



**T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı**

Yayın No : 3.0

Yayın Tarihi : 07.11.2018

Doküman No : TITCK-RRS-JSON

**T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı
Versiyon 3.0**




T.C. Sağlık Bakanlığı

T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

İçindekiler	
Versiyon Tarihçesi	3
Amaç ve Kapsam	4
Yetkilendirme	4
Entegrasyon Adımları	4
Yöntem 1: JSON Veri Gönderme Metodu	4
Renkli Reçete JSON Dosyası ile Bilgi Aktarımı	4
Şifreleme Algoritması	5
JSON Formatı Element Veri Seti	6
AHBS için JSON Formatı	8
AHBS için JSON Formatı Örneği	10
JAVA ile JSON Nasıl Oluşturulur?	11
.Net ile JSON Nasıl Oluşturulur?	13
Nasıl Şifreleme Yapacağım?	16
Örnek Şifreleme Kod Parçacıkları	16
Her İki Metot İçin İmza İşlemleri	17
Elektronik İmza	17

 T.C. Sağlık Bakanlığı	T.C. SAĞLIK BAKANLIĞI RENKLİ REÇETE SİSTEMİ AHBS Entegrasyon Dokümanı	Yayın No : 3.0 Yayın Tarihi : 07.11.2018 Doküman No : TITCK-RRS-JSON
--	--	---

Versiyon Tarihçesi

VERSİYON	DEĞİŞİKLİK TANIMI	TARİH
1.0	---	01.03.2017
2.0	JSON formatı ile veri gönderimi, kapsam genişlemesi	08.03.2018
2.1	Yatan Hasta, Günübirlik Reçeteleri için kullanılacak Başhekim ve Hastane Eczanesi giriş işlemleri ile EHU onay işlemleri	12.03.2018
3.0	İlaç kümesinin JSON formatı ile veri gönderim sürecine dahil edilmesi ve Versiyon 1 ile duyurulan yönlendirme metodunun yayından kaldırılması	07.11.2018



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

Amaç ve Kapsam

Elektronik Renkli Reçete Sisteminin amacı, hali hazırda matbu olarak basılan tüm renkli reçetelerin (kırmızı, yeşil, turuncu ve mor) ve normal reçete ile verilmesi gereken izlemeye tabi ilaçları ihtiva eden reçetelerin elektronik ortama aktarılmasını sağlamaktır. Bu dokümanın amacı ise geliştirilen Renkli Reçete Sistemine, AHBS'lerin entegre olmasını sağlayacak web api'lerin kullanımının anlatılmasıdır.

Yetkilendirme

Renkli Reçete Sistemine entegre olacak AHBS firmalarının ilk olarak Türkiye İlaç ve Tıbbi Cihaz Kurumu'na (bundan sonra Kurum olarak anılacaktır) başvurarak akredite olmaları gerekmektedir. Kurum tarafından firmalara iletilecek kullanıcı adı ve şifre bilgileri web api'lere erişim için zorunludur.

Servislerin test ortamında kullanılabilmesi için aşağıdaki bilgiler tanımlanmıştır. Test sürecinde tüm firmalar bu veriler üstünden işlem yapabilirler.

Parametre Adı	Değeri
Firma Kullanıcı Adı	11900012016
Firma Şifre	123456
Tesis Kodu	11000001 (TEST HASTANESİ)
Doktor Tc	11111111111

Tablo 1

Entegrasyon Adımları

Hekimler kullandıkları AHBS programlarında ilaç türü farketmeksizin reçetelerini oluşturacak, kaydet dedikleri anda otomatik olarak Renkli Reçete Sistemine yönlendirilecektir. Bu yönlendirme sırasında hasta, hekim, tesis, tanı ve ilaç bilgileri aktarılmış olacaktır. Eğer reçetede tüm ilaçlar normal reçete ile verilebilir ise hekim direkt olarak reçete imzalama aşamasına geçecektir. Ancak içerisinde kırmızı, yeşil, turuncu, mor veya kontrole tabi ilaç var ise hekim reçete oluşturma sayfasına gönderilecek, önceki versiyonlarda olduğu gibi reçeteyi Renkli Reçete Sisteminden kaydetmesi beklenecektir.

Bu yeni yönlendirme metodunun yanında Versiyon 2 olarak adlandırılan bir önceki yönlendirme metodu da hizmet vermeye devam edecektir.

Yöntem 1: JSON Veri Gönderme Metodu

Renkli Reçete JSON Dosyası ile Bilgi Aktarımı

Renkli Reçete Sistemine yönlendirme yapılırken AHBS sistemlerinden aktarılması gereken bilgilerin taşınması için JSON dosya kullanılmaktadır. Oluşturulacak olan JSON dosyanın içereceği alanlar ve bu alanların doldurulması ile ilgili kurallar ilerleyen bölümlerde verilmiştir.

AHBS sistemlerinden muayene ve reçete ekranları üzerinden reçete bilgileri doldurulduktan sonra reçete kaydetme işlemine geçilirken hazırlanacak olan JSON



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

dosyası PublicKey-PrivateKey tarzında şifrelenmiş olarak Renkli Reçete Sistemine gönderilecektir.

İlgili şifrelenmiş JSON dosyası yine PublicKey-PrivateKey tarzında çözülerek ilgili bilgiler Renkli Reçete Sistemi ekranlarında gösterilecek olup, Entegrasyon Adımları başlığı altında anlatıldığı gibi Renkli Reçete Sistemindeki kontrol sonrası ya imzalama sayfasına ya da reçete oluşturma sayfasına yönlendirilecektir. Her durumda imzalama işlemi Renkli Reçete Sistemi üzerinden devam ettirilecektir. .Net ve Java ortamlarında geliştirilen uygulamalar, Kurum tarafından paylaşılacak olan .dll ve .jar kütüphane dosyalarını kullanarak şifreleyebileceklerdir.

Şifreleme Algoritması

Renkli Reçete Sisteminde hibrit şifreleme algoritması kullanılmaktadır.

.Net ve Java ortamında geliştirilmemiş AHBS'lerin entegrasyonu veya şifreleme süreçlerini kendi tarafında geliştirmeyi tercih eden AHBS'ler için şifreleme algoritması aşağıda pseudo-code olarak verilmiştir.

```
var password = "ZJ=ENY'2H+0bm'oyIe6J";  
var RENKLIPUBLICFILE = "./RenkliRsaPublicKey.pem";  
private const int SALT_LENGTH = 32;  
private const string SALT_DAT = "./salt.dat";  
input = json;  
var saltBytes;  
// read salt file into byte array  
using (var instream = new FileStream(SALT_DAT, FileMode.Open))  
{  
    saltBytes = new byte[SALT_LENGTH];  
    instream.Read(saltBytes, 0, SALT_LENGTH);  
}
```

AES (Advanced Encryption Standard) algoritması kullanılarak (PKCS 5 V2.0 Scheme 2 tanımına göre SHA-1 HMac hesaplama fonksiyonu ile birlikte) bir private key üretilir. Üretilen private key ile JSON string'imiz şifrelenir.

Public Key kullanılarak RSA algoritması ile "private key şifrelenir". PADDING_PKCS1 = 11 alınacaktır). Anahtar boyutu 2048 bit.

Şifrelenmiş JSON aşağıdaki URL üzerinden test edilebilir :
<https://renklirecete.saglik.gov.tr/Auth/ApiLoginTest?encryptedJson=<<encryptedJson>>&encryptedAes=<<encodedAes>>>;

Versiyon 3:

Şifrelenmiş JSON ve şifrelenmiş AES Key kullanılarak <https://renklirecete.saglik.gov.tr/Api/GetToken> adresine HttpRequest oluşturulmalıdır.



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

Şifreleme sonrası elde edilen encryptedJson ve encryptedAes Json olarak serialize edilmeli ve body üzerinden gönderilmelidir. Alınan her token tek kullanımlık ve 15 dakikalık geçerlilik süresine sahip olacaktır.

Aşağıda PostMan uygulaması kullanılarak alınmış bir token örneği yer almaktadır.

```
{
  "encryptedJson": "...",
  "encryptedAes": "..."
}
```

Son olarak <https://renklirecete.saglik.gov.tr/Api/CheckPrescription?token={token}> şeklinde yönlendirme yapılmalıdır.

JSON Formatı Element Veri Seti

AHBS JSON formatları içerisinde adı geçen seçmeli elementlerin veri setleri *Tablo2*'de belirtilmiştir.

Veri setleri içerisinde seçilecek değere karşılık gelen rakamlar belirtilmiş olup, JSON formatı oluşturulurken seçilen değere karşılık gelen rakamlar gönderilecektir.

Element Adı	Veri Seti
doktorSertifikaKodu	0 – Yok 56 – Hemodiyaliz 109 – Aile Hekimliği
provizyonTip	1 – Normal 2 – Trafik 3 – Doğal Afet 4 – Adli Vaka 5 – İş Kazası 6 – Meslek Hastalığı 7 – Analık Hali

receteAltTuru	1 – Ayaktan Reçetesi 2 – Yatan Reçetesi 3 – Taburcu Reçetesi 4 – Günübirlik Reçetesi 5 – Acil Reçetesi 6 – Yeşil Alan Reçetesi 7 – Evde Bakım Reçetesi 8 – Gezici Sağlık Hizmeti Reçetesi
periyodBirimi	3 - Gün 4 - Hafta 5 - Ay 6 - Yıl
kullanımSekli	1 - Ağızdan(Oral) 2 - Cilt üzerine (Epidermal) 3 - Solunum yolu 4 - Ağız içi 5 - Burun içi(Intranazal) 6 - Dil altı(Sublingual) 7 - Dış kulak yolu 8 - Kolon 9 - Intra müsküler 10 - Göz üzerine 11 - Rektal 12 - Intra vajinal 13 - Intra dermal 14 - Subkutan 15 - Intra venöz 16 - Kalp içi 17 - Inhalasyon 18 - Trans dermal 19 - İntravezikal 20 - İntra-artiküler 21 - İntraperitoneal 22 - İntravitreal 23 - İntratekal 24 - İntraligamenter 25 - Perinöral 99 - Belirtilmemiş

Tablo 2



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

AHBS için JSON Formatı

AHBS sistemleri tarafından Renkli Reçete Sistemine gönderilmesi gereken JSON formatının alan adları ve tanımı *Tablo3*'de gösterildiği gibidir.

Alan Adı	Zorunlu	Açıklama
kullaniciKodu	Evet	Renkli Reçete Sistemi tarafından belirlenmiş olan AHBS kullanıcı kodudur.
parola	Evet	AHBS kullanıcısının parolası.
doktorTc	Evet	Reçeteyi yazan doktorun TC kimlik numarasıdır.
doktorMedulaPassword	Evet	Doktorun Medula şifresidir.
receteAltTuru	Evet	Medula tarafından belirlenmiş değer kümesinden seçilir. Değer kümesinden bir değer gönderilmezse 0 (sıfır) değeri gönderilir.
takipNo	Evet	Takip numarasını gösterir.
hastaTc	Evet	Hastanın TC kimlik numarasıdır.
hastaGsm	Evet	Hastanın cep telefonu numarasıdır. Başında 0 (sıfır) olmadan 10 (on) karakterden oluşan rakamlar bütünüdür. Boş ise 0 (sıfır) verilmelidir. Cep telefonu SMS gönderimi için kullanılmalıdır.
yuPass	Evet	Varsa, hastanın YUPASS numarasını gösterir. Yoksa, 0 (sıfır) değeri gönderilir.
protokolNo	Evet	Protokol numarasıdır.
provizyonTip	Evet	Aile hekimlikleri için provizyon tipi yoktur. Bu nedenle 0 (sıfır) değeri gönderilir.
doktorBransKodu	Evet	Medula tarafından belirlenmiş değerler kümesinden seçilir.
doktorSertifikaKodu	Evet	Medula tarafından belirlenmiş değerler kümesinden seçilir.
taniListesi	Evet	Reçeteye yazılacak tanıları içerir. Bir veya birden fazla <i>taniKodu</i> gönderilebilir. Boş olduğu durumda ise “[]” gönderilmelidir.
sysTakipNo	Hayır	Hastanın ilgili başvurusuna ait USS alınan takip numarasıdır. Gönderimi zorunlu değildir. Renkli Reçete Sistemindeki reçetelerin USS bildirimlerinin yapılabilmesi için hastane referans numarası ile birlikte kullanılmaktadır.
hastaneReferansNumarasi	Evet	Hastanın ilgili başvurusuna ait hizmet sunucusu bazında tekil referans numarası bilgisidir. USS (SYS) bildirimlerinde kullanılan değer ile aynı olmalıdır.

receteAciklama	Hayır	Hekim tarafından reçeteye girilen açıklama bilgisidir.
ilacListesi	Hayır	Reçeteye ait ilaç listesidir. Bir veya birden fazla ilaç gönderilebilir. Her bir ilaç nesnesi için kullanılacak alan adları ve tipleri tablo 4.1 de anlatılmıştır.
tesisKodu	Evet	AHBS için reçete yazan kurumun ÇKYS'de tanımlı olan tesis kodu belirtilecektir.

Tablo 3

ilacHariciUrunMu	int	Evet	Eğer reçetede yer alacak ürün ilaç ise 0, değil ise(hasta bezi vs.) 1 olarak gönderilmelidir.
urunAdi	string	Hayır	ilacHariciUrunMu değeri 1 ise doldurulmalıdır. Reçetede yer alacak ürünün adını gösterir.
gtin	string	Evet	İlaca ait barkod bilgisidir. 0 ile başlamalı ve 14 haneli olmalıdır. İlaç harici ürün ise "0" şeklinde gönderilmelidir.
doz1	int	Evet	Doz bilgisidir. İlacın günde kaç defa kullanılacağını gösterir. İlaç harici ürün ise 0 şeklinde gönderilmelidir.
doz2	float	Evet	Doz bilgisidir. İlacın her seferde kaç doz kullanılacağını gösterir. İlaç harici ürün ise 0 şeklinde gönderilmelidir.
periyod	int	Evet	İlacın kullanım periyodunu gösterir. İlaç harici ürün ise 0 şeklinde gönderilmelidir.
periyodBirimi	int	Evet	İlacın kullanım periyodunu gösterir. Veri kümesi yukarıda verilmiştir. SKRS kodları ile aynıdır. İlaç harici ürün ise 0 şeklinde gönderilmelidir.
kutu	int	Evet	İlaca ait kutu sayısını gösterir.
kullanimSekli	int	Evet	İlacın kullanım şeklini gösterir. Veri kümesi yukarıda verilmiştir. SKRS kodları ile aynıdır. İlaç harici ürün ise 0 şeklinde gönderilmelidir.
aciklama	string	Hayır	İlaca ait açıklama bilgisidir. "" şeklinde boş gönderilebilir.

Tablo 4.1



T.C. Sağlık Bakanlığı

T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

AHBS için JSON Formatı Örneği

```
{
  "kullaniciKodu":"11900012016",
  "parola":"123456",
  "takipNo":"2KHYLLD",
  "hastaGsm":"0",
  "tesisKodu":"11000001",
  "doktorTc":"11111111111",
  "doktorMedulaPassword":"0",
  "doktorBransKodu":"1400",
  "doktorSertifikaKodu":"109",
  "hastaTc":"11111111111",
  "yuPass":"0",
  "protokolNo":"107713",
  "provizyonTip":"1",
  "receteAltTuru":"1",
  "hastaneReferansNumarasi":"0",
  "sysTakipNo":"0",
  "receteAciklama":"Test reçetesidir",
  "taniListesi":[
    {
      "taniKodu":"R05"
    },
    {
      "taniKodu":"J39.9"
    }
  ],
  "ilacListesi":[
    {
      "ilacHariciUrunMu":0,
      "urunAdi":null,
      "gtin":"08699522705009",
      "doz1":2,
      "doz2":1.0,
      "periyod":1,
      "periyodBirimi":1,
      "kutu":1,
      "kullanimSekli":1,
      "aciklama":""
    },
    {
      "ilacHariciUrunMu":0,
      "urunAdi":null,

```



```
"gtin": "08681428010592",  
"doz1": 3,  
"doz2": 1.0,  
"periyod": 1,  
"periyodBirimi": 1,  
"kutu": 2,  
"kullanımSekli": 1,  
"acıklama": "Açıklama metni"  
},  
{  
"ilacHariciUrunMu": 1,  
"urunAdi": null,  
"gtin": "0",  
"doz1": 0,  
"doz2": 0.0,  
"periyod": 0,  
"periyodBirimi": 0,  
"kutu": 10,  
"kullanımSekli": 0,  
"acıklama": null  
}  
]  
}
```

JAVA ile JSON Nasıl Oluşturulur?

JAVA programlama dili ile yazılmış olan AHBS uygulamalarında kullanılmak üzere hazırlanmış kod örneği bu başlık altında verilmiştir.

Bu örnekte *JSONObject* ve *JSONArray Class*'larını kullanabilmek için **java-json.jar** kütüphanesi eklenmiştir.

```
public static void main(String []args){  
    JSONObject obj = new JSONObject();  
    obj.put("kullaniciKodu", "11900012016");  
    obj.put("parola", "123456");  
    obj.put("doktorBransKodu", "1400");  
    obj.put("doktorMedulaPassword", "0");  
    obj.put("hastaGsm", "5005434343");  
    obj.put("doktorSertifikaKodu", "109");  
    obj.put("doktorTc", "11111111111");  
    obj.put("hastaTc", "11111111111");  
}
```



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

```
obj.put("protokolNo", "PRTKL-1");  
obj.put("provizyonTip", "1");  
obj.put("receteAltTuru", "1");  
obj.put("takipNo", "0");  
obj.put("tesisKodu", "11000001");  
obj.put("yuPass", "0");  
obj.put("sysTakipNo", "0");  
obj.put("hastaneReferansNumarasi", "0");  
obj.put("receteAciklama", "Test reçetesidir.");
```

```
JSONArray taniList = new JSONArray();  
JSONObject taniObj = new JSONObject();  
for(int i = 0; i < taniListesiUzunlugu; i++)  
{  
    taniObj.put("taniKodu", taniKodu);  
    taniList.put(taniObj);  
}  
obj.put("taniListesi", taniList);
```

```
JSONArray ilacList = new JSONArray();  
JSONObject ilacObj = new JSONObject();  
for(int i = 0; i < ilacListesiUzunlugu; i++)  
{  
    ilacObj.put("gtin", gtin);  
    ilacObj.put("doz1", doz1);  
    ilacObj.put("doz2", doz2);  
    ilacObj.put("periyod", periyod);  
    ilacObj.put("periyodBirimi", periyodBirimi);  
    ilacObj.put("kutu", kutu);  
    ilacObj.put("kullanimSekli", kullanimSekli);  
    ilacObj.put("aciklama", aciklama);  
    ilacList.put(taniObj);  
}  
obj.put("ilacListesi", ilacList);  
}
```



.Net ile JSON Nasıl Oluşturulur?

.Net programlama dili ile yazılmış olan AHBS uygulamalarında kullanılmak üzere hazırlanmış kod örneği bu başlık altında verilmiştir.

Burada belirtilen kod sadece örnek amaçlıdır. Başka yöntemler kullanılarak da JSON format oluşturabilirsiniz.

Proje içerisinde JSON format elementlerini tutmak için bir Class oluşturacağız ardından JSON oluşturabilmek için Newtonsoft.Json paketini projemize aşağıdaki gibi ekleyeceğiz.

Install-Package Newtonsoft.Json -Version 11.0.2

Bu paketi kullanarak tek satır ile oluşturacağımız nesneyi json olarak serailize edeceğiz.

JSON elementlerinin tutulduğu *Class* aşağıdaki gibidir;

```
public class AhbsApiLoginModel
{
    public string kullanıcıKodu { get; set; }
    public string parola { get; set; }
    public string takipNo { get; set; }
    public string hastaGsm { get; set; }
    public string tesisKodu { get; set; }
    public string doktorTc { get; set; }
    public string doktorMedulaPassword { get; set; }
    public string doktorBransKodu { get; set; }
    public string doktorSertifikaKodu { get; set; }
    public string hastaTc { get; set; }
    public string yuPass { get; set; }
    public string protokolNo { get; set; }
    public string provizyonTip { get; set; }
    public string receteAltTuru { get; set; }
    public string hastaneReferansNumarasi { get; set; }
    public string sysTakipNo { get; set; }
    public string receteAciklama { get; set; }
    public AhbsApiTaniModel[] taniListesi { get; set; }
    public AhbsApiIlacModel[] ilacListesi { get; set; }
}
```

```
public class AhbsApiTaniModel
{
    public string taniKodu { get; set; }
}
```



T.C. Sağlık Bakanlığı

T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

```
public class AhbsApiIlacModel
{
    public int ilacHariciUrunMu { get; set; }
    public string urunAdi { get; set; }
    public string gtin { get; set; }
    public int doz1 { get; set; }
    public float doz2 { get; set; }
    public int periyod { get; set; }
    public int periyodBirimi { get; set; }
    public int kutu { get; set; }
    public int kullanimSekli { get; set; }
    public string aciklama { get; set; }
}
```

JSON oluşturabilmek için kullanılacak olan kod aşağıdaki gibidir;
var json = JsonConvert.SerializeObject(AhbsApiLoginModel);

Son olarak proje içerisinde verileri göndereceğimiz proje Class'ında kullanılacak olan örnek kodlar aşağıdaki gibidir;

```
var AhbsApiLoginModel = new AhbsApiLoginModel
{
    kullanıcıKodu = "11900012016",
    parola = "123456",
    doktorBransKodu = "1400",
    doktorMedulaPassword = "0",
    hastaGsm = "0",
    doktorSertifikaKodu = "109",
    doktorTc = "11111111111",
    hastaTc = "11111111111",
    protokolNo = "107713",
    provizyonTip = "1",
    receteAltTuru = "1",
    takipNo = "2KHILLD",
    tesisKodu = "11000001",
    yuPass = "0",
    sysTakipNo = "0",
    hastaneReferansNumarasi = "0",
    receteAciklama = "Test reçetesidir",
    taniListesi = new[]
    {
```



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

```
new AhbsApiTaniModel { taniKodu = "R05" },
new AhbsApiTaniModel { taniKodu = "J39.9" }
},
ilacListesi = new[]
{
new AhbsApiIlacModel()
{
gtin = "08699522705009",
doz1 = 2,
doz2 = 1,
periyod = 1,
periyodBirimi = 1,
kutu = 1,
kullanimSekli = 1,
aciklama = ""
},
new AhbsApiIlacModel()
{
gtin = "08681428010592",
doz1 = 3,
doz2 = 1,
periyod = 1,
periyodBirimi = 1,
kutu = 2,
kullanimSekli = 1,
aciklama = "Açıklama Metni"
},
new AhbsApiIlacModel()
{
ilacHariciUrunMu = 1,
gtin = "0",
doz1 = 0,
doz2 = 0,
periyod = 0,
periyodBirimi = 0,
kutu = 10,
kullanimSekli = 0
}
}
};
var json = JsonConvert.SerializeObject(AhbsApiLoginModel);
```



T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

Nasıl Şifreleme Yapacağız?

Buradaki bilgiler .Net veya Java üzerinden geliştirilmesi yapılmış AHBS uygulamaları için geçerlidir.

Şifreleme yapılabilmesi için Kurum tarafından gönderilecek olan **salt.dat** ve **RenkliRsaPublicKey.pem** dosyalarına ihtiyaç duyulmaktadır. Bu dosyalar **KESİNLİKLE** değiştirilmemelidir ve yine Kurum tarafından gönderilecek olan DLL dosyaları ile aynı dizinde yer almalıdır. Bu dosyaları başka bir lokasyonda tutmak isteyen firmalar veya web tabanlı uygulamalar Encrypt metodunda yer alan ikinci parametreyi kullanarak ilgili dosyaların kök dizinini belirtebilir.

Örnek Şifreleme Kod Parçacıkları

.Net için örnek kod parçacığı

.Net ortamında şifreleme işlemleri için BouncyCastle kütüphanesinin 1.7 sürümü kullanılmaktadır. Kütüphanenin bu sürümü RenkliEncryptor.dll kütüphanesi ile birlikte Kurum tarafından gönderilecektir.

Oluşturulan *string* tipindeki JSON nesnesinin şifrelenmesine ait örnek kod aşağıdaki gibidir;

```
"[{\"parola\":\"parola1\", \"kullaniciKodu\":\"kullanici1\", \"doktorTc\":\"1111111111\", \"doktorMedulaPassword\":\"pass1\", \"receteAltTuru\":\"1\", \"takipNo\":\"tkp1\", \"hastaTc\":\"1111111111\", \"hastaGsm\":\"0\", \"tesisKodu\":\"11000001\", \"protokolNo\":\"A-461942\", \"provizyonTip\":\"0\", \"doktorSertifikaKodu\":\"109\", \"doktorBransKodu\":\"9999\", \"hastaneReferansNumarasi\":\"0\", \"taniListesi\":[\"taniKodu\":\"K21\"}], \"ilacListesi\":[\"gtin\":\"08699536640013\", \"doz1\":\"1\", \"doz2\":\"1\", \"periyod\":\"1\", \"periyodBirimi\":\"1\", \"kutu\":\"1\", \"kullanimSekli\":\"1\", \"aciklama\":\"\"}], \"yuPass\":\"0\", \"sysTakipNo\":\"0\", \"receteAciklama\":\"Test reçetesidir\"]];
```

```
Var link = new RenkliEncryption.RenkliEncryptor().Encrypt(json, null, true, \"https://renklirecetedev.saglik.gov.tr\");
```

Encrypt metoduna ait parametreler sırasıyla aşağıda açıklanmıştır.

- Birinci parametre olarak hazırlanmış olan json string alınmaktadır.
- İkinci parametre string olarak dosya kök dizini almaktadır. Eğer size verilen public key ve salt.dat dosyası DLL ile aynı lokasyonda ise metoda null veya "" verilebilir.
- Üçüncü parametre hazırlanan json da ilaç kümesinin olup olmadığını belirtir. Entegrasyon sürecinde firmaların sorun yaşamaması için üçüncü parametre false olarak set edilirse aynı dll versiyon 2 yönlendirmesine de izin verecektir. İlaç kümesi ekleniyorsa true olarak set edilmelidir.
- Yönlendirme yapılacak url bilgisini gösterir. Testler sırasında "https://renklirecetedev.saglik.gov.tr" adresi kullanılabilir.



T.C. Sağlık Bakanlığı

T.C. SAĞLIK BAKANLIĞI
RENKLİ REÇETE SİSTEMİ
AHBS Entegrasyon Dokümanı

Yayın No : 3.0
Yayın Tarihi : 07.11.2018
Doküman No : TITCK-RRS-JSON

Java için örnek kod parçacığı

Java ile şifreleme için sağlanan kod için BouncyCastle kütüphanesinin 1.52 sürümü kullanılmaktadır. İstemci tarafında sürüm değişikliği Kurum ile koordine edilmeden yapılmamalıdır.

Projeye bcpkix-jdk15on-1.52.jar, bcprov-ext-jdk15on-1.52.jar, bcprov-jdk15on-1.52.jar ve ayrıca kurum tarafından gönderilecek olan .jar dosyası kullanılmalıdır.

Oluşturulan string tipindeki JSON nesnesinin şifrelenmesine ait metod c# örneğinde olduğu gibi 4 parametre almaktadır, kullanım yönünden farklılık yoktur.

Her İki Metot İçin İmza İşlemleri

Elektronik İmza

Elektronik imza olarak ArkSigner kullanılmaktadır.

MSI paketi;

<https://www.arksigner.com/dist/site/upload/ArkSigner-Installer-MSI-v2.2.1.msi>

Tüm Sürümler;

<https://www.arksigner.com/indir>