruklular için sırasıyla YKN ya da Pasaport Numarası bilgisini ifade etmektedir.
<b>PÖS:</b>	TCMB Ödeme Sistemleri Perakende Ödeme Sistemi
<b>TCKN:</b>	T.C. İçişleri Bakanlığı Nüfus ve Vatandaşlık İşleri Genel Müdürlüğü tarafından kişinin Türk vatandaşı olduğunu göstermesi amacıyla tahsis edilmiş Türkiye Cumhuriyeti kimlik numarasıdır.
<b>TCMB:</b>	Türkiye Cumhuriyet Merkez Bankası
<b>VKN:</b>	Vergi kimlik numarası, gerçek ve tüzel kişiler ile kamu ve özel kuruluşlar tarafından gerçekleştirilen işlemlerin kayıt altına alınması ve belli bir düzenle bu işlerin ilerlemesi için kişi ya da kuruma verilen numaradır.
<b>YKN:</b>	T.C. İçişleri Bakanlığı Nüfus ve Vatandaşlık İşleri Genel Müdürlüğü tarafından, ikamet ve çalışma izni almış yabancı uyruklu vatandaşlara ya da talep etmeleri halinde KKTC vatandaşlarına tahsis edilmiş yabancı kimlik numarasıdır.

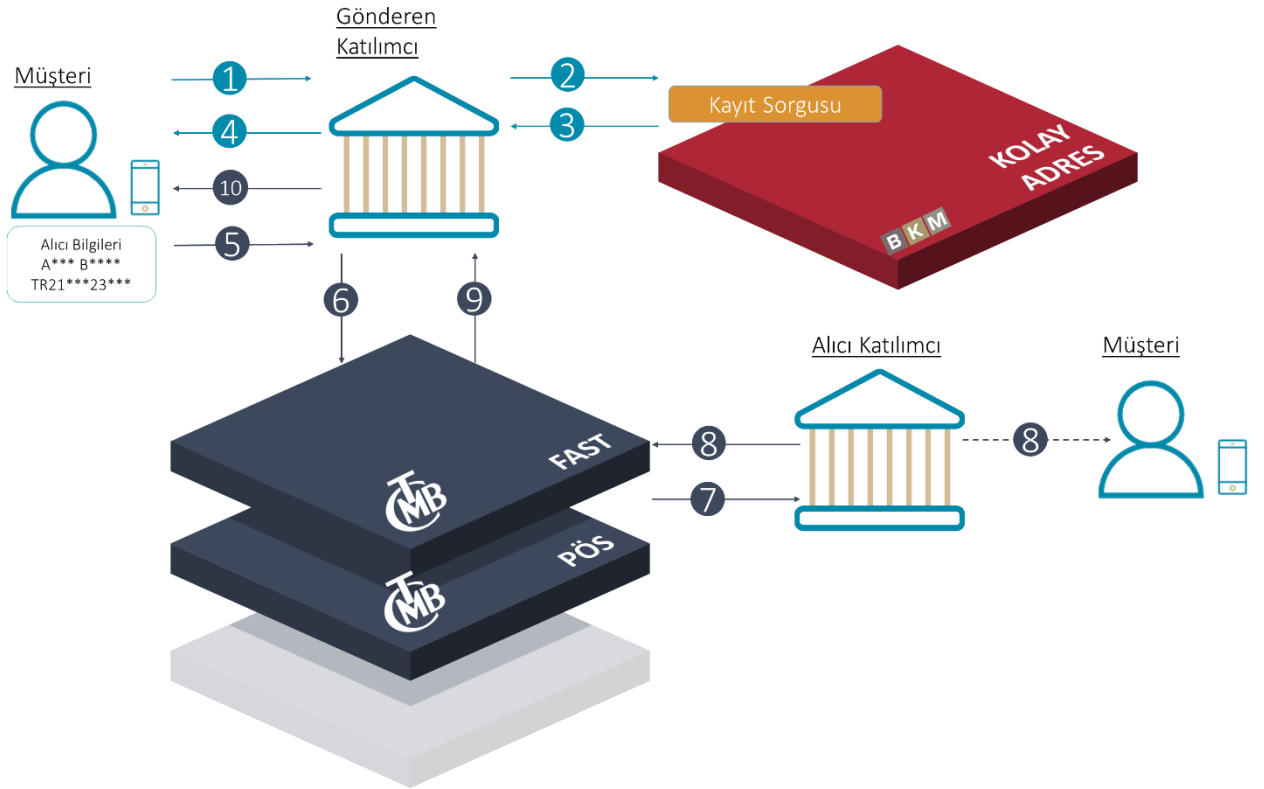
## 2. Amaç ve Kapsam

Bu dokümanın amacı, Fonların Anlık ve Sürekli Transferi (FAST) ödeme sisteminin katmanlı servislerinden biri olan “Kolay Adresleme Sistemi (KOLAS)” ve bu sistemin kullandığı alt yapının işleyişine ilişkin genel kuralları ve katılımcıların teknik entegrasyon gereksinimlerini açıklamaktır.

Dokümanda, “Kolay Adresleme Sistemi (KOLAS)” katılımcıları tarafından gerçekleştirilecek “kolay adres / hesap eşleştirmesi” ile ilgili servisler, iş akışları, uygulama mimarisi, entegrasyon gereksinimleri, mesaj standartları ve işletim kuralları tanımlanmaktadır.

Kolay Adresleme Sistemi; EFT, PÖS, FAST gibi fon transferi gerçekleştirilen ödeme sistemlerinde müşterilerin kullanım deneyimlerini geliştirme hedefi taşıyan katma değerli bir hizmettir. Bu hizmet, müşterilerin ödeme sistemi katılımcılarındaki hesap numaralarının (IBAN) basitçe hatırlanabilecek bir adres ile (**Kolay Adres**: telefon numarası, e-posta adresi, T.C. Kimlik numarası (TCKN), Vergi Kimlik Numarası (VKN), Yabancı Kimlik Numarası, Pasaport Numarası) eşleştirilmesi ve fon transferlerinde hesap numarası yerine bu adreslerin kullanılabilmesi prensibine dayanır.

### 2.1. Kolay Adres İle Ödeme Akışı (Alıcı Sorgulama)



- 1 Gönderen müşteri, gönderen katılımcının sunduğu kanallar (internet bankacılığı, dijital cüzdan...) üzerinden alıcının kolay adresini (telefon numarası, kimlik numarası veya e posta adresi) girerek ödeme talimatını iletir.
- 2 Gönderen Katılımcı, “Alıcı Sorgulama” mesajını KOLAS’a gönderir.
- 3 KOLAS, Alıcı’ya ait kolay adres bilgileri mevcut ise bu bilgileri (Ad Soyad, Hesap Numarası) mesajın cevabı olarak Gönderen Katılımcı’ya iletir.
- 4 Gönderen Katılımcı, ödeme yapılacak Alıcı bilgilerini teyit etmek ve onaylatmak üzere maskeli bir şekilde Gönderen Müşteri’ye gösterir.

- 5 Gönderen Müşteri, Alıcı bilgilerini ve gönderilecek tutarı kontrol ederek onaylar.
- 6 Gönderen Katılımcı, ödeme mesajını Alıcı bilgileriyle birlikte Ödeme Sistemine (FAST, PÖS, ...) ileterek ödeme akışını başlatır.
- 7 Ödeme Sistemi ödeme talebini Alıcı Katılımcı'ya iletir.
- 8 Alıcı Katılımcı, müşterisinin hesap kontrolünü yaparak işlemi onaylar ve müşterisini bilgilendirir.
- 9 Ödeme Sistemi, ödeme işleminin onaylandığını Gönderen Katılımcı'ya iletir.
- 10 Gönderen Katılımcı Gönderen Müşteri'yi bilgilendirir.

## 2.2. Kolay Adres Örnekleri

TCKN/VKN/YKN	Adres Tipi	Adres Değeri	Hesap Numarası	Müşteri Adı	Ticari?	Açıklama
21012345678	K	21012345678	TR0010...1234	Ali Şahin	N	A Bankası Maaş Hesabı
21012345678	T	5557775555	TR0010...1234	Ali Şahin	N	A Bankası Maaş Hesabı
21012345678	T	5441234567	TR0046...2468	Ali Sahin	N	Bbank Otomatik Ödeme
21012345678	E	<a href="mailto:alisahin34@gmail.com">alisahin34@gmail.com</a>	TR0067...3692	ALİ ŞAHİN	Y	Dükkanın hesabı-C Ödeme
21012345678	E	<a href="mailto:alisahin34@hotmail.com">alisahin34@hotmail.com</a>	TR0062...4826	M. ALİ ŞAHİN	Y	Dükkanın hesabı-D Bank
21012345678	E	<a href="mailto:alisahin@xbank.com.tr">alisahin@xbank.com.tr</a>	TR0111...5050	ALI SAHIN	Y	Dükkanın hesabı-X Bank
21012345678	S*	@AliSahin34	TR0111...5050	ALI SAHIN	Y	Dükkanın Twitter hesabı
21012345678	D*	A018^?d*q123	TR0134...6284	ALI ŞAHİN	N	
21012345678	R*	alisahin34@odegec	TR0067...7418	ALİ METE ŞAHİN	Y	Büfe

\*Dijital ID, sosyal medya hesap ismi, rumuz gibi bilgilerle hesap eşleştirmesi ilerleyen fazlarda değerlendirilecektir.

### 3. Temel Prensipler

- KOLAS, asıl olarak PÖS ve FAST sistemlerinde hızlı ödeme başlatmak amacıyla kullanılabileceği gibi, işletim kuralları çerçevesinde farklı iş ihtiyaçları için de kullanılabilir.
- 7x24 esasına göre çalışacaktır.
- Kolay adres olarak “Telefon Numarası”, “T.C. Kimlik Numarası (TCKN)”, “Vergi Kimlik Numarası (VKN)”, “Yabancı Kimlik Numarası (YKN)”, “Pasaport Numarası” ve “E-posta Adresi” kullanılabilecektir. İleride ihtiyaçlara göre yeni kolay adres tipleri eklenebilecektir.
- Bir müşteri **aynı tip** kolay adresi birden fazla kez kullanabilecektir.
- Kolay adres olarak kullanılacak bilgilerden **her birine yalnızca bir hesap** (IBAN) bağlanabilecek, bu hesaplar farklı katılımcılara ait hesaplar olabilecektir.
- **Farklı kolay adresler** aynı hesap numarasına bağlanabilecektir. (Örnek: Hem e-posta hem de telefon numarası aynı hesaba bağlanabilir)
- Bir müşteri **en fazla beş (5) adet kolay adrese** sahip olabilecektir. (Örnek: 3 Telefon numarası, 2 E-posta adresi). Bu sınır ileride ihtiyaçlara göre artırılabilir.
- Farklı katılımcılara ait hesaplar arasında adres taşıma işlemi, mevcut kolay adres kaydının arşivlenerek yeni bir kolay adres kaydı oluşturulması şeklinde olacaktır.
- Tüm akışlar anlık ve web servisler yardımıyla yürütülecek, dosya yoluyla toplu kayıt işleme desteklenmeyecektir.
- Katılımcı ile KOLAS arasındaki veri trafiği güvenli ve şifreli olarak gerçekleşecek, veriler şifreli olarak saklanacaktır.
- KOLAS, katılımcılara gerekli işlevleri aşağıdaki servisler aracılığıyla sunacaktır:
  - **Kolay Adres Ekleme:** Hesap sahibi katılımcının müşterisi tarafından tanımlanan yeni bir kolay adresi bildirmesi
  - **Kolay Adres Güncelleme:** Hesap sahibi katılımcının müşterisi tarafından mevcuttaki bir kolay adres tanımını güncellemesi
  - **Kolay Adres Silme:** Hesap sahibi katılımcının müşterisi tarafından silinmek istenen kolay adresi bildirmesi
  - **Kolay Adres Sorgulama:** Hesap sahibi katılımcının müşteri(ler)ine ait kolay adres(ler)i sorgulaması
  - **Alıcı Sorgulama:** Ödeme öncesinde gönderici katılımcının KOLAS’ta kayıtlı hesap bilgilerini sorgulaması
  - **Taşınan Adres Sorgulama:** Bir katılımcı tarafından bildirilmiş kolay adres(ler)in başka katılımcıya taşınması durumunda eski katılımcının arşivlenen kolay adres bilgi(ler)ini sorgulaması
- İleride ihtiyaca göre e-devlet entegrasyonu yapılarak, e-devlet tarafından doğrulanan müşterilerin katılımcı fark etmeksizin kolay adreslerinin bulunduğu tüm hesaplarını sorgulayıp görüntülemesi mümkün olabilecektir.
- Kolay adres ekleme/güncelleme işleminde, iletilen kolay adres değeri ile farklı bir Müşteri Kimlik Numarası’na ait halihazırda kolay adres kaydı varsa talep reddedilecektir.
- Katılımcı, Kolay Adres Güncelleme ve Silme servislerini kullanmadan önce mutlaka Kolay Adres Sorgulama servisi ile müşteriye ait KOLAS’ta yer alan güncel kolay adres bilgilerini alarak müşteriye göstermeli ve müşteri talebine göre güncelleme ya da silme işlemi yapmalıdır.
- Katılımcılar, KOLAS’a bildirilmiş bir hesabın kapanması durumunda anlık olarak KOLAS’a bildirmekle yükümlüdür.
- Katılımcılar, sahipliği değişen kolay adresleri (telefon numarası, e-posta adresi vs.) silinmek üzere, anlık olarak KOLAS’a bildirmekle yükümlüdür.
- Katılımcılar, müşterileri tarafından iletilen ekleme, silme, güncelleme vb. talimatları bekletmeksizin anlık olarak KOLAS’a, bu işlemlerin sonuçlarını ise bekletmeksizin müşterilerine bildirmekle yükümlüdür.
- Katılımcılar, müşterilerinin açık rızasını almadan KOLAS’a kayıt gönderemezler, müşteri adına yapılacak değişiklikler için müşteriye doğrulamakla yükümlüdür.

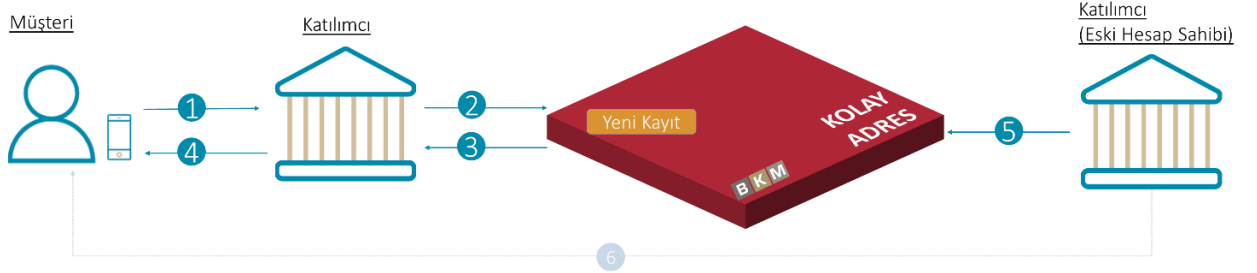
- Kolay Adres İle Ödeme işlemi öncesinde yapılan sorgulamada göndericiye ait Müşteri Kimlik Numarası da Alıcı Sorgulama servis isteğinde KOLAS'a gönderilmelidir.

## 4. KOLAS Servisleri

### 4.1. Kolay Adres Ekleme

Müşteri tarafından katılımcı nezdinde bulunan hesap bilgisi ile sistemde izin verilen kolay adres tiplerinden birini eşleştirmek için kullanılan servistir.

Müşteriler farklı katılımcılardaki hesaplarını farklı kolay adresler ile eşleştirebilirler. Kolay adres kayıtları sisteme başarılı olarak kaydedildiği andan itibaren kullanılabilir olacaktır. İleri tarihli kolay adres ekleme seçeneği bulunmamaktadır.



- 1 Müşteri, eşleştirmek istediği hesabın bağlı olduğu katılımcıya ilgili kolay adres ve hesap ikilisini seçerek yeni kayıt talimatını gönderir.
- 2 Katılımcı, müşterinin kimliği ve adres bilgilerini doğrular ve yeni kayıt mesajını oluşturarak KOLAS'a gönderir.
- 3 KOLAS gerekli kontrolleri yapar ve
  - ilgili kolay adres değerine ait halihazırda bir kayıt bulunmuyorsa yeni bir kolay adres kaydı oluşturur,
  - ilgili kolay adres başka bir Müşteri Kimlik Numarasına bağlı ise ekleme talebini reddeder,
  - iletilen Anahtar Alanlar ile aynı Katılımcıya ait bir adres kaydı tanımlı ise kayıt talebini reddeder,
  - iletilen Anahtar Alanlar ile farklı bir Katılımcı'ya ait adres kaydı tanımlı ise, mevcut kaydı arşivleyerek yeni bir kolay adres kaydı oluşturur (**Adres Taşıma**)
  - işlemin sonucu hakkında Katılımcıyı bilgilendirir.
- 4 Katılımcı, işlemin sonucu hakkında müşterisini bilgilendirir.
- 5 Ekleme işlemi Adres Taşıma olarak sonuçlanmışsa eski hesap sahibi katılımcı, Taşınan Adres Sorgulama servisini kullanarak dilediği periyotlarla kayıtları sorgulayarak bu değişiklikten haberdar olabilir.
- 6 Eski hesap sahibi katılımcı isterse müşterisi ile iletişime geçebilir ya da bu değişiklikle ilgili bilgilendirebilir.

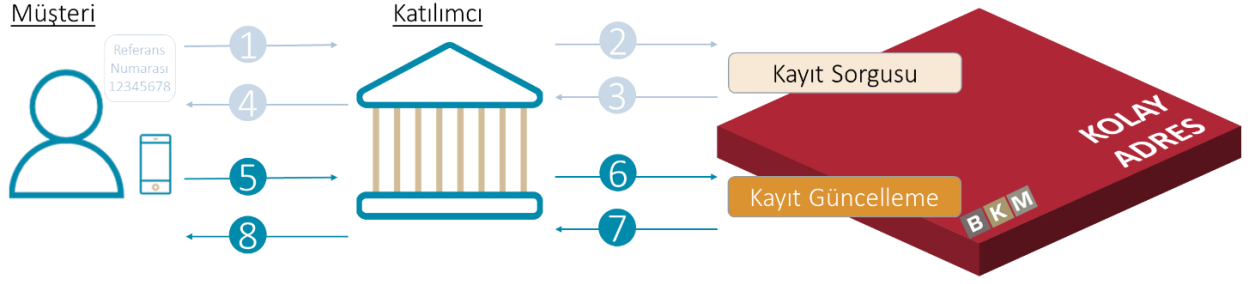
### 4.2. Kolay Adres Güncelleme

Kolay Adres Güncelleme servisi, kolay adres hesabının katılımcısı tarafından güncellenebilmesi amacıyla kullanılan servistir.

Kolay Adres Güncelleme servisinde tüm alanların güncel bilgilerle iletilmesi gerekmektedir. Güncelleme işleminden önce KOLAS'taki mevcut bilgiler, "Kolay Adres Sorgulama" servisiyle sorgulanarak, güncellenecek kayda ait "KOLAS Referans Numarası" (systemRefNo) temin edilir ve bir sonraki adımda güncelleme işlemi bu numara ile yapılır.

Kolay Adres Güncelleme servisi ile sistemdeki bir kayda ait Müşteri Kimlik Numarası güncellenemez.





1 - 4 Katılımcı, güncelleme öncesinde müşteriye ait mevcut bilgileri ve KOLAS Referans Numarasını sorgular.

5 Müşteri, daha önce tanımladığı kolay adres bilgisini güncelleme talebini Katılımcıya iletir.

6 Katılımcı gerekli kontrolleri yaparak müşteriden aldığı talimatı “Kolay Adres Güncelleme” mesajıyla KOLAS’a iletir.

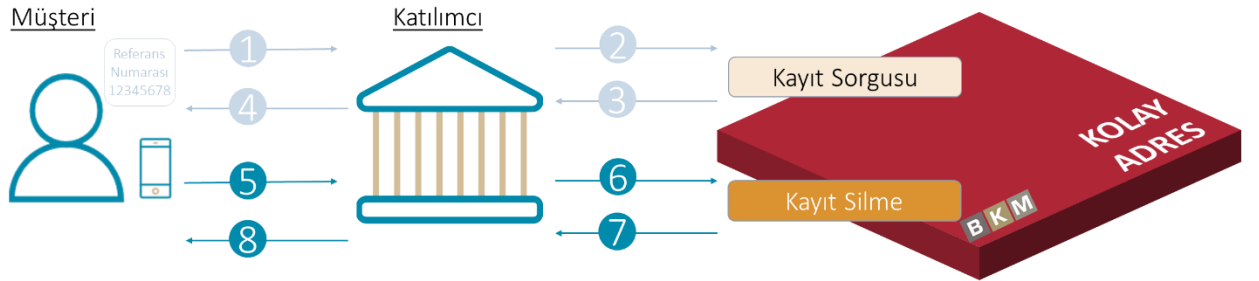
7 KOLAS, referans numarası üzerinden ilgili kolay adresin kayıtlarda mevcut olduğunu kontrol ederek güncelleme işlemini gerçekleştirerek Katılımcıya dönüş yapar.

8 Katılımcı, işlemin sonucu hakkında müşterisini bilgilendirir.

### 4.3. Kolay Adres Silme

Müşteri tarafından daha önce Katılımcı nezdinde bulunan kolay adres kaydının silinmesi amacıyla kullanılan servistir.

İşlem öncesinde güncel bilgiler KOLAS sisteminden “Kolay Adres Sorgulama” servisi ile sorgulanmalıdır. Silme işleminin, KOLAS Sistem Referans Numarası ile yapılması zorunludur.



1 - 4 Katılımcı, güncelleme öncesinde müşteriye ait mevcut bilgileri ve KOLAS Referans Numarasını sorgular.

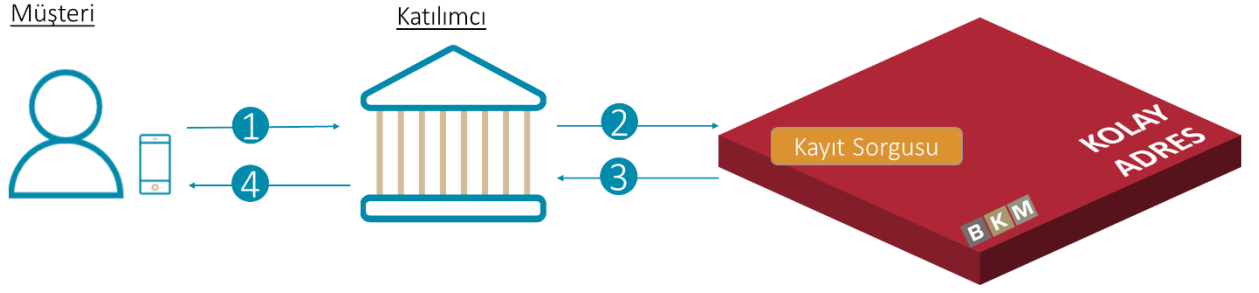
6 Katılımcı gerekli kontrolleri yaparak müşteriden aldığı talimatı “Kolay Adres Güncelleme” mesajıyla KOLAS’a iletir.

7 KOLAS, referans numarası üzerinden ilgili kolay adresin kayıtlarda mevcut olduğunu kontrol ederek silme işlemini gerçekleştirerek Katılımcıya dönüş yapar.

8 Katılımcı, işlemin sonucu hakkında müşterisini bilgilendirir.

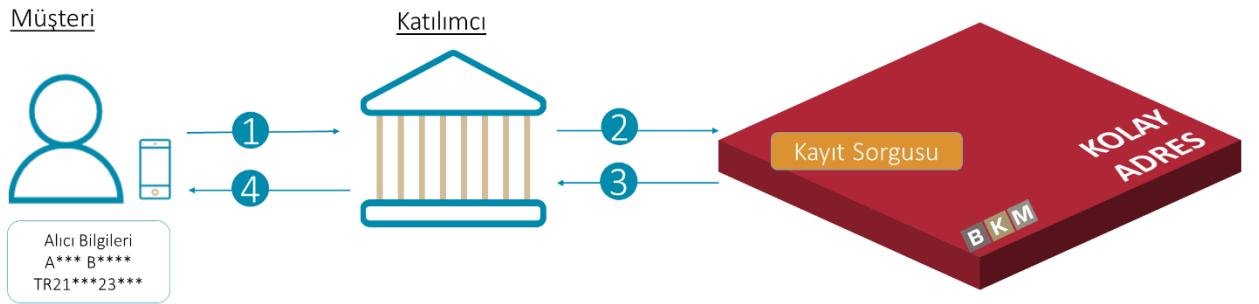
### 4.4. Kolay Adres Sorgulama

Katılımcı’nın KOLAS’ta yer alan kolay adres kayıtlarını sorguladığı servistir. Katılımcılar yalnızca kendisine ait kayıtlara erişebilir, sorgu yapılan müşterinin farklı Katılımcı’larda bulunan kolay adreslerine erişemez.



#### 4.5. Alıcı Sorgulama (Ödeme Akışı Öncesi)

Katılımcı'nın, ödeme işlemi öncesi Alıcıya ait bir kolay adres bilgisi ile eşleştirilen hesap bilgisini sorgulaması amacıyla kullanılan servistir. Sorgulama sonucunda hangi Katılımcı'ya ait olduğuna bakılmaksızın alıcı müşteriye ait Ad Soyad, Hesap Numarası bilgileri paylaşılır.

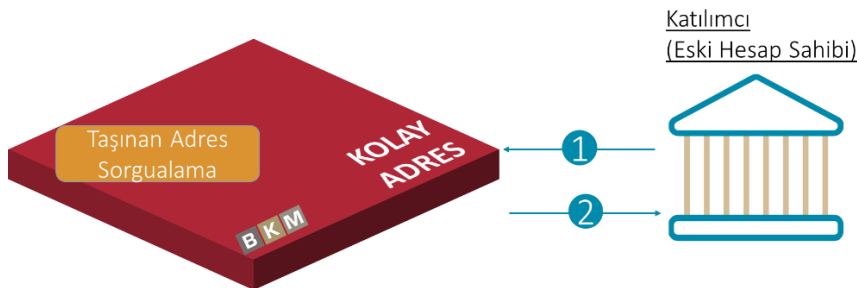


- 1 Gönderen Müşteri, Alıcı'nın kolay adresini (telefon numarası, kimlik numarası veya e posta adresi) girerek ödeme talimatını iletir.
- 2 Gönderen Katılımcı, "Alıcı Sorgulama" mesajını KOLAS'a gönderir.
- 3 KOLAS, Alıcı'ya ait kolay adres bilgileri mevcut ise bu bilgileri (Ad Soyad, Hesap Numarası) mesajın cevabı olarak Gönderen Katılımcı'ya iletir.
- 4 Gönderen Katılımcı, ödeme yapılacak Alıcı bilgilerini teyit etmek ve onaylatmak üzere maskeli bir şekilde Gönderen Müşteri'ye gösterir.

#### 4.6. Taşınan Adres Sorgulama

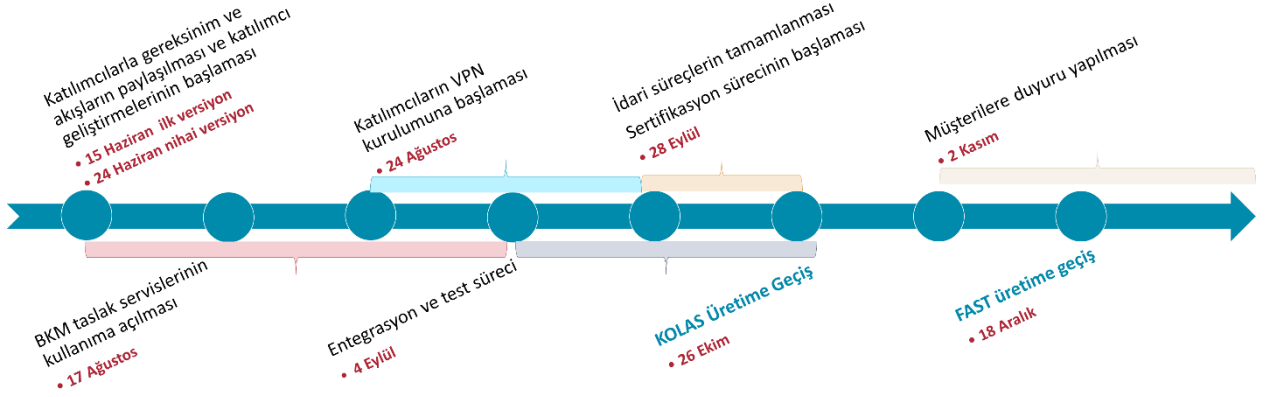
Daha önce müşteri tarafından bir hesap ile ilişkilendirilen bir kolay adres bilgisinin başka bir Katılımcı'ya ait hesaba değiştirilmesi (Hesap Taşıma) durumunda, eski Katılımcı'nın bu değişiklikten bilgi sahibi olması amacıyla kullanacağı servistir.

Aynı Katılımcı'nın farklı hesapları arasında bir taşıma söz konusu olması durumunda, bu bilgi "Kolay Adres Güncelleme" servisinin cevabında Katılımcı'ya bildirileceği için "Taşınan Adres Sorgulama" servisinin cevabında bu kayıtlar yer almaz.



## 5. Takvim ve Gereksinimler

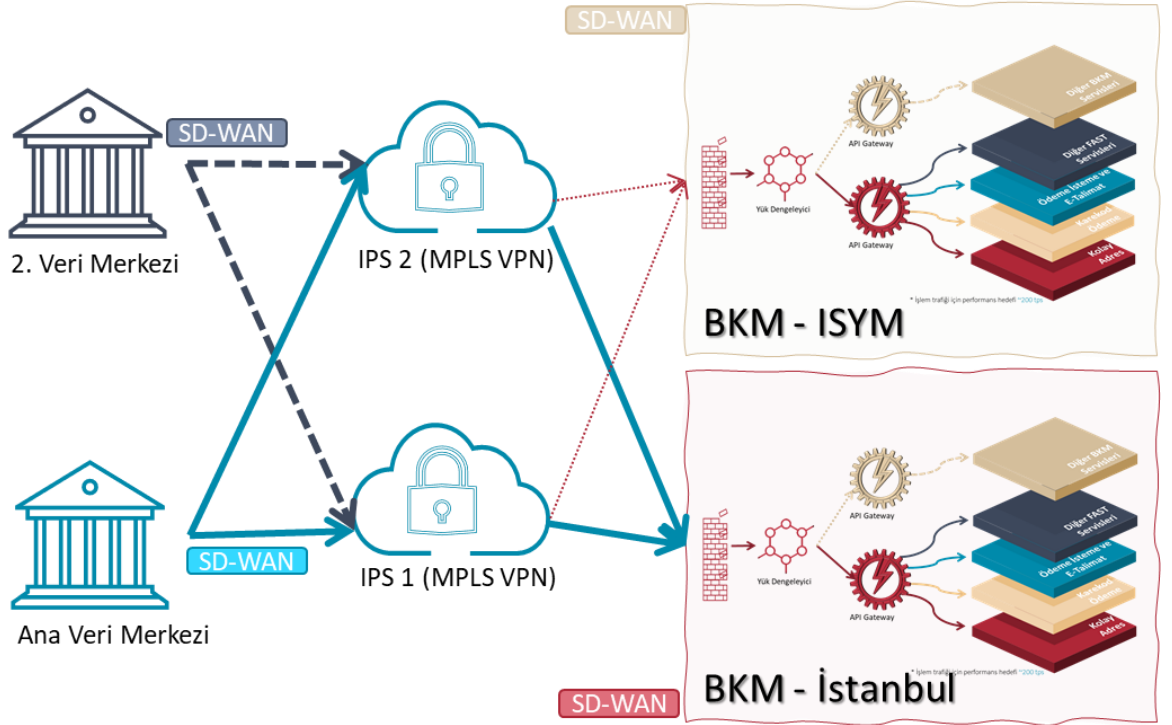
### 5.1. Proje Takvimi



- 12 Haziran'da "FAST Projesi Kapsamında Katman Servislerin Geliştirilmesi" konulu Katılımcı bilgilendirme toplantısının ardından 15 Haziran'da Katılımcı'larla "Kolay Adresleme Sistemi Referans Dokümanı" taslak olarak paylaşılacaktır.
- 24 Haziran'da Katılımcılardan gelecek geri bildirimler ışığında bu dokümanın son güncel sürümü Katılımcı'larla paylaşılacaktır. Bu tarihten itibaren Katılımcılar tarafından **planlama, analiz ve geliştirme** çalışmalarına başlanabilir.
- 17 Ağustos'tan itibaren BKM tarafından sağlanacak taslak servisler kullanılabilir.
- 4 Eylül'den itibaren geliştirmelerini tamamlayan Katılımcılar BKM ile Entegrasyon çalışmalarına başlayabilirler.
- Tüm Katılımcıların 24 Ağustos – 25 Eylül tarihleri arasında VPN kurulumlarını tamamlamaları gerekmektedir.
- 28 Eylül – 23 Ekim tarihleri arasında, Katılımcılarla daha önce paylaşılacak olan test senaryolarına uygun olarak sertifikasyon süreci işletilecektir.
- Sertifikasyon ve ardından entegrasyon süreci ve üretime geçiş adımları VPN bulutu üzerinden yürütüleceği için 28 Eylül tarihine kadar tüm üyelerin VPN kurulumlarını tamamlamaları büyük önem taşımaktadır.
- 26 Ekim'de KOLAS üretime alınarak Katılımcıların pilot olarak kullanması ve servislere erişimleri sağlanacaktır.
- 2 Kasım'da tüm Katılımcı'ların müşterileri ile iletişime geçerek Kolay Adres kayıtlarını yapmaları gerekmektedir. Bu nedenle bu tarihten önce müşterilerden bu bilgilerin alınabileceği ortam, arayüz ve süreçlerin de oluşturulmuş olması beklenmektedir.
- 18 Aralık tarihinde FAST'ın üretime geçişi ile birlikte müşterilerin kolay adreslerini kullanarak para transferi yapabilmelerine imkân sağlanmış olmalıdır.

## 5.2. Katılımcı Gereksinimleri

### 5.2.1. Network (VPN Bulutu) Gereksinimleri



KOLAS ile Katılımcı arasındaki mesaj trafiği güvenli **VPN bulutu** üzerinden işleyeceği için her Katılımcının aşağıdaki gereksinimleri belirtilen takvim doğrultusunda sağlamış olması beklenmektedir.

- Teknik detayları Katılımcı'ların network bölümleri ile daha önce paylaşılan **Router** (iki adet yedekli olmalıdır) ve **SD-WAN Lisansının** temin edilmesi gerekmektedir.
- En az iki farklı ISP'den (Vodafone, Turkcell ya da Türk Telekom) temin edilecek **data hatlarıyla** bağlantı sağlanmalıdır.
- **16 Temmuz** tarihine kadar BKM tarafından Katılımcı'lara devre numaraları ve izin yazıları verilecektir.
- Katılımcıların hat tesisi, donanım ve yazılım temini, konfigürasyon gibi hazırlıklarını **21 Ağustos** tarihine kadar tamamlaması gerekmektedir.
  - BKM, **17 Ağustos** tarihinden itibaren hazırlıklarını tamamlayan Katılımcı'lara VPN kurulumu için çalışma slotu vermeye başlayacaktır.
  - Tüm Katılımcıların **24 Ağustos – 25 Eylül** tarihleri arasında oluşturulan bu yeni VPN bulutuna dahil olmaları beklenmektedir.
- BKM Network Ekibi tarafından tüm teknik detaylar katılımcı network ekipleriyle paylaşılacaktır.

### 5.2.2. İş Sürekliliği ve Yedeklilik Gereksinimleri

KOLAS'a ait tüm sistem bileşenleri hem BKM'nin ana veri merkezi hem de ikinci veri merkezinde Aktif/Pasif prensibi ile çalışacak şekilde hazırlanacaktır.

- İş sürekliliği ya da yedeklilik açısından 2 Veri Merkezlerinden de BKM'ye bağlantı sağlamak isteyen Katılımcıların birincil veri merkezi ile aynı konfigürasyonu tesis etmeleri gerekmektedir.

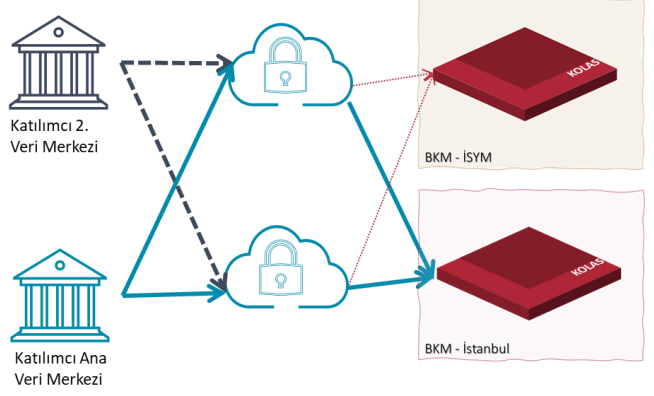
### 5.2.3. Güvenlik Gereksinimleri

Katılımcılar ile KOLAS arasındaki mesaj trafiği uçtan uca şifreli SD-WAN altyapısında MPLS VPN network üzerinden gerçekleştirilecektir.

KOLAS API'ları kimlik doğrulama için OAuth2 standardını kullanır.

### 5.2.4. Uygulama Gereksinimleri

Servisler REST API olarak sunulmaktadır. İstekleri JSON body olarak kabul eder, cevapları da JSON olarak döner. İşlemlerin başarılı olup olmadığını belirtmek için standart HTTP yanıt kodlarını kullanır. Dolayısıyla Katılımcıların bu standartlara uygun istemci geliştirmeleri beklenmektedir.



## 6. Entegrasyon Adımları

### 6.1. Servis Erişimi

Katılımcı'lara clientId ve clientSecret tahsis edilecektir. İlgili clientId ve clientSecret ile authorization servisi çağırılarak access token alınacak, izleyen servis çağrılarında kimlik doğrulaması bu token ile gerçekleştirilecektir.

#### 6.1.1. Test Ortamı :

Servis Adresi : <https://developer.bkm.com.tr/kolas>

IP Adresi : 31.145.171.90

Port : 443

#### 6.1.2. Üretim Ortamı

(Güncel IP adresi MPLS VPN network kurulumları tamamlandıktan sonra paylaşılacaktır)

Servis Adresi : [https://\[BirincilAdres\]/kolas](https://[BirincilAdres]/kolas)

IP Adresi : [BirincilAdres]

Port : 443

#### 6.1.3. İş Sürekliliği Yedekleme Merkezi Ortamı

(Güncel IP adresi MPLS VPN network kurulumları tamamlandıktan sonra paylaşılacaktır)

Servis Adresi : [https://\[İkincilAdres\]/kolas](https://[İkincilAdres]/kolas)

IP Adresi : [İkincilAdres]

Port : 443

Dokümanın devamında iletilen adres bilgileri, yukarıdaki tabloda yer alan ortam karşılıklarına göre servis adres bilgilerinin sonuna eklenerek kullanılmalıdır. (Örneğin, test ortam OAuth işlemleri için "Token Oluşturma – Authorization Servisi" başlığı altında belirtilen </api/v1/oauth/token> adres bilgisi için istek yapılacak tam adres için <https://developer.bkm.com.tr/kolas/api/v1/oauth/token> kullanılabilir. Aynı işlemin üretim ortamı karşılığı da [https://\[BirincilAdres\]/kolas/api/v1/oauth/token](https://[BirincilAdres]/kolas/api/v1/oauth/token) olacaktır.)

## 6.2. Servis Veri Yapıları

REST API’lerde kullanılacak Kolay Adres (proxyAccount) resource nesnesine ait bilgiler ile alanların servislerde kullanım durumları aşağıdaki tabloda detaylandırılmıştır. Kolay Adres Ekleme, Kolay Adres Güncelleme, Kolay Adres Sorgulama ve Taşınan Adres Sorgulama servis yanıtlarında ilişkili kayda ait aşağıda yer alan tüm bilgiler Katılımcıya dönüş yapılır.

Alan Adı	JSON Alan Adı	Örnek	Format*	Maks. Uzunluk	Kolay Adres Ekleme**	Kolay Adres Güncelleme**	Kolay Adres Silme**	Kolay Adres Sorgulama**	Alıcı Sorgulama**	Taşınan Adres Sorgulama**	Açıklama
Sistem Referans Numarası	systemRefNo	12345678901234567890	N20	20	-	Z	Z	K <sup>2</sup>	-	-	KOLAS tarafından üretilen sistem referans numarasıdır. " <b>Katılımcı Kodu, Müşteri Kimlik Numarası, Kolay Adres Tipi, Kolay Adres Değeri</b> " bazında tekildir. Kolay Adres Ekleme operasyonunda KOLAS tarafından üretilerek Katılımcı'ya dönüş yapılır. Güncelleme ve silme operasyonlarında Katılımcı tarafından gönderilmesi zorunludur,
Hizmet Sağlayıcı Kodu	serviceProviderNo	0800	N4	4	-	-	-	-	-	-	Katılımcı adına kaydı ileten hizmet sağlayıcı kuruluş bilgisidir. Servis çağrılarında iletimi beklenmemektedir. KOLAS tarafından çağrıyı gerçekleştiren token içerisinden tespit edilecektir.
Katılımcı Kodu	memberNo	0800	N4	4	-	-	-	-	-	-	Kayıd ileten Katılımcı bilgisidir. Servis çağrılarında iletimi beklenmemektedir. KOLAS tarafından çağrıyı gerçekleştiren token içerisinden alınacaktır.
Müşteri Kimlik Numarası	identityNo	1234567890	AN7..11	11	Z	Z	-	K <sup>2</sup>	-	-	Müşteriye ait geçerli bir TC Kimlik Numarası(N11), Vergi Kimlik Numarası(N10), Yabancı Kimlik Numarası(N11) veya Pasaport Numarası(AN7..9) bilgilerinden birinin bulunduğu alandır.
Kolay Adres Tipi	addressType	'T'	AN1	1	Z	Z	-	K <sup>3</sup>	Z	-	Müşterinin eklemek istediği kolay adres tipidir. KOLAS'ta tanımlı değerlerden biri olmalıdır.
Kolay Adres Değeri	addressValue	'5555555555'	AN7..50	50	Z	Z	-	K <sup>3</sup>	Z	-	Müşterinin eklediği, Katılımcı'sı tarafından doğrulanmış kolay adres değeridir. "Kolay Adres Tipi" ile uyumlu bir değer olmalıdır. TC Kimlik Numarası(N11), Vergi Kimlik Numarası(N10), Yabancı Kimlik Numarası(N11), Pasaport Numarası(AN7..9), Telefon(N10), E-Posta (AN7..50)
Ad-Soyad / Ticari Ünvan	accountOwner	'SERKAN EREN'	AN6..50	50	Z	Z	-	-	-	-	Adres kaydı yapan müşterinin ad-soyad veya ticari ünvan bilgisidir.
Hesap Numarası	accountNo	'TR120800001234567890123456'	AN26	26	Z	Z	-	-	-	-	"Kolay Adres Değeri" ile eşleştirilecek hesap bilgisidir. Gönderici Katılımcı'ya ait bir IBAN olmalıdır. TR ile başlamalı ve 26 karakter uzunluğunda iletilmelidir.
Hesap Tipi	accountType	'T'	AN1	1	Z	Z	-	-	-	-	Ticari-Bireysel hesap bilgisi ayrımının belirtildiği alandır. KOLAS'ta tanımlı değerlerden biri olmalıdır.
Tabelâ Ünvanı	tradeName	'BANKALARARASI KART MERKEZİ'	AN0..50	50	K <sup>1</sup>	K <sup>1</sup>	-	-	-	-	Ticari hesaplar için Tabelâ Ünvanı bilgisidir. "Hesap Tipi" Ticari olan hesaplar için iletimi zorunludur.
Bildirim Kanalı	informChannel	'M'	AN1	1	O	O	-	-	-	-	Adres kayıt/güncelleme işleminin gerçekleştirildiği kanal bilgisidir. KOLAS'ta tanımlı değerlerden biri olmalıdır.
Açıklama	description	'Maaş Hesabım'	AN0..50	50	O	O	-	-	-	-	Müşteri'nin adres kayıt/güncelleme sırasında girdiği, kolay adresi tanımlayabilecek Açıklama bilgisidir.
Gönderen Müşteri Kimlik Numarası	senderIdentityNo	1234567890	AN7..11	11	-	-	-	-	Z	-	Ödeme işlemini başlatan Gönderen Müşteriye ait geçerli bir TC Kimlik Numarası(N11), Vergi Kimlik Numarası(N10), Yabancı Kimlik Numarası(N11) veya Pasaport Numarası(AN7..9) bilgilerinden birinin bulunduğu alandır.

Kayıt Statüsü	recordStatus	1	N1..2	2	-	-	-	-	-	-	KOLAS'daki kaydın güncel durumunun yer aldığı alandır. Servis çağrılarında iletimi beklenmemektedir. KOLAS tarafından yönetilir.
Oluşturma Zamanı	createdDate	'2020-04-08T13:57:42'	AN19	19	-	-	-	-	-	-	Kaydın ilk oluşturulduğu gün ve zaman bilgisini içerir. YYYY-MM-DDThh:mm:ss formatında KOLAS tarafından oluşturulur.
Son Güncelleme Zamanı	lastModifiedDate	'2020-06-08T13:57:42'	AN19	19	-	-	-	-	-	-	Kaydın son güncellendiği gün ve zaman bilgisini içerir. YYYY-MM-DDThh:mm:ss formatında KOLAS tarafından oluşturulur.

\*N:Nümerik, AN:Alfanümerik

\*\*Z:Zorunlu, O:Opsiyonel, -:iletimi Beklenmiyor, K1:accountType='T' ise zorunlu, K2: systemRefNo ya da identityNo alanlarından en az birinin iletimi zorunlu, K3: Doluysa addressType ve addressValue alanlarının her ikisinin birlikte iletimi zorunlu



```

proxyAccount Object {
    @Size(min = 20, max = 20)
    @Pattern(regexp = "[0-9]*$")
    String systemRefNo;

    @Min(value = 0)
    @Max(value = 9999)
    Integer serviceProviderNo;

    @Min(value = 0)
    @Max(value = 9999)
    Integer memberNo;

    @NotNull
    @Size(min = 7, max = 11)
    @Pattern(regexp = "[0-9]*$")
    String identityNo;

    @NotNull
    @Size(min = 1, max = 1)
    String addressType;

    @NotNull
    @Size(min = 7, max = 50)
    String addressValue;

    @NotNull
    @Size(min = 6, max = 50)
    String accountOwner;

    @NotNull
    @Size(min = 26, max = 26)
    String accountNo;

    @NotNull
    @Size(min = 1, max = 1)
    String accountType;

    @Size(min = 0, max = 50)
    String tradeName;

    @Size(min = 0, max = 1)
    String informChannel;

    @Min(value = 0)
    @Max(value = 99)
    Integer recordStatus;

    @Size(min = 0, max = 50)
    String description;

    LocalDateTime createdAt;
    LocalDateTime lastModifiedDate;
}

```

## 6.3. Servis Metodları

### 6.3.1. Token Oluşturma – Authorization Servisi

Tüm servisler RESTful JSON olarak çalışır ve Bearer token yapısı ile yetkilendirilmiştir. Servisler çağrılmadan önce JWT formatında Access token oluşturulmalıdır. Token JWT formatında min 512 bit private key kullanılarak üretilenektir. Token geçerlilik süresi 60 dakikadır. Süre bitiminde yeni token üretilmesi gerekmektedir. Token oluşturma işlemi server-to-server authentication yapısına uygun olan OAuth2 client credentials akışına uygun gerçekleştirilecektir. Doğrulama servislerinin çağırım işlemlerinde alınan Access token http header'ın Authorization alanına aşağıdaki gibi eklenerek gönderilmelidir.

**POST** /api/v1/oauth/token

**Http Header**

Content-Type: application/json

**Body:**

```
{
  "client_id": "$clientId",
  "client_secret": "$clientSecret",
  "grant_type": "client_credentials"
}
```

**Response: (Örnek)**

```
{
  "access_token": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJlc2VyIiwiaXV0aCI6ImlJPTFVfVFNfUiIsImV4cCI6MTU4MDI3NTU0NH0.9lwcYGNnik13OgDE0Jg-iLmVOvEFwAnc7frujKstt7u23UBcE6h-QNdGcRj4mB-VcyWVL9Htmz0z1hNn70PeKQ",
  "token_type": "Bearer",
  "expires_in": 3599
}
```

Request Object	Response Object
<pre>ClientCredentials {   @NotNull   @Size(min = 4, max = 50)   @JsonProperty("client_id")   String clientId;    @NotNull   @Size(min = 4, max = 100)   @JsonProperty("client_secret")   String clientSecret;    @NotNull   @Size(min = 4, max = 20)   @JsonProperty("grant_type")   String grantType; }</pre>	<pre>MemberJWTToken {    @JsonProperty("access_token")   String accessToken;    @JsonProperty("token_type")   String tokenType;    @JsonProperty("expires_in")   Long expiresIn; }</pre>

**Http Header:**

Authorization: Bearer \$accessToken

**Örnek:**

Authorization: Bearer  
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJlc2VyIiwiaXV0aCI6ImlJPTFVfVFNfUiIsImV4cCI6MTU4MDI3NTU0NH0.9lwcYGNnik13OgDE0Jg-iLmVOvEFwAnc7frujKstt7u23UBcE6h-QNdGcRj4mB-VcyWVL9Htmz0z1hNn70PeKQ

**6.3.2. Kolay Adres Ekleme Servisi**

Yeni bir kolay adres kaydı ekleme işlevini yerine getiren servistir. serviceProviderNo ve memberNo alanları erişim gerçekleştiren api kullanıcısına ait bilgilerden alınacak; systemRefNo, recordStatus, createdAt ve lastModifiedDate alanları KOLAS tarafından doldurularak servis yanıtında dönüş yapılacaktır. Diğer tüm bilgilerin iletimi zorunludur. Alanlara ait detaylar “Servis Veri Yapıları” başlığı altında yer almaktadır.

**POST /api/v1/proxyAccount**

Örnek Request 1:	Örnek Response 1:
<pre>{   "identityNo": "12345678901",   "addressType": "T",   "addressValue": "5555555555",   "accountOwner": "SERKAN EREN",   "accountNo": "TR120800001234567890123456",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "Maaş Hesabım" }</pre>	<pre>{   "systemRefNo": "12345678901234567890",   "serviceProviderNo": 0800,   "memberNo": 0800,   "identityNo": "12345678901",   "addressType": "T",   "addressValue": "5555555555",   "accountOwner": "SERKAN EREN",   "accountNo": "TR120800001234567890123456",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "Maaş Hesabım", }</pre>

<pre> } </pre>	<pre> "recordStatus": 1, "createdDate": "2020-06-12T13:57:42", "lastModifiedDate": "2020-06-12T13:57:42" } </pre>
<p><b>Örnek Request 2:</b></p> <pre> {   "identityNo": "12345678901",   "addressType": "E",   "addressValue": "Serkan.eren@loremipsum.com",   "accountOwner": "M. SERKAN EREN",   "accountNo": "TR120800001234567890123459",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "Şahsi Hesabım" } </pre>	<p><b>Örnek Response 2:</b></p> <pre> {   "systemRefNo": "12345678901234567891",   "serviceProviderNo": 0800,   "memberNo": 0800,   "identityNo": "12345678901",   "addressType": "E",   "addressValue": "Serkan.eren@loremipsum.com",   "accountOwner": "M. SERKAN EREN",   "accountNo": "TR120800001234567890123459",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "Maaş Hesabım",   "recordStatus": 1,   "createdDate": "2020-06-12T13:59:49",   "lastModifiedDate": "2020-06-12T13:59:49" } </pre>
<p><b>Örnek Request 3:</b></p> <pre> {   "identityNo": "12345678901",   "addressType": "K",   "addressValue": "12345678901",   "accountOwner": "SERKAN EREN",   "accountNo": "TR120800001234567890123457",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "X Bank Hesabım" } </pre>	<p><b>Örnek Response 3:</b></p> <pre> {   "systemRefNo": "12345678901234567892",   "serviceProviderNo": 0800,   "memberNo": 0800,   "identityNo": "1234567890",   "addressType": "K",   "addressValue": "12345678901",   "accountOwner": "SERKAN EREN",   "accountNo": "TR120800001234567890123457",   "accountType": "B",   "tradeName": "",   "informChannel": "M",   "description": "X Bank Hesabım",   "recordStatus": 1,   "createdDate": "2020-06-13T16:01:07",   "lastModifiedDate": "2020-06-13T16:01:07" } </pre>

### 6.3.3. Kolay Adres Güncelleme Servisi

Var olan bir kolay adres kaydının güncellenmesi işlevini yerine getiren servistir. serviceProviderNo ve memberNo alanları erişim gerçekleştiren api kullanıcısına ait bilgilerden alınacak; recordStatus, createdDate ve lastModifiedDate alanları KOLAS tarafından doldurularak servis yanıtında dönüş yapılacaktır. Diğer tüm alanların iletimi zorunludur. Güncellenecek adres kaydının belirlenmesinde Katılımcı tarafından iletilen systemRefNo değeri kullanılacaktır. Katılımcı'lar yalnızca kendilerine ait kolay adres kaydını güncelleyebileceklerdir. Alanlara ait detaylar "Servis Veri Yapıları" başlığı altında yer almaktadır.

PUT /api/v1/proxyAccount

<p><b>Örnek Request:</b></p> <pre> {   "systemRefNo": "12345678901234567890",   "identityNo": "1234567890",   "addressType": "T",   "addressValue": "5555555555",   "accountOwner": "SERKAN EREN", </pre>	<p><b>Örnek Response:</b></p> <pre> {   "systemRefNo": "12345678901234567890",   "serviceProviderNo": 0800,   "memberNo": 0800,   "identityNo": "1234567890",   "addressType": "T",   "addressValue": "5555555555",   "accountOwner": "SERKAN EREN", </pre>
---	---

<pre> "accountNo": "TR120800001234567890123458", "accountType": "T", "tradeName": "BANKALARARASI KART MERKEZİ", "informChannel": "M", "description": "Şirket hesabım" } </pre>	<pre> "accountNo": "TR120800001234567890123458", "accountType": "T", "tradeName": "BANKALARARASI KART MERKEZİ", "informChannel": "M", "description": "Şirket hesabım", "recordStatus": 1, "createdDate": "2020-05-03T13:57:42", "lastModifiedDate": "2020-06- 12T16:32:27" } </pre>
--	---

#### 6.3.4. Kolay Adres Silme Servisi

Var olan bir kolay adres kaydının silinmesi işlevini yerine getiren servistir. Silinecek adres kaydının belirlenmesinde Katılımcı tarafından iletilen systemRefNo değeri kullanılacaktır. Katılımcı'lar yalnızca kendilerine ait kolay adres kaydını silebileceklerdir. systemRefNo alanına ait detay bilgiler "Servis Veri Yapıları" başlığı altında yer almaktadır.

**DELETE** /api/v1/proxyAccount/{systemRefNo}

<p><b>Örnek Request :</b></p> <p><b>DELETE</b> /api/v1/proxyAccount/12345678901234567890</p>	<p><b>Örnek Response: (HTTP_NO_CONTENT)</b></p> <p>{}</p>
--	---

#### 6.3.5. Kolay Adres Sorgulama Servisi

Katılımcı'nın KOLAS'ta yer alan kayıtlarını sorgulaması işlevini yerine getiren servistir. systemRefNo, identityNo ya da addressType ve addressValue değerlerinin her ikisi ile sorgulama seçeneği bulunmaktadır. Sorgu sonucunda yalnızca çağrıyı gerçekleştiren Katılımcı'ya ait kayıtlar dönüş yapılacaktır. Alanlara ait detaylar "Servis Veri Yapıları" başlığı altında yer almaktadır.

**GET** /api/v1/proxyAccount/{systemRefNo}

<p><b>Örnek Request 1:</b></p> <p><b>GET</b> /api/v1/proxyAccount/12345678901234567892</p>	<p><b>Örnek Response 1:</b></p> <pre> { "systemRefNo": "12345678901234567892", "serviceProviderNo": 0800, "memberNo": 0800, "identityNo": "1234567890", "addressType": "K", "addressValue": "12345678901", "accountOwner": "SERKAN EREN", "accountNo": "TR120800001234567890123457", "accountType": "B", "tradeName": "", "informChannel": "M", "description": "X Bank Hesabım", "recordStatus": 1, "createdDate": "2020-06-13T16:01:07", "lastModifiedDate": "2020-06-13T16:01:07" } </pre>
--	--

**POST** /api/v1/proxyAccount/query

<p><b>Örnek Request 2:</b></p> <pre> { "identityNo": "1234567890" } </pre>	<p><b>Örnek Response 2:</b></p> <pre> [ { "systemRefNo": "12345678901234567891", "serviceProviderNo": 0800, "memberNo": 0800, "identityNo": "12345678901", "addressType": "E", "addressValue": "Serkan.eren@loremipsum.com", "accountOwner": "M. SERKAN EREN", "accountNo": "TR120800001234567890123459", </pre>
--	--

	<pre> "accountType": "B", "tradeName": "", "informChannel": "M", "description": "Maaş Hesabım", "recordStatus": 1, "createdDate": "2020-06-12T13:59:49", "lastModifiedDate": "2020-06-12T13:59:49" }, { "systemRefNo": "12345678901234567892", "serviceProviderNo": 0800, "memberNo": 0800, "identityNo": "1234567890", "addressType": "K", "addressValue": "12345678901", "accountOwner": "SERKAN EREN", "accountNo": "TR120800001234567890123457", "accountType": "B", "tradeName": "", "informChannel": "M", "description": "X Bank Hesabım", "recordStatus": 1, "createdDate": "2020-06-13T16:01:07", "lastModifiedDate": "2020-06-13T16:01:07" } ] </pre>
<p><b>Örnek Request 3:</b></p> <pre> {   "addressType": "E",   "addressValue":   "Serkan.eren@loremipsum.com" } </pre>	<p><b>Örnek Response 3:</b></p> <pre> [   {     "systemRefNo": "12345678901234567891",     "serviceProviderNo": 0800,     "memberNo": 0800,     "identityNo": "12345678901",     "addressType": "E",     "addressValue":     "Serkan.eren@loremipsum.com",     "accountOwner": "M. SERKAN EREN",     "accountNo":     "TR120800001234567890123459",     "accountType": "B",     "tradeName": "",     "informChannel": "M",     "description": "Maaş Hesabım",     "recordStatus": 1,     "createdDate": "2020-06-12T13:59:49",     "lastModifiedDate": "2020-06-12T13:59:49"   } ] </pre>

### 6.3.6. Alıcı Sorgulama Servisi

Ödeme akışı öncesinde Gönderici Katılımcı'nın Alıcı Müşteri'ye ait hesap bilgilerini sorgulaması işlevini yerine getiren servistir. addressType ve addressValue değerlerinin her ikisinin iletimi zorunludur. Sorgu sonucunda hangi Katılımcı'ya ait olduğuna bakılmaksızın iletilen addressType ve addressValue değerlerine karşılık gelen accountOwner, accountNo, accountType ve tradeName bilgileri dönüş yapılacaktır. Alanlara ait detaylar "Servis Veri Yapıları" başlığı altında yer almaktadır.

POST /api/v1/proxyAccount/recipientQuery

<p><b>Örnek Request:</b></p> <pre> {   "addressType": "T",   "addressValue": "5555555555",   "senderIdentityNo": "21123456789" } </pre>	<p><b>Örnek Response:</b></p> <pre> {   "accountOwner": "SERKAN EREN",   "accountNo": "TR120800001234567890123458",   "accountType": "T",   "tradeName": "BANKALARARASI KART MERKEZİ" } </pre>
---	--

### 6.3.7. Taşınan Adres Sorgulama Servisi

Belirli bir zaman dilimi içerisinde servis erişimini gerçekleştiren Katılımcı'dan başka bir Katılımcı'ya taşınmış olan kolay adres bilgilerinin sorgulanması işlevini yerine getiren servistir. transferStartDate ve transferEndDate parametrelerinin iletimi zorunludur. İletilen başlangıç ve bitiş zaman dilimi içerisinde taşınmış olan kolay adres bilgileri dönüş yapacaktır. Request alanlarına ait detay bilgiler aşağıda, response alanlarına ait detaylar "Servis Veri Yapıları" başlığı altında yer almaktadır.

Alan Adı	JSON Alan Adı	Örnek	Format	Maks.Uzunluk	Taşınan Adres Sorgulama	Açıklama
Sorgu Başlangıç Zamanı	transferStartDate	'2020-06-12T00:00:00'	AN19	19	Z	Taşınan adres sorgusu için, sorgulama yapılacak zaman aralığı başlangıcı bilgisini içerir. YYYY-MM-DDThh:mm:ss formatında iletilmelidir.
Sorgu Bitiş Zamanı	transferEndDate	'2020-06-12T23:59:59'	AN19	19	Z	Taşınan adres sorgusu için, sorgulama yapılacak zaman aralığı bitiş bilgisini içerir. YYYY-MM-DDThh:mm:ss formatında iletilmelidir..

```
transferQueryRequest Object {
    LocalDateTime transferStartDate;
    LocalDateTime transferEndDate;
}
```

#### GET

/api/v1/proxyAccount?transferStartDate={transferStartDate}&transferEndDate={transferEndDate}

#### Örnek Request:

```
GET
/api/v1/proxyAccount?transferStartDate=2020-06-01T00:00:00&transferEndDate=2020-06-01T23:59:59
```

#### Örnek Response:

```
[
  {
    "systemRefNo":
"12345678901234567898",
    "serviceProviderNo": 0800,
    "memberNo": 0800,
    "identityNo": "654321789",
    "addressType": "T",
    "addressValue": "4444444444",
    "accountOwner": "TUNCAY SAHIN",
    "accountNo":
"TR120800001234567890123451",
    "accountType": "B",
    "tradeName": "",
    "informChannel": "M",
    "description": "",
    "recordStatus": 5,
    "createdDate": "2020-02-01T08:19:43",
    "lastModifiedDate": "2020-06-01T01:23:59"
  },
  {
    "systemRefNo":
"12345678901234567893",
    "serviceProviderNo": 0800,
    "memberNo": 0800,
    "identityNo": "963852741",
    "addressType": "E",
    "addressValue":
"zulal.cagcag@loremipsum.org",
    "accountOwner": "ZÜLAL CAĞCAĞ",
    "accountNo":
"TR120800001234567890123453",
    "accountType": "B",
    "tradeName": "",
    "informChannel": "I",
    "description": "",
    "recordStatus": 5,
    "createdDate": "2020-04-23T16:48:37",
```

	<pre> "lastModifiedDate": "2020-06- 01T07:35:11" } ] </pre>
--	---

## 6.4. Servis Hata Yönetimi

KOLAS, bir API isteğinin başarılı veya başarısız olduğunu belirtmek için geleneksel HTTP yanıt kodlarını kullanır. 2xx aralığındaki kodlar (Örnek:200 ok, 201 created vb) işlemin başarılı olduğunu gösterir. 4xx aralığındaki kodlar, servise gelen bilgiler göz önüne alındığında istemci taraflı bir hata olduğunu ve başarısız bir işlemi gösterir. (Örnek: 404 badrequest, 404 not found, 401 unauthorized vb). 5xx aralığındaki kodlar sunucu kaynaklı bir problemin olduğunu gösterir. (Örnek: 500 Internal Server Error)

200 - OK	Başarılı işlem
400 - Bad Request	<p>İstek kabul edilmedi, parametre eksik veya hatalı</p> <pre> {   "type": "constraint-violation",   "title": "Method argument not valid",   "status": 400,   "path": "/api/v1/proxyAccount",   "message": "error.validation",   "fieldErrors": [     {       "objectName": "verification",       "field": "memberNo",       "message": "Size"     }   ] } </pre>
401 - Unauthorized	<p>Geçerli bir token yok</p> <pre> {   "type": "problem-with-message",   "title": "Unauthorized",   "status": 401,   "detail": "Full authentication is required to access this resource",   "path": "/api/v1/proxyAccount",   "message": "error.http.401" } </pre>
402 - Request Failed	Parametreler geçerli, istek başarısız oldu.

403 - Forbidden	Token'ın isteği gerçekleştirme izni yok <pre>{   "type": "problem-with-message",   "title": "Forbidden",   "status": 403,   "detail": "Access is denied",   "path": "/api/v1/proxyAccount",   "message": "error.http.403" }</pre>
404 - Not Found	Talep edilen kaynak bulunamadı <pre>{   "timestamp": "2020-02-03T22:39:59.352+0000",   "status": 404,   "error": "Not Found",   "message": "Not Found",   "path": "/api/v1/account" }</pre>
429 - Too Many Requests	Çok fazla istek RESPONSE: HTTP/1.1 429 Too Many Requests API rate limit exceeded
500, 502, 503, 504 - Server Errors	Sunucu tarafında bir hata oluştu <pre>{   "type": "problem-with-message",   "title": "Internal Server Error",   "status": 500,   "detail": "",   "path": "/api/v1/proxyAccount",   "message": "error.http.500" }</pre>