

AKS Express Kurir d.o.o.

AKS Web Service

Verzija: 1.1.4.52

Sadržaj:

Lokacija aplikacije	3
Metoda za preuzimanje liste postanskih brojeva.....	3
Metoda za preuzimanje sifarnika sadržaja posiljke.....	3
Metoda za upisivanje nove posiljke	4
Metoda za upisivanje nove posiljke JSON protokolom	6
Metoda za pregled trenutnog statusa posiljke (po ID-u)	8
Metoda za pregled promene statusa posiljke (po ID-u).....	9
Metoda za pregled promene statusa posiljke (po referenci)	10
Metoda za pregled trenutnog statusa liste posiljaka	11
Metoda za pregled trenutnog statusa liste posiljaka JSON protokolom	12
Metoda za trajno otkazivanje posiljke	13
Metoda za trajno otkazivanje posiljke JSON protokolom	14
Metoda za preuzimanje podataka za posiljku	15
Metoda za preuzimanje podataka za posiljku JSON protokolom	17
Metoda za upisivanje nove pozivnice	19
Metoda za upisivanje nove pozivnice JSON protokolom	21
Metoda za dobijanje ID posiljke za pozivnicu	23
Metoda za dobijanje ID naloga povrata za posiljku (po ID-u posiljke).....	24
Metoda za dobijanje liste promene statusa za sve naloge.....	25
Lista gresaka	27

Lokacija aplikacije

<http://www.akskurir.com/AKSvipService/>

Metoda za preuzimanje liste postanskih brojeva

<http://www.akskurir.com/AKSvipService/PostanskiBrojList>

Protokol: output JSON

return List<PostanskiBroj >

```
public class PostanskiBroj
{
    public int ID { get; set; } //Postanski broj npr 11000
    public string Mesto { get; set; } //Mesto npr Beograd
    public int CentarID { get; set; } //AKS Centar
}
```

Metoda za preuzimanje sifarnika sadrzaja posiljke

<http://www.akskurir.com/AKSvipService/OpisSadrzajList>

Protokol: output JSON

return List<OpisSadrzaj >

```
public class OpisSadrzaj
{
    public int ID { get; set; } //ID Koji se koristi za posiljku
    public string Opis { get; set; } //Opis npr. Tehnicka roba
}
```

Metoda za upisivanje nove posiljke

<http://www.akskurir.com/AKSvipService/NovaPosiljka/ps>

Protokol: input `text/plain` output: JSON

Posiljka ps;

```
public class Posiljka
{
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public string PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PosiljalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PosiljalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PosiljalacPostanskiBroj { get; set; } //iz liste npr 11000

    public string PrimalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PrimalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PrimalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PrimalacPostanskiBroj { get; set; } //iz liste npr 11000
    public string PrimalacMesto { get; set; } //Mesto primaoca ukoliko se
koristi nekompletna posiljka i ignorese PrimalacPostanskiBroj (PrimalacPostanskiBroj=0).
Inace ostaviti prazan string ""

    public int BrojPaketa { get; set; } //Broj dodatnih paketa za posiljku 0
- default
    public double MasaPosiljke { get; set; } //Ukupna masa posiljke u kilogramima
    public int OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste

    public string KlijentRef { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef2 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef3 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef4 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef5 { get; set; } //Referenca za povezivanje posiljke
(max len 50) - Ukoliko je NekompletnaPosiljka = 0

    public double IznosVrednostiPouzece { get; set; } //Iznos vrednosti za placanje
pouzece
    public int ElektronskaUplata { get; set; } //Da li se za posiljku izvrsava
elektronska uplata (1-da, 0-ne)
    public int PovratOtpremnice { get; set; } //Da li je potreban povrat potpisane
otpremnice (1-da, 0-ne)
    public double VrednostPosiljkeOsiguranje { get; set; } //Vrednost posiljke za
iznos osiguranja
```

```

        public int SMSPotvrdaIsporuke { get; set; } //Prijem SMS poruke po uspesnoj
        isporuci posiljke (1-da 0-ne)

        public double CenaTezina { get; set; } // Cena transporta po tezini
        public int NacinPlacanja { get; set; } // Nacin placanja usluge transporta
        0-PosiljalacVirman, 1- PosiljalacGotovina, 2-Primalac
        public int PlacanjeDodatneUsluge { get; set; } //Nacin placanja dodatne usluge
        transporta 0-Posiljalac paketa, 1-PrimalacPaketa

        public string Napomena { get; set; } //<120 karaktera = napomena koja ce se
        prikazivati kuriru prilikom isporuke

        public int NekompletnaPosiljka { get; set; } // Da li se posiljka salje kao
        nekompletna i zahteva dodatnu proveru unutar AKSa (1-da 0-ne). Ukoliko se koristi
        nekompletna posiljka polje KlijentRef5 ce biti ignorisano

    }

```

Povratne informacije:

ErrorCode: (Int) - iz Liste

PosiljkaID: (Int64) Jedinstveni Barkod/Broj posiljke pod kojim ce se posiljka voditi u sistemu

Paketi: (Int64[]) Jedinstveni Barkod/Broj pod kojim ce se svaki prateci paket voditi u sistemu

```

return Json(new { ErrorCode = Res, PosiljkaID = PosiljkaID, Paketi= PaketiRes});

```

Metoda za upisivanje nove posiljke JSON protokolom

<http://www.akskurir.com/AKSVipService/NovaPosiljkaJSON/>

Protokol: input: JSON output: JSON

Ulazni parametar: Posiljka

```
public class Posiljka
{
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public string PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PosiljalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PosiljalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PosiljalacPostanskiBroj { get; set; } //iz liste npr 11000

    public string PrimalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PrimalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PrimalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PrimalacPostanskiBroj { get; set; } //iz liste npr 11000

    public int BrojPaketa { get; set; } //Broj dodatnih paketa za posiljku 0
- default
    public double MasaPosiljke { get; set; } //Ukupna masa posiljke u kilogramima
    public int OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste

    public string KlijentRef { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef2 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef3 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef4 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef5 { get; set; } //Referenca za povezivanje posiljke
(max len 50)

    public double IznosVrednostiPouzece { get; set; } //Iznos vrednosti za placanje
pouzecem
    public int ElektronskaUplata { get; set; } //Da li se za posiljku izvrsava
elektronska uplata (1-da, 0-ne)
    public int PovratOtpremnice { get; set; } //Da li je potreban povrat potpisane
otpremnice (1-da, 0-ne)
    public double VrednostPosiljkeOsiguranje { get; set; } //Vrednost posiljke za
iznos osiguranja

    public int SMSPotvrdaIsporuke { get; set; } //Prijem SMS poruke po uspesnoj
isporuci posiljke
```

```

        public double CenaTezina { get; set; }          // Cena transporta po tezini
        public int NacinPlacanja { get; set; }          // Nacin placanja usluge transporta
0-PosiljalacVirman, 1- PosiljalacGotovina, 2-Primalac
        public int PlacanjeDodatneUsluge { get; set; } //Nacin placanja dodatne usluge
transporta 0-Posiljalac paketa, 1-PrimalacPaketa

        public string Napomena { get; set; } //<120 karaktera = napomena koja ce se
prikazivati kuriru prilikom isporuke

    }

```

Povratne informacije:

ErrorCode: (Int) – iz Liste

PosiljkaID: (Int64) Jedinstveni Barkod/Broj posiljke pod kojim ce se posiljka voditi u sistemu

Paketi: (Int64[]) Jedinstveni Barkod/Broj pod kojim ce se svaki prateci paket voditi u sistemu

```
return Json(new { ErrorCode = Res, PosiljkaID = PosiljkaID, Paketi= PaketiRes});
```

Metoda za pregled trenutnog statusa posiljke (po ID-u)

<http://www.akskurir.com/AKSVipService/TrenutniStatus/ID>

Protokol: output: JSON

ID: Jedinstveni broj posiljke

Povratne informacije:

ErrorCode: (Int) – iz Liste

Status: (Int) – trenutni status posiljke

Metoda za pregled promene statusa posiljke (po ID-u)

<http://www.akskurir.com/AKSVipService/Pracenje/ID>

Protokol: output: JSON

ID: Jedintveni broj posiljke

Povratne informacije:

ErrorCode: (Int) – iz Liste

StatusList: StatusProm

```
public class StatusProm
{
    public DateTime Vreme { get; set; } //Datum i vreme skeniranja - promene statusa
    public int VremeInt { get; set; } //Datum i vreme kao sekunde od 01.01.2007.
    public string Centar { get; set; } //Centar u kom je posiljka skenirana
    public string StatusOpis { get; set; } //Opisni Status
    public int NStatus { get; set; } //INT Kod statusa po AKS sifarniku
}

return Json(new { ErrorCode = Res, StatusList = lstStatusProm });
```

Metoda za pregled promene statusa posiljke (po referenci)

<http://www.akskurir.com/AKSVipService/Pracenje/Ref>

Protokol: output: JSON

Ref: Referenca posiljke iz polja Klijent Ref

Povratne informacije:

ErrorCode: (Int) – iz Liste

StatusList: StatusProm

```
public class StatusProm
{
    public DateTime Vreme { get; set; } //Datum i vreme skeniranja - promene statusa
    public int VremeInt { get; set; } //Datum i vreme kao sekunde od 01.01.2007.
    public string Centar { get; set; } //Centar u kom je posiljka skenirana
    public string StatusOpis { get; set; } //Opisni Status
    public int NStatus { get; set; } //INT Kod statusa po AKS sifarniku
}

return Json(new { ErrorCode = Res, StatusList = lstStatusProm });
```

Metoda za pregled trenutnog statusa liste posiljaka

<http://www.akskurir.com/AKSVipService/TrenutniStatusList>

Protokol: input text/plain output: JSON

Parametri:

PosiljalacIdent: Identifikacija klijenta, koriscena prilikom kreiranja posiljke

lstBarcode: lista brojeva posiljaka (List<long>)

Povratne informacije:

ErrorCode: (Int) – iz Liste

StatusList: StatusProm

```
public class MDLStatusListRes
{
    public long Barcode { get; set; }
    public int Status { get; set; }
}

return Json(new { ErrorCode = Res, lstStatus= lstMDLStatus });
```

Metoda za pregled trenutnog statusa liste posiljaka JSON protokolom

<http://www.akskurir.com/AKSVipService/TrenutniStatusListJSON>

Protokol: input: JSON output: JSON

Parametri:

PosiljalacIdent: Identifikacija klijenta, koriscena prilikom kreiranja posiljke

lstBarcode: lista brojeva posiljaka (List<long>)

Povratne informacije:

ErrorCode: (Int) – iz Liste

StatusList: StatusProm

```
public class MDLStatusListRes
{
    public long Barcode { get; set; }
    public int Status { get; set; }
}

return Json(new { ErrorCode = Res, lstStatus= lstMDLStatus });
```

Metoda za trajno otkazivanje posiljke

<http://www.akskurir.com/AKSVipService/OtkaziPosiljku>

Protokol: input text/plain output: JSON

Parametri:

PosiljalacID: Identifikacija klijenta, koriscena prilikom kreiranja posiljke

ID: Jedinstveni broj posiljke

Povratne informacije:

ErrorCode: (Int) – iz Liste

Metoda za trajno otkazivanje posiljke JSON protokolom

<http://www.akskurir.com/AKSVipService/OtkaziPosiljkuJSON>

Protokol: input: JSON output: JSON

Parametri:

PosiljalacIdent: Identifikacija klijenta, koriscena prilikom kreiranja posiljke

ID: Jedinstveni broj posiljke

Povratne informacije:

ErrorCode: (Int) – iz Liste

Metoda za preuzimanje podataka za posiljku

<http://www.akskurir.com/AKSVipService/PreuzmiPosiljku>

Protokol: input text/plain output: JSON

Parametri:

PosiljalacIdent: Identifikacija klijenta, korisćena prilikom kreiranja posiljke

ID: Jedinstveni broj posiljke

Povratne informacije:

```
public class Posiljka
{
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public string PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PosiljalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PosiljalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PosiljalacPostanskiBroj { get; set; } //iz liste npr 11000

    public string PrimalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PrimalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PrimalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PrimalacPostanskiBroj { get; set; } //iz liste npr 11000

    public int BrojPaketa { get; set; } //Broj dodatnih paketa za posiljku 0
- default
    public double MasaPosiljke { get; set; } //Ukupna masa posiljke u kilogramima
    public int OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste

    public string KlijentRef { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef2 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef3 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef4 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef5 { get; set; } //Referenca za povezivanje posiljke
(max len 50)

    public double IznosVrednostiPouzece { get; set; } //Iznos vrednosti za placanje
pouzećem
    public int ElektronskaUplata { get; set; } //Da li se za posiljku izvsava
elektronska uplata (1-da, 0-ne)
```

```
        public int PovratOtpremnice { get; set; } //Da li je potreban povrat potpisane  
otpremnice (1-da, 0-ne)  
        public double VrednostPosiljkeOsiguranje { get; set; } //Vrednost posiljke za  
iznos osiguranja  
  
        public int SMSPotvrdaIsporuke { get; set; } //Prijem SMS poruke po uspesnoj  
isporuci posiljke  
  
        public double CenaTezina { get; set; } // Cena transporta po tezini  
        public int NacinPlacanja { get; set; } // Nacin placanja usluge transporta  
0-PosiljalacVirman, 1- PosiljalacGotovina, 2-Primalac  
        public int PlacanjeDodatneUsluge { get; set; } //Nacin placanja dodatne usluge  
transporta 0-Posiljalac paketa, 1-PrimalacPaketa  
  
        public string Napomena { get; set; } //<120 karaktera = napomena koja ce se  
prikazivati kuriru prilikom isporuke  
  
    }
```


Metoda za preuzimanje podataka za posiljku JSON protokolon

<http://www.akskurir.com/AKSVipService/PreuzmiPosiljkuJSON>

Protokol: input JSON output: JSON

Parametri:

PosiljalacIdent: Identifikacija klijenta, korisnica prilikom kreiranja posiljke

ID: Jedinstveni broj posiljke

Povratne informacije:

```
public class Posiljka
{
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public string PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PosiljalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PosiljalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PosiljalacPostanskiBroj { get; set; } //iz liste npr 11000

    public string PrimalacIme { get; set; } // <50 karaktera npr Pera Peric
    public string PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public string PrimalacKontaktOsoba { get; set; } // <50 karakteranpr Mika Peric
    public string PrimalacTelefon { get; set; } //<20 karaktera npr 063608323
    public int PrimalacPostanskiBroj { get; set; } //iz liste npr 11000

    public int BrojPaketa { get; set; } //Broj dodatnih paketa za posiljku 0
- default
    public double MasaPosiljke { get; set; } //Ukupna masa posiljke u kilogramima
    public int OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste

    public string KlijentRef { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef2 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef3 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef4 { get; set; } //Referenca za povezivanje posiljke
(max len 50)
    public string KlijentRef5 { get; set; } //Referenca za povezivanje posiljke
(max len 50)

    public double IznosVrednostiPouzece { get; set; } //Iznos vrednosti za placanje
pouzecem
    public int ElektronskaUplata { get; set; } //Da li se za posiljku izvrsava
elektronska uplata (1-da, 0-ne)
```

```

        public int PovratOtpremnice { get; set; } //Da li je potreban povrat potpisane
otpremnice (1-da, 0-ne)
        public double VrednostPosiljkeOsiguranje { get; set; } //Vrednost posiljke za
iznos osiguranja

        public int SMSPotvrdaIsporuke { get; set; } //Prijem SMS poruke po uspesnoj
isporuci posiljke

        public double CenaTezina { get; set; } // Cena transporta po tezini
        public int NacinPlacanja { get; set; } // Nacin placanja usluge transporta
0-PosiljalacVirman, 1- PosiljalacGotovina, 2-Primalac
        public int PlacanjeDodatneUsluge { get; set; } //Nacin placanja dodatne usluge
transporta 0-Posiljalac paketa, 1-PrimalacPaketa

        public string Napomena { get; set; } //<120 karaktera = napomena koja ce se
prikazivati kuriru prilikom isporuke
    }

```

Metoda za upisivanje nove pozivnice

<http://www.akskurir.com/AKSVipService/NovaPozivnica/ps>

Protokol: input `text/plain` output: JSON

Pozivnica ps;

```
public class Pozivnica
{
    public long ID { get; set; } //INTERNAL USE

    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public Int64 PozivnicaID { get; set; } //0

    public String PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric

    public String PosiljalacGrad { get; set; } //Grad posiljaoca

    public String PosiljalacMesto { get; set; } //Mesto posiljaoca

    public String PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22

    public String PosiljalacTelefon { get; set; } //<20 karaktera npr 063123456

    public String PosiljalacKontaktOsoba { get; set; } //<50 karaktera npr Pera Peric

    public String PosiljalacEmail { get; set; }

    public String PrimalacIme { get; set; } // <50 karaktera npr Pera Peric

    public String PrimalacGrad { get; set; } //Grad primaoca

    public String PrimalacMesto { get; set; } //Mesto primaoca

    public String PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22

    public String PrimalacTelefon { get; set; } //<20 karaktera npr 063123456

    public String PrimalacKontaktOsoba { get; set; } //<50 karaktera npr Pera Peric

    public Int16 BrojPaketa { get; set; } //Broj paketa za posiljku 1 - default

    public Double Masa { get; set; } //Ukupna masa posiljke u kilogramima

    public Byte OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste

    public Double IznosVrednostiPovrat { get; set; } //Iznos vrednosti za placanje
pouzece

    public Byte Placanje { get; set; } // Nacin placanja usluge transporta 0-
Primalac, 1- PosiljalacGotovina, 3-PosiljalacVirman

    public Byte PlacanjeDodatne { get; set; } // Placanje dodatne usluge 0-
PrimalacPaketa, 1-PosiljalacPaketa
```

```

        public String Napomena { get; set; }///<120 karaktera = napomena koja ce se
        prikazivati kuriru prilikom isporuke

        public String NapomenaPreuzimanje { get; set; }///<120 karaktera = napomena koja
        ce se prikazivati kuriru prilikom preuzimanja

        public String StatusNapomena { get; set; }

        public Boolean OTP { get; set; }///

```

Povratne informacije:

ErrorCode: (Int) – iz Liste

PosiljkaID: (Int64) Jedinstveni broj pozivnice pod kojim ce se zahtev voditi u sistemu

```
return Json(new { ErrorCode = Res, PozivnicaID = PozivnicaID });
```

Metoda za upisivanje nove pozivnice JSON protokolom

<http://www.akskurir.com/AKSVipService/NovaPozivnicaJSON/>

Protokol: input: JSON output: JSON

Ulazni parametar: Pozivnica

```
public class Pozivnica
{
    public long ID { get; set; } //INTERNAL USE
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira
    public Int64 PozivnicaID { get; set; } //0
    public String PosiljalacIme { get; set; } // <50 karaktera npr Pera Peric
    public String PosiljalacGrad { get; set; } //Grad posiljaoca
    public String PosiljalacMesto { get; set; } //Mesto posiljaoca
    public String PosiljalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public String PosiljalacTelefon { get; set; } //<20 karaktera npr 063123456
    public String PosiljalacKontaktOsoba { get; set; } //<50 karaktera npr Pera Peric
    public String PosiljalacEmail { get; set; }
    public String PrimalacIme { get; set; } // <50 karaktera npr Pera Peric
    public String PrimalacGrad { get; set; } //Grad primaoca
    public String PrimalacMesto { get; set; } //Mesto primaoca
    public String PrimalacAdresa { get; set; } // <50 karaktera npr Beogradska 22
    public String PrimalacTelefon { get; set; } //<20 karaktera npr 063123456
    public String PrimalacKontaktOsoba { get; set; } //<50 karaktera npr Pera Peric
    public Int16 BrojPaketa { get; set; } //Broj paketa za posiljku 1 - default
    public Double Masa { get; set; } //Ukupna masa posiljke u kilogramima
    public Byte OpisSadrzineID { get; set; } //Opis sadrzine posiljke iz liste
    public Double IznosVrednostiPovrat { get; set; } //Iznos vrednosti za placanje
pouzece
    public Byte Placanje { get; set; } // Nacin placanja usluge transporta 0-
Primalac, 1- PosiljalacGotovina, 3-PosiljalacVirman
    public Byte PlacanjeDodatne { get; set; } // Placanje dodatne usluge 0-
PrimalacPaketa, 1-PosiljalacPaketa
```

```

        public String Napomena { get; set; }///<120 karaktera = napomena koja ce se
        prikazivati kuriru prilikom isporuke

        public String NapomenaPreuzimanje { get; set; }///<120 karaktera = napomena koja
        ce se prikazivati kuriru prilikom preuzimanja

        public String StatusNapomena { get; set; }

        public Boolean OTP { get; set; }///

```

Povratne informacije:

ErrorCode: (Int) – iz Liste

PosiljkaID: (Int64) Jedinstveni broj pozivnice pod kojim ce se zahtev voditi u sistemu

`return` `Json(new { ErrorCode = Res, PozivnicaID = PozivnicaID });`

Metoda za dobijanje ID posiljke za pozivnicu

<http://www.akskurir.com/AKSVipService/PosiljkaZaPozivnicu/ID>

Protokol: output: JSON

ID: Jedinstveni broj pozivnice (dobijen prilikom kreiranja)

Povratne informacije:

ErrorCode: (Int) – iz Liste

PosiljkaID: (long) – ID Posiljke

Metoda za dobijanje ID naloga povrata za posiljku (po ID-u posiljke)

<http://www.akskurir.com/AKSVipService/PosiljkaPovrat/ID>

Protokol: output: JSON

ID: Jedinstveni broj posiljke

Povratne informacije:

ErrorCode: (Int) – iz Liste

PovratID: (long) – ID Naloga povrata

Metoda za dobijanje liste promene statusa za sve naloge

<http://www.akskurir.com/AKSVipService/SPCTrackAllChanges>

Protokol: input:JSON output: JSON

Ulazni parameter: STPromSpec_PRIDRequest

```
public class STPromSpec_PRIDRequest
{
    public string PosiljalacIdent { get; set; } //Identifikacija klijenta u bazi AKS
Express Kurira

    public long LastKeyID { get; set; } //ID dobijen u odgovoru prilikom prethodnog
poziva

    public int MaxRows { get; set; } //Limit broja redova u odgovoru (Default=10,
Max=100)
}
```

Povratne informacije:

ErrorCode: (Int) – iz Liste

LastKeyID: (long) – poslednji id/id koji ce biti deo narednog poziva

ChangesCount: (int) – broj promena

List<STPromSpec_Changeltem>: lista promena

```
public class STPromSpec_Changeltem
{
    public DateTime Vreme { get; set; } //Datum i vreme skeniranja - promene statusa
    public int VremeInt { get; set; } //Datum i vreme kao sekunde od 01.01.2007.
    public string Centar { get; set; } //Centar u kom je posiljka skenirana
    public string StatusOpis { get; set; } //Opisni Status
    public int NStatus { get; set; } //INT Kod statusa po AKS sifarniku

    public long ChangeKeyID { get; set; } // Unique ID promene u sistemu

    public long Barcode { get; set; } //Barcode - IDPosiljke/Paketa
}
```

Lista gresaka

```
public static int OK_NO_ERROR = 0;
public static int ERROR_DATABASE_CONNECTION = 100;
public static int ERROR_BAD_IDENT = 201;
public static int ERROR_BARCODE_NUMBERS = 202;
public static int ERROR_BAD_FORMAT = 203;

public static int ERROR_POST_CODE_1 = 204;
public static int ERROR_POST_CODE_2 = 205;

public static int ERROR_CONTENT_ID = 206;
public static int ERROR_SQL_INPUT = 207;

public static int ERROR_DATA_MISSING = 209;
public static int ERROR_BARCODE_EXISTS = 210;

public static int ERROR_BARCODE_DOES_NOT_EXIST = 301;
```