İçindekiler

[GELİŞTİRME SÜRECİ 2](#_Toc92200541)

[Dinamik Webservis Dokümantasyonu 4](#_Toc92200542)

[Veri Eşitleme 5](#_Toc92200543)

[Veri Aktarımı Öncesi 5](#_Toc92200544)

[Veri Edinmede İzlenmesi Gereken Yol 5](#_Toc92200545)

[Veri Aktarımında İzlenmesi Gereken Yol 6](#_Toc92200546)

[Webmethod Listesi: 7](#_Toc92200547)

[GsbDataResult 8](#_Toc92200548)

[Hata Kodları 9](#_Toc92200549)

# GELİŞTİRME SÜRECİ

|  |  |  |  |
| --- | --- | --- | --- |
| **Tarih** | **Versiyon** | **Değişiklik Yapılan Bölüm** | **Açıklama** |
| 01.01.2022 | v.2.0 |  | Swagger, yeni metotlar eklendi. |
| 10.05.2017 | v.1.0.2 |  | Vizeler için FederasyonSezonId zorunluluğu geldi. |
| 01.11.2016 | v.1.0.1 |  | Yeni methodlar eklendi. Hata kodları listesi eklendi. |
| 04.05.2016 | v.1.0 | ---- | Webservis yayına alındı. |

# Dinamik Webservis Dokümantasyonu

Gereksinimler:

* Webservis kullanımı için kullanıcı adı ve şifre

1. Webservisi kullanabilmek ve methodları çağırabilmek için örnek kod aşağıdaki gibidir.

**C#**

var client = new RestClient("https://servis3.gsb.gov.tr/fedprotokoltest/api/FederasyonServisleri/IlListesiGetir");

client.Timeout = -1;

var request = new RestRequest(Method.GET);

request.AddHeader("Authorization", "Basic U3BvckJpbGdpVGVzdDoxMjM0NQ==");

IRestResponse response = client.Execute(request);

Console.WriteLine(response.Content);

## Veri Eşitleme

### Veri Aktarımı Öncesi

Veri aktarımından önce bazı parametrik değerleri (il id, ilçe id, lisans işlem türü id vb.) alabilmek için önce aşağıdaki metotları çağırmanız gerekmektedir.

IlListesiGetir()  
IlceListesiGetir()  
FederasyonaAitSezonlar()  
LisansIslemTurleri()  
KulupleriGetir()  
KpsKisiDurumKodlari()  
HataKodlari()

### Veri Edinmede İzlenmesi Gereken Yol

**Sporcu**: Öncelikle SporculariGetir() metodu kullanılarak SporBilgi uygulaması lisans modülündeki federasyonunuza ait lisansı bulunan sporcuların T.C. kimlik numaraları çekilmelidir. Limit ve offset parametreleri kullanılarak belli aralıklarla bu veri çekme işlemi yapılmalıdır.

int limit: Çekmek istediğiniz veri limitini belirler. Max = 1000

int offset: Çekilen verinin sınırlı sayıda olmasından dolayı sayfalama uygulanmaktadır. Diğer sayfadaki verileri çekmek için offset parametresinin gönderilmesi gerekmektedir. Standart değer = 0

Bir sporcuya ait daha detaylı bilgiye ulaşmak için bu metottan edindiğiniz kimlik numaralarını SporcuGetir() metoduna parametre olarak eklerseniz kişi bilgilerine ulaşabilirsiniz. Bu işlemden sonra, eğer veri kaydetme işlemi yapacaksanız, kendi uygulamanızdaki bilgilerle karşılaştırılarak eksikliklerin ve farklılıkların tespit edilmesi gerekmektedir.

**Kulüp:** Öncelikle KulupleriGetir() metodu kullanılarak SporBilgi uygulaması kulüp modülündeki federasyonunuza ait spor dallarını yaptığını taahhüt eden kulüpler çekilmelidir. Limit ve offset parametreleri kullanılarak belli aralıklarla bu veri çekme işlemi yapılmalıdır.

int limit: Çekmek istediğiniz veri limitini belirler. Max = 1000

int offset: Çekilen verinin sınırlı sayıda olmasından dolayı sayfalama uygulanmaktadır. Diğer sayfadaki verileri çekmek için offset parametresinin gönderilmesi gerekmektedir. Standart değer = 0

Kulüp adı, derbis kütük no, il, ilçe, faaliyet durumu, kulüp no, Başkan bilgileri, adresi, telefonu, epostası, web sitesi ve faks gibi özet bilgileri edinebilirsiniz. Bir kulüp hakkında daha detaylı (Kuruluş tarihi, Kulüp logosu, Forma rengi vb…) bilgiye ulaşmak için KulupGetirDetayli() metodunu çağırabilirsiniz.

### Veri Aktarımında İzlenmesi Gereken Yol

**Sporcu:** Eğer federasyon lisanslı sporcu SporculariGetir() metodu ile gelmediyse, SporcuKaydet() metodu çağırılarak kişinin kaydı yapılmalı ve dönen bilgiler tarafınıza kaydedilmelidir. Kişi zaten kayıtlı uyarısı dönerse, SporcuGetir() metodu ile sporcunun lisans bilgisi ve lisans geçmişi tüm bilgileri alınmalıdır.

**Lisans:** Eğer ilk defa lisans kaydedilecek ise; LisansTescil metodu kullanılmalıdır.

SporcuGetir() metodundan edindiğiniz lisansı güncellemek için lisans guid’i kullanmanız gerekmektedir. Bu güncelleme metotları aşağıda listelenmiştir.

DonusumYapFerdidenKulube: Ferdi lisansı kulüp lisansına çevirmek için kullanılacak metot. KulupGetir() metodundan edindiğiniz KulupGuid değerini kullanmanız gerekmektedir.

KuluptenFerdiye: Kulüp lisansını, ferdi lisansa dönüştürmek için kullanılacak metot.

LisansIlDegisikligi: Bir lisansın il bilgisini değiştirmek için kullanılacak metot.

LisansSil: Hatalı kaydedilen lisansı silmek için kullanılacak metot.

SpordaliEkleCikar: Kayıtlı lisansa spor dalı eklemek ve çıkarmak için kullanılacak metot.

Transfer: Kulüp değişikliği yapmak için kullanılacak metot. KulupGetir() metodundan edindiğiniz KulupGuid değerini kullanmanız gerekmektedir.

Vize: Vize süresi dolmuş bir lisansı vize yapmak için kullanılacak metot.

## Webmethod Listesi:

GET /api/FederasyonServisleri/HataKodlari

GET /api/FederasyonServisleri/IlListesiGetir

GET /api/FederasyonServisleri/IlceListesiGetir

GET /api/FederasyonServisleri/FederasyonaAitSezonlar

GET /api/FederasyonServisleri/LisansIslemTurleri

GET /api/FederasyonServisleri/KulupleriGetir

GET /api/FederasyonServisleri/KpsKisiDurumKodlari

POST /api/FederasyonServisleri/SporcuKaydet

GET /api/FederasyonServisleri/KulupGetirDetayli

GET /api/FederasyonServisleri/SporcuGetir

GET /api/FederasyonServisleri/SporculariGetir

POST /api/FederasyonServisleri/LisansTescil

POST /api/FederasyonServisleri/DonusumYapFerdidenKulube

POST /api/FederasyonServisleri/KuluptenFerdiye

POST /api/FederasyonServisleri/LisansIlDegisikligi

POST /api/FederasyonServisleri/LisansSil

POST /api/FederasyonServisleri/SpordaliEkleCikar

POST /api/FederasyonServisleri/Transfer

POST /api/FederasyonServisleri/Vize

### GsbDataResult

Metotların dönüşünde kullanılan özel sınıf’a verdiğimiz isim.

Dönüş tipi: GsbDataResult()

GsbDataResult sınıfının içeriği aşağıdaki gibidir.

**bool Status :** Veriler çekilirken bir problem olup olmadığını kontrol etmek için Status alanını kontrol ediniz. True problem oluşmadığı anlamına gelmektedir.

**T Data :** Metodun dönüş tipinde belirtilen tipte gelen nesne.

**string Message :** Status alanı false olduğunda ErrorMessage alanını kontrol ederek oluşan hata ile ilgili daha detaylı bilgiye ulaşabilirsiniz.

**long ErrorCode :** Dönen hata mesajının hata kodu.

**string MessageType :** (Bu senaryoda kullanılmıyor).

### Hata Kodları

1 = TC Kimlik No, Cep Telefonu ve EPosta alanı boş bırakılamaz.

2 = Hatalı TC Kimlik No.

3 = EPosta alanı boş bırakılamaz.

4 = Hatalı eposta adresi.

5 = Cep telefonu alanı boş bırakılamaz.

6 = Hatalı Cep Telefonu girişi. Cep telefonu 11 haneli ve 05XXXXXXXXX formatında olmalıdır.

7 = Hatalı Ev Telefonu girişi. Ev telefonu 11 haneli ve 0XXXXXXXXXX formatında olmalıdır.

8 = Hatalı İş Telefonu girişi. İş telefonu 11 haneli ve 0XXXXXXXXXX formatında olmalıdır.

9 = Doğru kan gurubu bilgisini seçmek için ilgili methodu kullanmanız gerekmektedir.

10 = Geçersiz lisans türü girişi. Doğru lisans türü bilgisini seçmek için ilgili methodu kullanmanız gerekmektedir.

11 = Geçersiz il kodu girişi. Doğru il kodu seçmek için ilgili methodu kullanmanız gerekmektedir.

12 = Geçersiz ilçe kodu girişi. Doğru ilçe kodu seçmek için ilgili methodu kullanmanız gerekmektedir.

13 = Lisans tarihi boş bırakılamaz.

14 = Lisans tarihi bugünden büyük olamaz.

15 = Geçersiz kulüp girişi. Doğru kulüp bilgisini seçmek için ilgili methodu kullanmanız gerekmektedir.

16 = Geçersiz branş girişi. Seçmiş olduğunuz kulüp ilgili branşı taahhüt etmiyor.

17 = Geçersiz spor dalı girişi. Doğru spor dalı bilgisini seçmek için ilgili methodu kullanmanız gerekmektedir.

18 = Bu federasyondan lisans çıkarmaya yetkiniz yoktur.

19 = Geçersiz Guid değeri. Değişken adı: {0}

20 = Kişi bulunamadı.

21 = Sporcunun federasyonunuza ait lisansı vardır. Mevcut lisansı güncellemek için LisansGuid değerini gönderiniz.

22 = Lisans bulunamadı.

23 = İlgili lisansın geçerli vizesi var.

24 = Kişi bulunamadı.

25 = Lütfen pozitif bir sayı giriniz.

26 = Sadece yetkili federasyon kullanıcısı bu methodu kullanabilir.

27 = Geçersiz kullanıcı veya şifre.

28 = En fazla 1000 kayıt getirebilirsiniz.

29 = LİSANS GÜNCELLENEMEDİ!: Lisansta farklılık bulunamadı.

30 = Lisans işlem tarihi lisans tarihinden büyük olamaz.

31 = En az bir vize göndermelisiniz.

32 = Geçersiz SporcuKayitTarihi değeri.

33 = Sporcu kayıt tarihi bugünden büyük olamaz.

34 = Federasyon bulunamadı.

35 = Authentication Header vermediniz.

36 = izin verilen süreden önce istekte bulundunuz. {0} saniye sonra istek yapabilirsiniz.

37 = Kayıt bulunamadı.

38 = Bu Tc Kimlik No ile Kişi Zaten Kayıtlı.

39 = KPS Kişi bilgileri çekilirken hata oluştu.

40 = Sağlık Bakanlığı Sistemine İşlemiş Z02.05 tanı kodlu geçerli bir Sağlık Raporu Bulunamadı.

41 = Seçilen Federasyon için henüz sonuçlanmamış bir başvurunuz bulunmaktadır.

42 = Bu lisans için gerekli yaş şartını sağlamıyorsunuz.

43 = Zaten bu federasyona ait lisansınız bulunmaktadır.

44 = Kulübünüz bu branşı taahhüt ettiği için ferdi lisans çıkaramazsınız.

45 = E-Posta Göndermede Hata Oldu.

46 = Lisans Başvuru Kaydetmede Hata Oldu.

47 = Velayet Sorgulanamadı.

48 = Velayet Doğrulanamadı.

49 = Eklemek istediğiniz kişi vefat etmiştir.

50 = Göndermiş olduğunuz vize(ler) zaten kayıtlıdır.

51 = Lisans Vize Başvuru Kaydetmede Hata Oldu.

52 = Göndermiş olduğunuz sezona ait vize mevcuttur.

53 = Vize başlangıç tarihi ile vize bitiş tarihi arası bir yıldan fazla olamaz.

54 = Vize bitiş tarihi sezon bitiş tarihi olmalı.

55 = Vize bitiş tarihi vize başlangıç tarihinden küçük olamaz.

56 = Göndermiş olduğunuz Vize Guid değeri ile Lisans Guid değeri eşleşemedi.

57 = Girilen tarih geçersizdir. Lütfen tarih alanlarını yyyyMMddHHmmss formatında gönderiniz.

58 = Vize başlangıç tarihi lisans tescilden küçük tarihli olamaz.

59 = İleri tarihli vize gönderilemez.

60 = Kulüp faal olmadığı için işlem gerçekleştirilemez.

61 = Fotoğraf boyutu 150 x 200'den büyük olduğu için işlem gerçekleştirilemez.

62 = Kullanıcı bulunamadı.

63 = Göndermeye çalıştığınız lisansa ait sezon sistemde mevcut değildir. Kayıt yapabilmek için lütfen sistem yöneticinize başvurunuz.

64 = Göndermeye çalıştığınız FederasyonSezonId geçersizdir. Doğru FederasyonSezonId için lütfen FederasyonaAitSezonlar() metodunu kullanınız.

65 = Gönderdiğiniz vize başlangıç tarihi FederasyonSezonId ile uyuşmamaktadır. Lütfen bilgileri kontrol edip tekrar gönderiniz.

66 = Vefat eden sporcu için vefat tarihinden sonra işlem yapılamaz. Vefat tarihi : {0}.

67 = Sporcu kayıt tarihinden eski tarihli lisans veya vize kaydedilemez.

68 = Bu kişinin T.C. vatandaşlığına ait kimlik kaydı var.

69 = Başlangıç tarihi bitiş tarihinden büyük olamaz.

70 = Ek lisans sadece kulüp lisanslarına çıkarılabilir.

71 = Lisans tarihinden eski tarihli ek lisans basılamaz.

72 = Ek lisans veriliş tarihi bugünden büyük olamaz.

73 = Ek lisansın ait olduğu lisansın ek lisans veriliş tarihine denk gelen sezona ait vize kaydı olmak zorundadır.

74 = Sporcunun yaşı ikinci lisans için uygun değildir.

75 = Ek lisansın kulübü ile lisansın kulübü aynı olamaz.

76 = Kulüp ili ile lisans ili farklı olamaz.

77 = Kulüp lisansının işlem tarihi kulüp tescil tarihinden küçük olamaz.

78 = Spor Kart Vize Başvuru Kaydetmede Hata Oldu.

79 = Zaten bu federasyona ait sporkartınız bulunmaktadır.

80 = Kulüp bulunamadı.

81 = 1 Ekim itibariyle webservis kullanıma kapatılmıştır. Lütfen Spor Kuruluşları Dairesi ile iletişime geçiniz.

82 = Kullanıcınız federasyon yetkili kullanıcısı değil.

83 = Geçerli bir Base64 stringi giriniz.

84 = Geçersiz lisans işlem türü.

100 = İşlem başarıyla gerçekleştirildi.

0 = Exception.