



e-FATURA UYGULAMASI

(Yazılım Standartları ve Nesne Yapısı)

Nisan 2010

ANKARA

GELİR İDARESİ BAŞKANLIĞI
Uygulama ve Veri Yönetimi Daire Başkanlığı

Versiyon	Yayım Tarihi	Eklene/Silinen/Değişen Bölüm	Açıklama
1.0	05.03.2010	--	Kılavuzun ilk yayım tarihi
1.1	26.04.2010	3 XAdES	3. XMLDsign bölümü tamamen kaldırılarak XAdES olarak değiştirildi.

1	Giriş	5
2	MTOM (Message Transmission Optimization Mechanism)	5
3	XAdES	6
4	Mesaj Özeti Standardı.....	8
5	SSL Mekanizması	8
6	Kullanılan Metot ve Sınıf Yapıları	8
6.1	Metot Özetleri.....	8
6.2	Metot Detayları.....	9
6.2.1	sendDocument.....	9
6.2.2	getApplicationResponse	9
6.3	Kullanılan Sınıf Özetleri	9
6.4	DocumentRequest Sınıfı Metotları	10
6.5	DocumentResponse Sınıfı Metotları.....	10
6.6	DocumentReturnType Sınıfı Metotları.....	11
6.7	DocumentType Sınıfı Metotları.....	11
6.8	GetAppRespRequest Sınıfı Metotları	12
6.9	GetAppRespRequestType Sınıfı Metotları	12
6.10	GetAppRespResponse Sınıfı Metotları.....	12
6.11	GetAppRespResponseType Sınıfı Metotları.....	12

1 Giriş

Bu kılavuz, Elektronik Fatura Uygulamasına entegre olmak isteyenler ile uygulamaya ilişkin yazılım geliştireceklerin uymak zorunda oldukları yazılım standartları ve nesne yapısı hakkında bilgi vermek amacıyla hazırlanmıştır.

Elektronik Fatura Uygulamasının genel özellikleri aşağıda yer almaktadır.

- Uygulama içerisinde teknolojik iletişim altyapısı olarak Web servis kullanılmıştır.
- Web servis iletişim altyapısı SOAP mesajları ile sağlanmaktadır. SOAP metotları için iki tür kodlama metodu bulunmaktadır. Bunlar RPC ve DOC kodlamalarıdır. Uygulamada performans açısından DOC kodlaması benimsenmiştir. SOAP mesajlarının iletiminde büyük dosyalar söz konusu olduğundan MTOM (Message Transmission Optimization Mechanism) mekanizması kullanılmıştır.
- Web Servis belirtiminin diğer web servis gerçekleştirmeleri ile sorunsuz çalışabilmesi için WS-I ve WS-I Basic Profile standartları web servis gerçekleştirmede dikkate alınmıştır.

Kılavuzun ilk bölümünde MTOM mekanizması genel olarak tanıtılacak, sonraki bölümlerde ise tanımlanan Web Servis metotları ve metotların kullandığı parametre, dönüş değeri ve nesnelere ilgili bilgilere yer verilecektir.

2 MTOM (Message Transmission Optimization Mechanism)

MTOM ikilik verilerin etkin bir biçimde web servislerden veya web servislere gönderimini sağlayan bir metottur. İkilik veriyi göndermek için XOP(XML-binary Optimized Packaging) kullanır. MTOM büyük miktarda ikilik veriyi SOAP mesajıyla gönderme imkânı sağlamakta olup aşağıda sıralanan 3 önemli yararı bulunmaktadır.

- Güvenlik: MTOM, güvenlikle oluşur, verilerin güvenliği SOAP mesaj kadar iyi korunur. Ayrıca DIME'nin desteklemediği bazı güvenlik seviyeleri de MTOM ile desteklenmektedir. (Örneğin: TCP analizi DIME ile yapılan transferde gerçekleştirilirse, dosyanın görüldüğü görünür ancak MTOM, dosya içine SOAP mesajına uygulanan güvenliği uygulayabilmektedir.)
- Dosya Boyutunun Korunması: MTOM sayesinde hat üzerinde ekstra yük oluşmaz ve böylece dosya boyutu korunmuş olur.
- Basit Programlama Modeli: MTOM desteği etkinleştirilerek servisteki bütün byte transferleri otomatik olarak şifrelenir.

MTOM'in web servis ile uygulamaları çeşitli web servis gerçekleştirim ürünleriyle sağlanmıştır. Bunlara örnek axis2 ve Microsoft web servisleri örnek verilebilir.

3 XAdES

XAdES, Elektronik Fatura Uygulaması kapsamında gönderilecek olan belgelerin (Fatura veya Uygulama Yanıtı) imzalanması veya onaylanmasında kullanılmaktadır. XAdES imzalamada minimum olarak XAdES-BES kullanılmaktadır.

e-Fatura onaylama işleminde bir faturanın XML imza olarak minimum aşağıdaki XML etiketlerini kullanması gerekmektedir.

```
<ds:Signature Id="Signature">
  <ds:SignedInfo Id="SignedInfo">
    <ds:CanonicalizationMethod/>
    <ds:SignatureMethod/>
    <ds:Reference URI="" >
      <ds:Transforms>
        <ds:Transform/>
      </ds:Transforms>
      <ds:DigestMethod/>
      <ds:DigestValue/>
    </ds:Reference>
    <ds:Reference URI="#SignedProperties" >
      <ds:DigestMethod />
      <ds:DigestValue />
    </ds:Reference>
  </ds:SignedInfo>
</ds:Signature>
```

```

</ds:SignedInfo>
<ds:SignatureValue/>
<ds:KeyInfo>
  <ds:KeyValue/>
  <ds:X509Data>
    <ds:X509SubjectName/>
    <ds:X509Certificate/>
  </ds:X509Data>
</ds:KeyInfo>
<ds:Object>
  <xades:QualifyingProperties Target="Signature">
    <xades:SignedProperties Id="SignedProperties">
      <xades:SignedSignatureProperties>
        <xades:SigningTime/>
        <xades:SigningCertificate>
          <xades:Cert>
            <xades:CertDigest>
              <ds:DigestMethod/>
              <ds:DigestValue/>
            </xades:CertDigest>
            <xades:IssuerSerial>
              <ds:X509IssuerName/>
              <ds:X509SerialNumber/>
            </xades:IssuerSerial>
          </xades:Cert>
        </xades:SigningCertificate>
        <xades:SignerRole>
          <xades:ClaimedRoles>
            <xades:ClaimedRole/>
          </xades:ClaimedRoles>
        </xades:SignerRole>
      </xades:SignedSignatureProperties>
    </xades:SignedProperties>
  </xades:QualifyingProperties>
</ds:Object>
</ds:Signature>

```

Mesaj özetlerinin güvenlik açısından SHA256 olması önerilmektedir. Yukarıda verilen XAdES xml imza iskeletinde bulunan alanlar mutlaka kullanılmalıdır. İskelette verilen nitelik(attribute) değerleri örnek teşkil etmesi amacıyla verilmiştir.

4 Mesaj Özeti Standardı

Elektronik Fatura Uygulaması aracılığı ile gönderilecek olan belgelerin Zarf XML'i içerisine konulması gerekmektedir. Bu XML, zip formatında sıkıştırıldıktan sonra Web servis aracılığı ile gönderilecektir. Web servis ile bu zip belgesine ait mesaj özeti de (hash) gönderilmesi uygulama içerisinde beklenen bir durumdur.

Mesaj özeti standardı olarak MD5 standardına uyulacaktır. Alınan ve gönderilen belgelere ait özet değerleri hesaplanacak ve gelen değerler ile karşılaştırılacaktır. Özet değerleri 32 karakter uzunluğunda olacaktır.

5 SSL Mekanizması

Web servis iletişimi HTTPS protokolü üzerinden SSL mekanizması ile sağlanacaktır. Buna göre istemci uygulaması SSL sunucu uygulamasının kimliğini doğrular. Bu durumda Elektronik Fatura Uygulamasında bulunan birimlerin güvenilir sunucu sertifikası temin etmeleri gerekmektedir.

6 Kullanılan Metot ve Sınıf Yapıları

Bu bölümde anlatılan metot ve sınıf yapıları Java programlama dili ve axis2 gerçekleştirimi ile oluşmuş örnek yapılarıdır. Oluşacak metot ve sınıf yapıları değişik programlama dilleri ve web servis gerçekleştirmeleri ile değişebilir.

6.1 Metot Özetleri

DocumentResponse	sendDocument (DocumentRequest request) Kullanıcının zarfların göndereceği metottur.
GetAppRespResponse	getApplicationResponse (GetAppRespRequest request) Kullanıcının zarfların durumunu sorguladığı metottur.

6.2 Metot Detayları

6.2.1 sendDocument

```
public DocumentResponse sendDocument(DocumentRequest request)
    throws EFaturaFaultMessage
```

Kullanıcının zarfların gönderileceği metottur. İkili veriyi(Binary Data), dosya adını ve ikilik verinin özetini içeren DocumentRequest nesnesi alır ve kendi hesapladığı özet değeri ile mesajı içeren DocumentResponse nesnesini döner.

Throws: EFaturaFaultMessage

6.2.2 getApplicationResponse

```
public GetAppRespResponse
getApplicationResponse(GetAppRespRequest request)
    throws EFaturaFaultMessage
```

Kullanıcının zarfların durumunu sorguladığı metottur. Zarf ID'yi içeren GetAppRespRequest nesnesini alır ve sistem yanıtını içeren GetAppRespResponse nesnesini döner.

Throws: EFaturaFaultMessage

6.3 Kullanılan Sınıf Özetleri

DocumentRequest	sendDocument metoduna parametre olarak verilen sınıftır.
DocumentResponse	sendDocument metodundan dönen sınıftır.
DocumentReturnType	Mesaj ve özet niteliklerini içeren sınıftır.
DocumentType	İkili veri, dosya adı ve özet niteliklerini içeren sınıftır.

GetAppRespRequest	getApplicationResponse metoduna parametre olarak verilen sınıftır.
GetAppRespRequestType	getApplicationResponse metodundan dönülen sınıftır.
GetAppRespResponse	Zarf id niteliğini içeren sınıftır.
GetAppRespResponseType	Sistem Uygulama Yanıtını içeren sınıftır.

6.4 DocumentRequest Sınıfı Metotları

DocumentType	getDocumentRequest () DocumentType nesnesini döner.
void	setDocumentRequest (DocumentType param) DocumentType nesnesine parametre ile verilen nesneyi atar.

6.5 DocumentResponse Sınıfı Metotları

DocumentReturnType	getDocumentResponse () DocumentReturnType nesnesini döner.
Void	setDocumentResponse (DocumentReturnType param) DocumentReturnType nesnesine parametre ile verilen nesneyi atar.

6.6 DocumentReturnType Sınıfı Metotları

String	getHash() İkili verinin özet değerini döner.
String	getMsg() Mesajı döner.
void	setHash(String param) param ile verilen değeri Hash'e atar.
void	setMsg(String param) param ile verilen değeri Msg'a atar.

6.7 DocumentType Sınıfı Metotları

Base64Binary	getBinaryData() İkili (binary) veriyi döner.
String	getFileName() Gönderilen zarfa ait dosya adını döner.
String	getHash() İkili verinin özet değerini döner.
void	setBinaryData(Base64Binary param) Param ile verilen değeri ikili veriye atar.
void	setFileName(String param) Param ile verilen değer dosya adına atar.
void	setHash(String param) Param ile verilen değeri özete atar

6.8 GetAppRespRequest Sınıfı Metotları

GetAppRespRequestType	getGetAppRespRequest () GetAppRespRequestType nesnesini döner
Void	setGetAppRespRequest (GetAppRespRequestType param) GetAppRespRequestType nesnesine parametre ile verilen nesneyi atar.

6.9 GetAppRespRequestType Sınıfı Metotları

String	getInstanceIdentifier () Zarf id'yi döner.
void	setInstanceIdentifier (String param) Zarf id'ye parametre ile verilen değeri atar.

6.10 GetAppRespResponse Sınıfı Metotları

GetAppRespResponseType	getGetAppRespResponse () GetAppRespResponseType nesnesini döner.
Void	setGetAppRespResponse (GetAppRespResponseType param) GetAppRespResponseType nesnesine paramere ile verilen nesneyi atar.

6.11 GetAppRespResponseType Sınıfı Metotları

String	getApplicationResponse () Uygulama Yanıtını döner.
--------	--

void	setApplicationResponse (String param) Uygulama Yanıtına parametre ile verilen değeri atar.
------	--