

Sovos Bulut e-İrsaliye Web Servis API Kılavuzu V1.3



Sovos Bulut e-İrsaliye Web Servis API Kılavuzuna **Hoşgeldiniz!**

e-İrsaliye Çözümünün özelliklerini, işlevlerini ve standart çalıştırma talimatlarını öğrenmek için bu belgeye bakın.

Feragatname

BU YAYIN KAPSAMINDAKİ KONUYLA İLGİLİ DOĞRU VE YETKİLİ BİLGİLER SUNMAK AMACIYLA TASARLANMIŞTIR. YAYINCININ HUKUKİ, MUHASEBE VEYA DİĞER PROFESYONEL HİZMET SUNMADIĞININ ANLAYIŞI İLE SAĞLANIR. HUKUKİ TAVSİYE VEYA BAŞKA BİR UZMAN YARDIMI GEREKİRSE, YETKİN BİR PROFESYONEL KİŞİDEN HİZMETLER ARAŞTIRILMALIDIR.

Burada atıfta bulunulan tüm ticari isimler, ilgili şirketlerin ticari markaları veya tescilli ticari markalarıdır. Kılavuzun yeni baskıları, önceki baskıdan bu yana güncellenen tüm materyalleri içerecektir.

Doküman Hakkında

Bu doküman Sovos Bulut e-İrsaliye Özel Entegratörlüğü sisteminin sağladığı Web Servis entegrasyonunun kullanım detaylarını içermektedir. Web servise ve güncel UBL-TR e-İrsaliye belgelerine ait teknik dokümanlar (WSDL, XSD, XML vb.) bu dokümanın bulunduğu pakete eklenmiştir.

Bu dokümanın en güncel versiyonuna [linkten](#) ulaşabilirsiniz.

Kısaltmalar

Kısaltma	Açıklama
Bulut e-İrsaliye	Sovos Bulut e-İrsaliye Özel Entegratörlük Platformu
Bulut OM	Sovos Bulut Operasyon Merkezi
GİB	Gelir İdaresi Başkanlığı
VKN	Vergi Kimlik Numarası
TCKN	T.C. Kimlik Numarası
Identifier (Alias)	Etiket Bilgisi
WS API	Web Servis Uygulama Programlama Arabirimi (Web Service Application Programming Interface)
Portal	Sovos Bulut e-İrsaliye Portalı

Destek ve Servis

Bu ürünle ilgili sorun yaşıyorsanız veya sorularınız varsa, lütfen Sovos Destek ile iletişime geçin veya aşağıdaki URL aracılığıyla Sovos Portalı'nda bir vaka gönderin: <https://sovos.com/portal>. Sovos Desteęi hakkında genel daha fazla bilgi için lütfen aşağıdaki URL'yi ziyaret edin: <https://sovos.com/support>.

DOKÜMAN İÇERİĞİ

FERAGATNAME	2
DOKÜMAN HAKKINDA	3
Kısaltmalar	3
DESTEK VE SERVİS	4
GENEL BİLGİLER	6
e-İrsaliye Gönderilen Zarf/Belge Süreci	6
E-İRSALİYE WEB SERVİS API	7
Zarf, İrsaliye ve İrsaliye Yanıtı Gönderme (sendDesUBL())	8
Genel Bilgiler	8
e-İrsaliye için Sanal VKN ve Sanal Etiket Bilgileri	9
Dijital İmzalar	10
Sistem Yanıtları	10
İrsaliye ID'leri ve Otomatik Üretilmesi	10
Müşteri İrsaliye Numaraları (CustDesID)	11
İrsaliye-İrsaliye Yanıtı Görüntüleri ve XSLT Dosyaları	12
Zarf Durum Sorgusu ve Tekrar Gönderimi	13
Gönderilen zarf, irsaliye ve irsaliye yanıtı kontrolleri	13
Hata Yönetimi	13
IS_DRAFT Kullanımı	14
Resend Kullanımı	15
Gelen/Gönderilen UBL Belge Listesini Alma (getDesUBLList())	16
Belge İndirme (getDesUBL())	18
Zarf Durum Sorgulama (getDesEnvelopeStatus())	19
Belge Görüntüsü Alma (getDesView())	20
Kullanıcı Listesi (getDesUserList())	21
Parçalı Kullanıcı Listesi (getDesPartialUserList())	22
İrsaliye Yanıtı Sorgulama (getDesReceipts())	23
API LIMITLERİ VE OPTİMİZASYON	24
WS API HATA KODLARI VE AÇIKLAMALARI	26
GİB E-FATURA SİSTEM YANITI KODLARI	27
TEST YÖNTEMİ	28
EK1 – BAĞLANTI KLAVUZU	29
EK2 – YETKİLENDİRME (AUTHENTICATION)	29
EK3 - .NET İLE API PROJESİ YARATMA VE METODLARI ÇAĞIRMA	30
EK-4 SOAP UI AUTHENTICATION VE TLS AYARLARI	34

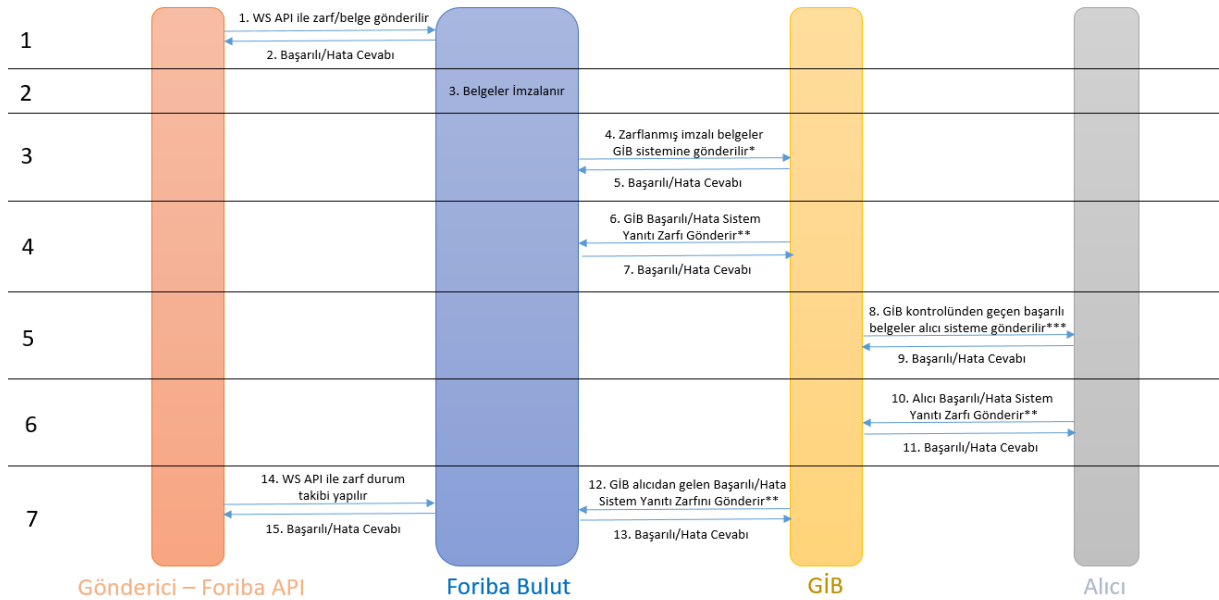
Genel Bilgiler

Bulut e-İrsaliye WS API, SOAP standardıyla uyumlu olarak GİB'in Türkiye'de e-İrsaliye için kullanıma sunduğu UBL-TR doküman formatlarıyla çalışmaktadır. API entegrasyonu öncesinde, e-İrsaliye'nin kullanımı hakkında yazılımcıların mutlaka efatura.gov.tr adresinde bulunan e-İrsaliye teknik klavuzları, belge alan açıklamalarını ve UBL-TR teknik belgeleri (XML şemalar ve XML örnekleri) incelemeleri ve GİB'in yayınladığı güncellemeleri takip etmeleri gerekmektedir.

Bulut e-İrsaliye sistemi, GİB'in güncel şema-şematron dosyaları ve diğer zorunluluklara uygun şekilde çalışmaktadır. Dokümanın yayım tarihinden sonra UBL-TR klavuzları ve paketlerinde GİB'in yayınlayacağı güncellemeleri takip ederek sistemlerini güncel tutmak entegre sistemlerin sorumluluğundadır. GİB klavuzlarında belirtilen bilgilere, bu doküman paketinde tekrar yer verilmemiştir.

API'ye erişim için öncelikle Sovos Bulut OM ile iletişime geçilerek müşteri kaydı yapılmalı ve kullanılacak VKN/TCKN'ye ait bir web servis kullanıcı adı ve şifre alınmalıdır.

e-İrsaliye Gönderilen Zarf/Belge Süreci



* GİB sistemine erişilemediği durumlarda, sistem tekrar açılana ve dokümanlar başarıyla gönderilene kadar tekrar deneme süresiz otomatik olarak yapılır

** GİB ya da alıcı sisteminden sistem yanıtı alınamayan/son duruma ulaşmayan gönderilen belgeler, bulut sistem tarafından düzenli aralıklarla süresiz sorgulanmaya devam eder

*** GİB sisteminden alıcı sisteme gönderilemeyen zarflar GİB sisteminde 1210 kodunu alır ve 24 saat boyunca 3 defa daha tekrar denemir. Denemeler sonucu hala gönderim başarısızsa 1215 koduyla süreç sonlanır

E-İrsaliye Web Servis API

Sovos Bulut e-İrsaliye çözümünü web servisler aracılığıyla kullanmak isteyen mükellefler, sunulan entegrasyon altyapısı aracılığıyla tüm e-İrsaliye ihtiyaçlarını başka bir arayüz/servis kullanmaksızın gerçekleştirebilirler. Sovos Bulut e-İrsaliye web servisi, Sovos Bulut Portali ile entegre çalışmakta ve tüm gönderilen/alınan belgelere bu portal aracılığıyla da erişilebilmektedir.

Gelir İdaresi Başkanlığı (GİB)'nin tanımladığı UBL-TR formatına uygun şema-şematron kontrolleri, kullanıcı-yetki kontrolleri, imzalama/imza validasyonları ve gönderim/alım ile ilgili tüm işlemler arkaplanda sistem tarafından otomatik olarak gerçekleştirilmektedir. Belge alıcılarının kontrolleri GIB'in kayıtlı e- İrsaliye kullanıcı listelerine göre otomatik yapıp kayıtlı olmayan mükelleflere belge gönderilmesi önlenmektedir.

API erişimleri kötü niyetli/yetkisiz erişim denemelerine karşı güvenlik katmanları tarafından sürekli olarak izlenmekte ve otomatik erişim kısıtlamaları uygulanmaktadır. Müşteriler başvuru esnasında ilettikleri bilgi formlarında bulunan VKN/TCKN, etiket, kullanıcı adı ve şifre bilgileriyle "HTTP Basic Authentication" yöntemiyle sistemlere bağlanabilmektedir ve bağlantı HTTPS üzerinden gerçekleşmektedir. Müşteriler servisler üzerinden yalnızca API Key'lerinin bağlı olduğu VKN/TCKN'ye ait belgelerle ilgili işlemleri gerçekleştirebilirler.

Dışarıdan gelen tüm paketler ağ güvenlik donanımlarıyla da sürekli olarak izlenmektedir, herhangi bir otomatik blokajdan korunmak için sisteme yüksek sayıda ya da döngü halinde hatalı çağrılar yapılmamalıdır. Servis metodlarına tüm başarılı ve başarısız erişimler kayıt altına alınmaktadır ve hizmet kalitesi 7x24 takip edilmektedir.

Web servis tanımı ekli **ClientEDespatchServices-1.0.wsdl** ve referans verilen şema dosyalarında bulunmaktadır. Servis metodlarının nasıl kullanılacağı bu doküman içerisinde açıklanmıştır. Sunucular üzerinden WSDL/XSD dosyaların alımı güvenlik ve uyumluluk sebebiyle engellenmiştir. Web servis adresleri, bu klavuzun da bulunduğu <https://api.fitbulut.com/docs> adresinden sağlanmaktadır. Servis durumları da yine <https://api.fitbulut.com/> adresinden takip edilebilmektedir.

Bulut e-İrsaliye web servis metodları, toplu işlemlerde hem istemci hem de sunucu sistemlerde her zaman toplamda çok daha yüksek performans sunmaktadır. Bu yüzden tüm servis metodları için mümkün olduğunca metodlarda belirtildiği şekilde toplu işlemler yapılarak müşteri ve bulut sistemleri için daha az çağrı sayısı ve yüksek performans sağlanması hedeflenmelidir.

Sistemdeki ETTN (UUID)'ler, GIB sisteminde de olduğu gibi büyük-küçük harf duyarlıdır. Tüm metodlar için en az 1 dakikalık gönderim ve 5 dakikalık alım zaman aşımı (timeout) süresi belirlenmesi, zaman aşımı hatalarını engellemek için tavsiye edilmektedir.

Zarf, İrsaliye ve İrsaliye Yanıtı Gönderme (sendDesUBL())

İstek	
VKN_TCKN	Gönderici VKN/TCKN
SenderIdentifier (Opsiyonel)	Gönderici Etiketi (yalnızca zarfsız gönderimlerde)
ReceiverIdentifier (Opsiyonel)	Alıcı Etiketi (yalnızca zarfsız gönderimlerde)
DocType	Belge Türü (ENVELOPE, DESPATCH, RECEIPT)
Parameters (Opsiyonel)	Ek Parametreler
DocData	Ziplenmiş UBL XML verisi

Cevap (Çoklu)	
EnvUUID	Zarf ETTN
UUID	Belge ETTN
ID	Belge ID
CustDesID(Opsiyonel)	Müşteri İrsaliye ID (İrsaliye ID otomatik yaratılıyorsa)

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

Genel Bilgiler

Bu metodla irsaliye(DespatchAdvice) ve irsaliye yanıtı(ReceiptAdvice) belgeleri toplu olarak zarflı ya da tek tek zarfsız gönderilebilir. Zarfsız gönderimlerde etiket bilgilerinin iletilebilmesi için SenderIdentifier ve ReceiverIdentifier parametreleri gönderici ve alıcının etiket bilgileriyle doldurulmuş olmalıdır.

DocType parametresi irsaliye/irsaliye yanıtı zarfı gönderimlerinde “ENVELOPE”, irsaliye gönderimlerinde “DESPATCH” ve irsaliye yanıtı gönderimlerinde “RECEIPT” olmalıdır. Mümkün olan her durumda, tekil belge gönderimi yerine gönderici/alıcı başına gruplanarak zarflı gönderimler yapılmalıdır.

Zip verisi sıkıştırılmış tek bir UBL XML dosyası içermelidir, ve bu XML dosyanın ismi gönderilen doküman türüne bağlı olarak belge UUID’si ile aynı olmalıdır. Örneğin irsaliye UUID’si “c674822e-cce4-48a9-a4e2-e7552aeee83a” olan bir UBL irsaliye gönderimi yapılıyorsa, zip verisi içerisinde “c674822e-cce4-48a9-a4e2-e7552aeee83a.xml” isimli tek bir XML dosya bulunmalıdır ve bu XML’de irsaliye şemasına uygun tek bir irsaliye bulunmalıdır.

Her gönderim sonrasında cevap yapısında sistemde oluşmuş olan her bir belge için zarf UUID, belge UUID & ID ve eğer otomatik ID oluşturuluyorsa, ID oluşturmakta kullanılan müşteri irsaliye numaraları (CustDesID – bkz 5.1.4 & 5.1.5) dönülmektedir.

UBL formatına uygun olarak tüm belge UUID’leri, diğer tüm verilerle birlikte müşteri sisteminde oluşturulup gönderilmelidir. Bulut e-İrsaliye sistemi gönderilen UBL XML’lerin kontrollerini yaptıktan sonra herhangi bir veri zenginleştirme işlemi yapmadan kaydetmekte ve GIB’e iletilmektedir. Buna istisna olan durumlar irsaliye ID’lerinin sistem tarafından oluşturulması ve var olan dijital imzaların sistem tarafından silinerek entegratör mali mührü ile belgelerin imzalanmasıdır.

e-İrsaliye için Sanal VKN ve Sanal Etiket Bilgileri

10.01.2020 tarihli yayınlanan schematron dosyası üzerinde e-İrsaliye için sanal vkn ve sanal etiket yayınlanmıştır. Bu yayınlanan schematron ile birlikte alıcısı e-İrsaliye mükellefi olmayan, irsaliye belgesi için GIB tarafında sanal posta kutusu oluşturulmuştur.

Sanal VKN:	3900892152
Sanal Alıcı Etiket:	urn:mail:irsaliyek@gib.gov.tr

sendDesUBL ws metodu ile zarflı e-İrsaliye gönderiminde, e-İrsaliye mükellefi olmayan müşteriler için, İrsaliye belgesi içerisinde bulunan DeliveryCustomerParty alanına e-İrsaliye göndermesi istenen müşteri bilgileri yazılmalıdır. Zarfın Receiver elementinin identifier (urn:mail:irsaliyek@gib.gov.tr) ve contact (3900892152) alanları sanal alıcı bilgileri ile doldurulmalıdır.

sendDesUBL ws metodu ile zarflı e-İrsaliye gönderimi için;

Örnek sendDesUBL Request:

```
<edes:sendDesUBLRequest>
<edes:VKN_TCKN>1234567890</edes:VKN_TCKN>
<edes:DocType>ENVELOPE</edes:DocType>
<edes:DocData>cid:303213566433</edes:DocData>
</edes:sendDesUBLRequest>
```

sendDesUBL ws metodu ile zarfsız e-İrsaliye gönderiminde, e-İrsaliye mükellefi olmayan müşteriler için, e-İrsaliye belgesi içerisinde bulunan DeliveryCustomerParty alanına e-İrsaliye gönderilmek istenen müşteri bilgileri yazılmalıdır. Zarfsız gönderim de, sendDesUBL requesti üzerinde ReceiverIdentifier (urn:mail:irsaliyek@gib.gov.tr) sanal etiket yazılarak gönderilmelidir. e-İrsaliye mükellefi olmayan müşteriler için receiverIdentifier parametresine sanal etiket yazılarak request yapılması durumunda, zarf oluşturulurken irsaliye belgesi için zarfın receiver elementinin contact alanına sanal vkn (3900892152) otomatik olarak eklenir.

sendDesUBL ws metodu ile zarfsız e-İrsaliye gönderimi için;

Örnek sendDesUBL Request:

```
<edes:sendDesUBLRequest>
<edes:VKN_TCKN>1234567890</edes:VKN_TCKN>
<edes:SenderIdentifier>urn:mail:defaultgb@test.com</edes:SenderIdentifier>
<edes:ReceiverIdentifier>urn:mail:irsaliyek@gib.gov.tr</edes:ReceiverIdentifier>
<edes:DocType>DESPATCH</edes:DocType>
<edes:DocData>cid:146147510762</edes:DocData>
</edes:sendDesUBLRequest>
```

Dijital İmzalar

Gönderilen zarflı/zarfsız irsaliye ve irsaliye yanıtı belgelerinde dijital imzalara ait düğümler (UBLExtensions ve Signature) bulunmamalıdır. GİB'in yayınladığı UBL-TR şemalarında bu alanlar zorunlu olmakla birlikte, bulut e-İrsaliye sistemi XML verisinde bu alanların varlığını kontrol etmez. Bu doküman paketine eklenen UBL-TR belge şemalarında ilgili alanlar opsiyonel hale getirilmiştir. Sistemsel zorunluluklardan dolayı bu alanları dolu göndermesi gerekebilecek sistemler, UBL şemaya uygun olmak kaidesiyle geçersiz imza bilgileriyle doldurabilirler, bunlar sistem tarafından daha sonra silinecektir. Gerçek imzalar bu metod ile yapılan gönderimler sonrasında zamanlanmış görevlerle özel entegratöre ait mali mühür ile atılmaktadır ve GİB'e sonrasında düzgün imzalı belgeler gönderilmektedir. Sistem dışarıdan imzalı belge kabul etmemektedir.

Sistem Yanıtları

Bulut e-İrsaliye sistemi, GİB'den gelen tüm zarflara otomatik olarak sistem yanıtı zarfları oluşturarak dönüş yapmaktadır. Bu sebeple sistem yanıtı belgeleri bu ya da başka bir API metodu ile gönderilmemektedir. Otomatik oluşturulup dönülen sistem yanıtları ya da zarfları getDesUBLList() ile listesi alınıp sonrasında getDesUBL() metoduyla müşteri sistemine de aktarılması sağlanabilir. GİB'in yayınladığı güncel şema & şematron kontrollerine uygun, geçerli imza taşıyan tüm GİB'en gelen zarflara sistem başarılı yanıt dönmetedir.

İrsaliye ID'leri ve Otomatik Üretilmesi

e-İrsaliye'deki irsaliye ID'leri, kağıt irsaliyelerdeki seri numarasına tekabül etmektedir. E-İrsaliye ID'leri UBL irsaliye belgeleri içerisinde bulunan ID alanında gönderilip alınır. Bu alan müşteri sistemleri tarafından üretilebileceği gibi, platforma mükellefin kayıt aşaması sırasında otomatik üretilmesi seçeneğiyle gönderici birim etiketi başına da tanım yapılabilir. Bir mükellefe ait VKN/TCKN'ye tanımlı bir gönderici birim etiketi için, irsaliye ID'lerinin otomatik ya da manuel işaretlenmesi, gerek web servislerle entegrasyon, gerekse portalden kullanımda aynı şekilde geçerli olacak ve mükellef ya her zaman irsaliye ID'lerini kendisi üretecek, ya da her zaman platform tarafından otomatik ve sıralı olarak oluşturulacaktır. Tanımlar etiket başına yapıldığı için aynı VKN/TCKN'ye ait bir GB etiketi irsaliye ID'lerini manuel olarak oluştururken, ikinci bir GB etiketi için otomatik üretilebilir. Ancak tek bir VKN/TCKN'ye ait aynı irsaliye ID'sini içeren iki farklı irsaliye oluşturulmamalıdır, bu irsaliyeler GIB sisteminden hata alacaktır.

Sovos Bulut E-İrsaliye Platformu tarafından irsaliye ID'lerinin otomatik oluşturulması istenen durumlarda, müşteriler ID'lerin otomatik oluşturulacağı her bir GB etiketi için iki alfanumerik karakterden oluşan (ör. AC, 5D vb) ayrı birer "ön ek" tanımlarlar. Portal, adaptör, ERP ya da web servislerden irsaliye ya da irsaliye zarfı gönderimlerinde, sistem bu irsaliye ID öneki, irsaliye düzenlenme tarihi (irsaliyedeki IssueDate alanı), düzenlenme tarihi yılı ve 9 haneli numerik sıra kullanılarak her bir irsaliye için bir irsaliye ID yaratır. Yaratılan irsaliye ID'leri "DA02018000000001" formatındadır; 3 alfanumerik karakterle başlar, düzenlenme tarihi yılı ve sıralı numerik 9 hane ile devam eder ve toplamda 16 karakterdir. Başlangıçtaki 2 alfanumerik karakter GB etiketi için tanımlanmış olan ön ektir.

Oluşturulan irsaliye ID'lerindeki 3. alfanumerik karakter, irsaliyenin sistem tarafından alınma tarihi ve düzenlenme tarihi arasındaki gün farkı kullanılarak seçilen irsaliye serisini alır. Ör. eğer platform, düzenlenme tarihi aynı gün olan bir irsaliye alırsa (düzenlenme günü = alınma günü), "0" serisi kullanılır ve irsaliye ID'si "DA02018000000001" formatında yaratılır. Eğer platform bir önceki günde düzenlenmiş bir irsaliye alırsa, bu durumda "1" serisi kullanılır ve "DA12018000000001", "DA12018000000002" formatında ID'ler yaratılır. Bu numerik seriler 0 ile 7 arasındadır ve düzenlenme tarihi irsaliyenin platform tarafından alınma tarihinden 7 günden daha eski olan irsaliyeler için de yine 7. seri kullanılır.

Başlangıçtaki 3 alfanumerik karakter bu mantığa göre seçildikten sonra, düzenlenme tarihinin yılı eklenir ve sonrasındaki 9 haneli sıra numarası, her VKN/TCKN ve etiket için bu önek + yıl kombinasyonuna göre sıralı devam eder.

İrsaliye ID'lerinin sistem tarafından otomatik oluşturulması durumunda, müşterinin gönderdiği her irsaliye için bir “**Müşteri İrsaliye No - CustDesID**” belirtmesi gereklidir. Bu numara, başarılı irsaliyelerin tekrar gönderilmesini engellediği gibi, GIB ya da karşı taraftan hata alan zarfların içerisindeki irsaliyelerin de tekrar aynı irsaliye ID'yi alarak gönderilmesini sağlar.

Kayıt esnasında belirlenen irsaliye ID'lerinin otomatik oluşturulması seçeneği ve GB etiketleri için belirlenen ön ekler, tutarlılığın sağlanabilmesi için bu GB etiketiyle ilk irsaliye gönderimi gerçekleştirildikten sonra değiştirilmemelidir.

İrsaliye ID'leri sistemde her etiket ve VKN/TCKN çifti için tekil üretilmekle birlikte, e-İrsaliye ekosisteminde tekil bir numara değildir, farklı mükelleflerin aynı ID'leri kullanarak irsaliye gönderebilmeleri mümkündür. Okunabilir formatı sebebiyle mükellefler birbirlerine UUID alanı yerine irsaliye ID'leri aracılığıyla irsaliye referansı vermektedirler.

Müşteri İrsaliye Numaraları (CustDesID)

İrsaliye ID'lerinin bir GB etiketi için platform tarafından otomatik üretilmesi durumunda, her yeni irsaliye gönderiminde yeni bir irsaliye ID'sinin oluşmasının önüne geçmek ve hata alan irsaliyeleri yeni zarflarıyla ve aynı ID'lerle gönderebilmek için müşteri irsaliye numaraları(CustDesID) kullanılmaktadır. Bu numara, müşterilerin kendi sistemlerinde bulunan irsaliyeye ait belirleyici numarayı içermelidir ve irsaliye ID'leri otomatik oluşan bir GB etiketiyle yapılan her irsaliye gönderiminde, irsaliye UBL XML'inde bulunmalıdır.

Gönderilen müşteri irsaliye numarası, GB etiketi ve VKN/TCKN çifti için sistemde tekil olmalıdır. CustDesID'ler, irsaliye ID'leri ile birebir eşleşecek şekilde oluşturulur; aynı CustDesID'ye sahip irsaliyeler sisteme gönderildiğinde sistem tarafından her zaman aynı irsaliye ID'lerini alırlar. Örneğin müşteri sisteminde “A1000” irsaliye numarasıyla tutulan ve bulut e-irsaliye sisteminden gönderim esnasında “DA02018000000085” irsaliye ID'sini alan bir irsaliye varsa, aynı VKN/TCKN ve GB etiketiyle başka bir “A1000” müşteri irsaliye numarasıyla gönderim gerçekleşirse, platform daha önce bu numaraya atanmış olan irsaliye ID'yi bulup (“DA02018000000085”) yine aynı ID ile GIB'e göndermektedir.

Bu mekanizma hata durumlarında irsaliyelerin aynı ID ile tekrar gönderilebilmesini sağlar ve başarıyla gönderilmiş irsaliyelerin tekrar gönderilmesini de engeller. Tüm irsaliye ve irsaliye yanıtları için olduğu gibi zarfının durumu henüz sonuçlanmamış olan ya da karşı tarafta başarıyla işlenen (1200/1300) irsaliyelere ait CustDesID'ler tekrar gönderildiğinde sistem hata dönmektedir.

İrsaliye ID'si otomatik oluşan bir GB etiketi kullanılarak gönderilen bir UBL İrsaliye içerisinde CustDesID, AdditionalDocumentReference düğümü içerisinde gönderilmelidir:

```
<cac:AdditionalDocumentReference>
  <cbc:ID>A1000</cbc:ID>
  <cbc:IssueDate>2018-01-01</cbc:IssueDate>
  <cbc:DocumentTypeCode>CUST_DES_ID</cbc:DocumentTypeCode>
</cac:AdditionalDocumentReference>
```

Bu alan alıcılara da aynen iletilerek standart irsaliye görüntüsünde müşteri muhasebe irsaliye numarası olarak görüntülenmektedir.

Servisin cevap yapısında oluşan irsaliye ID'ler yine zarf-irsaliye UUID ve CustDesID'lerle birlikte tüm irsaliyeler için dönmektedir. Böylece sistemde oluşan irsaliye ID'leri, gönderici sisteme de kaydedilebilir.

İrsaliye-İrsaliye Yanıtı Görüntüleri ve XSLT Dosyaları

İrsaliye ve irsaliye yanıtı belgelerinin görüntüleri, UBL-TR irsaliye ve irsaliye yanıtı XML'lerinin, içerisindeki **AdditionalDocumentReference** düğümüne eklenen XSLT belgelerinden geçirilerek HTML dosyası oluşturulmasıyla elde edilmektedir. Sistemden getDesView() metodu ile WS API üzerinden ya da portalden alınabilen, gelen ve gönderilen irsaliye & irsaliye yanıtlarına ait HTML görüntüleri de bu şekilde oluşturulmaktadır, PDF görüntüler ise HTML belgelerden çevrilmektedir.

GİB standart irsaliye ve irsaliye yanıtı görüntüleme XSLT'lerini tasarlayıp kullanıcılara sunmuştur. Bulut e-İrsaliye sistemi, müşterinin GB/PK etiketlerine XSLT tanımları yapılmamışsa ve bu metod ile gönderilen belgelerin içerisine XSLT eklenmemişse, gönderilen belgelere en güncel GİB irsaliye/irsaliye yanıtı XSLT'lerini otomatik olarak eklemektedir.

GİB'in sunduğu standart XSLT'ler yerine kendilerine özel XSLT kullanmak isteyen müşteriler:

1. Bulut Portal üzerinden ilgili e-İrsaliye GB/PK etiketlerine XSLT dosyası tanımlayabilirler. İrsaliyeler için XSLT dosyaları GB etiketlerinde, irsaliye yanıtları ise PK etiketlerinde tanımlanmaktadır.
2. Bulut OM ile iletişime geçerek bu işlemin kendileri için gerçekleştirilmesini talep edebilirler
3. Dinamik olarak aynı GB/PK etiketinde farklı XSLT'ler kullanılmak isteniyorsa, irsaliye ve irsaliye yanıtı belgelerinin her birinin içerisinde gönderim yapabilirler:

```
<cac:AdditionalDocumentReference>
<cbc:ID>0li6ryg3xe1a2z</cbc:ID>
<cbc:IssueDate>2018-01-01</cbc:IssueDate>
<cbc:DocumentTypeCode>XSLT</cbc:DocumentTypeCode>
<cbc:DocumentType>XSLT</cbc:DocumentType>
<cac:Attachment>
<cbc:EmbeddedDocumentBinaryObject
characterSetCode="UTF-8"
encodingCode="Base64"
filename="İrsaliyeID.xslt" mimeType="application/xml">XSLT_BASE64</cbc:EmbeddedDocumentBinaryObject>
</cac:Attachment>
</cac:AdditionalDocumentReference>
```

Burada fileName niteliği mutlaka **İrsaliyeID.xslt** formatında olmalıdır, alıcı sistemler buradaki dokümanın XSLT'si olduğunu bu nitelik ile anlamaktadır. **XSLT_BASE64** yazılı yere ise belgeye eklenecek XSLT dosyası **base64** formatında eklenmelidir.

XSLT belgeleri UBL-TR XML belge boyutlarının en büyük kısmını oluşturduğu için, performans açısından dinamik görüntü gereği olmadığı durumlarda XSLT'ler etiketlere tanımlanmalı ve API üzerinden belge gönderimleri XSLT'siz olarak minimum boyutta tutulmalıdır. XSLT boyutları e-İrsaliye sürecine dahil olan tüm sistemlerin performansını doğrudan etkilemektedir.

Alınan belgelerde yaşanan herhangi bir görüntülenme probleminde, eğer GİB e-Fatura görüntüleyici ile de belge görüntülenemiyorsa, gönderici mükellef ve sistemin sorumlularıyla irtibata geçmek gereklidir. Platform gönderilen ya da alınan belgelerin görüntülerini oluştururken başka bir veri kullanmamaktadır.

Zarf Durum Sorgusu ve Tekrar Gönderimi

Karşı tarafa gönderilen zarflar, GIB ya da karşı sistemde meydana gelen herhangi bir problemten ötürü hata durumlarına düşerek tekrar gönderimleri gerekebilmektedir. (örneğin geçici bir durumu ya da başarılı aktarımı belirten kodlar olan 1200, 1210, 1220 ve 1300 hariç tüm sistem yanıtı kodları: 1150, 1160, 1163, 1171, 1172, 1176, 1195, 1215, 1230 vb). Gönderilen her bir zarfın durumu başarılı ya da hatalı bir son statü koduna ulaşana kadar belirli aralıklarla (ör. 8 saatte bir, her gece vs) `getDesEnvelopeStatus()` metoduyla sorgulandıktan sonra, hata durumuna düşen zarflar gerekli bir düzeltme varsa yapıldıktan sonra tekrar gönderilmelidir. Tekrar gönderimlerde irsaliye ID/CustDesID'ler ve diğer bilgiler aynı tutulmakla birlikte, zarf ve belge UUID'leri yeniden oluşturulmalıdır.

Eğer zarfı geçici bir statüde beklerken ya da başarıyla karşı tarafa iletilmiş olmasına rağmen aynı gönderilen irsaliyeler ya da gelen irsaliyelere verilen irsaliye yanıtları sisteme tekrar yeni bir zarfla gönderilirse, sistem ilgili hata kodlarıyla gönderimleri engelleyecektir. Gönderilen bir irsaliyenin tekrar gönderimi ya da gelen bir irsaliyeye verilen irsaliye yanıtının tekrar gönderimi için öncelikle ilk gönderimlere ait zarfların son duruma ulaşarak başarılı ya da hata durumlarında işlemlerini tamamlamış olması gerekir. Gelen bir irsaliyeye birden fazla irsaliye yanıtı verebilmek için de öncelikle her bir irsaliye yanıtı gönderiminin son duruma ulaşmış olması gereklidir. Bu kontroller müşteri ve özel entegratörlük sistemlerinde tutarlılığın sağlanabilmesi sebebiyle eklenmiştir.

Gönderilen zarf, irsaliye ve irsaliye yanıtı kontrolleri

`sendDesUBL()` metoduyla gönderilen her belge, aşağıdaki kontrollerden geçmektedir:

1. Göndericilerin Bulut E-İrsaliye platformuna kayıt, yetki ve erişim kontrolleri
2. Genel servis parametreleri ve belge boyutları ile tanımlayıcı alanlarının kontrolü
3. Gönderici ve alıcıların GIB'e kayıtlarının kontrolü
4. UBL-TR şema validasyonu
5. UBL-TR şematron validasyonu
6. Zarf ve belgelerin tekiliği & durum kontrolleri

Hata Yönetimi

`sendDesUBL()` metodu, diğer metodlar gibi senkron çalıştığı için cevap yapısı herhangi bir ağ ya da sunucu kesintisi sebebiyle istemci sisteme ulaşmayabilir. Bu durum zarfsız gönderimlerde zarf UUID'nin, irsaliye ID'sinin otomatik oluştuğu irsaliye veya irsaliye zarfı gönderimlerinde ise oluşan irsaliye ID'lerinin istemci sistem tarafından alınamamasına sebep olacaktır.

Sistemden cevap alınamayan bir durumda (tanımlı bir hata kodu ya da başarılı cevabın gelmediği durumlar), sisteme gönderilen belgeler oluşmuş ya da olmamış olabilir, ve aynı veri gönderilirse UUID mevcut hatası alınacaktır.

Bu duruma düşen gönderimler için, `getDesUBLList()` metodu UUID bazlı sorgulanarak gönderilmiş bir irsaliye/irsaliye yanıtı zarf UUID'sine ya da irsaliye ID'sine erişilebilir. Gönderimler bir kuyruklama mantığıyla çoklu olarak yapılıyorsa, bu araftaki belgelerin sorgulanması da yine toplu olarak yapılabilir ve belgelerin durumları topluca güncellenebilir. Ancak sistemden cevap alınamayan, bilgileri dönmeyen gönderimlerin durumları mutlaka kontrol edilerek gönderici sistemde durumları güncellenmelidir.

IS_DRAFT Kullanımı

sendDesUBL WS metodu ile sisteme taslak irsaliye gönderilmek istendiğinde sendDesUBL metodu parameters alanına IS_DRAFT parametresi geçilmelidir, örnek request ve response'u aşağıdaki gibidir. Taslak irsaliyeler portalden tetiklenmediği sürece GİB'e gönderilmez.

IS_DRAFT Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:edes="http://Sovos.com/eDespatch/">
  <soapenv:Header/>
  <soapenv:Body>
    <edes:sendDesUBLRequest>
      <edes:VKN_TCKN>1234567801</edes:VKN_TCKN>
      <!--Optional:-->
      <edes:SenderIdIdentifier>urn:mail:irsaliyegb@sovos.com</edes:SenderIdIdentifier>
      <!--Optional:-->
      <edes:ReceiverIdentifier>urn:mail:irsaliyekp@sovos.com</edes:ReceiverIdentifier>
      <edes:DocType>DESPATCH</edes:DocType>
      <!--Zero or more repetitions:-->
      <edes:Parameters>IS_DRAFT</edes:Parameters>
      <edes:DocData>cid:1086071435206</edes:DocData>
    </edes:sendDesUBLRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

IS_DRAFT Response:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <sendDesUBLResponse xmlns="http://Sovos.com/eDespatch/">
      <Response>
        <EnvUUID>cd5a31cd-6c2e-4595-b4f4-2dbe817db3c2</EnvUUID>
        <UUID>264d3255-5837-4a6a-9589-883bedbd3fc8</UUID>
        <ID>ABC2024000000021</ID>
        <CustDesID>30be09e3-2d3f-4824-8a2f-eed4abe9c52d</CustDesID>
      </Response>
    </sendDesUBLResponse>
  </s:Body>
</s:Envelope>
```

Resend Kullanımı

GIB'den hata alan ve aynı belge UUID ile tekrar gönderilmesi gereken belgeler için e-İrsaliye istemci web servisindeki sendDesUBL methodunda Parameters alanında aşağıdaki gibi bir değer gönderilmelidir.

RESEND:d6013d4d-a611-1111-a2dd-b24f79eeab59

Bu değerin ":" nin soldaki kısmı (RESEND) yapılacak işlemin bir yeniden gönderme işlemi olduğunu belirtmekte, ":" nin sağındaki kısmı (d6013d4d-a611-1111-a2dd-b24f79eeab59) ise yeniden gönderilecek belge zarfının zarf UUID bilgisini belirtmektedir. Ek olarak RecieverIdentifier alanı gönderilmemelidir. Ayrıca Docdata alanı boş veya Docdata alanı gönderilmemelidir.

Örnek Request :

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:edes="http://Sovos.com/eDespatch/">
  <soapenv:Header/>
  <soapenv:Body>
    <edes:sendDesUBLRequest>
      <edes:VKN_TCKN>3880718497</edes:VKN_TCKN>
      <edes:SenderIdentifier>urn:mail:defaultgb@fitcons.com</edes:SenderIdentifier>
      <edes:DocType>ENVELOPE</edes:DocType>
      <edes:Parameters>RESEND:d6013d4d-a611-45cd-a2dd-b24f79eeab59</edes:Parameters>
    </edes:sendDesUBLRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Örnek Response :

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <sendDesUBLResponse xmlns="http://Sovos.com/eDespatch/">
      <Response>
        <EnvUUID>0f060263-8f91-4977-b44d-36dc9f19f77d</EnvUUID> <!--İrsaliye belgesinin yeni zarf
uuid bilgisi-->
        <UUID>d6013d4d-a611-45cd-a2dd-b24f79eeab59</UUID> <!--Hata alan irsaliye belgesi uuid
bilgisi-->
        <ID>FD72020000000067</ID>
        <CustDesID>TST2020090014257</CustDesID>
      </Response>
    </sendDesUBLResponse>
  </S:Body>
</S:Envelope>
```

RESEND edilebilecek statü kodları için aşağıdaki bölümü inceleyiniz.

Zarf Durum Sorgusu ve Tekrar Gönderimi

Gelen/Gönderilen UBL Belge Listesini Alma (getDesUBLList())

İstek	
Identifier	Belgenin gönderici ya da alıcısına ait Gönderici Birim (GB) ya da Posta Kutusu (PK) etiketi
VKN_TCKN	Belgenin gönderici ya da alıcısına ait VKN/TCKN
DocType	Belge Türü (ENVELOPE, DESPATCH, RECEIPT, SYS_RESP)
Type	Gelen/Gönderilen Belge (OUTBOUND, INBOUND)
UUID (Opsiyonel-Çoklu)	Belge UUID listesi
FromDate	Başlangıç tarihi ve zamanı
ToDate	Bitiş tarihi ve zamanı

Cevap (Çoklu)	
UUID	Belge ETTN
Identifier	Belgenin gönderici ya da alıcısına ait Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	Belgenin gönderici ya da alıcısına ait VKN/TCKN
EnvType(Opsiyonel)	Zarf Tipi (SENDERENVELOPE, POSTBOXENVELOPE, SYSTEMENVELOPE)
EnvElementType(Opsiyonel)	Zarf Eleman Tipi (DESPATCHADVICE, RECEIPTADVICE, APPLICATIONRESPONSE)
EnvUUID(Opsiyonel)	Zarf ETTN
ID(Opsiyonel)	Belge ID
CustDesID(Opsiyonel)	Gönderilen irsaliyenin müşteri irsaliye numarası
InsertDateTime	Belge kayıt tarihi ve zamanı

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

Gelen ve gönderilen UBL belgelerin listesi bu metod ile UUID listesiyle ya da bir tarih aralığıyla sistemden alınabilmektedir.(Saklama statüsündeki irsaliyelere ait zarflar bu metod ile alınamamaktadır.) Gönderilen belgelerin listesini alırken Type parametresi “OUTBOUND” gönderilmeli ve belgenin göndericisine ait VKN/TCKN ve etiket kullanılmalıdır. Alınan belgeleri alırken ise Type parametresi “INBOUND” gönderilmeli ve belgenin alıcısına ait VKN/TCKN ve etiket istek parametrelerinde gönderilmelidir.

Gönderilen DocType parametresine göre zarf (ENVELOPE), irsaliye (DESPATCH), irsaliye yanıtı (RECEIPT) ya da sistem yanıtı (SYS_RESP) belgelerinin listesi dönülmektedir. Böylece sistemdeki bir VKN/TCKN-etikete ait istenilen belge türleri liste olarak alınabilir.

Belgelerin detayları UUID listesi ya da tarih aralığıyla alınabilir, ancak her ikisi de aynı anda kullanılamaz. getDesUBLList ws metodu UUID parametresi ile kullanımda, sendDesUBL ile gönderimde response olarak dönen ENVUUID bilgisini ve irsaliye belgesine ait bilgileri (envUUID,identifier, VKN/TCKN,EnvType,EnvElementType,InsertDateTime) tekrar edinmek için kullanılır. UUID ile alımlarda tek bir istekte en fazla 20 UUID gönderilebilir. Tarih aralığıyla belgelerin alımında, XML tarih saat formatıyla(ör.2018-01-01T00:00:00.00+03:00) FromDate-ToDate parametreleri gönderilmelidir.

Tarih aralığıyla arama yapılacaksa FromDate ve ToDate parametreleri aramanın başlangıç-bitiş tarihleri olarak kullanılır ve belgelerin bulut sisteme geliş/gönderim tarihine göre tarama yapılır. Bu tarih, gönderilen belgelerin sistemde oluşma tarihleri, gelen belgelerin ise GIB'den alınma ve işlenme tarihleridir. FromDate ve ToDate tarihleri arasında maksimum 1 günlük süre bulunmalıdır, bu sayede 1 günlük belge listeleri tek seferde alınabilir.

Cevap yapısında istenilen kriterlerdeki belgelere ait ETTN(UUID) ve kayıt tarihi(InsertDateTime) çoklu olarak dönlür. Zarf listelemelerinde, EnvType parametresiyle zarf tipi ve EnvElementType ile zarf eleman tipi dönlür. İrsaliye, irsaliye yanıtı ve sistem yanıtı listelemelerinde ise EnvUUID parametresiyle belgenin içinde bulunduğu zarfa ait ETTN bilgisi, belgenin ID bilgisi ve ID'si sistem tarafından oluşturulmuş gönderilen bir irsaliye alınıyorsa CustDesID dönlür.

Gönderilen belgeler listeleniyorsa alıcıya ait bilgiler cevap yapısındaki Identifier(GB ya da PK etiketi) ve VKN_TCKN parametrelerinde bulunur, gelen belgeler listeleniyorsa da bu alanlarda göndericiye ait bilgiler bulunur.

Bulut e-İrsaliye sistemine mükellef adına GIB'den gelen zarflara istinaden sistem yanıtı zarfları otomatik olarak oluşturulup GIB'e geri gönderilmektedir. Bu sebeple gönderilen sistem yanıtları alınmak istendiğinde bu otomatik oluşup gönderilen yanıtların listesi dönlür. Gönderilen tüm zarflar listelendiğinde, bu otomatik oluşan sistem yanıtı zarfları(SYSTEMENVELOPE) da liste içerisinde dönlmektedir.

Bu metodun ana kullanım amacı, GIB ve karşı sistemden mükellefe gelen belgelerin listesini almaktır. Başlangıç-bitiş tarihleri sorgulanan aralıklar tekrar tekrar sorgulanmamalı, ve birkaç saatlik periyodu bulunan zamanlanmış görevlerle (ör. 6 saatte bir son 6 saat) alınacak şekilde düzenli belge listesi çekilebilir. Tavsiye edilen kullanım yöntemi her zaman belge listesi yerine sadece zarf listesi çekerek belgeler için ayrı ayrı sorgu yapılmamasıdır. Çok sık gelen belge sorgulanması gereken sistemler bu aralığı 1 saate kadar düşürebilir, kullanıcı taleplerine bağlı anlık kontroller ise manuel olarak daha kısa aralıklara sahip ek sorgularla yapılabilir. Belgelerin listesi alındıktan sonra istenilen belgelere ait UBL'ler getDesUBL() metoduyla toplu olarak indirilebilir.

Belge İndirme (getDesUBL())

İstek	
Identifier	Belgenin gönderici ya da alıcısına ait Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	Belgenin gönderici ya da alıcısına ait VKN/TCKN
UUID(Çoklu)	Belge ETTN
DocType	Belge Türü (ENVELOPE, DESPATCH, RECEIPT, SYS_RESP)
Type	Gelen/Gönderilen Belge (OUTBOUND, INBOUND)

Cevap(Çoklu)	
UUID	Belge ETTN
DocData	Zipli belge ikilik verisi

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

Gönderilen ve alınan zarf, irsaliye, irsaliye yanıtı ve sistem yanıtı belgelerine ait UBL XML verileri bu metot ile çoklu olarak alınabilmektedir. Parametrelerin kullanımı getDesUBLList() metoduyla benzer şekildedir, gönderilen belgeler indirilirken gönderici etiket ve VKN/TCKN, alınan belgeler indirilirken alıcı etiket ve VKN/TCKN ilgili istek parametrelerinde gönderilmelidir. getDesUBLList() ile alınan UUID listelerindeki belgeler bu metotla müşteri sistemine indirilebilir, ve tek bir istekte en fazla 20 UUID gönderilebilir.

Belgeler her zaman DocData cevap parametresi içerisinde ziplenmiş olarak dönlür. Tüm entegrasyonlarda, özellikle zamanlanmış görevlerle gelen belgelerin UBL'lerinin sisteme alınmasında, bu metotla belgelerin tek tek yerine toplu olarak indirilmesi performans açısından oldukça avantaj sağlamaktadır ve tavsiye edilmektedir.

Zarf Durum Sorgulama (getDesEnvelopeStatus())

İstek	
Identifier	Zarflar göndericisine ait Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	Zarflar göndericisine ait VKN/TCKN
UUID(Çoklu)	Zarf ETTN
Parameters (Opsiyonel-Çoklu)	Ek istek parametreleri

Cevap(Çoklu)	
UUID	Zarf ETTN
IssueDate	Sistem Yanıtı Düzenlenme Tarihi
DocumentTypeCode (Opsiyonel)	Sorgulanan zarfların tipi (SENDERENVELOPE, POSTBOXENVELOPE)
DocumentCode(Opsiyonel)	Sorgulanan zarfların tipi (SENDERENVELOPE, POSTBOXENVELOPE)
ResponseCode(Opsiyonel)	Sistem Yanıtı Kodu
Description(Opsiyonel)	Sistem Yanıtı Açıklaması
DocData(Opsiyonel)	Zarflar sistem yanıtı zarflarının ikilik verisi

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

sendDesUBL() metoduyla gönderimi yapılan zarfların/belgelerin GİB ve karşı tarafa iletim/işlem durum takibi, zarflar durumları son duruma ulaşana kadar ([bkz. zarfların tekrar gönderimi](#)) bu metot ile yapılmalıdır. Zarf durumları her sorguda GİB sisteminden doğrudan çekilmektedir. GİB sisteminden çekilen zarflar durumu ile Sovos e-İrsaliye sistemindeki mevcut durumun aynı olmaması durumunda e-İrsaliye sistemindeki zarflar durumu GİB sistemindeki zarflar durumuyla aynı olacak şekilde güncellenmektedir.

Dönülen sistem yanıtı kodları GİB e-Fatura sistem yanıtı kılavuzunda belirtilmiştir. Henüz sistemde işlenmekte olan zarflar için "1000", GİB'e gönderilmiş ancak herhangi bir durum bilgisi dönmemiş zarflar için ise "1100" kodu kullanılmaktadır. Bu kodlar hariç dönülen tüm kodlar GİB ya da alıcı sistemden dönülen sistem yanıtından alınmaktadır.

Parameters parametresinde "DOC_DATA" parametresi de gönderildiğinde, çoklu yapıdaki her sistem yanıtıyla birlikte aynı zamanda zarflanmış bir sistem yanıtı UBL dokümanı zip içerisinde dönlümlenmektedir. Bu kullanım performans açısından tavsiye edilmemekle birlikte, sistem yanıtlarını UBL olarak almak zorunda olan sistemlerde kullanılabilir. Bu durumlar hariç, standart cevap yapısında dönülen statü kodu ve açıklaması durum takibi için yeterli olacaktır.

Zarflar durumları zamanlanmış görevlerle çoklu olarak güncellenmelidir, ve bir çağrıyla bu metoda en fazla 20 adet UUID gönderilebilir. Zarf durumlarının GİB sisteminde son statüye gelmesi günler sürebilmektedir, bu sebeple zamanlanmış görevlerle yapılan durum güncelleme işlemlerinde mutlaka toplu sorgularla sistem yanıtları alınmalı ve aynı zarflar durumu bir günde zamanlanmış görevlerle en fazla birkaç defa otomatik

sorgulanmalıdır. İhtiyaç durumunda ek manuel sorgulama yapılabilir. Bu özellikle yüksek hacimli entegrasyonlarda sistemlerin uzun vadeli performansı için oldukça önemlidir.

Belge Görüntüsü Alma (getDesView())

İstek	
Identifier	Belgenin gönderici ya da alıcısına ait Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	Belgenin gönderici ya da alıcısına ait VKN/TCKN
Belge Detayları(Çoklu)	
UUID(Opsiyonel)	Belge ETTN
ID(Opsiyonel)	Belge ID
CustDesID(Opsiyonel)	Müşteri İrsaliye ID
Type	Gelen/Gönderilen Belge (OUTBOUND, INBOUND)
DocType	Belge Türü (DESPATCH/RECEIPT)
ViewType	Doküman Türü (HTML, PDF, XSLT, HTML_DEFAULT, PDF_DEFAULT)

Cevap(Çoklu)	
UUID(Opsiyonel)	Belge ETTN
ID(Opsiyonel)	Belge ID
CustDesID(Opsiyonel)	Müşteri İrsaliye ID
Type	Gelen/Gönderilen Belge (OUTBOUND, INBOUND)
DocType	Belge Türü (DESPATCH/RECEIPT)
ViewType	Belge Görüntü Türü (HTML, PDF, XSLT, HTML_DEFAULT, PDF_DEFAULT)
DocData	Zipli görüntü ikilik verisi
Result	Durum Kodu (0: Başarılı, 1: Hatalı)
ResultDescription	Durum Açıklaması

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

Gelen/gönderilen irsaliye ve irsaliye yanıtı belgelerinin HTML/PDF görüntüleri ve XSLT'lerini almak için kullanılacak metoddur. Gönderilen belgeler için gönderici etiket ve VKN/TCKN, alınan belgeler için ise alıcı etiket ve VKN/TCKN istek parametrelerinde gönderilmelidir.

İrsaliye belgelerinin görüntüleri UUID ve ID bilgileriyle, irsaliye yanıtı görüntüleri ise yalnızca UUID ile alınabilir. Gönderilen irsaliye görüntüleri, irsaliye numarası otomatik oluşan GB etiketleri için müşteri irsaliye numarası (CustDesID) bilgisiyle de alınabilir.

Bu metodla belge görüntüleri, her belge için istenilen farklı formatlarda 20 belgeye kadar topluca alınabilir. Her belgenin detayı ve ziplenmiş görüntüsü/XSLT verisi cevap yapısında dönmektedir. Görüntü oluşturma işleminin başarılı olduğu her belge için Result parametresi 0, hata durumlarında ise 1 döner ve hatanın

sebebi ResultDescription alanından alınabilir. Görüntüsü oluşturulamayan belgeler için, GİB varsayılan (HTML_DEFAULT/PDF_DEFAULT) XSLT'leri ile görüntüler oluşturulabilir.

Kullanıcı Listesi (getDesUserList())

İstek	
Identifier	Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	VKN/TCKN bilgisi
Role	Kullanıcı Tipi (GB/PK)

Cevap	
DocData	Zipli GİB kayıtlı e-Fatura & e-İrsaliye kullanıcı listesi XML'i

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

GİB'in sağladığı E-Fatura & e-İrsaliye sistemlerine ait kullanıcı listesi XML'ini döndürür. İstek parametrelerinde bulut sistemde kayıtlı herhangi bir VKN/TCKN ve etiket çifti kullanılabilir. Role parametresi gönderici birim etiket listesini almak için "GB", posta kutusu etiket listesini almak için ise "PK" olmalıdır.

Alıcı listesi (PK) günlük olarak alınıp istemci sistemlerde yerel olarak kaydedilerek, belge gönderimleri için alıcı kontrolleri ilgili VKN/TCKN ve uygulama tipi üzerinden lokal olarak yapılmalıdır. Günde tek bir PK listesi alınması yeterlidir. Bulut sistem de yine bu liste içerisinde aktif etiketlere göre GİB kayıt kontrolü yapmaktadır.

İndirilen kullanıcı listeleri üzerinde Document Type alanı "DespatchAdvice" ise e-irsaliye, Document Type alanı "Invoice" ise e-fatura etiketleri olduğu ayırt edilebilir.

e-Fatura

```
<Documents>
  <Document type="Invoice">
    <Alias>
      <Name>urn:mail:testgb@test.com</Name>
      <CreationTime>2020-07-06T17:01:50</CreationTime>
    </Alias>
  </Document>
```

e-İrsaliye

```
<Document type="DespatchAdvice">
  <Alias>
    <Name>urn:mail:testgb@test.com.tr</Name>
    <CreationTime>2020-07-02T15:51:16</CreationTime>
  </Alias>
</Document>
```

</Documents>

Parçalı Kullanıcı Listesi (getDesPartialUserList())

İstek	
Identifier	Gönderici Birim (GB) ya da Posta Kutusu (PK) etiketi
VKN_TCKN	VKN/TCKN bilgisi
Role	Kullanıcı Tipi (GB/PK)
IncludeBinary	true ya da false değerini alabilir. (true olarak gönderildiğinde binaryData alanında kullanıcı listesi bölümü(zipli olarak) response'da dönülür, Default değer true'dur false olarak gönderildiğinde ise binaryData boş olarak dönülür. Listenin toplam kaç bölümden oluştuğu ve bölümlerin dosya isimleri talep edildiğinde false olarak kullanılabilir.)
FileNameList(Opsiyonel)	Dosya ismi (Çoklu) (Bütün bölümler tek seferde işlenemediğinde ve/veya bölümler işlenirken spesifik bir bölümde hata alınıp tekrar istendiğinde tüm bölüm listesini tekrar almak yerine dosya adı verilerek sadece ilgili bölüm/bölümler alınabilir.)
Parameters(Opsiyonel)	Parametre listesi, şu anda herhangi bir parametre alınmamaktadır, ileride doğabilecek ekstra durumlar için parametre alanı kullanılacaktır.

Cevap	
totalPartCount	Toplam bölüm sayısı
userCountPerPart	Her bir bölüm (XML) başına içerisinde bulunan kullanıcı (user) sayısı
totalUserCount	Tüm bölümlerinde toplamında bulunan kullanıcı (user) sayısı
correlationID	Tüm listenin bütünlüğünü temsil eden benzersiz UUID (Kullanıcı listesi periyodik olarak güncellenmektedir, her güncelleme sonrası bu UUID değişecektir)
lastUpdatedAt	Kullanıcı listesinin en son güncellendiği tarih ve saat.
fileName	Zipli dosya ismi.
binaryData	İçerisinde 100.000 kullanıcı bulunan zipli XML, base64 encode edilmiş binaryData.

Hata	
Code	Sistem hata kodu
Message	Sistem hata açıklaması

Bilindiği üzere Gelir İdaresi Başkanlığı kayıtlı e-Fatura/ e-İrsaliye kullanıcı listesini periyodik olarak güncelleyip yayınlanmaktadır. Mevcutta sisteme kayıtlı mükellef sayısı yaklaşık 1.5 milyona ulaşmıştır ve hızlı bir şekilde büyümektedir. Kullanıcı listesi XML formatta yayınlanmaktadır ve mevcutta bu XML'in

zipsiz açık hali 1GB + büyüklüğe ulaşmıştır. XML dosyanın bu boyutlara ulaşması, bu listeye ihtiyacı olan, listeyi periyodik olarak indirip işleyen müşterilerimiz için client tarafında yüksek kaynak (memory) ihtiyaçları gerektirmektedir ve büyüme sürekli olarak devam ettiğinden kaynak tedarik edilse bile periyodik olarak kaynak arttırılma ihtiyacı doğurmaktadır.

Bu probleme daha efektif bir çözüm geliştirmek amacıyla Sovos Bulut e-İrsaliye API'ye mevcutta olan `getDesUserList()` metodundan daha efektif ve kullanışlı, yüksek kaynak ihtiyacı gerektirmeyen `getDesPartialUserList()` metodu eklenmiştir.

Bu metot ile 1.5 milyon kullanıcı tek bir XML dosya ile değil, 100.000'lik kullanıcı listesine bölünerek 15 ayrı parçada yine zipli olarak verilmektedir. Yani web servis yanıtında, 15 kullanıcı listesi parçası zipli olarak verilir, bu her bir zipin içerisinde 100.000 adet kullanıcı içeren XML dosyası bulunur. Kullanıcı listesinin büyümesiyle birlikte parça sayısı da büyüyecektir. Örneğin e-Fatura/ e-İrsaliye kullanıcı sayısı 3 milyon olduğunda, `getDesPartialUserList()` metodu 30 zip parçası dönecektir.

`getDesPartialUserList()` metodunun döndüğü kullanıcı listesi parçaları ayrı ayrı zipten çıkartıp işlediğinde, memory problemine yol açmayacaktır, çünkü her bir parça zipli binary data olarak 7-8MB, zipten çıkarıldığında ise 70-80MB boyutlarını aşmayacaktır. (`getDesUserList()` ile alınan tek parça XML dosyasının zipten çıkarılmış hali 1GB'tır).

Not: Mevcut `getDesUserList` metodu client tarafında problemlere yol açabilecek duruma gelmiştir, bu yüzden müşterilerin bu metoda (`getDesPartialUserList()`) entegre olup kullanıcı listesi süreçlerini bu metot üzerinden yürütmesi beklenmektedir. `getDesUserList()` metodu ilerleyen zamanlarda kapatılacaktır.

Not: Sovos Bulut e-İrsaliye sistemindeki kullanıcı listesi periyodik olarak GİB sisteminden alınan kullanıcı listesi ile güncellenmektedir. Sovos Bulut e-İrsaliye sistemindeki kullanıcı listesi bir `correlationID` ile ilişkilendirilmektedir. Her güncelleme sonrası güncellenen listeye ait `correlationID` değiştirilmektedir. Kullanıcı listesi bölümler halinde birden fazla istek yapılarak alınırken (örneğin 15 parça var ise, her bir istekte, dosya isimleri istekte belirtilerek, 3 parça talep edip toplamda 5 istekte tüm liste çekilebilir) dikkat edilmesi gereken tüm isteklerin cevaplarındaki `correlationID` değerinin aynı olmasıdır. Eğer `correlationID` değişirse süreç baştan tekrarlanmalı ve tüm bölümler yeniden alınmalıdır. Çekilen ilk parçaların eski, sonraki parçaların yeni `correlationID`'ye sahip olması durumunda ilk parçalar stale(bayat) durumda kalacaktır.

İrsaliye Yanıtı Sorgulama (`getDesReceipts()`)

İstek	
Identifier	İrsaliyenin göndericisine/alıcısına ait ait Gönderici Birim(GB) ya da Posta Kutusu(PK) etiketi
VKN_TCKN	İrsaliyenin göndericisine/alıcısına ait VKN/TCKN
UUID(Çoklu)	İrsaliye ETTN
Type	Gelen/Gönderilen Belge (OUTBOUND, INBOUND)

Cevap(Çoklu)	
DespatchUUID	İrsaliye ETTN
İrsaliye Yanıtı Detayları(Çoklu)	
EnvUUID	Zarf ETTN
UUID	Belge ETTN

ID	Belge ID
InsertDateTime	Belge kayıt tarihi ve zamanı
IssueDate	Belge düzenleme tarihi
DocData(Opsiyonel)	Zipli sistem yanıtı zarfının ikilik verisi

Hata	
HataKodu	Sistem hata kodu
HataAçıklaması	Sistem hata açıklaması

Bu metod ile manuel olarak herhangi bir gelen/gönderilen irsaliye belgesine ait gönderilen/gelen irsaliye yanıtlarının listesi alınabilir. İrsaliye yanıtı belgelerinin detayları ve ziplenmiş UBL dosyaları ziplenmiş olarak cevap yapısında dönmektedir.

Bu metod zamanlanmış görevlerle gelen/gönderilen irsaliye yanıtları takibi için kesinlikle kullanılmamalıdır. Yalnızca ihtiyaç durumlarında manuel kullanıcı sorgulamaları için API'a dahil edilmiştir. Tüm irsaliye yanıtları, irsaliye belgeleri gibi getDesUBList() zarf sorgularıyla takip edilmeli ve düzenli olarak istemci sisteme aktarılmalıdır.

Tek bir sorguda 20 adet irsaliye belgesi sorgulanabilmektedir. Çoklu belge sorgulamalarında optimizasyon için tek tek sorgulamadan kaçınarak toplu yapılması önemlidir.

API Limitleri ve Optimizasyon

Yüksek hacimli gönderim yapılması gereken durumlarda, performans optimizasyonu için tüm belgeler zarflanarak sendDesUBL() metoduyla toplu olarak gönderilmelidir. Zip içerisinde gönderilen XML dosyanın açılmış boyutu 5MB ile sınırlanmıştır, ve genellikle 25-30 adet belge içeren zarflar boyut limiti aşılmadan gönderilebilmektedir. Ancak bu rakam ağ hızlarına, donanım kaynaklarına ve müşterinin ortalama belge boyutlarına göre optimize edilmelidir.

getDesUBList() metoduyla alınan gelen belgeler, mecburi durumlar haricinde her zaman zarflı olarak alınmalıdır. Sorgulanan zaman aralıkları tekrar sorgulanmamalı, zamanlanmış görevlerle metodun kullanımı minimum 2-4 saat olarak ayarlanmalıdır. Manuel ek sorgular, kullanıcı taleplerine göre yapılabilir. Gönderilen belgelerin ve listelerinin, özel ihtiyaçlar dışında alınmasına gerek yoktur.

getDesUBL() metodu kullanımlarında, transfer boyutları ciddi rakamlara çıkmaktadır. UBL belgeler her zaman toplu olarak indirilmeye çalışılmalı, tek tek belge alımından kaçınılmalıdır. Bir çağrıyla alınan UBL sayıları istemci sisteme ve ağ koşullarına göre optimize edilmelidir.

Gönderilen zarfların getEnvelopeStatus() metoduyla durum sorguları da yine mutlaka toplu olarak uzun aralıklarla, mümkünse sadece günlük olarak yapılmalıdır. Zamanlanmış görevler gecelik olarak çalıştırılarak sistemlerin gün içi performanslarını korumak için özen gösterilmelidir. Son kullanıcının anlık güncel bilgiye ihtiyaç duyduğu durumlar için sadece sınırlı sayıda belge için manuel tetiklenebilecek bir sorgulama yapısı uygun olacaktır. Son durumuna ulaşmış zarfların sistemden tekrar tekrar sorgulaması kesinlikle yapılmamalıdır. GİB sistem yanıtı gelmeyen belgeleri 8 günden sonra otomatik kabul durumuna çekmektedir, bu sebeple 10 günden daha eski gönderilen zarflar zaten son duruma ulaşmış zarflar olacağı için bu belgeler sorgu listesinden çıkartılmalıdır.

Kullanıcı listeleri, GİB ve bulut sistemler tarafından birkaç saatte bir güncellenmektedir. Genel kullanımda bu liste her gece bir defa tam liste indirilerek alıcı VKN/TCKN ve etiket listeleri lokal veritabanına kaydedilmeli ve kontroller buradan yapılmalıdır. Bulut ve GİB sistemleri zaten her zaman güncel listelerle

kontrolleri yaparak geçersiz alıcılara belge gönderimini engellemektedir. Bu sebeple getDesPartialUserList() metodunun günlük zamanlanmış görevle kullanımı tek bir kez olmalıdır.

getDesReceipts() metodu zamanlanmış görevlerle kullanılmamalı, gelen irsaliye yanıtları getUBLList() metoduyla düzenli zarf listesiyle beraber alınmalıdır. Bu metod yalnızca manuel sınırlı sayıda kullanıcı işlemi için ihtiyaç durumlarında kullanılmalıdır.

Tüm API metodlarında hata durumlarında hata kodları ve açıklamaları dönülmektedir, hatalı isteklerde tüm istek hatalı kabul edilmektedir. Bulut sistemden gelen hatalar mutlaka loglanarak takip edilmeli ve gerekli durumlarda hata açıklamaları son kullanıcılara gösterilebilmelidir. Otomatik hata yakalama mekanizmalarında tanımlı hata kodlarından faydalanılmalıdır. Sisteme sürekli olarak aynı hatalı isteklerin gönderilmesi kesinlikle engellenmelidir.

Bu API ve açıklanan uygun kullanım yöntemlerine uyum, hacimden bağımsız olarak sağlanmalıdır. 10 ya da 10 milyon belge transferi, burada açıklanan kurallara uygun olduğu sürece aynı şekilde ve sorunsuz gerçekleştirilebilir. Tüm entegrasyonlarda, belge adetlerinin tüm ekosistemde her yıl arttığı unutulmamalı ve birkaç kat artması durumlarında bile sorun yaşanmayacak şekilde optimizasyon testleri yapılmalıdır.

API kullanımı, servis durumu ve kalitesi 7x24 izlenmektedir. Bu dokümanda açıklanan uygun kullanımlara aykırı, sistem performansını ya da yükünü negatif etkileyen durumlarda **erişim kesintileri sistem tarafından otomatik olarak uygulanmaktadır.**

WS API Hata Kodları ve Açıklamaları

Hata Kodu	Açıklama
1000	Parametre Hatası
1010	Şema validasyonu hatası
1020	Şematron hatası
1080	UTF-8 validasyonu hatası
1100	Gönderici VKN/TCKN ve etiketi GIB'e kayıtlı değil
1101	Alıcı VKN/TCKN ve etiketi GIB'e kayıtlı değil
1110	Gönderici VKN/TCKN ve etiketi kayıtlı değil
1111	Alıcı VKN/TCKN ve etiketi kayıtlı değil
1112	VKN/TCKN ve etiket kayıtlı değil
1200	İstemci IP adresinin bu işleme yetkisi yok
1201	İstemci IP adresi için WS kullanıcı adı/şifre hatalı
1202	Kullanıcının bu VKN/TCKN için işlem yetkisi yok
1300	Çağrı limiti aşıldı
3010	Zarf UUID sistemde mevcut
3216	Fatura göndericiye ait değil
4011	İrsaliye UUID sistemde mevcut
4012	İrsaliye ID sistemde mevcut
4013	İrsaliye ID otomatik üretiliyor, gönderilmemeli
4014	İrsaliye ID otomatik üretiliyor, müşteri irsaliye numarası gönderilmeli
4016	RESEND edilmek istenilen İrsaliye zarfı hatalı durumda değildir tekrar gönderilemez
4201	İrsaliye yanıtı UUID sistemde mevcut
4210	İrsaliye yanıtı verilen irsaliye bulunamadı
4215	İrsaliye alıcıya ait değil
4216	RESEND edilmek istenilen İrsaliye yanıtı zarfı hatalı durumda değildir tekrar gönderilemez
4230	İrsaliyeye gönderilen önceki irsaliye yanıtı sonuçlanmamış
4610	UUID'ye ait belge bulunmadı
4630	Belge gönderilen VKN/TCKN ve etikete ait değil
4730	Belge gönderilen VKN/TCKN ve etikete ait değil
4910	UUID'ye ait belge bulunamadı
4920	Belge gönderilen VKN/TCKN ve etikete ait değil
4950	UUID'ye ait zarf bulunamadı
4960	Zarf gönderilen VKN/TCKN ve etikete ait değil

GIB e-Fatura Sistem Yanıtı Kodları

e-Fatura Uygulaması Sistem Yanıtı Şema Yapısı - v1.5 dokümanından alınmıştır:

Durum Kodu	Durum Açıklaması
1000	ZARF KUYRUGA EKLENDİ
1100	ZARF İSLENİYOR
1110	ZIP DOSYASI DEĞİL
1111	ZARF ID UZUNLUĞU GECERSİZ
1120	ZARF ARSİVDEN_KOPYALANAMADI
1130	ZIP AÇILAMADI
1131	ZIP BİR DOSYA İÇERMELİ
1132	XML DOSYASI DEĞİL
1133	ZARF ID VE XML DOSYASININ ADI AYNI OLMALI
1140	DOKÜMAN AYRISTIRILAMADI
1141	ZARF ID YOK
1142	ZARF ID VE ZIP DOSYASI ADI AYNI OLMALI
1143	GECERSİZ VERSİYON
1150	SCHEMATRON KONTROL SONUCU HATALI
1160	XML SEMA KONTROLUNDEN GECEMEDİ
1161	İMZA SAHİBİ TCKN VKN ALINAMADI
1162	İMZA KAYDEDİLEMEDİ
1163	GÖNDERİLEN ZARF SİSTEMDE DAHA ÖNCE KAYITLI OLAN BİR FATURAYI İÇERMEKTEDİR.
1164	GÖNDERİLEN ZARF SİSTEMDE DAHA ÖNCE KAYITLI OLAN BİR BELGEYİ İÇERMEKTEDİR.
1170	YETKİ KONTROL EDİLEMEDİ
1171	GÖNDERİCİ BİRİM YETKİSİ YOK

Durum Kodu	Durum Açıklaması
1172	POSTA KUTUSU YETKİSİ YOK
1175	İMZA YETKİSİ KONTROL EDİLEMEDİ
1176	İMZA SAHİBİ YETKİSİZ
1177	GEÇERSİZ İMZA
1180	ADRES KONTROL EDİLEMEDİ
1181	ADRES BULUNAMADI
1182	KULLANICI EKLENEMEDİ
1183	KULLANICI SİLİNEMEDİ
1190	SİSTEM YANITI HAZIRLANAMADI
1195	SİSTEM HATASI
1200	ZARF BASARIYLA İŞLENDİ
1210	DOKUMAN BULUNAN ADRESE GÖNDERİLEMEDİ
1215	DOKUMAN GÖNDERİMİ BASARISIZ. TERKAR GÖNDERME SONLANDI
1220	HEDEFTEN SİSTEM YANITI GELMEDİ
1230	HEDEFTEN SİSTEM YANITI BASARISIZ GELDİ
1235	FATURA İPTAL'E KONU EDİLDİ
1300	BASARIYLA TAMAMLANDI

Test Yöntemi

Entegrasyon testleri Bulut E-İrsaliye platformuna kaydolunan tek bir VKN/TCKN'ye ait GB ve PK etiketleri arasında yapılabilir.

Tüm metodların birim entegrasyon testleri tamamlandığında, aşağıdaki akış takip edilerek WS API ile tam entegrasyonun çalıştığı doğrulanabilir:

İrsaliye Gönderim Testi:

1. Gönderici ve alıcısı test VKN/TCKN'si olan ve test GB etiketinden test PK etiketine hazırlanmış, zarflanmış bir irsaliye belgesi sendDesUBL() metoduyla gönderilir. Herhangi bir şema-şematron ya da validasyon hatası alınmadığı kontrol edilir, alınıyorsa geri dönen hata mutlaka düzgün yakalanıp hata kodu ve açıklaması kullanıcıya görünür olmalıdır.
2. getDesUBLList() gönderilen irsaliye UUID'si ile çağrılır ve oluşan zarf-irsaliye detayları kontrol edilir.
3. getDesEnvelopeStatus() metoduyla gönderilen zarf sorgulanır ve durumu 1300 oluncaya kadar aralıklarla kontrole devam edilir. 1300 kodu döndüğünde zarf PK etiketine ulaşmış ve başarıyla işlenmiş, yani GIB'den tekrar geri sisteme gönderilmiş ve sistem yanıtı dönülmüş demektir. getDesEnvelopeStatus() metodu ile sadece gönderilen irsaliye ve irsaliye yanıtı zarflarının durumları takip edilebilir. Gönderilen irsaliyeye gelen irsaliye yanıtları bu metod ile alınamaz.

İrsaliye Yanıtı Gönderim Testi:

4. Test VKN/TCKN'si ve test PK etiketinden test GB etiketine hazırlanmış, gelen irsaliye UUID'si için bir irsaliye yanıtı zarflanarak sendDesUBL() metoduyla sisteme gönderilir.
5. getDesEnvelopeStatus() metoduyla gönderilen zarf sorgulanır ve durumu 1300 oluncaya kadar kontrole devam edilir. 1300 kodu döndüğünde zarf GB etiketine ulaşmış ve başarıyla işlenmiş demektir.

İrsaliye ve İrsaliye Yanıtları Alım Testi:

6. getDesUBLList() metoduyla test VKN/TCKN'si ve PK etiketi kullanılarak gelen zarflar sorgulanır, ve GB etiketinden gönderilen irsaliye zarfı ile PK etiketinden gönderilen irsaliye yanıtı zarfının listede bulunduğu teyit edilir.
7. getDesUBL() metoduyla gelen irsaliye & irsaliye yanıtı belgeleri ya da zarfları indirilir

İrsaliye ve İrsaliye Yanıtları Görüntüleri Alım Testi:

8. getDesView() metodu belge UUID'leri ile birlikte kullanılarak belgelerin HTML/PDF görüntüleri alınır.

e-İrsaliye Kullanıcı Listesi Alım Testi:

9. getDesPartialUserList() metoduyla kullanıcı listelerinin sisteme indirilerek lokal veritabanına kaydedilebilirliği ve sendDesUBL() belge gönderimlerinde ilgili kullanıcı kayıt kontrollerinin düzgün yapılabilirliği kontrol edilmeli.

EK1 – Bağlantı Klavuzu

Bulut e-İrsaliye sistemlerine HTTPS (TLS 1.1+) protokolü ve 443 portu üzerinden erişim sağlanır. Entegrasyon geliştirmelerine başlamadan önce, platforma erişeceğiniz IT sistemleriniz üzerinden API hostname/port'larına telnet komutuyla erişim testi yaparak sisteminizin web servis adreslerine erişebildiğini kontrol etmeniz gerekmektedir. Bu adım öncesinde, platformun public DNS ve IP adreslerine, kendi ağ yönetim ve güvenlik duvarı yazılımlarınızdan öncelikle erişim sağlamanız gerekmektedir.

Windows Command Prompt ile API'ye erişecek sunucuda çalıştırılacak örnek komut:

telnet efaturaws.fitbulut.com 443

WS API, SOAP 1.1 standardıyla uyumlu olarak çalışmaktadır. Web servis istemci uygulamalarının bu ayarı kendi altyapılarına uygun şekilde yapmaları gerekmektedir.

SSL ve TLS 1.1 öncesi versiyonlar, güvenlik gereği desteklenmemektedir. API'ye erişim sağlayacak tüm uygulamaların, TLS 1.1 ve sonrası versiyonları destekliyor olması gerekmektedir.

Bulut WS API'ye erişimde kullanılan public HTTPS sertifikası ve kök sertifikalar, api.fitbulut.com adresi ile aynıdır. Bu adresten her zaman güncel sertifikalar indirilebilir.

Sistem erişimlerinde hiçbir zaman doğrudan WS API'nin IP adresleri kullanılmamalıdır, bu adresler felaket kurtarma merkezine geçiş gibi durumlarda değişmektedir. Erişim her zaman DNS adresleri üzerinden sağlanmalıdır.

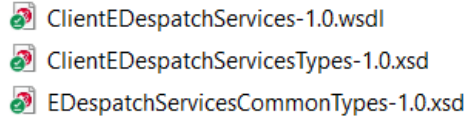
EK2 – Yetkilendirme (Authentication)

Web servislere erişim HTTPS üzerinden "HTTP Basic Authentication" yöntemi kullanılarak VKN/TCKN bazlı kullanıcı adı ve şifre kontrolleriyle güvenlik altına alınmıştır. Kayıt ve aktivasyon işlemlerinin tamamlanmasından ve sistem erişim testlerinin başarıyla yapılmasından sonra, eğer servislerden "HTTP 401 - Unauthorized" alınıyorsa, öncelikle başvuru formlarında bildirilen web servis kullanıcı adı ve şifreleri kontrol edilmelidir. WS şifreleri süreli olarak

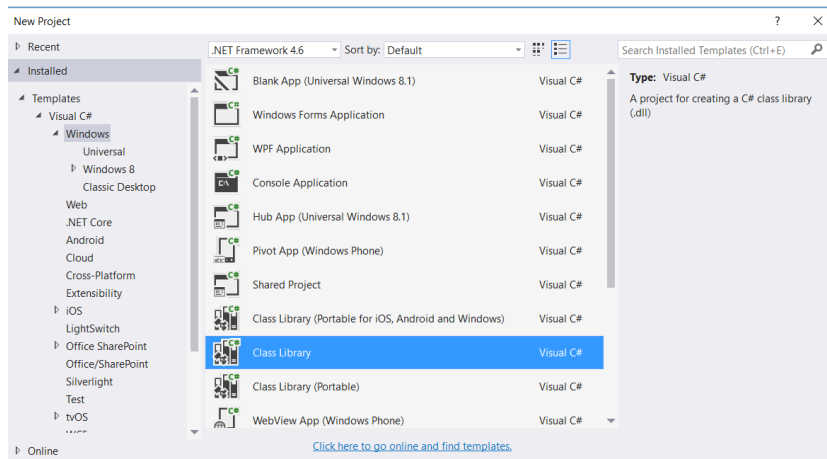
verilmektedir ve firma yetkilisi olarak belirlenen kişinin e-posta adresine ilgili şifrenin kullanım süresinin dolacağı ve dolduğu uyarıları gönderilmektedir.

EK3 - .Net ile API projesi yaratma ve metodları çağırma

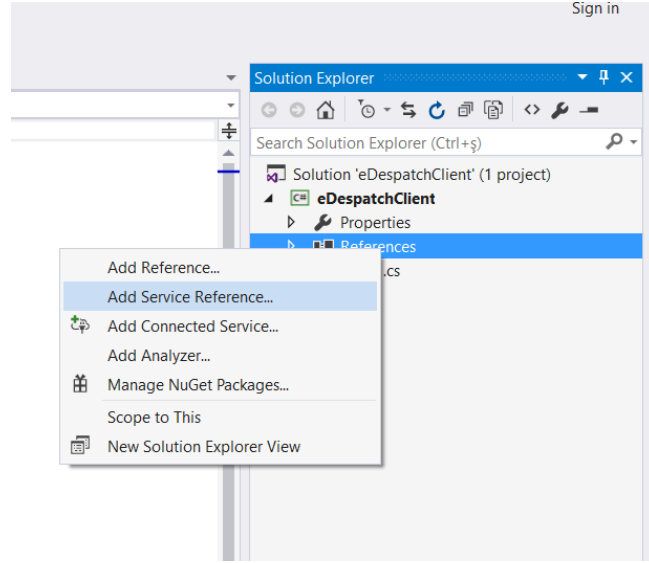
- 1- e-İrsaliye API WSDL ve XSD dosyaları, API paketinde EK2-Teknik Belgeler klasöründe bulunmaktadır:



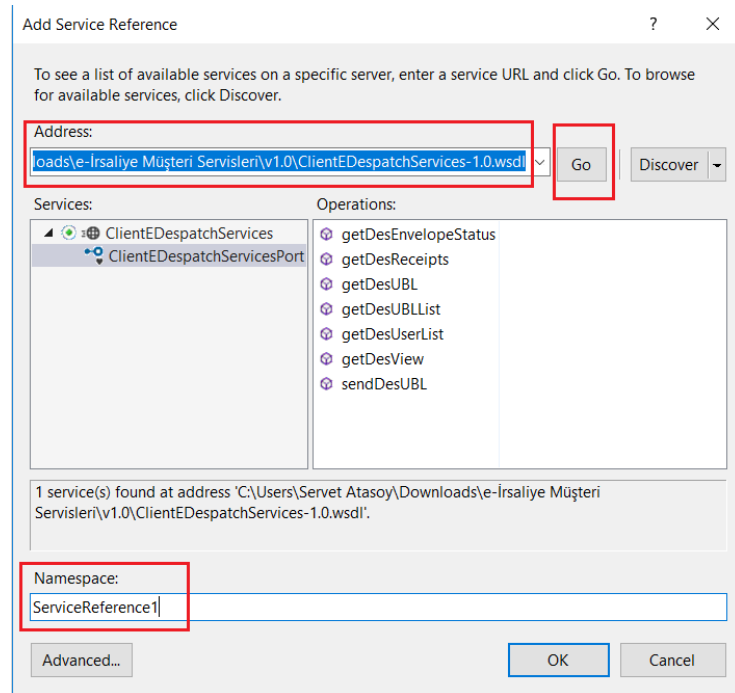
- 2- Visual Studio yardımı ile varsa mevcut solution veya yeni bir solution içerisine boş bir Class Library projesi açınız:



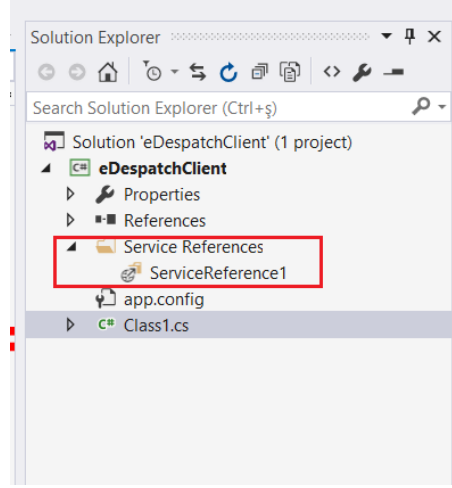
- 3- Yeni açtığınız proje içerisinde References tabına giderek, sağ click Add Service Reference sekmesine tıklayınız:



- 4- Açılan pencere üzerinde Address alanına, zip dosyasından çıkarttığınız WSDL dosyasının yolunu göstererek Go butonuna basınız.
- 5- Go butonuna bastıktan sonra aşağıdaki resimde görüldüğü gibi Operations bölmesine servise ait metodlar gelecektir.
- 6- Alt kısımda bulunan Namespace alanına projenize uygun bir isim veriniz:



7- OK butonuna bastıktan sonra aşağıdaki gibi servisiniz eklenmiş olacaktır:



8- Kod tarafından aşağıdaki şekilde erişim sağlayabilirsiniz. 6. Madde de vermiş olduğunuz namespace ismi ServiceReference1 kısmına karşılık gelmektedir:

```
ServiceReference1.ClientEDespatchServicesPortClient client = new ServiceReference1.ClientEDespatchServicesPortClient();
```

9- WSDL'ı projenize eklendikten sonra app.config veya web.config dosyasına otomatik olarak binding ayarları eklenecektir fakat değiştirilmesi gereken bir kaç ayar mevcuttur.

10- Otomatik olarak eklenen basicBinding düğümü silenerek customBinding düğümü eklenmelidir.

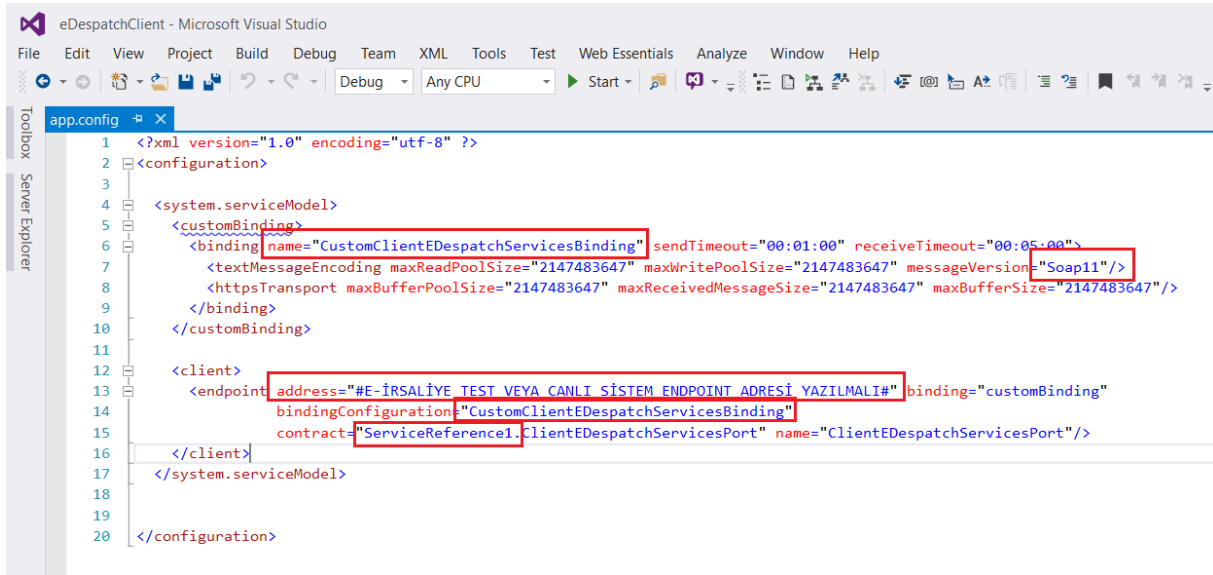
11- customBinding düğümünde name alanına bir isim verilmelidir.

12- textMessageEncoding ve httpsTransport düğümlerindeki ayarları değiştirmeden ekleyiniz.

13- endpoint düğümünde address alanına test veya canlı sistemin adresi yazılmalıdır.

14- endpoint düğümünde bindingConfiguration alanına verdiğiniz isim ile customBinding düğümünde name alanına verdiğiniz isim aynı olmalıdır.

15- endpoint düğümünde contract alanındaki ServiceReference1 ifadesi 6. Madde de verdiğiniz namespace ile aynı olmalıdır:



.Net ile HTTP Basic Authentication

e-İrsaliye müşteri servislerine erişim sağlayabilmeniz için web servis kullanıcı adı ve şifrenizi her request çağrısında HttpHeaders'a ekleyerek göndermeniz gerekmektedir. Aşağıdaki kod örneğini kendi WS API kullanıcı adı ve şifrenize göre uyarlayıp kullanabilirsiniz:

```
ServiceReference1.ClientEDespatchServicesPortClient client = new ServiceReference1.ClientEDespatchServicesPortClient();

using (new System.ServiceModel.OperationContextScope((System.ServiceModel.IClientChannel)client.InnerChannel))
{
    string authorization = "UserName:Password";

    byte[] byteArray = System.Text.Encoding.ASCII.GetBytes(authorization);
    string base64authorization = Convert.ToBase64String(byteArray);

    System.ServiceModel.Web.WebOperationContext.Current.OutgoingRequest.Headers.Add(
        HttpRequestHeader.Authorization, String.Format("Basic {0}", base64authorization));

    client.sendDesUBL(new sendDesUBLRequest
    {
        //REQUEST PARAMETRELERİ
    });
}
```

Not: System.ServiceModel.Web ve System.Net kütüphanelerini eklemeniz gerekmektedir.

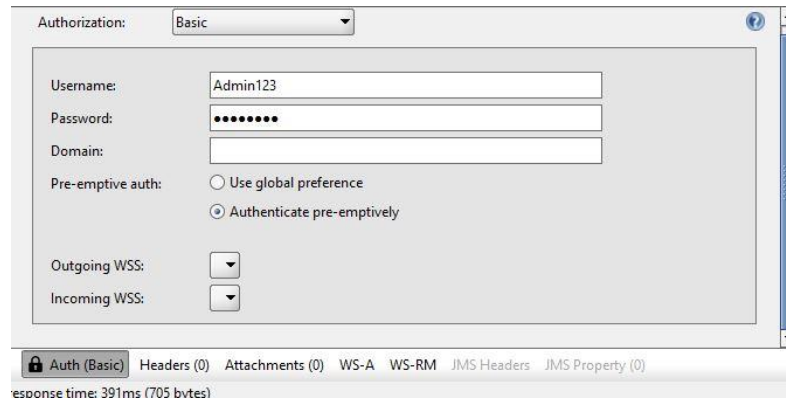
.Net ile TLS 1.1 ayarı

.Net projesinde web servisleri çağırılmadan önce aşağıdaki kodun eklenmesi gerekmektedir:

System.Net.ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls11;

EK-4 SOAP UI Authentication ve TLS Ayarları

SoapUI programı ile web servislere bağlanabilmek için, Auth tabından “Authenticate pre-emptively” modunda HTTP Basic Authentication ayarlarının yapılması ve kullanıcı adı şifre bilgilerinin girilmesi gerekmektedir. Bu ayar sayesinde, gönderilen HTTP isteklerin HTTP Header’larına “Authorization:Basic” bilgisi, base64 formatında kullanıcı adı-şifreyle birlikte eklenecektir:



HTTP Raw Request tabından başarılı bir Authorization HTTP Header örneği:

```
POST https://efaturawtest.fitbulut.com/ClientEDespatchServicePort.svc HTTP/1.1
Accept-Encoding: gzip,deflate
Content-Type: text/xml; charset=UTF-8
SOAPAction: "getDesEnvelopeStatus"
Authorization: Basic METLcHlqVno6WGhnbyFIM0w=
Content-Length: 664
Host: efaturawtest.fitbulut.com
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
```

Doğru VKN-TCKN ve web servis kullanıcı bilgileri ile bağlantı sağlandığı durumda web servislerden başarılı şekilde sonuç dönüyor olacaktır. Bu ayarlar yapılmadığında ise “unauthorized” hatası dönecektir.

SOAP UI TLS ayarı:

SoapUI testlerinde TLS/HTTPS hatası alınırsa, Program Files\SmartBear\SoapUI-5.3.0\bin\SoapUI-5.3.0.vmoptions adresindeki SoapUI-5.3.0.vmoptions (kendi versiyonunuza göre ilgili dizini bulunuz) adlı dosyayı Notepad++ ile admin olarak açarak aşağıdaki alanı dosyaya ekleyip kaydederek TLS ayarlarını güncelleyebilirsiniz:

-Dsoapui.https.protocols=SSLv3,TLSv1.1,TLSv1.2